Cat. No. W451-RU2-03

Серия SYSMAC CP
CP1H-X40D□-□, CP1H-XA40D□-□,
CP1H-Y20DT-D
CP1L-L14D□-□, CP1L-L20D□-□,
CP1L-M30D□-□, CP1L-M40D□-□,
CP1L-L10□-□, CP1L-M60□-□

Модули ЦПУ CP1H/CP1L

РУКОВОДСТВО ПО ПРОГРАММИРОВАНИЮ

OMRON

Серия SYSMAC CP

CP1H-X40D□-□, CP1H-XA40D□-□, CP1H-Y20DT-D

Модули ЦПУ CP1H

CP1L-L14D□-□, CP1L-L20D□-□, CP1L-M30D□-□, CP1L-M40D□-□ CP1L-L10□-□, CP1L-M60□-□

Модули ЦПУ CP1L

Руководство по программированию

Версия: май, 2007 г.

Предварительная версия
Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

Замечание:

Продукты компании OMRON должны использоваться надлежащим образом, только для целей, описанных в настоящем руководстве, и только квалифицированным персоналом.

В настоящем руководстве для обозначения различных типов опасности используются следующие предупреждающие знаки и надписи. Обязательно учитывайте информацию, которую они содержат. Пренебрежение этой информацией может стать причиной несчастного случая или материального ущерба.

⚠ **ОПАСНОСТЬ** Указывает на чрезвычайно опасную ситуацию, которая, если не принять меры к ее устранению, приведет к смерти или серьезной травме. Кроме того, может быть нанесен значительный материальный ущерб.

⚠ **ВНИМАНИЕ** Указывает на потенциально опасную ситуацию, которая, если не принять меры к ее устранению, может привести к смерти или серьезной травме. Кроме того, может быть нанесен значительный материальный ущерб.

⚠ **Предупреждение** Указывает на потенциально опасную ситуацию, которая, если не принять меры к ее устранению, может привести к травме средней или легкой степени тяжести либо нанесению материального ущерба.

Вспомогательные обозначения

Сокращение «Ch», которое появляется на некоторых дисплеях и на некоторых продуктах OMRON, часто означает «слово» и в документации в этом смысле имеет сокращение «Wd».

Сокращение «PLC» (ПЛК) означает «Программируемый контроллер». Однако на некоторых экранах CX-Programmer может встречаться сокращение «PC», которое также означает «Программируемый контроллер».

Информационные знаки

Для выделения информации различного типа в левой колонке настоящего руководства используются следующие заголовки и обозначения.

Примечание. Особенно интересная и полезная информация о наиболее эффективных и удобных способах работы с изделиями.

1,2,3... 1. Обозначение последовательности действий, перечня или любого другого списка.

© OMRON, 2005

Все права защищены. Воспроизведение, размещение в информационно-поисковой системе или передача третьему лицу какой-либо части настоящего руководства в какой-либо форме и каким-либо способом (механическим, электронным, путем ксерокопирования, записи на носитель или иным способом) не допускается без предварительного письменного разрешения компании OMRON.

Использование информации, содержащейся в настоящем руководстве, не сопряжено с какой-либо патентной ответственностью. Кроме того, поскольку компания OMRON неуклонно стремится к совершенствованию своей продукции, информация, содержащаяся в настоящем руководстве, может быть изменена без предупреждения. Подготовка настоящего руководства выполнялась с надлежащей тщательностью. Тем не менее, данная версия является предварительной, компания OMRON не несет ответственности за какие-либо ошибки и упущения в данном руководстве.

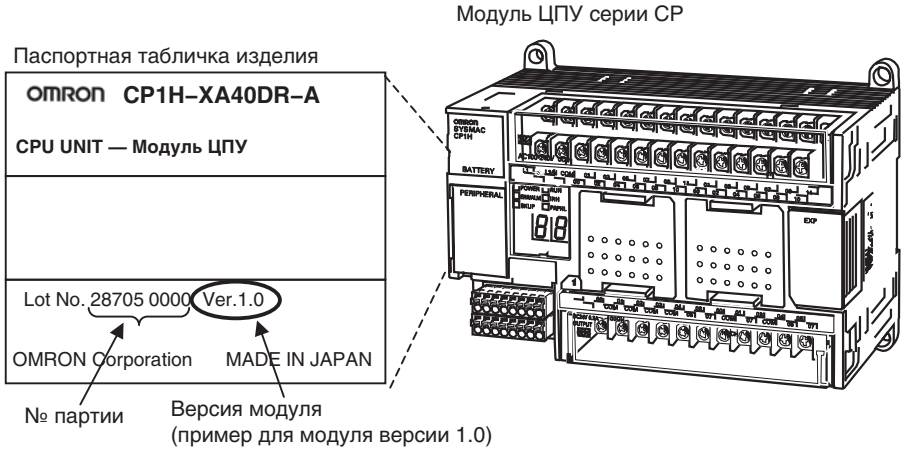
Версии модулей ЦПУ серии CP

Версии модулей

Понятие «версия модуля» («исполнение модуля») было введено для классификации модулей ЦПУ серии CP в соответствии с различиями в их функциональности, возникающими по мере обновления модулей.

Обозначение версии модуля на изделии

Версия модуля указывается справа от номера партии на паспортной табличке изделия, для которого введено различение по версиям (см. рис. ниже).



Определение версии модуля с помощью программного обеспечения на ПК

Способы определения версии модуля для модулей ЦПУ серии CP1H и модулей ЦПУ серии CP1L немного отличаются.

Модули ЦПУ CP1H

Версию модуля можно определить с помощью программы CX-Programmer версии 6.1 и выше, используя один из двух следующих способов (см. примеч.).

- С помощью диалогового окна ***PLC Information (Информация о ПЛК)***.
- С помощью диалогового окна ***Unit Manufacturing Information (Производственная информация модуля)***.

Примечание. Версию модулей ЦПУ CP1H невозможно определить в CX-Programmer версии ниже 6.1.

Модули ЦПУ CP1L

Для определения версии модуля можно использовать диалоговое окно ***PLC Information (Информация о ПЛК)*** программы CX-Programmer версии 7.1 и выше (см. примеч.). Диалоговое окно ***Unit Manufacturing Information (Производственная информация модуля)*** использовать невозможно.

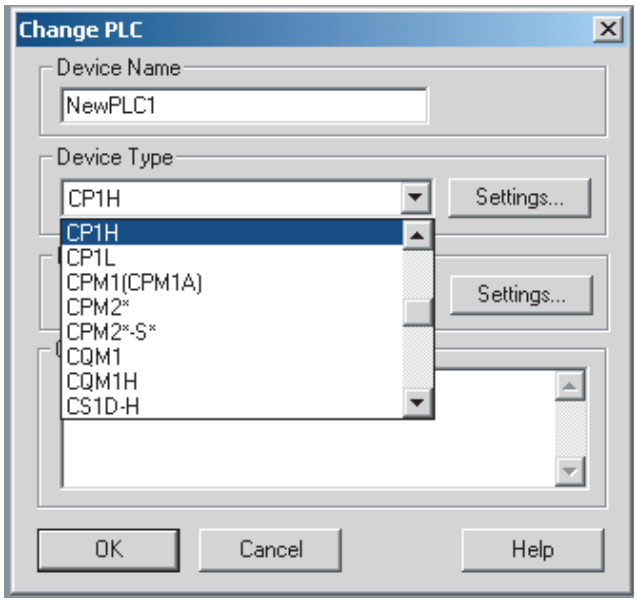
Примечание. Версию модулей ЦПУ CP1L невозможно определить в CX-Programmer версии ниже 7.1.

■ Диалоговое окно «PLC Information» (Информация о ПЛК)

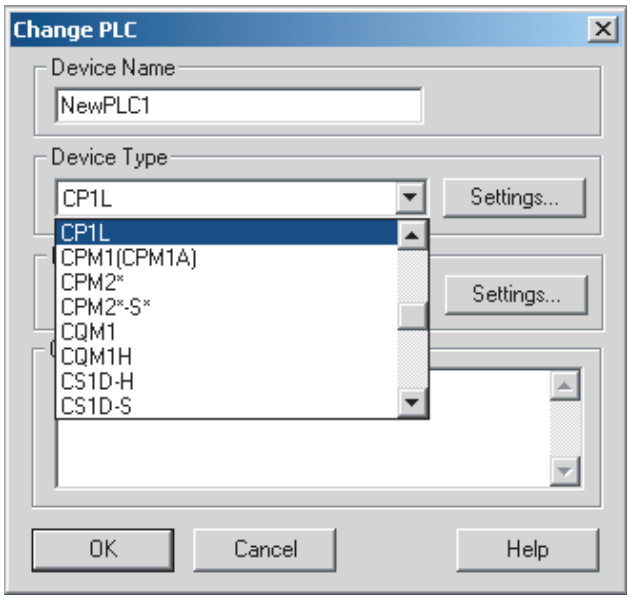
Процедура при известном типе устройства и типе ЦПУ

1,2,3... 1. Если известен тип устройства и тип ЦПУ, выберите их в диалоговом окне «Change PLC» (Изменение ПЛК), перейдите в онлайн-режим и выберите команду меню ***PLC (ПЛК) — Edit (Правка) — Information (Информация)***. Откроется показанное ниже окно изменения ПЛК.

Пример для CP1H

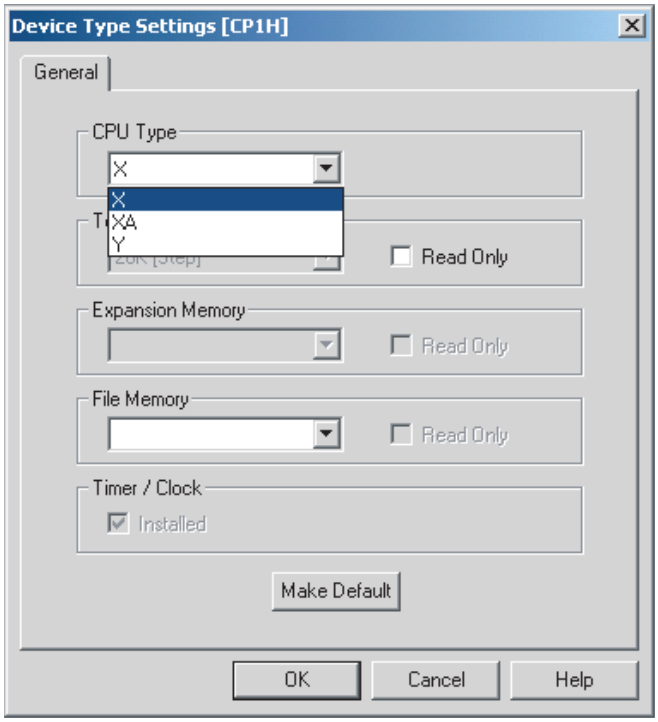


Пример для CP1L

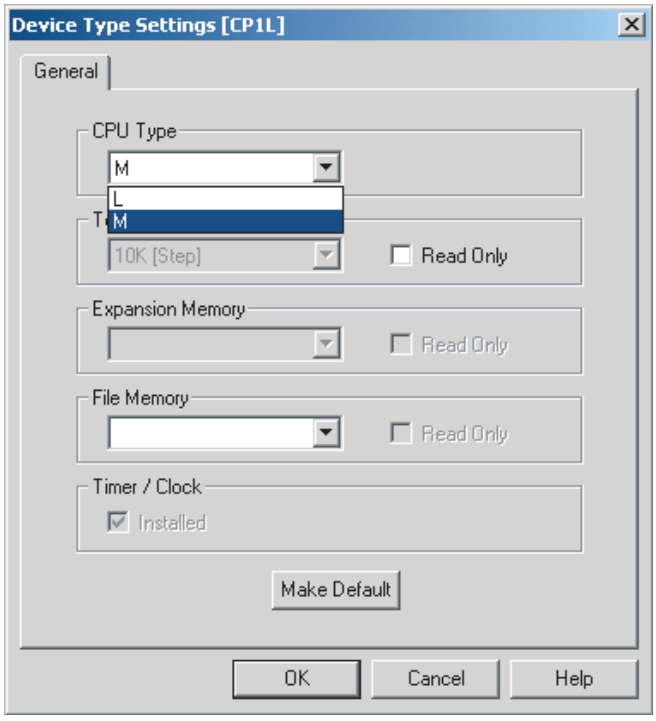


2. Щелкните кнопку **Settings (Параметры)**. В отобразившемся диалоговом окне «Device Type Settings» (Параметры типа устройства) выберите тип ЦПУ.

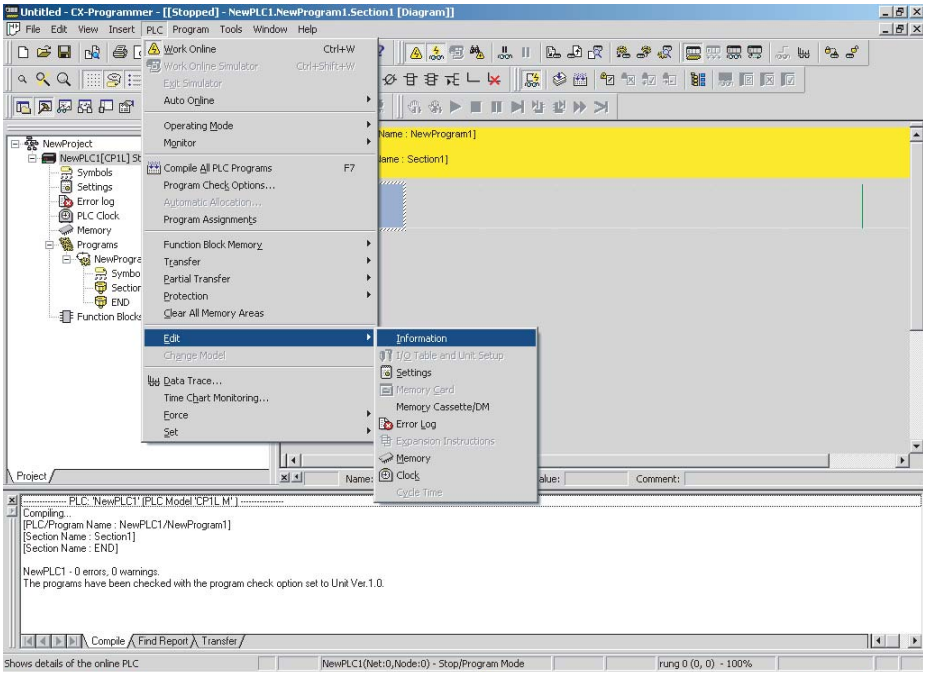
Пример для CP1H



Пример для CP1L

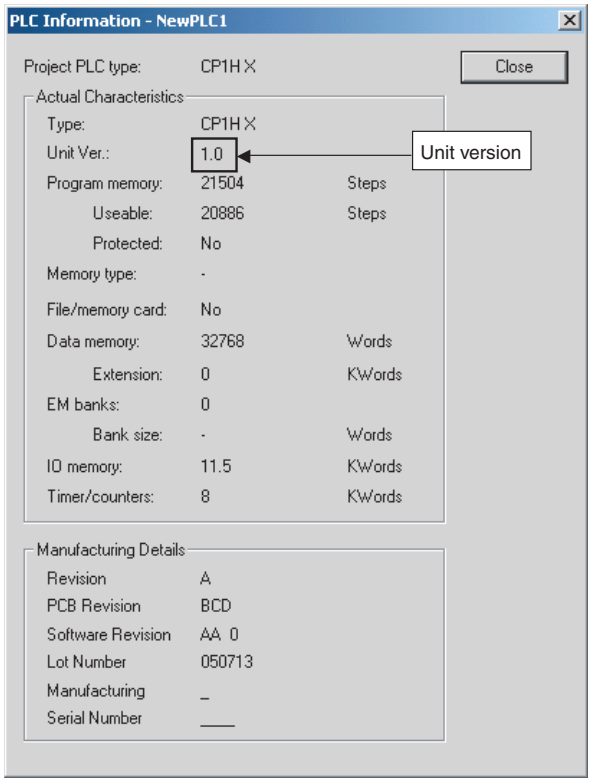


3. Перейдите в режим онлайн и выберите ***PLC (ПЛК) — Edit (Правка) — Information (Информация)***

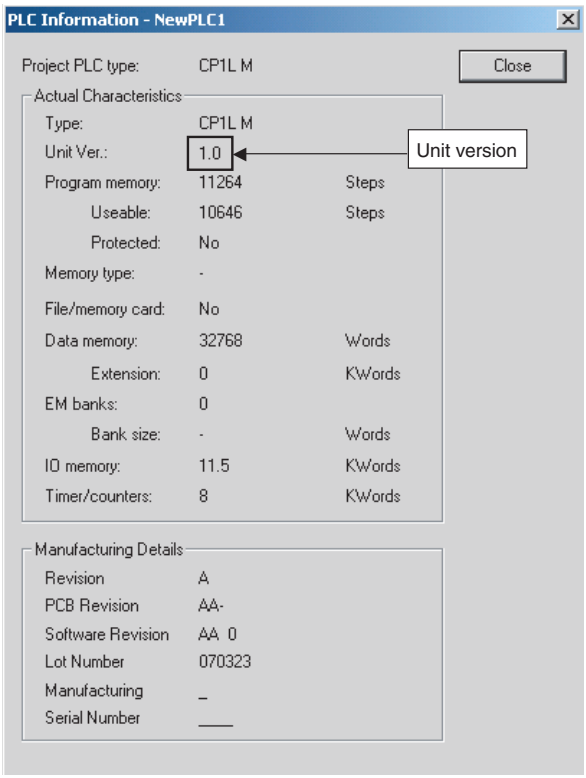


Откроется диалоговое окно «PLC Information» (Информация о ПЛК).

Пример для CP1H



Пример для CP1L



Приведенное выше окно позволяет определить версию модуля ЦПУ.

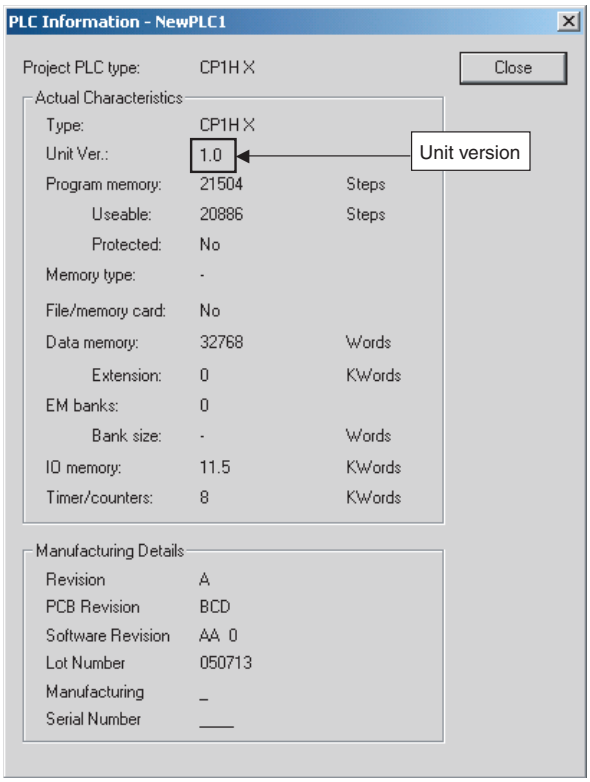
Предварительная версия
Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

Процедура при неизвестном типе устройства и типе ЦПУ

Применение данного способа возможно только при прямом подключении к модулю ЦПУ по последовательному интерфейсу.

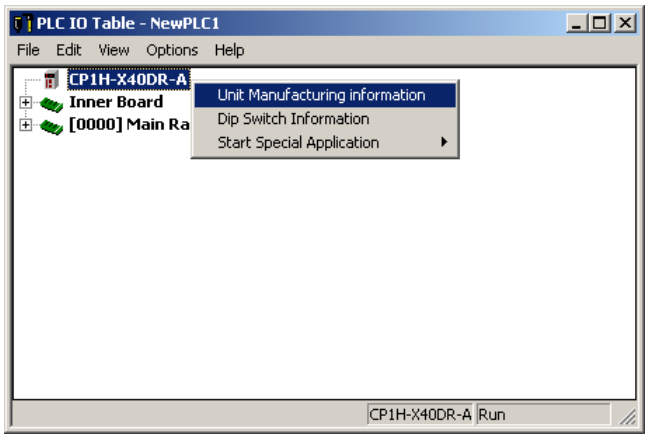
Если тип устройства и тип ЦПУ неизвестны, но есть возможность прямого подключения к модулю ЦПУ по последовательному интерфейсу, выберите ***PLC (ПЛК) — Auto Online (Автоматическое соединение с ПЛК)*** для перехода в онлайн-режим, после чего выберите команду меню ***PLC (ПЛК) — Edit (Правка) — Information (Информация)***.

Откроется диалоговое окно «PLC Information» (Информация о ПЛК), которое можно использовать для определения версии модуля ЦПУ.

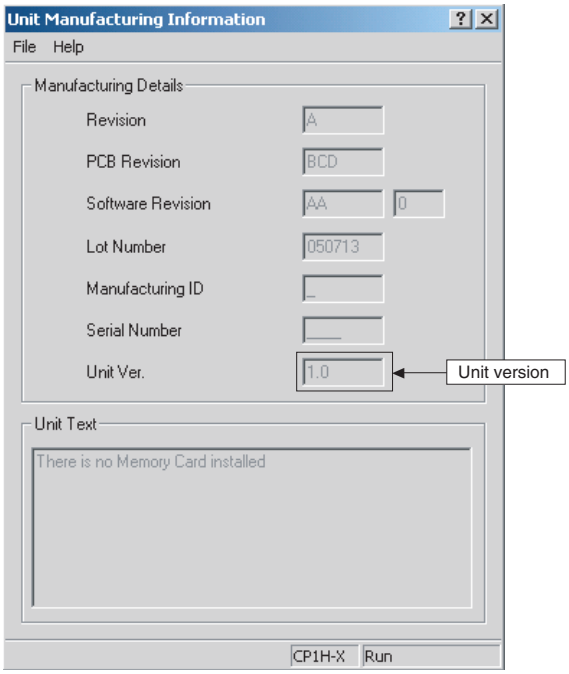


■ Диалоговое окно «Unit Manufacturing Information» (Производственная информация модуля) (только для модулей ЦПУ CP1H)

1,2,3... 1. В окне «IO Table» (Таблица ввода/вывода) щелкните правой кнопкой мыши и выберите ***Unit Manufacturing information — CPU Unit (Производственная информация модуля — модуль ЦПУ)***.



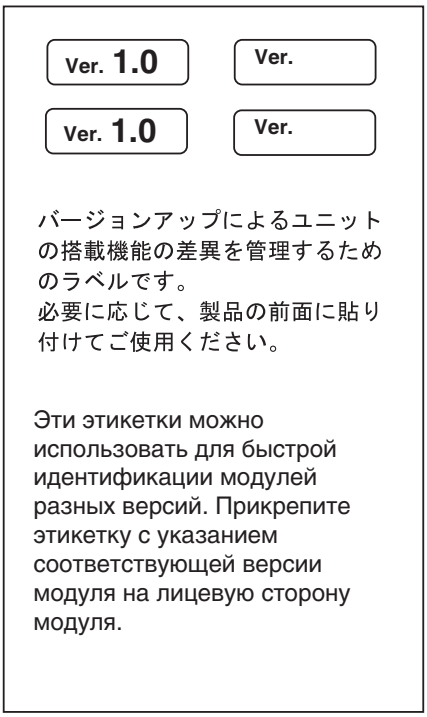
2. Отобразится диалоговое окно «Unit Manufacturing information» (Производственная информация модуля) следующего вида.



Приведенное выше окно позволяет определить версию модуля ЦПУ, с которым установлена связь.

Использование этикеток с версией модуля

В комплекте с модулем ЦПУ поставляются следующие этикетки, на которых указывается версия модуля.



Эти этикетки можно прикрепить с лицевой стороны каждого модуля ЦПУ, чтобы можно было отличить модули ЦПУ разных версий.

СОДЕРЖАНИЕ

МЕРЫ ПРЕДОСТОРОЖНОСТИ xxiii

1 Для кого предназначено руководство. xxiv
2 Меры предосторожности общего характера xxiv
3 Меры предосторожности и обеспечения безопасности xxiv
4 Меры предосторожности в связи с условиями эксплуатации xxvi
5 Меры предосторожности при эксплуатации xxvii
6 Соответствие Директивам ЕС xxxi

РАЗДЕЛ 1
Основные принципы программирования 1

1-1 Основные принципы программирования 2
1-2 Меры предосторожности 36
1-3 Проверка программ 45
1-4 Функциональные блоки 51

РАЗДЕЛ 2
Задачи 55

2-1 Программирование с использованием задач 56
2-2 Применение задач 65
2-3 Задачи обработки прерываний 77
2-4 Использование CX-Programmer для задач. 85

РАЗДЕЛ 3
Команды 87

3-1 Условные обозначения и структура информации при описании команд 95
3-2 Входные битовые команды 99
3-3 Выходные битовые команды. 126
3-4 Команды управления последовательностью выполнения 146
3-5 Команды управления таймерами и счетчиками 184
3-6 Команды сравнения 230
3-7 Команды передачи данных 271
3-8 Команды сдвига данных 301
3-9 Команды увеличения/уменьшения. 349
3-10 Символьные математические команды 366
3-11 Команды преобразования 425
3-12 Логические команды 477
3-13 Специальные математические команды 493
3-14 Команды математических операций над числами с плавающей запятой 516
3-15 Команды для чисел с плавающей запятой двойной точности. 573
3-16 Команды обработки табличных данных 620
3-17 Команды управления данными. 671

СОДЕРЖАНИЕ

3-18 Команды подпрограмм 730
3-19 Команды управления прерываниями 755
3-20 Команды управления скоростными счетчиками и импульсными выходами 770
3-21 Команды для пошагового выполнения 824
3-22 Команды базовых модулей ввода/вывода 843
3-23 Команды последовательного интерфейса 882
3-24 Сетевые команды 924
3-25 Команды дисплея 997
3-26 Команды для работы с часами 1006
3-27 Команды для отладки программы 1020
3-28 Команды для диагностики неисправностей 1025
3-29 Прочие команды 1051
3-30 Команды для создания программных блоков 1067
3-31 Команды обработки текстовых строк 1103
3-32 Команды управления задачами 1137
3-33 Команды преобразования моделей 1146

РАЗДЕЛ 4
Время выполнения и размер команд 1165

4-1 Время выполнения и размер команд 1166
4-2 Время выполнения экземпляра функционального блока 1191

ПРИЛОЖЕНИЯ

A Классификация команд по функциям 1193
B Список команд в порядке возрастания кодов функций 1203
C Список команд в алфавитном порядке 1219

Указатель 1231

Перечень версий 1241

О данном руководстве:

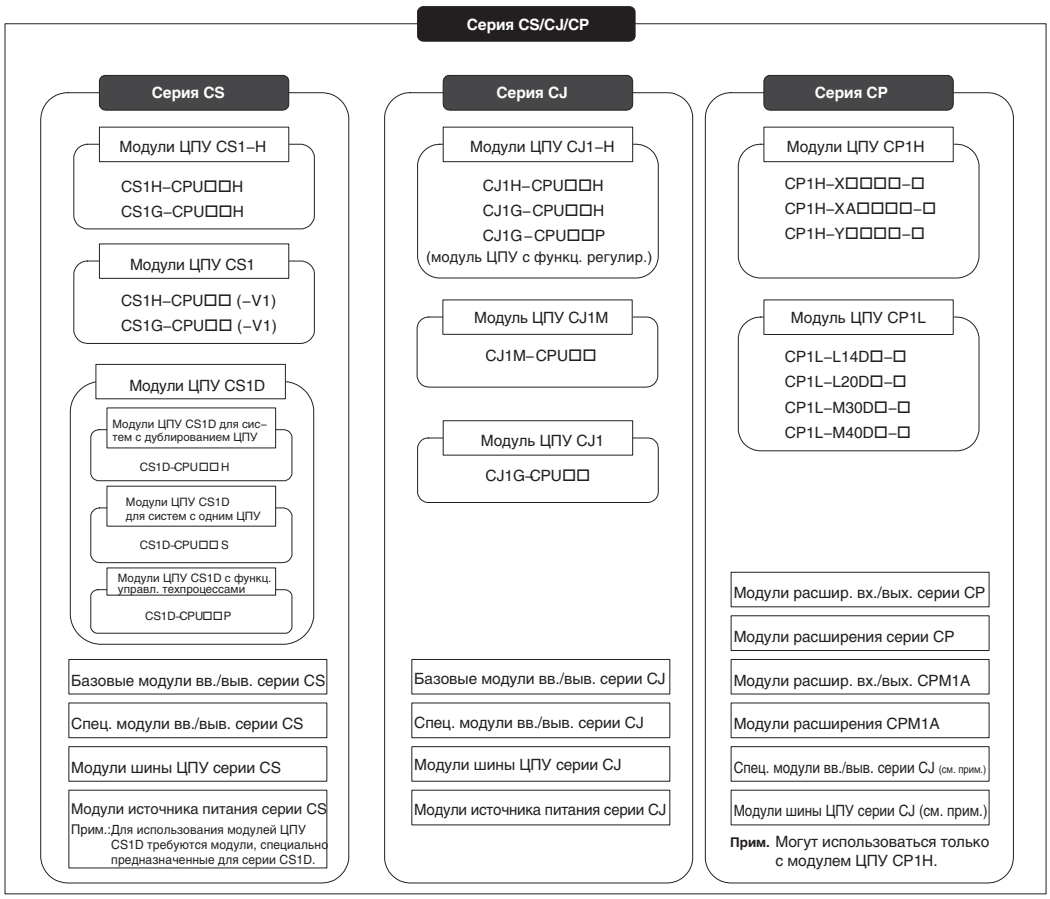
Данное руководство описывает программирование программируемых контроллеров (ПЛК) серии CP и содержит разделы, описанные ниже. CP1H и CP1L — это усовершенствованные компактные ПЛК, в основу которых легли передовые технологии управления и обширный опыт компании OMRON в области промышленной автоматизации.

Пожалуйста, внимательно прочитайте данное руководство и подробно изучите содержащуюся в нем информацию, прежде чем приступать к монтажу или эксплуатации ПЛК серии CP1H или CP1L. Обязательно ознакомьтесь с мерами предосторожности, которые приведены в следующем разделе.

Краткая характеристика серии CP

Серия CP, сердцем которой являются два модуля центрального процессора — CP1H и CP1L, базируется на той же архитектуре, что легла в основу ПЛК серий CS и CJ. С модулями ЦПУ серии CP1H могут использоваться специальные модули ввода/вывода и модули шины ЦПУ серии CJ. Базовые модули ввода/вывода серии CJ, однако, не поддерживаются. Для увеличения количества входов/выходов в ПЛК CP1H или CP1L должны использоваться только модули расширения серии CPM1A и модули расширения входов/выходов серии CPM1A.

Слова памяти для входов/выходов распределяются точно так же, как в ПЛК CPM1A/CPM2A: для входов и выходов используются фиксированные области памяти.



Меры предосторожности: приведены меры предосторожности общего характера, которые должны соблюдаться при эксплуатации программируемого контроллера и связанных с ним устройств.

Раздел 1 описывает основные понятия и принципы программирования контроллеров серии CP1H.

Раздел 2 описывает основные принципы работы задач и поясняет применение задач при создании программ.

Раздел 3 содержит подробное описание каждой команды, которая может быть использована для программирования ПЛК серии CP. Описываемые команды сгруппированы по выполняемым функциям.

Раздел 4 содержит значения времени выполнения и количества шагов для каждой команды, которая поддерживается программируемыми контроллерами серии CP1H. Также содержит методику расчета времени выполнения и количества шагов для экземпляра функционального блока.

Приложения предоставляют перечни команд программирования, рассортированные по выполняемым функциям и в порядке возрастания номеров функций.

Сопутствующие руководства

Ниже перечислены руководства, относящиеся к модулям ЦПУ серии CP. Используйте эти руководства по мере необходимости.

Cat. No.	Номера моделей	Название руководства	Описание
W451	CP1H-X40D☐-☐ CP1H-XA40D☐-☐ CP1H-Y20DT-D CP1L-L14D☐-☐ CP1L-L20D☐-☐ CP1L-M30D☐-☐ CP1L-M40D☐-☐	Серия SYSMAC CP: Модули ЦПУ CP1H и CP1L — Руководство по программированию (это руководство)	Содержит следующую информацию о серии CP: • Команды языка программирования • Методы программирования • Задачи • Память файлов • Функции Используйте это руководство вместе с *Руководством по эксплуатации модулей ЦПУ CP1H* (W450) и *Руководством по эксплуатации модулей ЦПУ CP1L* (W462).
W450	CP1H-X40D☐-☐ CP1H-XA40D☐-☐ CP1H-Y20DT-D	Серия SYSMAC CP: Модули ЦПУ CP1H — Руководство по эксплуатации	Содержит следующую информацию о серии CP: • Краткий обзор, конструкция, монтаж, обслуживание и другие основные сведения • Свойства • Конфигурация системы • Механический и электрический монтаж • Адресное пространство памяти ввода/вывода • Поиск и устранение неисправностей Используйте это руководство вместе с *Руководством по программированию ПЛК CP1H* (W451).
W462	CP1L-L14D☐-☐ CP1L-L20D☐-☐ CP1L-M30D☐-☐ CP1L-M40D☐-☐	Серия SYSMAC CP: Модули ЦПУ CP1L — Руководство по эксплуатации	
W461	CP1L-L14D☐-☐ CP1L-L20D☐-☐ CP1L-M30D☐-☐ CP1L-M40D☐-☐	Серия SYSMAC CP: Модули ЦПУ CP1L — Вводное руководство	Содержит основные сведения о монтаже, настройке и вводе в эксплуатацию ПЛК CP1L. • Основная конфигурация и названия деталей и компонентов. • Порядок выполнения механического и электрического монтажа. • Порядок действий при программировании, загрузке программы и отладке с помощью CX-Programmer. • Примеры прикладных программ для ЦПУ серии CP1L.
W342	CS1G/H-CPU☐H CS1G/H-CPU☐☐-V1 CS1D-CPU☐☐H CS1D-CPU☐☐S CS1W-SCU21 CS1W-SCB21-V1/41-V1 CJ1G/H-CPU☐☐H CJ1G-CPU☐☐P CP1H-CPU☐☐ CJ1G-CPU☐☐ CJ1W-SCU21-V1/41-V1	Серия SYSMAC CS/CJ — Справочное руководство по командам связи	Описывает команды связи, предназначенные для модулей ЦПУ серии CS и серии CJ, в т. ч. команды C-режима и команды FINS. **Примечание.** Данное руководство описывает команды связи, предназначенные для модулей ЦПУ, безотносительно интерфейса, посредством которого они передаются (могут использоваться последовательные порты модулей ЦПУ, порты модулей/плат последовательного интерфейса и порты модулей связи). Информацию о командах, предназначенных для специальных модулей ввода/вывода и модулей шины ЦПУ, см. в соответствующих руководствах.

Cat. No.	Номера моделей	Название руководства	Описание
W446	WS02-CXPC1-E-V70	SYSMAC CX-Programmer, версия 7.0 — Руководство по работе	Содержит сведения об установке и работе с программой CX-Programmer и обо всех ее функциях, за исключением функциональных блоков.
W447	WS02-CXPC1-E-V70	SYSMAC CX-Programmer, версия 7.0 — Руководство по работе. Функциональные блоки.	Содержит описание и методы работы с функциональными блоками. Функциональные блоки могут использоваться при наличии программы CX-Programmer версии 6.1 и выше, для модулей ЦПУ CS1-H/CJ1-H с версией модуля 3.0 или модулей ЦПУ CP1H. Сведения о работе с другими функциями, не связанными с функциональными блоками, содержатся в руководстве W446.
W444	CXONE-AL□□C-E	CX-One — Руководство по установке и настройке	Содержит краткое описание комплекта программного обеспечения CX-One для систем промышленной автоматизации и порядок действий по его установке на ПК.
W445	CXONE-AL□□C-E	CX-Integrator — Руководство по работе	Описывает порядок работы с программой CX-Integrator, в том числе предоставляет информацию о конфигурировании сетей обмена данными (настройка логических связей, таблиц маршрутизации, модулей связи и др.).
W344	WS02-PSTC1-E	CX-Protocol — Руководство по работе	Подробно описывает порядок действий по созданию макросов протоколов (процедур обмена данными) с помощью программы CX-Protocol и содержит другие сведения о программировании макросов для реализации протоколов связи. Для создания макросов протоколов с целью реализации специфических процедур обмена данными по последовательному интерфейсу или адаптации стандартных системных протоколов требуется программа CX-Protocol.

Предварительная версия
Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

Внимательно прочитайте настоящее руководство

Пожалуйста, внимательно прочитайте это руководство, прежде чем приступать к использованию продукта. В случае если у Вас имеются какие-либо вопросы или комментарии, обращайтесь, пожалуйста, к региональному представителю компании OMRON.

Гарантийные обязательства и ограничение ответственности

ГАРАНТИЙНЫЕ ОБЯЗАТЕЛЬСТВА

Компания OMRON дает исключительную гарантию того, что в течение одного года (если не оговорен иной период) с даты продажи изделия компанией OMRON в изделии будут отсутствовать дефекты, связанные с материалами и изготовлением изделия.

КОМПАНИЯ OMRON НЕ ДАЕТ НИКАКИХ ГАРАНТИЙ ИЛИ ОБЯЗАТЕЛЬСТВ, ЯВНЫХ ИЛИ ПОДРАЗУМЕВАЕМЫХ, В ОТНОШЕНИИ СОБЛЮДЕНИЯ ЗАКОНОДАТЕЛЬСТВА ПРИ ИСПОЛЬЗОВАНИИ ИЗДЕЛИЯ, В ОТНОШЕНИИ КОММЕРЧЕСКОГО УСПЕХА ИЗДЕЛИЙ ИЛИ ИХ ПРИГОДНОСТИ ДЛЯ КОНКРЕТНОГО ПРИМЕНЕНИЯ. КАЖДЫЙ ПОКУПАТЕЛЬ ИЛИ ПОЛЬЗОВАТЕЛЬ ПРИЗНАЕТ, ЧТО ОПРЕДЕЛЕНИЕ СООТВЕТСТВИЯ ИЗДЕЛИЙ ТРЕБОВАНИЯМ, ПРЕДЪЯВЛЯЕМЫМ ПОКУПАТЕЛЕМ ИЛИ ПОЛЬЗОВАТЕЛЕМ, НАХОДИТСЯ В КОМПЕТЕНЦИИ САМОГО ПОКУПАТЕЛЯ ИЛИ ПОЛЬЗОВАТЕЛЯ. КОМПАНИЯ OMRON НЕ ПРИЗНАЕТ КАКИЕ-ЛИБО ИНЫЕ ЯВНЫЕ ИЛИ ПОДРАЗУМЕВАЕМЫЕ ГАРАНТИЙНЫЕ ОБЯЗАТЕЛЬСТВА.

ОГРАНИЧЕНИЕ ОТВЕТСТВЕННОСТИ

КОМПАНИЯ OMRON НЕ НЕСЕТ ОТВЕТСТВЕННОСТИ ЗА ПРЯМЫЕ, КОСВЕННЫЕ ИЛИ ВЫТЕКАЮЩИЕ УБЫТКИ, ПОТЕРЮ ПРИБЫЛИ ИЛИ КОММЕРЧЕСКИЕ ПОТЕРИ, КАКИМ БЫ ТО НИ БЫЛО ОБРАЗОМ СВЯЗАННЫЕ С ИЗДЕЛИЯМИ, НЕЗАВИСИМО ОТ ТОГО, ПРЕДЪЯВЛЯЕТСЯ ЛИ ИСК НА ОСНОВАНИИ КОНТРАКТА, ГАРАНТИЙНЫХ ОБЯЗАТЕЛЬСТВ, В СВЯЗИ С НЕБРЕЖНЫМ ОБРАЩЕНИЕМ ИЛИ НА ОСНОВАНИИ БЕЗУСЛОВНОГО ОБЯЗАТЕЛЬСТВА.

Ни при каких обстоятельствах ответственность компании OMRON по какому-либо иску не может превысить собственную стоимость изделия, на которое распространяется ответственность компании OMRON.

НИ ПРИ КАКИХ ОБСТОЯТЕЛЬСТВАХ КОМПАНИЯ OMRON НЕ НЕСЕТ ОТВЕТСТВЕННОСТИ ПО ГАРАНТИЙНЫМ ОБЯЗАТЕЛЬСТВАМ, РЕМОНТУ ИЛИ ДРУГИМ ИСКАМ В ОТНОШЕНИИ ИЗДЕЛИЙ, ЕСЛИ В РЕЗУЛЬТАТЕ АНАЛИЗА, ПРОВЕДЕННОГО КОМПАНИЕЙ OMRON, УСТАНОВЛЕНО, ЧТО В ОТНОШЕНИИ ИЗДЕЛИЙ НАРУШАЛИСЬ ПРАВИЛА ЭКСПЛУАТАЦИИ, ХРАНЕНИЯ, МОНТАЖА И ТЕХНИЧЕСКОГО ОБСЛУЖИВАНИЯ, ЧТО В ИЗДЕЛИЯХ ИМЕЮТСЯ ЗАГРЯЗНЕНИЯ, ЛИБО ИЗДЕЛИЯ ИСПОЛЬЗОВАЛИСЬ НЕ ПО НАЗНАЧЕНИЮ ИЛИ ПОДВЕРГАЛИСЬ НЕДОПУСТИМОЙ МОДИФИКАЦИИ ИЛИ РЕМОНТУ.

Замечания по применению

ПРИГОДНОСТЬ ДЛЯ КОНКРЕТНОГО ПРИМЕНЕНИЯ

Компания OMRON не несет ответственности за соответствие каким-либо стандартам, нормативам или правилам, которые действуют в случае применения изделий в составе оборудования заказчика или при использовании изделий.

По запросу заказчика компания OMRON предоставляет соответствующие сертификаты, выдаваемые сторонними организациями, в которых перечисляются обеспечиваемые номинальные параметры и указываются ограничения на применение изделий. Сама по себе эта информация не является достаточной для полного определения пригодности изделий для применения в конечной системе, машине, оборудовании или в других областях применения.

Ниже приведены некоторые примеры применения, требующие особого внимания. Этот перечень не является исчерпывающим перечнем возможного применения изделий и не гарантирует пригодность изделий для целей, в нем перечисленных.

- Использование вне зданий, использование в условиях возможного химического загрязнения или электрических помех, либо при условиях эксплуатации, не описанных в настоящем документе.
- Системы управления объектами ядерной энергетики, тепловые системы, железнодорожные системы, авиация, медицинское оборудование, игровые автоматы, транспортные средства, оборудование защиты и системы, эксплуатация которых регулируется отдельными промышленными или государственными нормативами.
- Системы, машины и оборудование, представляющие угрозу для жизни или имущества.

Следует ознакомиться и соблюдать все запреты, распространяющиеся на данные изделия.

НИ В КОЕМ СЛУЧАЕ НЕ ИСПОЛЬЗУЙТЕ ИЗДЕЛИЕ В СИСТЕМАХ, ПРЕДСТАВЛЯЮЩИХ СЕРЬЕЗНУЮ УГРОЗУ ДЛЯ ЖИЗНИ ИЛИ ИМУЩЕСТВА, НЕ ОБЕСПЕЧИВ БЕЗОПАСНОСТИ ВО ВСЕЙ СИСТЕМЕ В ЦЕЛОМ, А ТАКЖЕ НЕ УБЕДИВШИСЬ В ТОМ, ЧТО ИЗДЕЛИЯ OMRON ИМЕЮТ НАДЛЕЖАЩИЕ НОМИНАЛЬНЫЕ ХАРАКТЕРИСТИКИ, НАДЛЕЖАЩИМ ОБРАЗОМ СМОНТИРОВАНЫ И ИСПОЛЬЗУЮТСЯ ПО НАЗНАЧЕНИЮ.

ПРОГРАММИРУЕМЫЕ ИЗДЕЛИЯ

Компания OMRON не несет ответственности за программы пользователя, создаваемые для программируемых изделий, и за какие-либо последствия, возникшие в результате их применения.

Отказ от ответственности

ИЗМЕНЕНИЕ ХАРАКТЕРИСТИК

Характеристики изделия и дополнительные принадлежности могут быть изменены в любое время в целях улучшения параметров и по другим причинам.

Мы практикуем изменение номера модели в случае изменения ранее заявленных номинальных характеристик или свойств, либо в случае существенного изменения конструкции. Тем не менее, некоторые технические характеристики изделий могут быть изменены без какого-либо уведомления. В спорном случае по вашему запросу модели может быть присвоен специальный номер, идентифицирующий или определяющий ключевые характеристики, требуемые для вашей задачи. Актуальные сведения о технических характеристиках приобретаемых изделий всегда можно получить в региональном представительстве OMRON.

ГАБАРИТНЫЕ РАЗМЕРЫ И МАССЫ

В настоящем документе приведены номинальные значения габаритов и масс, и их нельзя использовать в конструкторской документации, даже если приведены значения допусков.

ЭКСПЛУАТАЦИОННЫЕ ХАРАКТЕРИСТИКИ

Приведенные в настоящем документе эксплуатационные характеристики служат в качестве ориентира для пользователей при определении пригодности изделий для задач пользователей и не являются предметом гарантийного обязательства. Эти характеристики могли быть получены в результате испытаний, проведенных компанией OMRON, и пользователи должны соотносить их с требованиями к реальным прикладным задачам. Фактические эксплуатационные характеристики являются предметом «Гарантийных обязательств» и «Ограничения ответственности» компании OMRON.

ОШИБКИ И ОПЕЧАТКИ

Информация, содержащаяся в настоящем руководстве, была тщательно проверена и, вероятнее всего, является точной; тем не менее, данная версия является предварительной, компания OMRON не несет ответственности за какие-либо ошибки и упущения в данном руководстве.

Предварительная версия
Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

МЕРЫ ПРЕДОСТОРОЖНОСТИ

Данный раздел содержит общие указания по использованию программируемых контроллеров (ПЛК) серии CP и связанных с ними устройств.

Данный раздел содержит важную информацию о безотказном и безопасном применении программируемых контроллеров (ПЛК). Обязательно прочитайте этот раздел и подробно изучите содержащуюся в нем информацию, прежде чем приступать к монтажу, настройке, программированию или эксплуатации системы ПЛК.

1 Для кого предназначено руководство xxiv

2 Меры предосторожности общего характера. xxiv

3 Меры предосторожности и обеспечения безопасности xxiv

4 Меры предосторожности в связи с условиями эксплуатации xxvi

5 Меры предосторожности при эксплуатации xxvii

6 Соответствие Директивам ЕС xxxi

6-1 Применимые Директивы xxxi

6-2 Общие принципы xxxi

6-3 Соответствие Директивам ЕС xxxi

6-4 Способы подавления помех на релейных выходах xxxii

6-5 Условия соответствия Директивам по ЭМС при использовании модулей расширения входов/выходов CP1/CP/CPM1A с релейными выходами. xxxiii

1 Для кого предназначено руководство

Данное руководство предназначено для лиц, обладающих специальными знаниями в области электрических систем (инженер-электрик и т. п.).

- Персонал, ответственный за установку промышленных систем автоматизации.
- Персонал, ответственный за разработку промышленных систем автоматизации.
- Персонал, ответственный за администрирование оборудования промышленных систем автоматизации.

2 Меры предосторожности общего характера

Пользователь должен применять изделие в соответствии с эксплуатационными характеристиками, приведенными в руководствах по эксплуатации.

Прежде чем использовать изделие в условиях, которые не описаны в руководстве, а также в случае применения изделия в системах управления объектами ядерной энергетики, в железнодорожных системах, в авиации, в транспортных средствах, в тепловых системах, в медицинском оборудовании, в игровых автоматах и аттракционах, в оборудовании защиты и других системах, машинах и установках, которые могут серьезно повлиять на здоровье людей и привести к повреждению имущества при условии неправильной эксплуатации, обязательно проконсультируйтесь в ближайшем представительстве компании Omron.

Убедитесь в том, что номинальные значения и рабочие характеристики изделия достаточны для систем, машин и оборудования, и предусматривайте в системах, машинах и оборудовании механизмы удвоенной надежности.

В данном руководстве содержатся сведения по программированию и эксплуатации модуля. Прежде чем приступать к эксплуатации модуля, обязательно прочитайте данное руководство и храните его в легко доступном место, чтобы использовать во время работы.

⚠ **ВНИМАНИЕ** Очень важно, чтобы ПЛК и все его модули использовались только для оговоренных целей и только при оговоренных условиях эксплуатации, особенно в тех приложениях, в которых они могут прямым или косвенным образом повлиять на здоровье человека. Прежде чем применять систему ПЛК в описанных выше приложениях, обязательно проконсультируйтесь в ближайшем представительстве компании Omron.

3 Меры предосторожности и обеспечения безопасности

⚠ **ВНИМАНИЕ** Не пытайтесь разбирать какой-либо модуль при поданном напряжении питания. Это может привести к поражению электрическим током.

⚠ **ВНИМАНИЕ** Никогда не касайтесь клемм или клеммных колодок при поданном напряжении питания. Это может привести к поражению электрическим током.

⚠ **ВНИМАНИЕ** Не пытайтесь разбирать, ремонтировать или модифицировать какой-либо модуль. Это может привести к поражению электрическим током, возгоранию или сбою в работе.

⚠ **ВНИМАНИЕ** Предусмотрите надлежащие меры защиты во внешних цепях (т. е. вне программируемого контроллера), включая перечисленные ниже, для обеспечения безопасности в нештатном режиме, который может возникнуть из-за ошибки в работе программируемого контроллера или иных внешних факторов, влияющих на его работу. Невыполнение этого требования может стать причиной несчастного случая с тяжкими последствиями.

- Во внешних схемах управления должны быть предусмотрены схемы аварийного останова, схемы блокировки, ограничительные схемы и другие меры безопасности.
- ПЛК выключит все свои выходы, если встроенная в него функция самодиагностики обнаружит какую-либо ошибку или будет выполнена команда FALS (предупреждение о серьезной неисправности). На случай возникновения таких ошибок должны быть предусмотрены внешние средства защиты, обеспечивающие безопасность системы.
- Выходы ПЛК могут оставаться включенными или выключенными из-за перегорания или сваривания контактов выходных реле или повреждения выходных транзисторов. На случай возникновения таких неисправностей должны быть предусмотрены внешние меры защиты, обеспечивающие безопасность в системе.
- Перегрузка или короткое замыкание на выходе напряжения 24 В= (напряжение питания ПЛК) может вызвать падение напряжения, что приведет к выключению выходов. На случай возникновения таких неисправностей должны быть предусмотрены внешние меры защиты, обеспечивающие безопасность в системе.

⚠ **ВНИМАНИЕ** Конечным пользователем должны быть предусмотрены меры защиты для обеспечения безопасности в случае ошибочной подачи, отсутствия или недопустимого уровня сигналов из-за неисправностей в сигнальных линиях, кратковременного прерывания питания или по другим причинам. Невыполнение этого требования может стать причиной несчастного случая с тяжкими последствиями.

⚠ **Предупреждение** Редактирование в режиме онлайн можно производить лишь в том случае, если увеличение времени цикла не приведет к нежелательному воздействию на систему. В противном случае входные сигналы могут оказаться нечитаемыми.

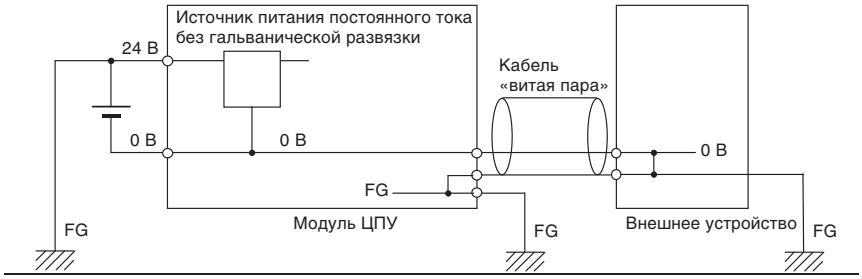
⚠ **Предупреждение** Прежде чем передавать в другой узел или изменять содержимое программы или области ввода/вывода, убедитесь в безопасности данной операции. Невыполнение этого требования может стать причиной несчастного случая.

⚠ **Предупреждение** При затяжке винтов клеммной колодки источника питания переменного тока соблюдайте момент затяжки, указанный в данном руководстве. Ослабление затяжки винтов может привести к пожару или неправильной работе устройства.

⚠ **Предупреждение** Не дотрагивайтесь до токоведущих частей и элементов, клемм входных и выходных цепей, а также прилегающих к ним участков при включенном напряжении питания или сразу после его отключения. Прикосновение к горячей поверхности может вызвать ожог.

⚠ **Предупреждение** Будьте внимательны и соблюдайте полярность при подключении цепей источника питания постоянного тока. Неправильное подключение может привести к нарушению работы системы.

⚠ **Предупреждение** При подключении персонального компьютера или другого внешнего устройства к ПЛК либо заземляйте внешний источник питания по цепи 0 В, либо не заземляйте внешний источник питания вообще. В противном случае клеммы внешнего источника питания могут быть замкнуты накоротко через цепи внешнего устройства. Ни в коем случае не заземляйте цепь 24 В внешнего источника питания, как показано на следующей схеме.



⚠ **Предупреждение** Если в новой (или измененной) программе используется команда IOWR, перед началом работы удостоверьтесь, что ПЛК будет правильно работать с новой программой и данными. Любые отклонения и нарушения могут привести к остановке работы ПЛК, что, в свою очередь, может привести к непредсказуемой работе производственного оборудования.

⚠ **Предупреждение** Модули ЦПУ серии CP автоматически создают резервную копию программы пользователя и значений параметров во флэш-памяти при записи этих данных в модуль ЦПУ. Копия содержимого памяти ввода/вывода (включая область DM, текущие значения счетчиков, флаги завершения и область регистров хранения) во флэш-память не записывается. Содержимое области DM, текущих значений счетчиков, флагов завершения и области регистров хранения может сохраняться при пропадании питания за счет подпитки батареей. Однако при наличии ошибки батареи содержимое этих областей после прерывания питания может быть недостоверным. Если содержимое области DM, текущих значений счетчиков, флагов завершения и области регистров хранения используется для управления внешними выходами, позаботьтесь об установлении допустимых состояний на этих выходах при включенном флаге ошибки батареи (A402.04).

4 Меры предосторожности в связи с условиями эксплуатации

⚠ **Предупреждение** Не эксплуатируйте систему управления в следующих местах:

- В местах воздействия прямого солнечного света.
- В местах, где температура окружающей среды или влажность не соответствуют требованиям.
- В местах возможного образования конденсата вследствие резких перепадов температуры.
- В местах возможного присутствия коррозионных или воспламеняющихся газов.
- В местах скопления пыли (особенно, металлического порошка) или солей.
- В местах возможного воздействия воды, масла или химических реактивов.
- В местах возможного воздействия ударов или вибрации.

⚠ **Предупреждение** При монтаже системы в перечисленных ниже местах следует предусматривать надлежащие и достаточные меры защиты:

- В местах воздействия статического электричества или любых других помех.
- В местах воздействия интенсивных электромагнитных полей.
- В местах возможного воздействия радиоактивных излучений.
- Вблизи источников электропитания.

⚠ **Предупреждение** Условия эксплуатации ПЛК могут оказать значительное влияние на срок службы и надежность системы. Не соответствующие требованиям условия эксплуатации могут привести к выходу из строя, сбоям или другим непредвиденным проблемам в системе ПЛК. Необходимо следить за тем, чтобы условия эксплуатации соблюдались при монтаже системы, а также поддерживались в пределах установленных норм в течение всего срока службы системы.

5 Меры предосторожности при эксплуатации

При эксплуатации ПЛК должны соблюдаться меры предосторожности, указанные ниже.

⚠ **ВНИМАНИЕ** Всегда соблюдайте данные меры предосторожности. Несоблюдение данных мер предосторожности может привести к тяжелой травме, возможно, со смертельным исходом.

- При установке модулей всегда выполняйте заземление через цепь с сопротивлением не более 100 Ом. Невыполнение этого требования может привести к поражению электрическим током.
- Обязательно выключите питание ПЛК, перед тем как выполнить одно из следующих действий. Невыполнение этого требования может привести к возникновению сбоев или поражению электрическим током.
  - Монтаж или демонтаж модулей расширения или любых других модулей.
  - Подсоединение или отсоединение карты памяти или дополнительной платы.
  - Настройка DIP- или поворотных переключателей.
  - Подсоединение кабелей или выполнение проводных соединений.
  - Подсоединение или отсоединение разъемов.

⚠ **Предупреждение** Несоблюдение следующих мер предосторожности может привести к возникновению сбоев при работе ПЛК или системы, а также к выходу ПЛК или его модулей из строя. Всегда соблюдайте данные меры предосторожности.

- Предусматривайте внешние автоматические выключатели, а также другие устройства для защиты от коротких замыканий во внешней проводке. Недостаточные меры защиты от коротких замыканий могут быть причиной пожара.
- Перед установкой модуля тщательно проверьте клеммные колодки и разъемы.
- Затяните все клеммные винты и винты разъемов кабелей с соблюдением моментов затяжки, указанных в соответствующих руководствах. Несоблюдение этого требования может привести к сбоям в работе.
- Выполняйте все проводные соединения в точном соответствии с указаниями в настоящем руководстве.
- Используйте для модулей только те напряжения питания, которые указаны в руководствах по эксплуатации. Недопустимое напряжение может быть причиной сбоя или пожара.
- Примите надлежащие меры по обеспечению подачи питания требуемой мощности, с требуемым номинальным напряжением и частотой, особенно, при работе с нестабильными источниками

питания. Не соответствующий требованиям источник питания может приводить к сбоям в работе оборудования.

- При подключении проводов не снимайте защитную этикетку, прикрепленную к модулю. Удаление этикетки может привести к возникновению сбоев во время работы.
- Завершив подключение электрических цепей, удалите этикетку, чтобы модуль не перегревался при работе. Перегрев модуля может привести к возникновению сбоев во время работы.
- Применяйте для подключения цепей обжимные наконечники. Не вставляйте в клеммы скрученные многожильные провода без обжимных наконечников. Подключение проводов без обжимных наконечников может быть причиной пожара.
- Не подавайте на клеммы входов напряжения, выходящие за номинальный диапазон. Чрезмерно высокое напряжение может стать причиной пожара.
- Подавайте напряжение и подключайте нагрузку к клеммам выходов в пределах нагрузочной способности выходов. Недопустимо высокие напряжения или токи нагрузки могут стать причиной пожара.
- Следите за тем, чтобы клеммные колодки, разъемы, дополнительные платы и другие части, снабженные механизмами фиксации, были надежно зафиксированы на своих местах. Ненадежная фиксация может привести к сбоям во время работы.
- Перед проведением испытания на электрическую прочность изоляции отсоедините клемму функционального заземления. Невыполнение этого требования может быть причиной пожара.
- Не допускайте ошибок при подключении цепей. Перед включением питания дважды проверьте все проводные соединения и положения переключателей. Ошибки в подключении электрических цепей могут стать причиной пожара.
- Перед началом работы убедитесь в правильности настройки DIP-переключателей и содержимого области DM.
- Проверьте правильность выполнения программы пользователя перед тем, как запустить ее на модуле в рабочем состоянии. Невыполнение этого требования может привести к непредсказуемой работе оборудования.
- После замены модуля ЦПУ и/или специальных модулей ввода/вывода возобновляйте работу только после загрузки в новый модуль ЦПУ содержимого областей DM, HR и CNT, необходимого для возобновления работы. Невыполнение этого требования может привести к непредсказуемой работе оборудования.
- Убедитесь в том, что ни одна из следующих операций не приведет к нежелательным последствиям для системы. Невыполнение этого требования может привести к непредсказуемой работе оборудования.
  - Изменение режима работы ПЛК (включая настройку режима работы при запуске).
  - Принудительная установка/сброс любого бита в памяти.
  - Изменение текущего значения любого слова или любого установленного значения в памяти.
- Не тяните за кабели и не перегибайте их сверх допустимого радиуса изгиба. Любое из этих действий может привести к обрыву кабеля.
- Не размещайте поверх кабелей какие-либо предметы. Это может привести к повреждению кабелей.
- При замене узлов используйте только узлы с подходящими номинальными параметрами. Несоблюдение этого требования может привести к возникновению сбоев или пожару.

- Перед прикосновением к модулю обязательно дотроньтесь до заземленного металлического предмета, чтобы снять электростатический заряд. Несоблюдение этого требования может привести к возникновению сбоев или выходу оборудования из строя.
- Не дотрагивайтесь до соединительного кабеля модуля расширения входов/выходов при поданном напряжении питания во избежание повреждения устройства статическим электричеством.
- Не выключайте напряжение питания модуля во время передачи данных.
- При транспортировке или хранении изделий печатные платы и модули должны быть обернуты или помещены в антистатический пакет из электропроводящего материала для защиты элементов (БИС и ИС) от статического электричества. Кроме того, должна соблюдаться надлежащая температура транспортировки и хранения.
- Не дотрагивайтесь до установленных элементов или тыльной поверхности печатных плат, чтобы не порезаться об острые края и выступы (например, выводы электрических элементов).
- Дважды проверяйте номера выводов при сборке узлов и соединении разъемов.
- Правильно подключайте электрические цепи, соблюдая установленный порядок действий.
- Не подсоединяйте вывод 6 (+5 В) порта RS-232C дополнительной платы, установленной в модуль ЦПУ, к какому-либо внешнему устройству, за исключением конвертера интерфейсов NT-AL001 или CJ1W-CIF11. Внешнее устройство или модуль ЦПУ могут быть повреждены.
- Для подключения модулей используйте специальные соединительные кабели, указанные в данном руководстве. Применение обычных, серийно выпускаемых компьютерных кабелей RS-232C может привести к повреждению внешних устройств или модуля ЦПУ.
- Прежде чем начинать работу, убедитесь в том, что таблицы логических связей и параметры настроены надлежащим образом. Невыполнение этого требования может привести к непредсказуемой работе оборудования. Даже если таблицы и параметры настроены правильно, заранее убедитесь в том, что запуск или остановка логических связей не повлияют отрицательно на работу управляемой системы.
- Прежде чем загружать таблицу маршрутизации в модуль ЦПУ, убедитесь в том, что перезапуск модулей шины ЦПУ, осуществляемый автоматически для вступления новых таблиц в силу, не приведет к нежелательным последствиям.
- Резервная копия программы пользователя и содержимого области параметров модуля ЦПУ серии CP сохраняется во встроенную флэш-память. В процессе выполнения резервного копирования на передней панели модуля ЦПУ светится индикатор «BKUP». Не выключайте напряжение питания модуля ЦПУ, пока светится индикатор «BKUP». Если питание будет отключено, резервная копия данных создана не будет.
- Не выключайте питание ПЛК, пока производится запись данных на карту памяти. Это может привести к повреждению данных на карте памяти. В процессе записи данных на карту памяти светится индикатор «BKUP». В случае модуля ЦПУ CP1H процесс записи также сопровождается свечением 7-сегментного индикатора. Прежде чем выключать напряжение питание ПЛК, дождитесь выключения индикатора «BKUP» и 7-сегментного индикатора.

- Непосредственно перед заменой батареи подайте на модуль ЦПУ питание минимум на 5 минут, произведите замену батареи в течение максимум 5 минут с момента выключения питания. При несоблюдении данного порядка действий содержимое памяти может быть повреждено.
- Для подсоединения модулей ввода-вывода, специальных модулей ввода-вывода и модулей шины ЦПУ используйте только провода следующего сечения: AWG22...AWG18 (0,32...0,82 мм^2).
- Согласно стандартам UL замену батарей должен производить опытный технический специалист. Не допускайте замену батарей лицами, не имеющими соответствующей квалификации. Всегда соблюдайте порядок действий по замене батареи, приведенный в данном руководстве.
- Никогда не замыкайте накоротко положительные и отрицательные клеммы батареи, не заряжайте, не разбирайте, не нагревайте и не сжигайте батарею. Защищайте батарею от воздействия сильных ударов и деформирующего сжатия. Несоблюдение любого из этих требований может привести к вытеканию электролита, разгерметизации, нагреву или воспламенению батареи. Утилизируйте любую батарею, упавшую на пол или иным образом подвергшуюся сильному удару. Батареи, подвергшиеся сильному удару, могут потечь в процессе эксплуатации.
- Организуйте работу внешних электрических цепей таким образом, чтобы питание на ПЛК подавалось до подачи питания на управляемую систему. В случае включения ПЛК после подачи питания на управляемую систему в последней могут наблюдаться кратковременные ошибки из-за поступления неправильных сигналов. Это связано с тем, что сразу после включения ПЛК происходит кратковременное включение выходов модулей выходов постоянного тока и других модулей.
- Конечным пользователем должны быть предусмотрены меры защиты для обеспечения безопасности в случае, если выходы модулей выходов останутся во включенном состоянии из-за повреждения реле, транзисторов и других элементов внутренних цепей модулей.
- Если включен бит удержания входов/выходов, выходы ПЛК не выключаются и сохраняют свои прежние состояния при переключении ПЛК из режимов «Выполнение» или «Мониторинг» в режим «Программирование». Позаботьтесь о том, чтобы устройства, подключенные к этим выходам, не создавали опасных условий в такой ситуации. (Если работа останавливается из-за критической ошибки, включая выполнение команды FALS(007), все физические выходы модуля выходов выключаются, а сохраняются только их внутренние состояния.)
- Производите утилизацию изделия и батарей в соответствии с предписаниями местных законодательных органов.
  Утилизация отработанных батарей должна производиться квалифицированным персоналом в соответствии с правилами утилизации промышленных отходов.




「廢電池請回收」

6 Соответствие Директивам ЕС

6-1 Применимые Директивы

- Директивы по ЭМС
- Директива по низковольтному оборудованию

6-2 Общие принципы

Директивы по ЭМС

Изделия OMRON, выполняющие требования Директив ЕС, также удовлетворяют соответствующим стандартам на ЭМС, что облегчает задачу их совместного использования с другими устройствами или применение всей системы в целом. Все выпущенные изделия протестированы на соответствие стандартам ЭМС (см. примечание ниже). В то же время, соответствие изделий стандартам системы, используемой покупателем, должно проверяться самим покупателем.

Относящиеся к ЭМС характеристики изделий OMRON, соответствующих Директивам ЕС, могут изменяться в зависимости от конфигурации, схемы соединений и прочих условий, связанных с оборудованием или панелью управления, в которые установлены изделия OMRON. Поэтому конечный пользователь должен проводить финальное тестирование на соответствие этих изделий и всей системы в целом стандартам ЭМС.

Примечание. Применимый стандарт электромагнитной совместимости (ЭМС): EN61131-2.

Директива по низковольтному оборудованию

Всегда следите за тем, чтобы устройства, работающие с напряжениями 50...1000 В~ и 75...1500 В=, удовлетворяли стандартам безопасности, применяемым для ПЛК (EN61131-2).

6-3 Соответствие Директивам ЕС

ПЛК серии CP1H/CP1L соответствуют Директивам ЕС. Для того чтобы машина или устройство, в составе которых используется ПЛК серии CP1H/CP1L, соответствовали Директивам ЕС, при установке ПЛК должны соблюдаться следующие требования.

1,2,3...

1. ПЛК CP1H/CP1L должен быть установлен внутри шкафа управления.
2. Для источников питания постоянного тока, подключаемых к модулям ввода/вывода и модулям ЦПУ, для которых требуется напряжение питания постоянного тока, должна предусматриваться усиленная или двойная изоляция. Для питания модулей, требующих напряжения питания постоянного тока, должен использоваться источник питания постоянного тока, способный удерживать напряжение на своем выходе в течение минимум 10 мс.
3. ПЛК CP1H/CP1L, удовлетворяющие Директивам ЕС, также соответствуют стандарту EN61131-2. Характеристики излучений (нормативы 10 м) могут изменяться в зависимости от конфигурации используемого шкафа управления, других устройств, установленных в шкафу управления, схем подключения и других условий. Поэтому на соответствие Директивам ЕС должна проверяться вся машина или все оборудование целиком.

6-4 Способы подавления помех на релейных выходах

ПЛК CP1H/CP1L соответствуют стандартам на общие излучения (EN61131-2), оговоренным в Директивах по ЭМС. Однако уровень помех, генерируемых при переключении релейного выхода, может не удовлетворять этим стандартам. В этом случае в нагрузке должен быть предусмотрен фильтр помех либо другие меры защиты, помимо предусмотренных в ПЛК.

Меры защиты, предпринимаемые с целью удовлетворения стандартам, зависят от нагрузки, схемы подключения, конфигурации системы и т. п. Ниже приведены примеры способов подавления генерируемых помех.

Меры защиты

Применение дополнительных мер не требуется, если коммутация нагрузки в системе, в состав которой входит ПЛК, производится не чаще, чем 5 раз в минуту.

Дополнительные меры защиты требуются лишь в том случае, когда частота коммутации нагрузки в системе, содержащей ПЛК, превышает 5 раз в минуту.

Примечание. Более подробная информация содержится в стандарте EN61131-2.

Примеры способов подавления помех

При коммутации индуктивной нагрузки параллельно нагрузке или контактам реле следует включить демпфирующую цепочку, диоды и т. п. Схемы подключения показаны ниже.

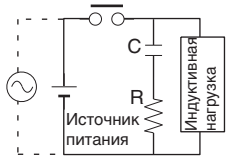
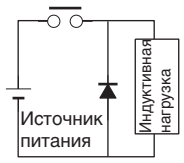
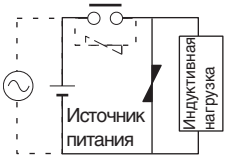
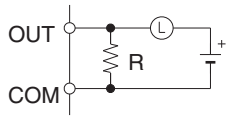
Схема	Ток		Описание	Требования к элементу
	В~	В=		
RC-цепочка 	Да	Да	Если нагрузкой является реле или электромагнит, ток в нагрузке пропадает не сразу после размыкания цепи питания, а с некоторой задержкой. При напряжении питания 24 или 48 В демпфирующую цепочку следует подключить параллельно нагрузке. При напряжении питания от 100 до 200 В цепочку следует подключить между контактами.	Емкость конденсатора должна составлять 1...0,5 мкФ на 1 А коммутируемого тока, а сопротивление резистора должно быть 0,5...1 Ом на 1 В напряжения на контактах. Эти значения, однако, могут меняться в зависимости от нагрузки и характеристик реле. Их можно подобрать экспериментально, принимая во внимание, что емкость влияет на гашение искрового разряда в момент размыкания контактов, а сопротивление — на ограничение тока нагрузки в момент замыкания контактов. Электрическая прочность конденсатора должна составлять 200...300 В. В случае переменного тока электролитические конденсаторы использовать не следует.

Схема	Ток		Описание	Требования к элементу
	В~	В=		
Диод 	Нет	Да	Подключенный параллельно нагрузке диод преобразует накопленную катушкой энергию в ток, который, протекая через катушку, затухает с выделением тепла вследствие омического сопротивления индуктивной нагрузки. По сравнению с искрогасящей RC-цепочкой время затухания тока в нагрузке после размыкания цепи питания в данном случае больше.	Электрическая прочность диода в обратном направлении должна, по меньшей мере, в 10 раз превышать рабочее напряжение цепи. Максимальный прямой ток диода должен быть равен или должен превышать ток нагрузки. В случае шунтирования электронных схем с низкими напряжениями достаточно, чтобы электрическая прочность диода в обратном направлении превышала рабочее напряжение хотя бы в 2–3 раза.
Варистор 	Да	Да	Подключение варистора предотвращает появление высокого напряжения между контактами за счет способности варистора ограничивать и стабилизировать напряжение на своих контактах. Ток в нагрузке пропадает с некоторой задержкой после размыкания цепи питания. При напряжении питания 24 В или 48 В варистор следует подключать параллельно нагрузке. При напряжении питания 100...200 В варистор подключают параллельно контактам.	---

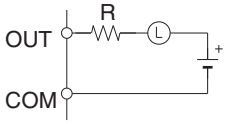
При коммутации нагрузок с высоким пусковым током (например, ламп накаливания) для подавления первоначального броска тока необходимо применять следующие схемы.

Способ 1



Отвод тока (примерно 1/3 от номинального тока лампы) на шунтирующий резистор

Способ 2



Включение ограничительного резистора

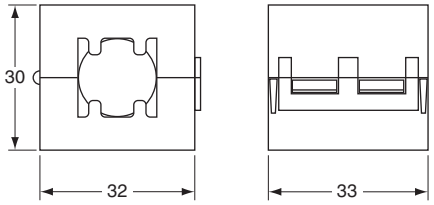
6-5 Условия соответствия Директивам по ЭМС при использовании модулей расширения входов/выходов CP1/CP/CPM1A с релейными выходами

Ниже приведены условия проведения испытаний на помехоустойчивость по стандарту EN61131-2 при использовании модулей CP1W-40EDR, CPM1A-40EDR, CP1W-16ER или CPM1A-16ER с соединительным кабелем ввода/вывода CP1W-CN811.

Рекомендуемое ферритовое кольцо

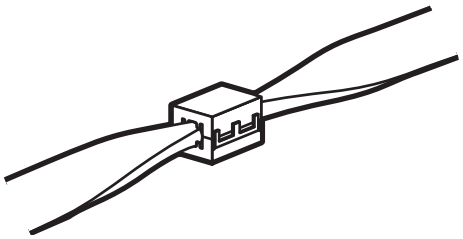
Ферритовое кольцо (фильтр линии передачи данных): 0443-164151, производство Nisshin Electric

Минимальное полное сопротивление: 90 Ом при 25 МГц, 160 Ом при 100 МГц

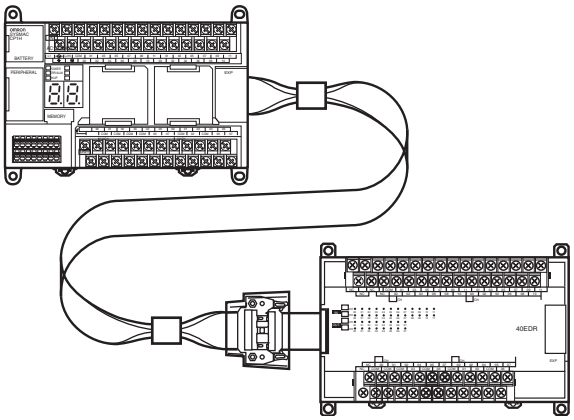


Рекомендуемый способ подключения

1,2,3... 1. Способ подключения кабеля



2. Способ подключения
Установите ферритовое кольцо с обеих сторон соединительного кабеля ввода/вывода CP1W-CN811, как показано на рисунке ниже.



РАЗДЕЛ 1
Основные принципы программирования

В данном разделе описаны основные понятия и принципы программирования контроллеров серии CP1H.

1-1 Основные принципы программирования . . . 2
1-1-1 Программы и задачи . . . 2
1-1-2 Основные сведения о командах программирования . . . 4
1-1-3 Местоположение команд и условия выполнения . . . 7
1-1-4 Обращение к областям памяти ввода/вывода . . . 7
1-1-5 Указание операндов . . . 9
1-1-6 Форматы данных . . . 14
1-1-7 Варианты выполнения команд . . . 18
1-1-8 Условия выполнения . . . 18
1-1-9 Время обновления входов и выходов по отношению к времени выполнения команд 20
1-1-10 Момент обновления входов/выходов . . . 22
1-1-11 Объем программы . . . 24
1-1-12 Основные принципы программирования на языке релейно-контактных схем . . . 25
1-1-13 Ввод мнемонических обозначений . . . 29
1-1-14 Примеры программ . . . 31
1-2 Меры предосторожности . . . 36
1-2-1 Флаги условий . . . 36
1-2-2 Специальные разделы программ . . . 41
1-3 Проверка программ . . . 45
1-3-1 CX-Programmer . . . 45
1-3-2 Проверка программы с помощью CX-Programmer . . . 45
1-3-3 Проверка выполнения программы . . . 48
1-3-4 Проверка на наличие критических ошибок . . . 50
1-4 Функциональные блоки . . . 51
1-4-1 Общие сведения и основные свойства . . . 51
1-4-2 Технические характеристики функциональных блоков . . . 52
1-4-3 Выходные файлы программы CX-Programmer . . . 53

Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

1-1 Основные принципы программирования

1-1-1 Программы и задачи

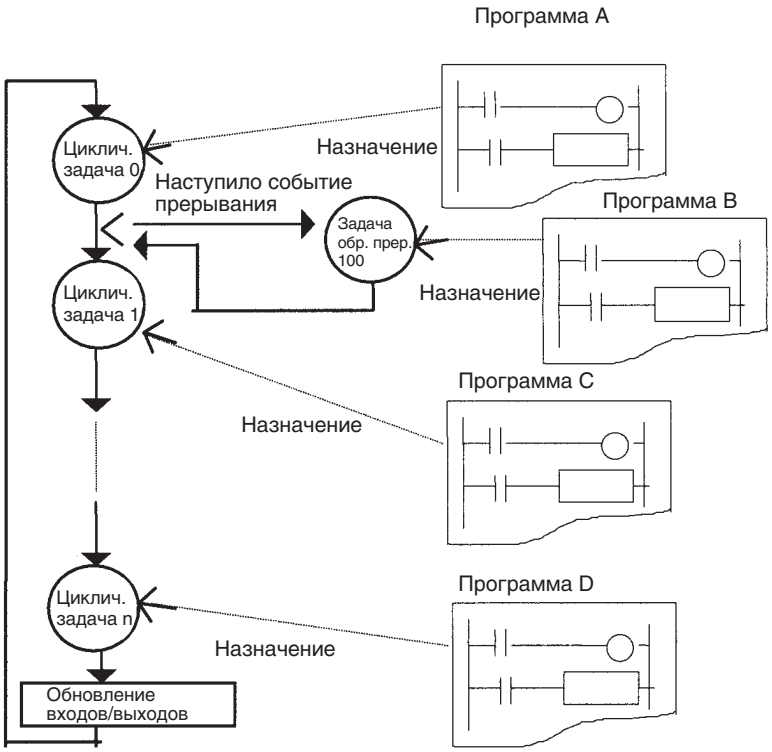
Различают два основных типа задач.

1,2,3...

1. Задачи, которые выполняются последовательно, одна за другой, называются «циклическими задачами».
2. Задачи, которые выполняются по прерыванию, называются «задачами обработки прерываний».

Примечание. Задачи, выполняемые по прерыванию, могут также выполняться циклически, как и обычные циклические задачи. В этом случае они называются «дополнительными циклическими задачами».

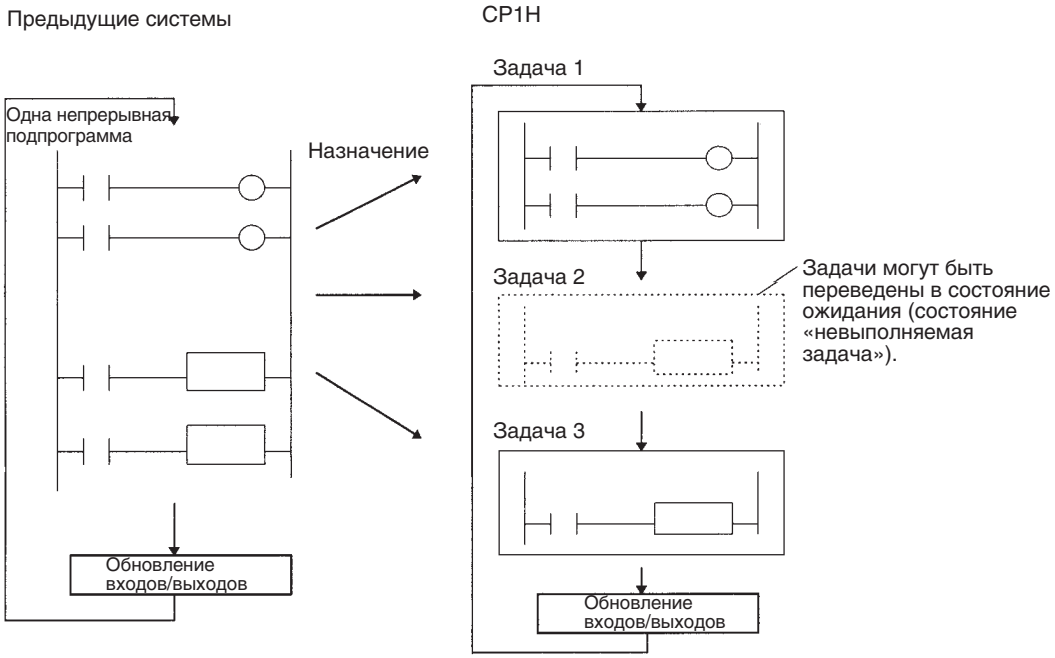
Программы, назначенные циклическим задачам, выполняются последовательно, в порядке возрастания номеров задач, а обновление входов/выходов происходит однократно в каждом цикле после завершения всех задач (если более точно, то после завершения задач, которые являются выполняемыми). Если во время выполнения циклических задач удовлетворяются условия формирования прерывания, выполнение текущей циклической задачи прерывается и выполняется программа, назначенная для задачи обработки прерывания.



В приведенном выше примере программы выполняются в следующем порядке: первая часть программы A, B, остаток программы A, C, а затем D. Предполагается, что во время выполнения программы A было сформировано прерывание, обрабатываемое задачей 100. Когда выполнение программы B завершается, выполняется остаток программы A, начиная с того места, на котором прервалось выполнение.

В более ранних версиях ПЛК OMRON одна непрерывная программа состояла из нескольких, переходящих одна в другую частей. В отличие от этого подхода, программы, назначаемые задачам, являются отдельными программами. Каждая из них завершается командой END, точно так же, как отдельные программы в ПЛК предыдущих поколений.

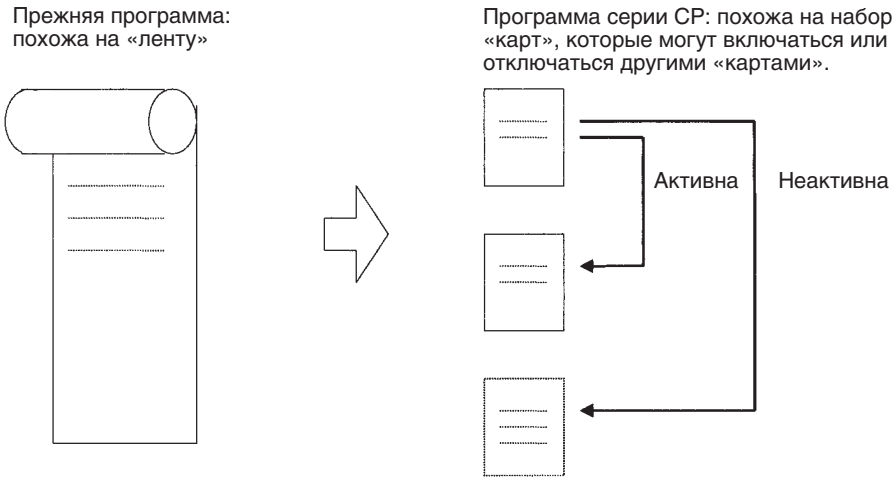
Одним из свойств циклических задач является возможность их включения (статус «выполняемая задача») и отключения (статус «ожидание») с помощью команд управления задачами. Составив программу из нескольких задач, впоследствии можно активизировать и выполнять только определенные программы (задачи), необходимые для текущей модели выпускаемого изделия или технологического процесса (переключать «шаги» программы). Выполняя только требуемые программы, можно существенно повысить производительность системы (за счет сокращения времени цикла).



Задача, уже выполненная в текущем цикле, будет выполняться вновь в последующих циклах, тогда как невыполняемая задача будет сохранять в последующих циклах статус «ожидание» до тех пор, пока она не будет выполнена в какой-либо другой задаче.

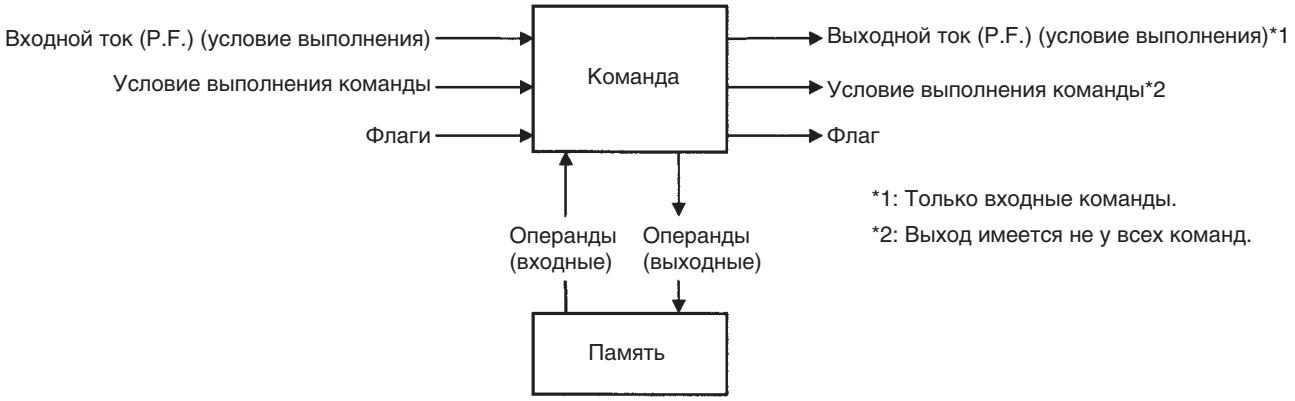
Примечание. В отличие от прежнего подхода к выполнению программы, который можно сравнить со чтением непрерывной «ленты», выполнение программы в виде задач можно сравнить со чтением последовательности отдельных «карт».

- Все «карты» считываются в заданной последовательности, начиная с «карты» с наименьшим номером.
- «Карты» делятся на активные и неактивные. Неактивные «карты» пропускаются («карты» активизируются и деактивизируются с помощью команд управления задачами).
- Активизированная «карта» остается активной и считывается в последующих циклах (последовательностях). Деактивизированная «карта» остается неактивной и пропускается до тех пор, пока она не активизируется повторно другой «картой».



1-1-2 Основные сведения о командах программирования

Программа состоит из команд. В общем случае команда содержит входы и выходы. Принцип построения команды показан на рисунке ниже.

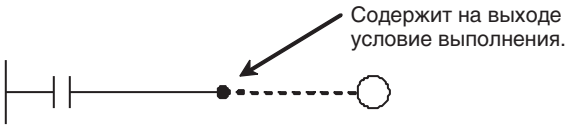


Протекание тока

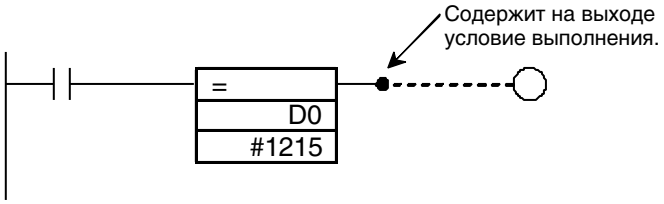
В языке релейно-контактных схем, по аналогии с настоящими электрическими цепями, используется понятие «тока». Протекание тока по цепи является условием выполнения, которое служит для управления командами, когда программы выполняются в обычном режиме. Таким образом, в релейно-контактных схемах (также часто называемых «лестничными диаграммами») протекание тока отражает состояние условия выполнения.

Входные команды

- Входные команды запускают работу логической цепи и содержат на выходе условие выполнения.

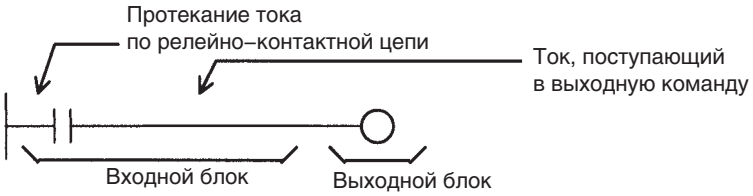


- Промежуточные команды «впускают» ток в качестве условия выполнения и «отдают» ток следующей промежуточной команде или выходной команде.



Выходные команды

Выходные команды выполняют те или иные функции, используя в качестве условия выполнения входной ток.



Условия выполнения команд

Условия выполнения команд — это специальные условия, которые относятся ко всему выполнению команды целиком и которые присутствуют на выходе команд, указанных ниже. При принятии решения о выполнении или невыполнении той или иной команды условия выполнения команды обладают более высоким приоритетом, нежели «протекание тока». В зависимости от условий выполнения команды, команда может стать невыполняемой или может быть выполнена иным образом. Условия выполнения команды сбрасываются (отменяются) в начале каждой задачи. Другими словами, они сбрасываются при смене задачи.

Для установки и отмены определенных условий выполнения команд служат следующие парные команды. Обе команды пары всегда должны находиться в пределах одной и той же задачи.

Условие выполнения команды	Описание	Устанавливающая команда	Отменяющая команда
Блокирован	Блокировка запрещает выполнение части программы. При этом также происходит выключение выходных битов, сброс таймеров и сохранение содержимого счетчиков.	IL(002)	ILC(003)
Выполнение BREAK(514)	Завершает цикл FOR(512) – NEXT(513) во время выполнения (исключает выполнение всех команд до следующей команды NEXT(513)).	BREAK(514)	NEXT(513)
	Выполняет переход от JMP0(515) к JME0(516).	JMP0(515)	JME0(516)
Выполнение программного блока	Выполняет программный блок между BPRG(096) и BEND(801).	BPRG(096)	BEND(801)

Флаги

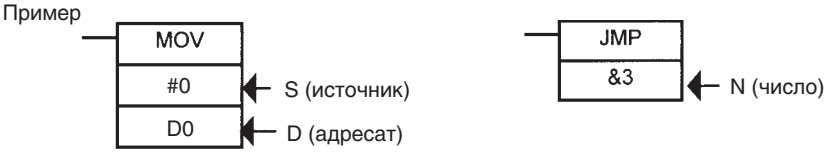
В контексте программирования ПЛК под флагом понимается бит, посредством которого отдельные команды взаимодействуют между собой.

Входные флаги	Выходные флаги
• Флаги распознавания смены состояний Флаги результатов распознавания смены состояний. Состояния этих флагов автоматически поступают на входы выходных команд, различающих направление переключения условия выполнения, а также команд DIFU(013)/DIFD(014). • Флаг переноса (CY) Флаг переноса служит в качестве операнда (не указываемого специально) в командах сдвига данных и в командах сложения/вычитания. • Флаги для специальных команд Сюда входят обучающие флаги для команд FPD(269) и флаги разрешения сетевых коммуникаций.	• Флаги распознавания смены состояний Флаги результатов распознавания смены состояний. Состояния этих флагов автоматически поступают на выходы выходных команд, различающих направление переключения условий выполнения, а также команд UP(521)/DOWN(522). • Флаги условий В состав флагов условий входят флаги «Всегда ВКЛ» и «Всегда ВЫКЛ», а также флаги, состояние которых обновляется в результате выполнения команды. В программе пользователя для обозначения этих флагов вместо адресов можно использовать символы, например: ER, CY, >, =, A1, A0. • Флаги для специальных команд К ним относятся флаги завершения выполнения MSG (046).

Операнды

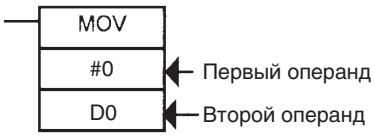
С помощью операндов указываются параметры команды (ячейки таблиц в РКС-символах). В качестве операнда может быть указано содержимое области памяти ввода/вывода или константа. Операнды подразделяются на входные операнды (также называемые «исходными» или «источниками»), выходные операнды (также называемые «конечными» или «адресатами») и численные значения.

Пример



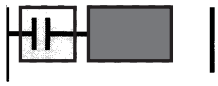
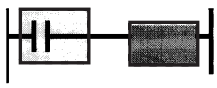
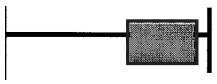
Типы операндов		Символьное обозначение	Описание	
Источник	Указывает адрес, по которому должны быть прочитаны данные, или константу.	S	Входной операнд	Любой входной операнд, кроме управляющих данных.
		C	Управляющие данные	Составные входные данные, значения и состояния битов которых определяют ход выполнения команды.
Выходной операнд (результат)	Указывает адрес, по которому должны быть записаны данные.	D (R)	---	
Число	Указывает определенное число, которое должно использоваться в команде (например, номер для перехода или номер подпрограммы).	N	---	

Примечание. Для операндов также используются понятия «первый операнд», «второй операнд» и так далее, начиная с верхнего операнда команды.



1-1-3 Местоположение команд и условия выполнения

В следующей таблице показаны возможные места расположения команд в программе. Команды разбиты на отдельные группы: 1) команды, для которых требуется условие выполнения; 2) команды, для которых условие выполнения не требуется.

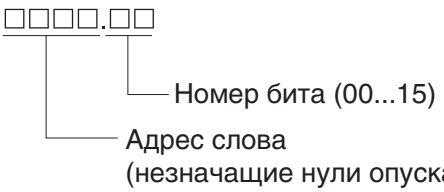
Тип команды		Возможное местоположение	Условие выполнения	Диаграмма	Примеры
Входные команды	Начало логической цепи (команды загрузки)	Подключаются непосредственно к левой шине или размещаются в начале блока команд.	Не требуется		LD, LD TST(350), LD > (и другие символьные команды сравнения)
	Промежуточные команды	Размещаются между начальной командой логической цепи и выходной командой.	Требуется		AND, OR, AND TEST(350), AND > (и другие символьные команды сравнения с AND), UP(521), DOWN(522), NOT(520) и др.
Выходные команды		Подключаются непосредственно к правой шине.	Требуется		Большинство команд, включая OUT и MOV(021).
			Не требуется		END(001), JME(005), FOR(512), ILC(003) и др.

Примечание. (1) Имеется еще одна группа команд, предназначенная для выполнения последовательности мнемонических команд, имеющих один общий вход. Это так называемые команды для программирования блоков. Программирование блоков подробно описано в руководстве *Модули ЦПУ CP1H/CP1L — Руководство по программированию*.

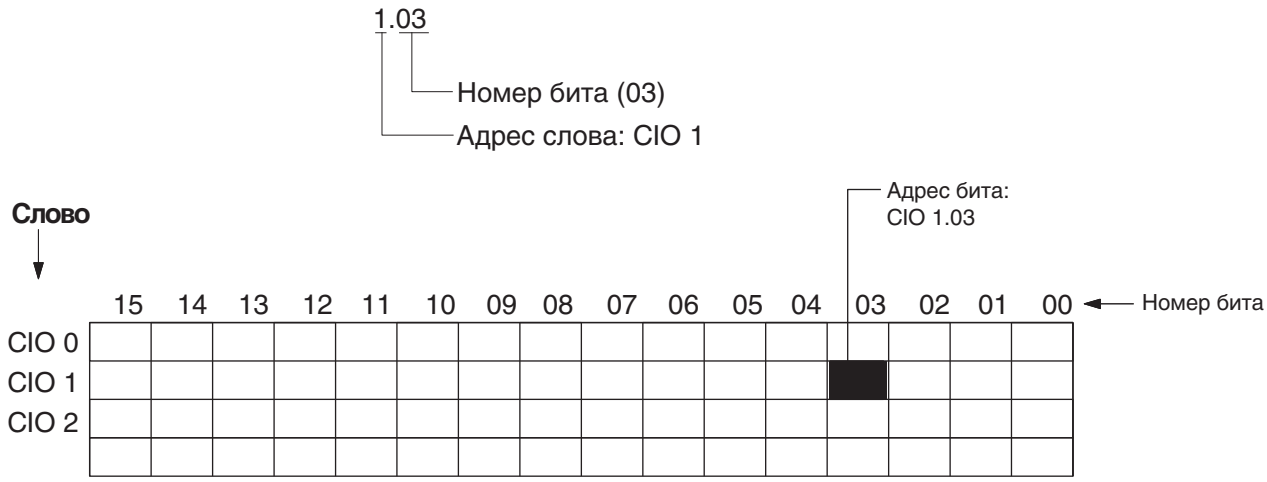
(2) Если команда, для которой требуется условие выполнения, будет подключена непосредственно к левой шине без команды запуска логической цепи спереди, при проверке программы в CX-Programmer будет обнаружена ошибка программы.

1-1-4 Обращение к областям памяти ввода/вывода

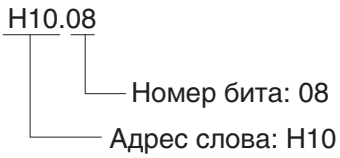
Адреса битов



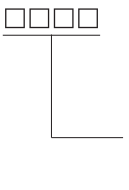
Пример: адрес бита 03 в слове 0001 области CIO имеет следующий вид. В настоящем руководстве такой адрес будет указываться следующим образом: CIO 1.03.



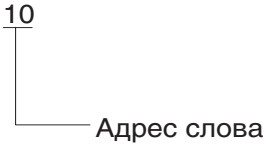
Пример: адрес бита 08 в слове H010 области HR имеет следующий вид.



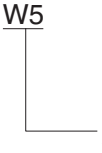
Адреса слов



Пример: адрес слова 0010 (биты 00...15) области CIO имеет следующий вид. В настоящем руководстве такой адрес будет указываться следующим образом: CIO 10.



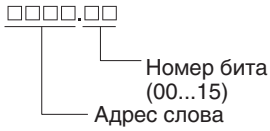
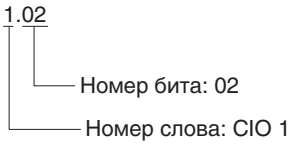
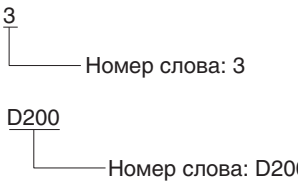
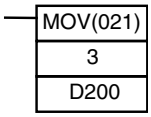
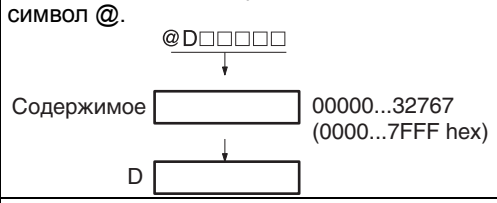
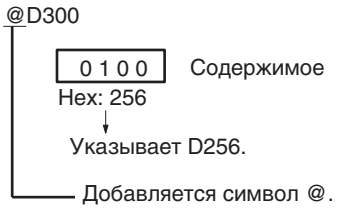
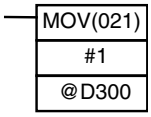
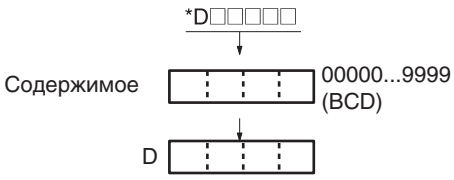
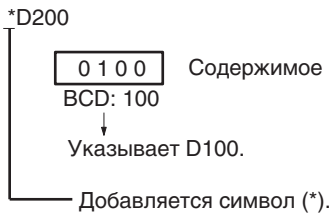
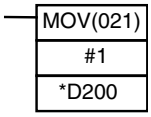
Пример: адрес слова W5 (биты 00...15) рабочей области имеет следующий вид.



Пример: адрес слова D200 (биты 00...15) области DM имеет следующий вид.



1-1-5 Указание операндов

Операнд	Описание	Представление	Примеры применения
Указание адресов битов	Для обращения к биту (биту входа) указываются непосредственно номер слова и номер бита.  **Примечание.** Для обращения к флагам завершения таймеров/счетчиков и для обращения к текущим значениям используются одни и те же адреса. Для флага задачи выделен только один адрес.	 	
Указание адресов слов	Для обращения к 16-битовому слову указывается непосредственно номер слова.  	 	
Косвенное указание адресов области DM в двоичном коде	В качестве адреса указывается величина смещения от начала области. Двоичное значение (00000...32767), содержащееся по указанному адресу, является адресом слова в памяти данных (DM). Чтобы указать косвенный адрес в двоичном формате, впереди адреса следует добавить символ @.  	 	
Косвенное указание адресов области DM в двоично-десятичном коде	В качестве адреса указывается величина смещения от начала области. Двоично-десятичное значение (0000...9999), содержащееся по указанному адресу, является адресом слова в памяти данных (DM). Чтобы указать косвенный адрес в двоично-десятичном формате, спереди адреса следует добавить символ (*).  	 	

Операнд	Описание	Представление	Примеры применения
Непосред-ственное указание регистра	Для обращения к регистру указателей (IR) или регистру данных (DR) указывается непосредственно номер регистра: IR☐ (☐: 0...15) или DR☐ (☐: 0...15).	IR0	MOVR(560) 1.02 IR0 Запись адреса памяти ПЛК слова CIO 10 в регистр IR0.
		IR1	MOVR(560) 10 IR1 Запись адреса памяти ПЛК слова CIO 10 в регистр IR1.

Операнд	Описание		Представ-ление	Примеры применения
Указание косвен-ного адреса с помощью регистра	Косвен-ный адрес (без смещения)	Обращение к биту или слову в памяти ПЛК, адрес которого содержится в IR☐. Для битов и слов в операндах команд должна использоваться следующая форма записи: ,IR☐.	,IR0	,IR0 Загрузка бита, расположенного в памяти ПЛК по адресу, который содержится в IR0.
			,IR1	MOV(021) #1 ,IR1 Запись значения #0001 в слово, расположенное в памяти ПЛК по адресу, который содержится в IR1.
	Фиксиро-ванное смещение	Обращение к биту или слову в памяти ПЛК, адрес которого определяется как: IR☐ + или - указанная константа. Используется следующая форма записи: *+/- константа* ,IR☐. Константа может находиться в диапазоне от -2048 до +2047 (десятичный формат). При выполнении команды значение константы преобразуется в двоичный формат.	+5,IR0	+5,IR0 Загрузка бита, расположенного в памяти ПЛК по адресу: IR0 + 5.
			+31,IR1	MOV(021) #1 +31 ,IR1 Запись значения #0001 в слово, расположенное в памяти ПЛК по адресу: IR1 + 31.

Предварительная версия
Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

Операнд	Описание		Представление	Примеры применения
Указание косвенного адреса с помощью регистра	Смещение в DR	Обращение к биту или слову в памяти ПЛК, адрес которого определяется как: IR☐ + содержимое DR☐. Используется следующая форма записи: DR☐ ,IR☐. Содержимое регистра данных (DR) обрабатывается как двоичное число со знаком. Если двоичное число со знаком является отрицательным, сдвиг от IR☐ будет сделан в обратном направлении.	DR0 ,IR0 DR0 ,IR1	DR0,IR0 ⊣⊢ Загрузка бита, расположенного в памяти ПЛК по адресу: IR0 + значение в DR0. MOV(021) #1 DR0 ,IR1 Запись значения #0001 в слово, расположенное в памяти ПЛК по адресу: IR1 + значение в DR0.
	Автоматический сдвиг вперед	После обращения к значению регистра как к адресу памяти ПЛК содержимое IR☐ автоматически увеличивается на +1 или +2. +1: используется следующая форма записи: ,IR☐+ +2: используется следующая форма записи: ,IR☐ + + **Примечание.** Если во время выполнения команды P_ER или P_AER в модуле ЦПУ CP1L происходит ошибка, автоматическое приращение адреса не производится.	,IR0 ++ ,IR1 +	,IR0 ++ ⊣⊢ Загрузка бита, расположенного в памяти ПЛК по адресу в IR0, и увеличение содержимого IR0 на 2. MOV(021) #1 ,IR1 + Запись значения #0001 в слово, расположенное в памяти ПЛК по адресу в IR1, и увеличение содержимого IR1 на 1.
	Автоматический сдвиг назад	После обращения к значению регистра как к адресу памяти ПЛК содержимое IR☐ автоматически уменьшается на 1 или 2. –1: используется следующая форма записи: ,–IR☐ -2: используется следующая форма записи: ,– –IR☐ **Примечание.** Если во время выполнения команды P_ER или P_AER в модуле ЦПУ CP1L происходит ошибка, автоматическое уменьшение адреса не производится.	,– –IR0 ,–IR1	,—IR ⊣⊢ Загрузка бита, расположенного в памяти ПЛК по адресу в IR0, и уменьшение содержимого IR0 на 2. MOV(021) #1 ,–IR1 Запись значения #0001 в слово, расположенное в памяти ПЛК по адресу в IR1, и уменьшение содержимого IR1 на 1.

Данные	Операнд	Формат данных	Символ	Диапазон	Пример применения
16-битовая константа	Любые двоичные значения или ограниченный диапазон двоичных значений	Двоичное без знака	#	#0000...#FFFF	MOV(021) #5A D100
		Десятичное со знаком	±	-32768...+32767	+(400) D200 −128 D300
		Десятичное без знака	&	&0...&65535	CMP(020) D400 &999
	Любые двоично-десятичные значения или ограниченный диапазон двоично-десятичных значений	BCD	#	#0000...#9999	−B(414) D500 #2000 D600
32-битовая константа	Любые двоичные значения или ограниченный диапазон двоичных значений	Двоичное без знака	#	#00000000...#FFFFFFFF	MOVL(498) #17FFF D100
		Двоичное со знаком	+	-2147483648...+2147483647	+L(401) D200 −65536 D300
		Десятичное без знака	& (см. примеч.)	&0...&429467295	CMPL(060) D400 &99999
	Любые двоично-десятичные значения или ограниченный диапазон двоично-десятичных значений	BCD	#	#00000000...#99999999	−BL(415) D500 #1000000 D600

Данные	Операнд	Формат данных	Символ	Диапазон	Пример применения
Текстовая строка	**Описание**		**Символ**	**Примеры**	
	Данные текстовой строки хранятся в виде ASCII-кодов (1 символ — 1 байт, за исключением спецсимволов), в следующем порядке: от старшего байта к младшему, от младшего слова к старшему. В случае нечетного количества символов младший байт (крайний правый байт) последнего слова содержит значение 00 Hex (код NUL). В случае четного количества символов в крайнем левом и крайнем правом свободных байтах последнего слова +1 содержится значение 0000 hex (2 кода NUL).		---	↓ 'ABCDE' 'A' 'B' 'C' 'D' 'E' NUL = 41 42 43 44 45 00 'ABCD' 'A' 'B' 'C' 'D' NUL NUL = 41 42 43 44 00 00	MOV$(664) D100 D200 D100 41 42 D101 43 44 D102 45 00 ↓ D200 41 42 D201 43 44 D202 45 00
	В состав символов ASCII, которые можно использовать в текстовой строке, входят алфавитно-цифровые символы, японский алфавит катакана и символы (за исключением спецсимволов). Символы показаны в следующей таблице.				

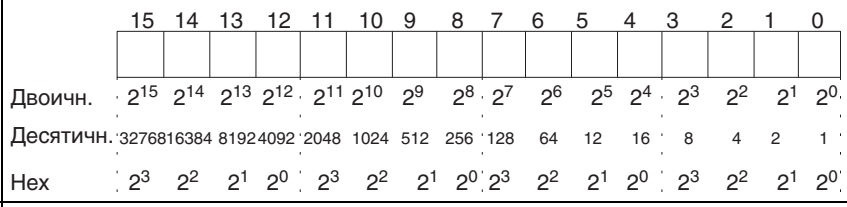
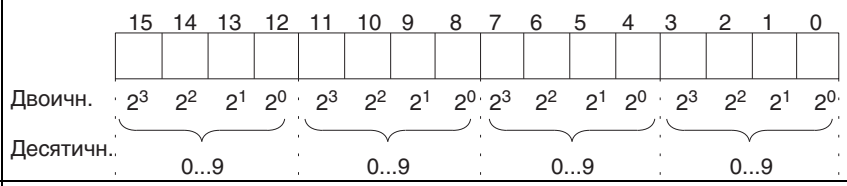
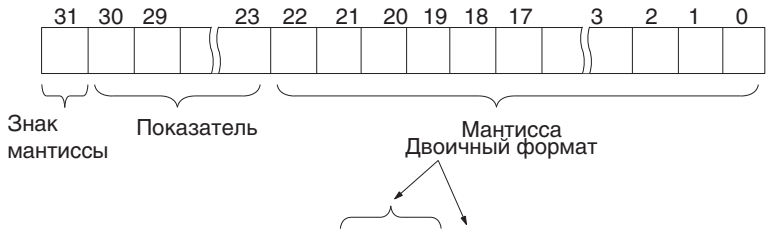
ASCII-символы

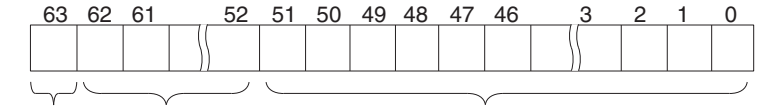
Биты 0...3		Биты 4...7															
Двоичное	Hex	0000 0	0001 1	0010 2	0011 3	0100 4	0101 5	0110 6	0111 7	1000 8	1001 9	1010 A	1011 B	1100 C	1101 D	1110 E	1111 F
0000	0			Пробел	0	@	P	`	p								
0001	1			!	1	A	Q	a	q								
0010	2			"	2	B	R	b	r								
0011	3			#	3	C	S	c	s								
0100	4			$	4	D	T	d	t								
0101	5			%	5	E	U	e	u								
0110	6			&	6	F	V	f	v								
0111	7			'	7	G	W	g	w								
1000	8			(	8	H	X	h	x								
1001	9			)	9	I	Y	i	y								
1010	A			*	:	J	Z	j	z								
1011	B			+	;	K	[	k	{								
1100	C			,	<	L	¥	l	|								
1101	D			-	=	M	]	m	}								
1110	E			.	>	N	^	n	~								
1111	F			/	?	O	_	o									

Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

1-1-6 Форматы данных

В следующей таблице перечислены форматы данных, с которыми могут работать ПЛК серии CP.

Тип данных	Формат данных	Десятичное значение	4-разрядное 16-ричное значение
Двоичное, без знака		0...65535	0000...FFFF
Двоичное, со знаком		0... -32768 0... +32767	Отрицательное: 8000...FFFF Положительное: 0000...7FFF
Двоично-десятичное (BCD)		0...9999	0000...9999
Десятичное с плавающей запятой, одинарной точности	 Значение = $(-1)^{Знак}$ x 1.[Мантисса] x $2^{Показатель}$ Знак (бит 31): 1: отрицательный или 0: положительный Мантисса: 23 бита (биты 00...22) содержат двоичное значение мантиссы, то есть той части, которая находится за десятичной запятой в числе 1.□□□..... . Показатель: 8 битов (биты 23...30) содержат значение показателя. Показатель представляется в двоичном коде как 127 + n в 2n. **Примечание.** Этот формат соответствует стандартам IEEE754, описывающим представление данных в формате с плавающей запятой, одинарной точности. Он предназначен только для команд, которые осуществляют преобразования или вычисления с числами в формате с плавающей запятой. Его можно использовать для ввода или отображения значений на экране редактирования и контроля памяти ввода/вывода (I/O memory Edit and Monitor) в программе CX-Programmer. Пользователю не требуется знать структуру формата, следует лишь учитывать, что значения в этом формате занимают два слова.	---	---

Тип данных	Формат данных	Деся-тичное значение	4-разрядное 16-ричное значение
Десятич-ное с плаваю-щей запятой, двойной точности	 Знак мантиссы Показатель Мантисса Двоичный формат Значение = $(-1)^{Знак}$ x 1.[Мантисса] x $2^{Показатель}$ Знак (бит 63) 1: отрицательный или 0: положительный Мантисса 52 бита (биты 00...51) содержат двоичное значение мантиссы, то есть той части, которая находится за десятичной запятой в числе 1, □□□..... . Показатель 11 битов (биты 52...62) содержат значение показателя. Показатель представляется в двоичном коде как 1023 + n в 2n. **Примечание.** Этот формат соответствует стандартам IEEE754, описывающим представление данных в формате с плавающей запятой, двойной точности. Он предназначен только для команд, которые осуществляют преобразования или вычисления с числами в формате с плавающей запятой. Его можно использовать для ввода или отображения значений на экране редактирования и контроля памяти ввода/вывода (I/O memory Edit and Monitor) в программе CX-Programmer. Пользователю не требуется знать структуру формата, следует лишь учитывать, что значения в этом формате занимают четыре слова.	---	---

Двоичные данные со знаком

У данных в двоичном формате со знаком старший (крайний левый) бит соответствует знаку двоичного 16-битового числа. Само значение представляется в виде 4-разрядного 16-ричного числа.

Положительные значения: значение является положительным или нулевым (0), если старший бит = 0 (ВЫКЛ). В этом случае 4-разрядное 16-ричное число находится в диапазоне 0000...7FFF Hex.

Отрицательные значения: значение является отрицательным, если старший бит = 1 (ВКЛ). В этом случае 4-разрядное 16-ричное число находится в пределах 8000...FFFF Hex. Для представления отрицательного (десятичного) числа в виде абсолютного значения используется дополнение до двух.

Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

Пример: для представления десятичного числа -19 в двоичном формате со знаком значение 0013 Hex (модуль числа -19) отнимается от FFFF Hex, после чего к результату добавляется 0001 Hex, что дает FFED Hex.

	F	F	F	F
	1111	1111	1111	1111
Действительное значение	0	0	1	3
	0000	0000	0001	0011
-)				
	F	F	E	C
	1111	1111	1110	1100
	0	0	0	1
	0000	0000	0000	0001
+)				
Дополнение до двух	F	F	E	D
	1111	1111	1110	1101

Операция дополнения

В общем случае под дополнением некоторого числа до основания x понимается операция, при которой все разряды данного числа вычитаются из значения x-1, после чего к младшему разряду добавляется 1 (пример: дополнение числа 7556 до 10: 9999 - 7556 + 1 = 2444). Представление отрицательных чисел в дополнительном коде позволяет обращаться с отрицательными числами как с положительными, в частности, вместо операции вычитания выполнять сложение и т. п.

Пример: 8954 - 7556 = 1398. При использовании дополнения: 8954 + (дополнение 7556 до 10) = 8954 + 2444 = 11398. Отбросив «лишний» старший разряд, получим результат вычитания 1398.

Дополнение до двух

Дополнение до двух — это дополнение до основания «2». В данном случае все разряды вычитаются из числа 1 (2 - 1 = 1), после чего добавляется число 1.

Пример: дополнение двоичного числа 1101 до двух: 1111 (F hex) - 1101 (D hex) + 1 (1 hex) = 0011 (3 hex). Ниже показано, как это значение представляется в виде 4-разрядного 16-ричного числа.

Дополнение числа a (hex) до двух = FFFF hex - a hex + 0001 hex = b hex. Для представления числа a (hex) как дополнения до двух также можно использовать следующий способ: b hex = 10000 hex - a hex.

Пример: чтобы определить дополнение числа 3039 hex до двух, следует выполнить операцию: 10000 hex - 3039 hex = CFC7 hex.

Аналогичным образом, для определения фактического значения a hex по числу b hex, являющемуся дополнением числа a hex до двух, следует использовать выражение: a hex = 10000 hex - b hex.

Пример: чтобы определить фактическое значение числа CFC7 Hex, являющегося дополнением до двух, следует использовать выражение: 10000 hex - CFC7 hex = 3039 hex.

В ПЛК серии CP предусмотрены две команды: NEG (160) (ДОПОЛНЕНИЕ ДО 2) и NEGL (161) (ДОПОЛНЕНИЕ ДВОЙНОГО СЛОВА ДО 2). Эти команды можно использовать для преобразования некоторого действительного значения в дополнение до двух и для обратного преобразования.

Данные в двоично-десятичном формате со знаком

Двоично-десятичный формат со знаком — это специальный формат, служащий для представления отрицательных значений в двоично-десятичном коде. Хотя этот формат и находит применение, он не имеет строгого определения и зависит от конкретного случая использования. В ПЛК серии CP предусмотрены следующие команды для конвертирования форматов данных: ДВОИЧНОЕ СО ЗНАКОМ В BCD: BINS(470), ДВОЙНОЕ BCD СО ЗНАКОМ В ДВОИЧНОЕ: BISL(472), ДВОИЧНОЕ СО ЗНАКОМ В BCD: BCDS(471) и ДВОЙНОЕ BCD СО ЗНАКОМ В ДВОИЧНОЕ: BDSL(473). Более подробно эти команды будут описаны в соответствующем разделе данного руководства.

Десятичное значение	16-ричное значение	Двоичное значение	Двоично-десятичное значение	
0	0	0000	0000	
1	1	0001	0001	
2	2	0010	0010	
3	3	0011	0011	
4	4	0100	0100	
5	5	0101	0101	
6	6	0110	0110	
7	7	0111	0111	
8	8	1000	1000	
9	9	1001	1001	
10	A	1010	0001	0000
11	B	1011	0001	0001
12	C	1100	0001	0010
13	D	1101	0001	0011
14	E	1110	0001	0100
15	F	1111	0001	0101
16	10	10000	0001	0110

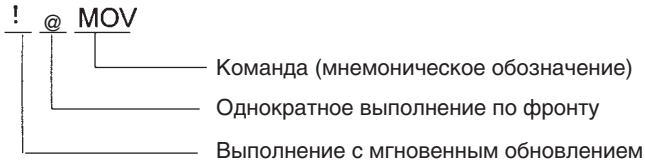
Десятичное значение	Двоичное значение без знака (4-разрядное 16-ричное число)	Двоичное значение со знаком (4-разрядное 16-ричное число)
+65535	FFFF	Не может быть представлено.
+65534	FFFE	
. . .	. . .	
+32769	8001	
+32768	8000	
+32767	7FFF	7FFF
+32766	7FFE	7FFE
. . .	. . .	. . .
+2	0002	0002

Десятичное значение	Двоичное значение без знака (4-разрядное 16-ричное число)	Двоичное значение со знаком (4-разрядное 16-ричное число)
+1	0001	0001
0	0000	0000
-1	Не может быть представлено.	FFFF
-2		FFFE
. . .		. . .
-32767		8001
-32768		8000

1-1-7 Варианты выполнения команд

В следующей таблице перечислены варианты выполнения команд, которые позволяют различать условия выполнения команд и обновлять данные при выполнении команд (мгновенное обновление).

Вариант выполнения		Символ	Описание
Определение направления переключения (различение фронта)	ВКЛ	@	Команда выполняется по включению (т. е. по положительному фронту) условия выполнения.
	ВЫКЛ	%	Команда выполняется по выключению (т. е. по отрицательному фронту) условия выполнения.
Мгновенное обновление		!	Обновление указанных операндами данных в области ввода/вывода или слов специального модуля ввода/вывода происходит при выполнении команды.



1-1-8 Условия выполнения

В ПЛК серии CP предусмотрены следующие типы основных и специальных команд.

- Команды без различения фронтов (многократные команды): выполняются в каждом цикле.
- Команды с различением фронтов (однократные команды): выполняются только один раз.

Команды без различения фронтов

Выходные команды, для которых требуются условия выполнения, выполняются один раз в каждом цикле, пока действует условие их выполнения (0 или 1).



Предварительная версия
Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

Входные команды (начальные команды логических цепей) и промежуточные команды считывают состояния битов, выполняют операции сравнения, проверяют состояния битов или выполняют другие типы операций в каждом цикле. В случае положительного результата (ВКЛ на выходе) открывается «путь для тока» (т. е. включается условие выполнения для следующей команды).

Многократная выходная команда

Пример

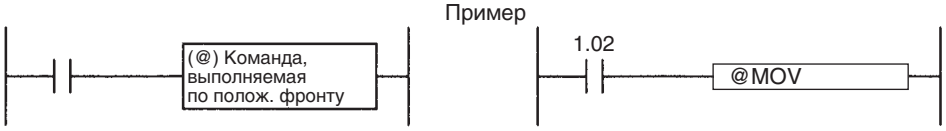


Команды с различением фронтов

Команды, выполняемые по положительному фронту (команды с префиксом @)

- **Выходные команды**: команда выполняется только в том цикле, в котором установилось (0 → 1) условие выполнения, и не выполняется в остальных циклах.

Пример



Команда MOV выполняется однократно, когда CIO 1.02 переходит из «0» в «1».

- **Входные команды (начальные команды логических цепей и промежуточные команды)**: команда считывает состояние бита, выполняет операции сравнения, проверяет состояния битов или выполняет другие операции в каждом цикле и устанавливает на выходе условие выполнения («замыкает токовую цепь»), когда результат операции переходит из состояния «0» в состояние «1». В следующем цикле условие выполнения вновь сбрасывается.

Входная команда, выполняемая по положительному фронту

Пример



Условие выполнения включается только для одного цикла, когда CIO 1.03 переходит из «0» в «1».

- **Входные команды (начальные команды логических цепей и промежуточные команды) с инверсией**: команда считывает состояние бита, выполняет операции сравнения, проверяет состояния битов или выполняет другие операции в каждом цикле и сбрасывает выходное условие выполнения («размыкает токовую цепь»), когда результат операции переходит из состояния «0» в состояние «1». В следующем цикле условие выполнения вновь устанавливается.

Входная команда, выполняемая по положительному фронту

Пример

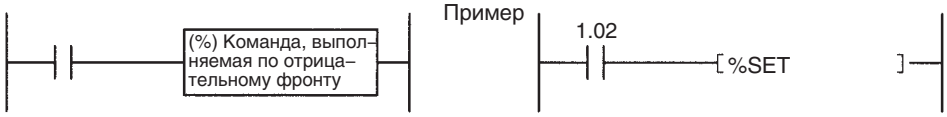


Условие выполнения выключается только для одного цикла, когда CIO 1.03 переходит из «0» в «1».

Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

Команды, выполняемые по отрицательному фронту (команды с префиксом %)

- **Выходные команды**: команда выполняется только в том цикле, в котором сбросилось (1 → 0) условие выполнения, и не выполняется в остальных циклах.



Команда SET выполняется однократно, когда CIO 1.02 переходит из «1» в «0».

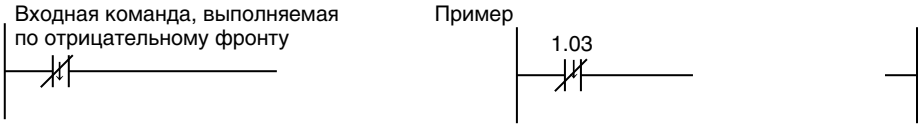
- **Входные команды (начальные команды логических цепей и промежуточные команды)**: команда считывает состояние бита, выполняет операции сравнения, проверяет состояния битов или выполняет другие операции в каждом цикле и устанавливает на выходе условие выполнения («замыкает токовую цепь»), когда результат операции переходит из состояния «1» в состояние «0». В следующем цикле условие выполнения вновь сбрасывается.



Выход включается, когда CIO 1.03 переходит из «1» в «0», и выключается в следующем цикле.

Примечание. В отличие от варианта выполнения по положительному фронту, вариант выполнения по отрицательному фронту можно выбрать только для команд LD, AND, OR, SET и RSET. Для других команд вариант выполнения по отрицательному фронту можно реализовать, используя команду DIFD или DOWN.

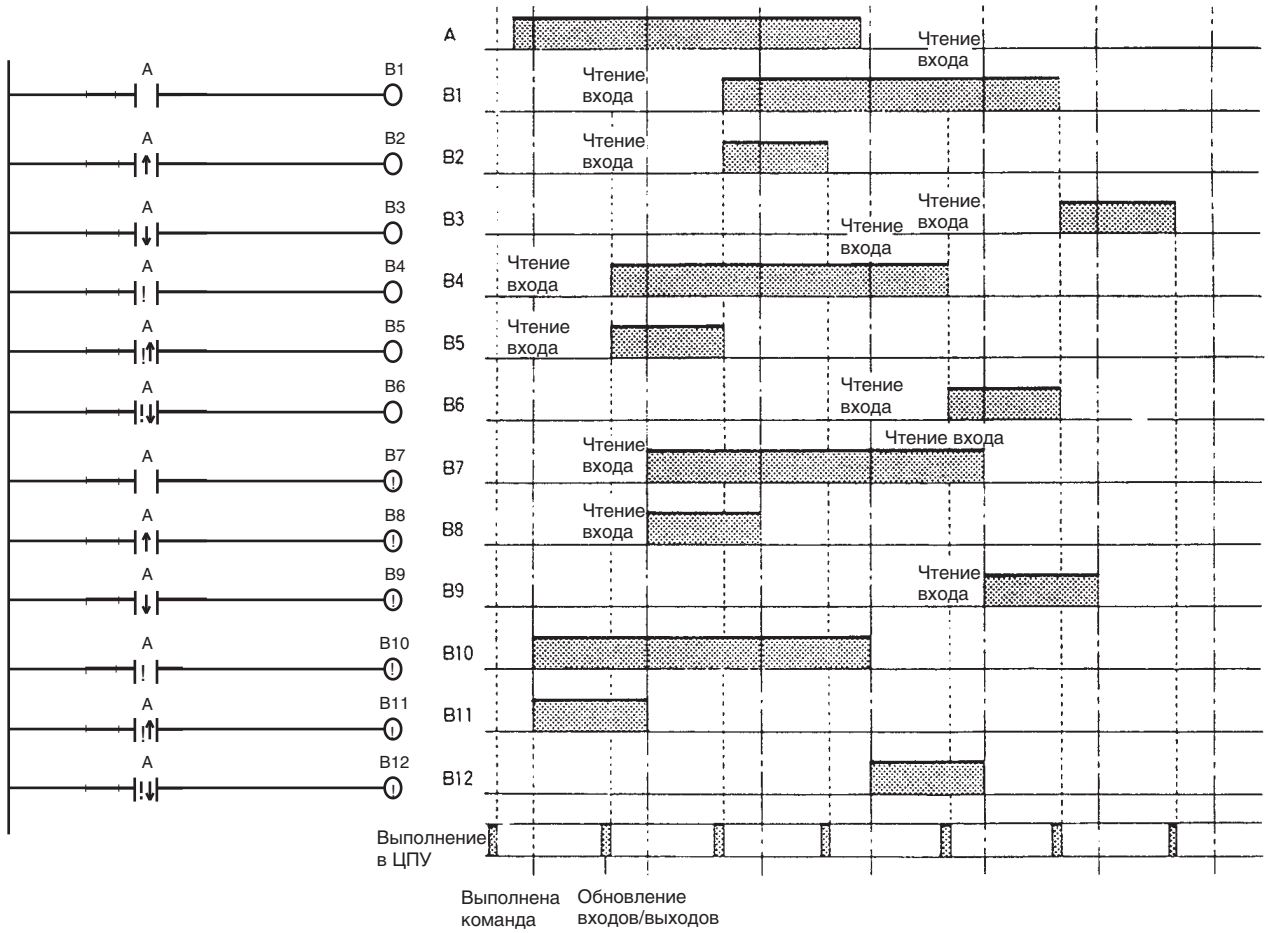
- **Входные команды (начальные команды логических цепей и промежуточные команды) с инверсией**: команда считывает состояние бита, выполняет операции сравнения, проверяет состояния битов или выполняет другие операции в каждом цикле и сбрасывает выходное условие выполнения («размыкает токовую цепь»), когда результат операции переходит из состояния «1» в состояние «0». В следующем цикле условие выполнения вновь устанавливается.



Условие выполнения выключается только для одного цикла, когда CIO 1.03 переходит из «1» в «0».

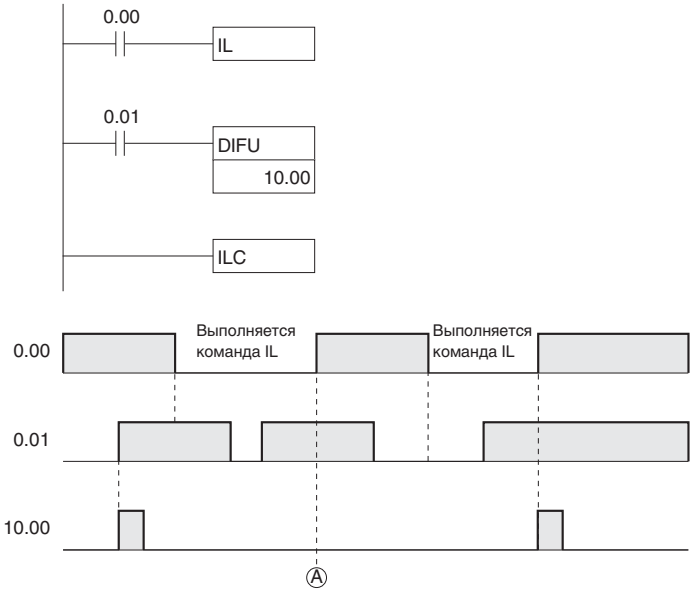
1-1-9 Время обновления входов и выходов по отношению к времени выполнения команд

Представленные ниже временные диаграммы иллюстрируют различия в работе входных (LD) и выходных (OUT) команд с разными вариантами выполнения и обновления.



Команды с различением фронтов

- У команды, выполняемой однократно по переключению состояния, имеется внутренний флаг, хранящий предыдущее состояние (0 или 1). В начале работы флаги предшествующих состояний команд, различающих переключение 0 -> 1 (DIFU и команды с префиксом @), устанавливаются, а флаги предшествующих состояний команд, различающих переключение 1 -> 0 (DIFD и команды с префиксом %), сбрасываются. Это предотвращает непредусмотренное включение выходов команд в начале работы.
- Выход однократной команды, различающей переключение 0 -> 1 (DIFU и команды с префиксом @), включается (переходит в состояние «1») лишь в том случае, когда включено условие выполнения и сброшен флаг предшествующего значения.
- **Применение с командами блокировки (команды IL — ILC)**
  В приведенном ниже примере флаг предшествующего значения для команды, различающей переключение состояний, хранит предшествующее (заблокированное) значение, которое не обновляется, пока действует блокировка. Поэтому выход в момент A не включается, хотя и имеет место переключение состояния.



- **Применение с командами перехода (команды JMP — JME):** как и в случае с блокировкой, флаг предшествующего значения для команды, различающей переключение состояний, не изменяется, если команда обходится, то есть сохраняет предыдущее значение. На выходе команд, различающих переключение 0 -> 1 и 1 -> 0, условие выполнения устанавливается, только если на входе появляется состояние, отличающееся от состояния, сигнализируемого флагом предшествующего состояния.

Примечание. (a) Для однократной команды, различающей переключение 0 -> 1, не следует использовать в качестве входного бита флаг «Всегда ВКЛ» или флаг A200.11 (флаг первого цикла). Команда никогда не будет выполнена.

(b) Для однократной команды, различающей переключение 1 -> 0, не следует использовать в качестве входного бита флаг «Всегда ВЫКЛ». Команда никогда не будет выполнена.

1-1-10 Момент обновления входов/выходов

Внешние входы/выходы могут обновляться в одном из следующих режимов:

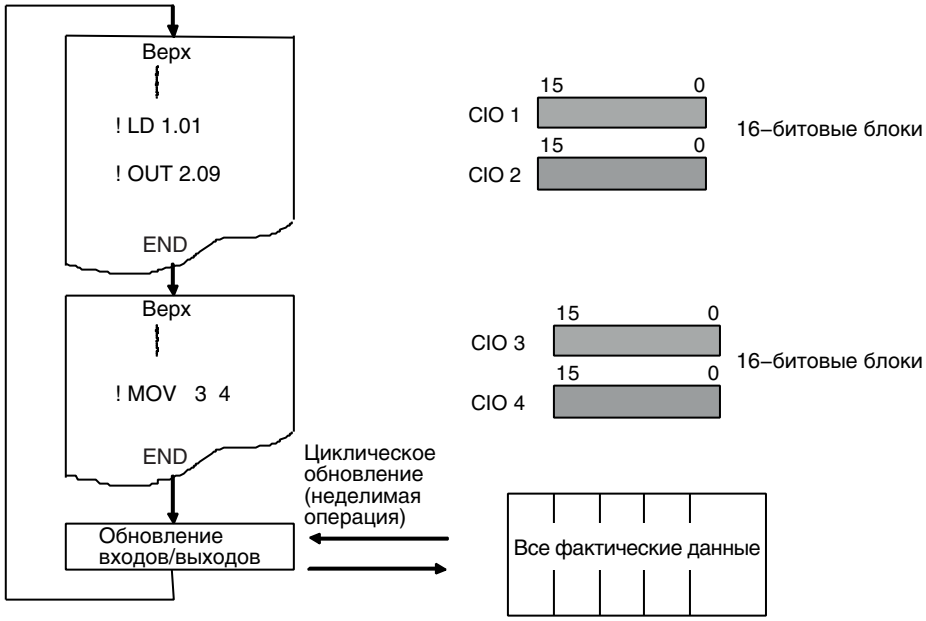
- циклическое обновление;
- мгновенное обновление (команда с префиксом «!», команда IORF).

Циклическое обновление

Каждая программа, принадлежащая готовой к выполнению циклической задаче или вызванной задаче обработки прерывания, выполняется, начиная с начального адреса программы вплоть до команды END (001). После того как все готовые к выполнению циклические задачи или вызванные задачи обработки прерывания выполнены, наступает процедура циклического обновления, которая обновляет одновременно все точки ввода/вывода.

Примечание. Прикладная программа может состоять из нескольких отдельных задач. Обновление входов/выходов происходит лишь после достижения самой последней команды END (001) в программе, принадлежащей циклической задаче с наибольшим номером (среди всех готовых к

выполнению циклических задач), и не происходит при достижении команд END (001) в программах других циклических задач.



Если в других задачах требуется обновить состояния входов/выходов до достижения последней команды END (001), следует выполнить команду IORF(097) для всех требуемых слов либо использовать команды с префиксом мгновенного обновления.

Мгновенное обновление

Выполнение команд с мгновенным обновлением (!)

Ниже описано, как происходит обновление входов/выходов во время выполнения команды, если в качестве операнда указан бит физического (встроенного) входа или выхода модуля ЦПУ.

- Если для команды в качестве операнда указан бит, обновляются все 16 битов слова входов или выходов, содержащего этот бит.
- Если для команды в качестве операнда указано слово, обновляются 16 битов указанного слова входов или выходов.
- Биты входов, используемые во входных командах или во входных операндах, обновляются (то есть принимают текущие состояния соответствующих им физических входов) непосредственно перед выполнением команды.
- Биты выходов, используемые в выходных командах или в выходных операндах, обновляются (то есть передают свои текущие состояния на соответствующие им физические выходы) сразу после выполнения команды.

Перед командой следует разместить восклицательный знак (!) (вариант мгновенного обновления).

Примечание. Префикс мгновенного обновления не действует для физических входов/выходов модулей расширения или модулей расширения входов/выходов серии CP/CPM1A. Для этих модулей можно использовать команду IORF(097).

Применение команды IORF(097) для модулей расширения

Для модулей расширения и модулей расширения входов/выходов серии CP/CPM1A предусмотрена команда IORF(097) (ОБНОВИТЬ ВХОДЫ/ВЫХОДЫ), которая обновляет физические входы/выходы в указанном диапазоне слов. С помощью этой команды в пределах цикла можно обновить либо все, либо определенную группу битов физических входов/выходов.

Примечание. Команду IORF(097) невозможно использовать для битов встроенных входов и выходов самого модуля ЦПУ. Для этих входов и выходов следует использовать команды с префиксом мгновенного обновления.

Команду IORF(097) также можно использовать для обновления слов, отведенных для специальных модулей ввода/вывода серии CJ.

DLNK(226) (модули ЦПУ серии CP1H)

Для обновления слов, отведенных модулям шины ЦПУ серии CJ в областях CIO и DM, а также для обновления других данных, связанных с модулями шины ЦПУ (например, логических связей), можно воспользоваться командой DLNK(226) (ОБНОВИТЬ ВХОДЫ/ВЫХОДЫ МОДУЛЯ ШИНЫ ЦПУ). При выполнении команды DLNK(226) с целью одновременного обновления всех перечисленных ниже данных должен указываться номер модуля шины ЦПУ.

- Слова, выделенные для модуля в области CIO.
- Слова, выделенные для модуля в области DM.
- Особые операции обновления данных (например, для модулей Controller Link при обмене через механизм логических связей или для модулей удаленных входов/выходов шины DeviceNet).

1-1-11 Объем программы

В следующей таблице приведены максимальные размеры программ (то есть суммарный объем всех задач) для различных модулей ЦПУ серии CP. Все размеры исчисляются максимальным количеством шагов. Максимальный размер программы превышать нельзя. При достижении максимально допустимого размера все последующие попытки программирования будут блокироваться.

Каждая команда состоит из шагов количеством от 1 до 7. Точное количество шагов, из которых состоит каждая команда, приведено в разделе *РАЗДЕЛ 4 Время выполнения и размер команд* . (Длина каждой команды увеличивается на один шаг, если используется операнд двойной длины.)

Серия	Тип модуля ЦПУ	Модель	Макс. размер программы
Модули ЦПУ серии CP1H	XA	CP1H-XA40D☐-☐	20K шагов
	X	CP1H-X40D☐-☐	
	Y	CP1H-Y20DT-D	
Модули ЦПУ серии CP1L	M	CP1L-M40D☐-☐	10K шагов
		CP1L-M30D☐-☐	
	L	CP1L-L20D☐-☐	5K шагов
		CP1L-L14D☐-☐	

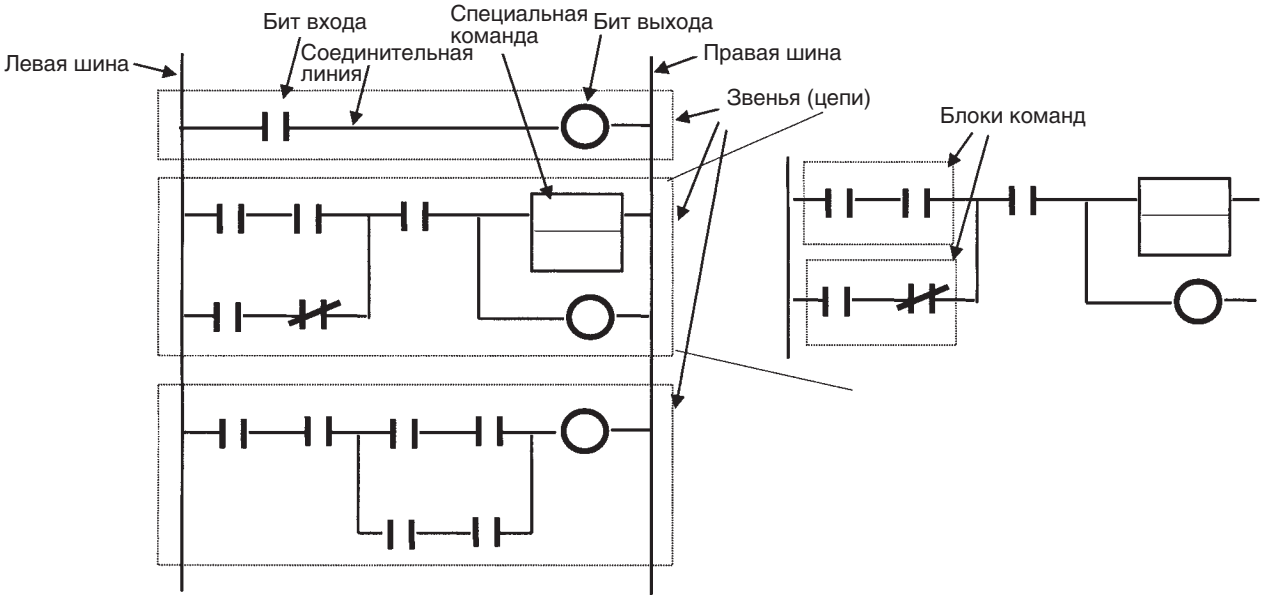
Примечание. Объем памяти для ПЛК серии CP измеряется в шагах, хотя для предыдущих ПЛК производства OMRON, таких как C200HX/HG/HE и CV, объем памяти измерялся в словах. Методика конвертации размеров программ предшествующих ПЛК производства OMRON описана в разделе *РАЗДЕЛ 4 Время выполнения и размер команд*.

1-1-12 Основные принципы программирования на языке релейно-контактных схем

Команды выполняются в том порядке, в котором они перечислены в памяти (т. е. в порядке следования мнемонических обозначений). Необходимо соблюдать основные принципы программирования, а также порядок выполнения.

Общая структура релейно-контактной схемы

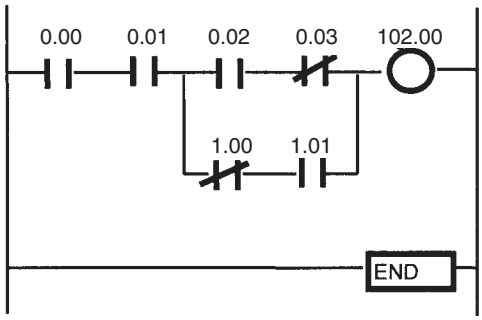
Программа, составленная на языке релейно-контактной логики (другие используемые в настоящем руководстве названия: РКС и «лестничная диаграмма»), состоит из левой и правой объединительных шин (bus bar), соединительных линий, битов входов, битов выходов и специальных команд. Программа может содержать одну или несколько цепей (rung, в настоящем руководстве также может называться «звеном» или «строкой программы»; в прежних руководствах также использовались термины «ветвь» и «логическая ветвь»). Цепью релейно-контактной схемы является модуль, который можно отделить от других модулей горизонтальной линией. В случае представления программы в мнемонической форме цепь объединяет все команды, расположенные между командой LD/LD NOT и выходной командой, которая предшествует следующим командам LD/LD NOT. Цепь программы состоит из блоков команд, которые начинаются с команды LD/LD NOT, выступающей в качестве логического условия начала выполнения.



Мнемоническая форма

В мнемонической форме программа имеет вид последовательности РКС-команд, представленных своими мнемоническими обозначениями. Программа в мнемонической форме содержит адреса, при этом один адрес эквивалентен одной команде.

Пример

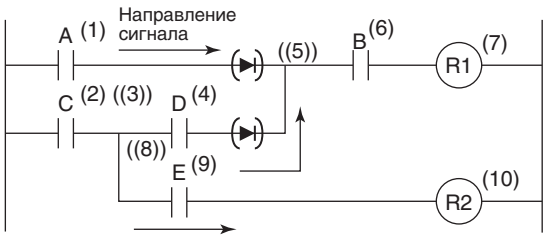


Адрес программы	Команда (мнемоническое обозначение)	Операнд
0	LD	0.00
1	AND	0.01
2	LD	0.02
3	AND NOT	0.03
4	LD NOT	1.00
5	AND	1.01
6	OR LD	
7	AND LD	
8	OUT	102.00
9	END	

Основные принципы построения РКС-программ

1,2,3... 1. Когда «лестничная диаграмма» выполняется программируемым логическим контроллером (ПЛК), сигнал («ток») всегда направлен слева направо. Нельзя использовать программы, в которых «ток» направлен справа налево. Таким образом, распространение сигнала в РКС-программах отличается от протекания тока в цепях, реализованных с помощью настоящих реле. Например, если схема «a» реализована в виде программы ПЛК, ток протекает так, как если бы в цепь были вставлены диоды (показанные в скобках), и катушку R2 невозможно запитать через контакт D. Фактический порядок выполнения показан справа с помощью мнемонических обозначений. Чтобы программа работала без этих воображаемых диодов, схему необходимо реорганизовать. Аналогичным образом, протекание тока, предусмотренное в схеме «b», нельзя запрограммировать непосредственно, поэтому и эту схему также необходимо реорганизовать.

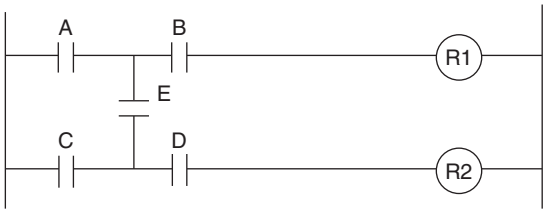
Схема «a»



Порядок выполнения (мнемонические обозначения)

(1) LD A	(6) AND B
(2) LD C	(7) OUT R1
(3) OUT TR0	(8) LD TR0
(4) AND D	(9) AND E
(5) OR LD	(10) OUT R2

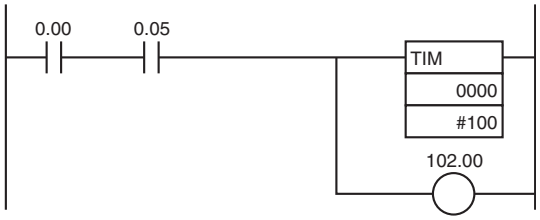
Схема «b»



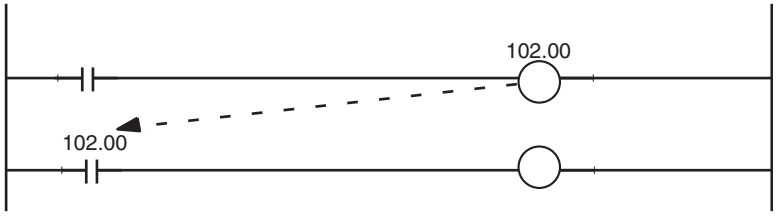
В схеме «a» катушку R2 невозможно запитать через предусмотренный контакт D.

Контакт E, предусмотренный в схеме «b», невозможно запрограммировать в «лестничной диаграмме». Программу необходимо видоизменить.

2. Количество битов входов и выходов, рабочих битов, таймеров и других входных битов не ограничено. С другой стороны, цепи программы должны быть как можно более простыми и понятными, даже если для этого потребуется применить побольше входных битов.
3. Количество битов входов, которое может быть включено последовательно или параллельно в последовательных или параллельных цепях программы, не ограничено.
4. Два или больше битов выходов можно соединять параллельно.



5. Биты выходов также можно использовать в качестве входных битов.

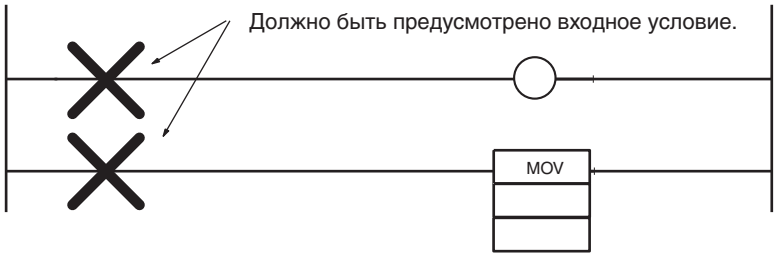


Ограничения

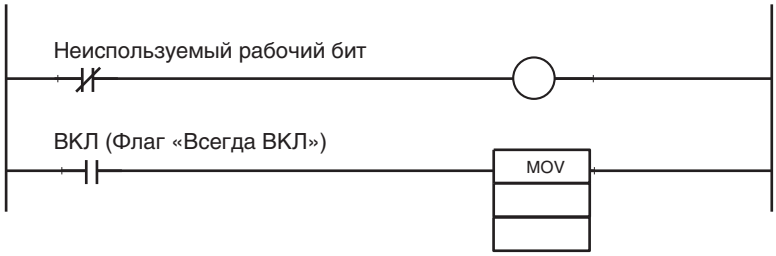
1,2,3... 1. Лестничная диаграмма должна быть замкнута таким образом, чтобы ток (сигнал) протекал от левой шины к правой шине. Если цепь не будет замкнута, произойдет ошибка цепи (однако программа может быть выполнена).



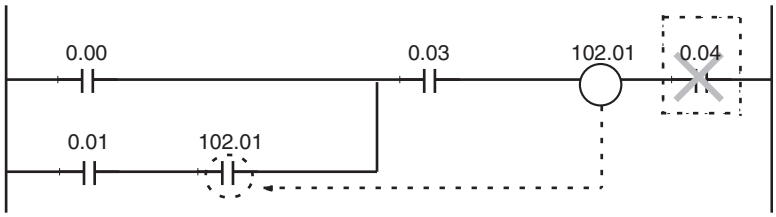
2. Биты выходов, таймеры, счетчики и другие выходные команды нельзя подсоединять непосредственно к левой шине. Если какой-либо из этих элементов будет подсоединен непосредственно к левой шине, во время проверки программы в CX-Programmer будет обнаружена ошибка цепи (программа может быть выполнена, но команды OUT и MOV(021) выполняться не будут).



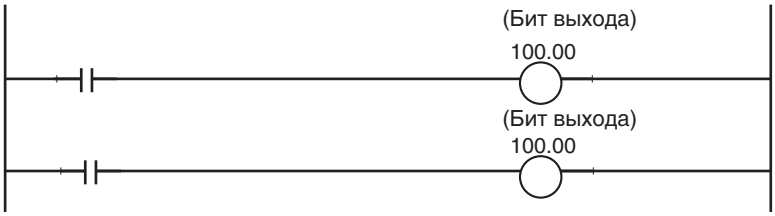
Если вход должен быть включен все время, следует вставить неиспользуемый нормально-замкнутый (НЗ) рабочий бит или флаг «Всегда ВКЛ».



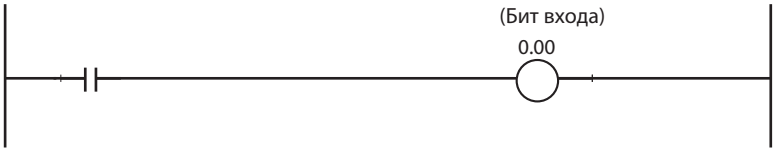
3. Входной бит следует всегда размещать до, а не после выходной команды (например, до выходного бита). Если он будет вставлен после выходной команды, при проверке программы в CX-Programmer будет обнаружена ошибка размещения.



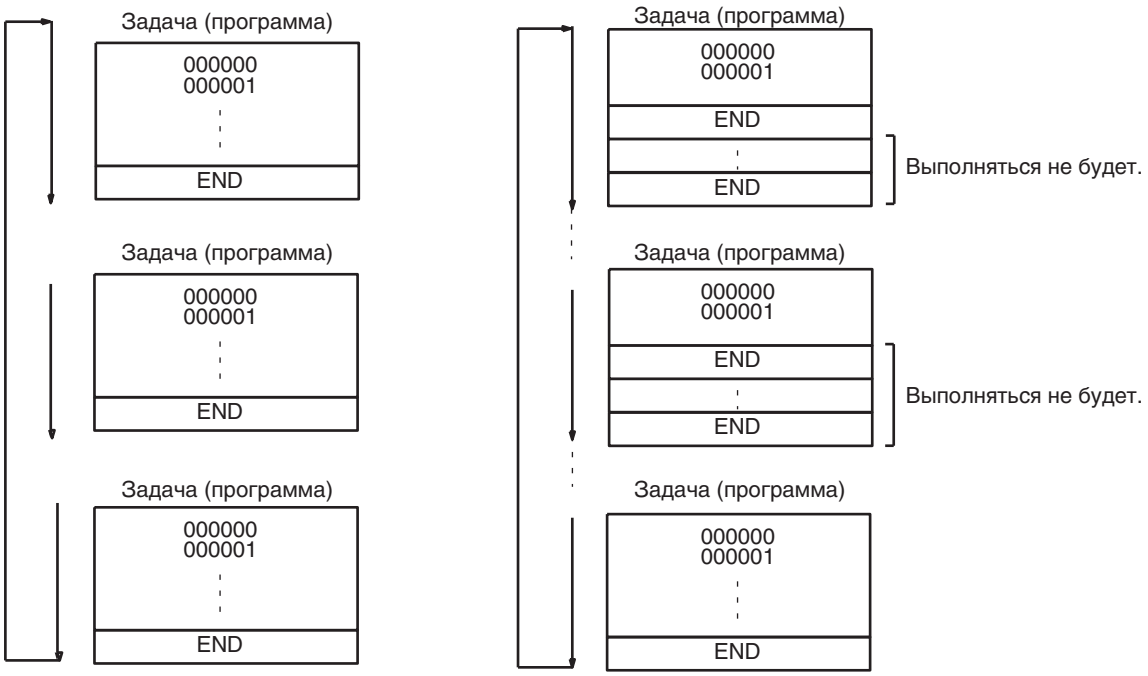
4. При создании программы нельзя использовать в выходной команде один и тот бит выхода. Команды в лестничной диаграмме выполняются в пределах одного цикла по порядку, начиная с самой верхней цепи, поэтому бит выхода в конечном итоге будет содержать результат выполнения выходной команды самой нижней цепи, содержащей этот бит, а результаты выполнения любых предшествующих команд, управляющих тем же битом, будут перезаписаны и утрачены.



5. Бит входа невозможно использовать в команде ВЫВОД (OUT).



6. Программа каждой задачи должна обязательно завершаться командой END(001).
   - Если начнется выполнение программы, не содержащей команду END(001), произойдет ошибка программы «No End Instruction» («Нет команды END»), на лицевой панели модуля ЦПУ будет светиться светодиод «ERR/ALM» и программа выполняться не будет.
   - Если в программе имеется несколько команд END(001), будет выполнена часть программы до первой команды END(001).
   - Для ускорения и упрощения отладки команду END(001) можно последовательно вставлять в различные точки программы, отделяя одни цепи программы от других. По завершении отладки все промежуточные команды END(001) необходимо удалить.

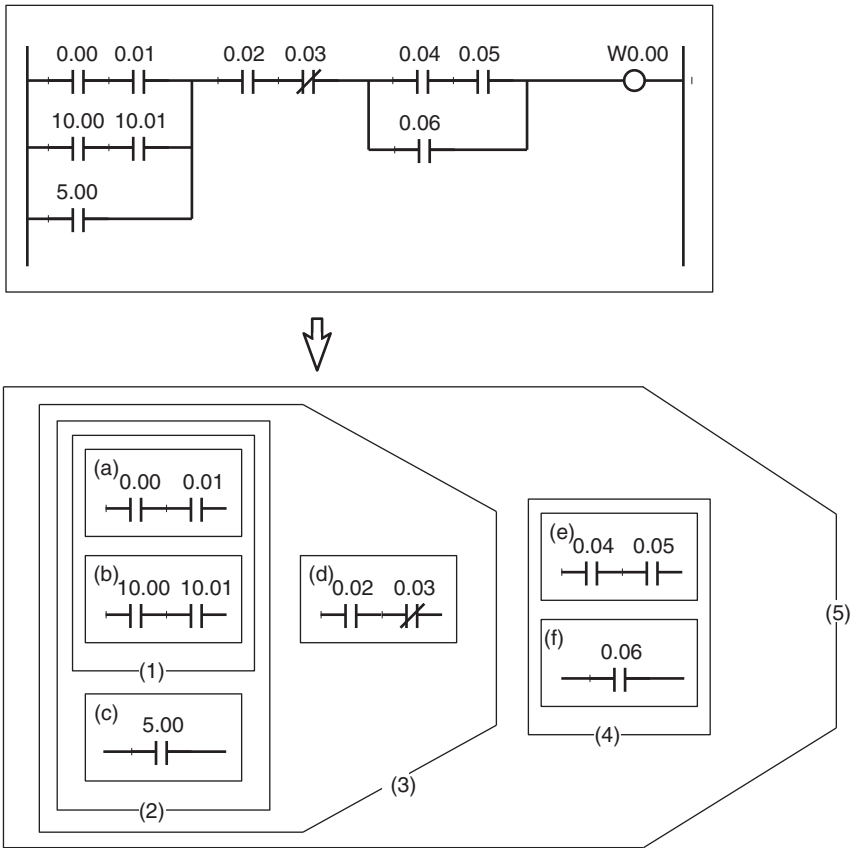


1-1-13 Ввод мнемонических обозначений

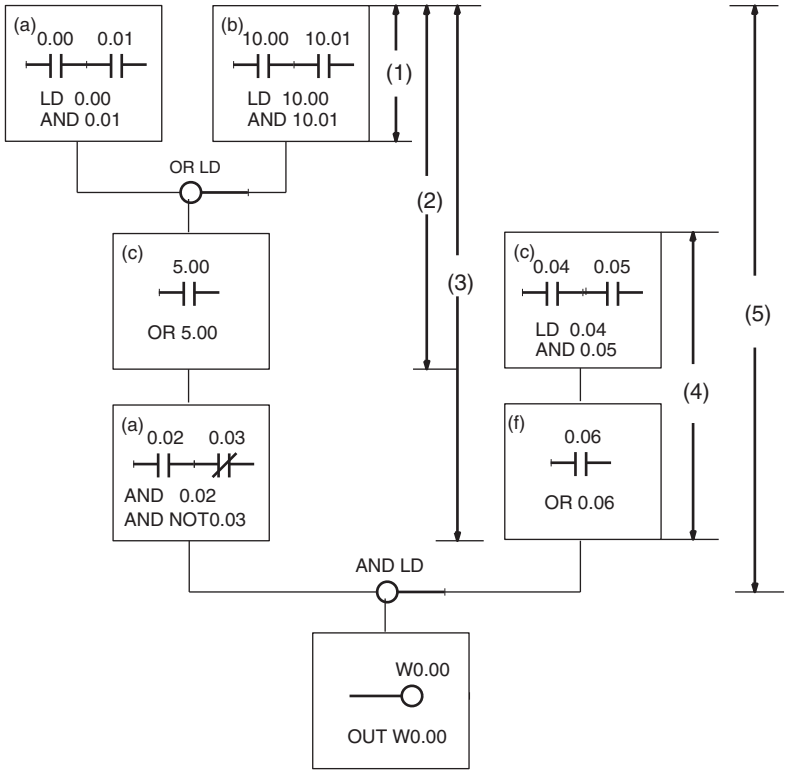
Логическим условием начала выполнения является команда LD/LDNOT. Область, заключенная между логическим условием начала выполнения и следующей командой LD/LD NOT, считается единым блоком команд.

Для создания отдельной цепи, состоящей из двух блоков команд, следует использовать команду AND LD (последовательное соединение блоков) или команду OR LD (параллельное соединение блоков). Процедура ввода мнемонических обозначений показана ниже на примере сложной цепи.

1,2,3... 1. Сначала разбейте цепь на небольшие блоки (a)...(f).



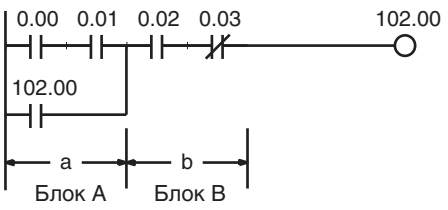
2. Составьте программу из блоков, располагая их в направлении сверху вниз, слева направо.



	Адрес	Команда	Операнд
(a)	200	LD	0.00
	201	AND	0.01
(b)	202	LD	10.00
	203	AND	10.01
	204	OR LD	---
(c)	205	OR	5.00
(d)	206	AND	0.02
	207	AND NOT	0.03
(e)	208	LD	0.04
	209	AND	0.05
(f)	210	OR	0.06
	211	AND LD	---
	212	OUT	W0.00

1-1-14 Примеры программ

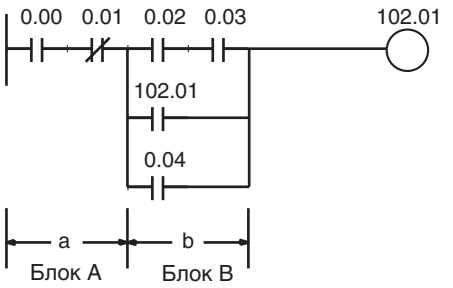
Цепи с параллельно-последовательным соединением



Команда	Операнды	
LD	0.00	a
AND	0.01	
OR	102.00	
AND	0.02	b
AND NOT	0.03	
OUT	102.00	

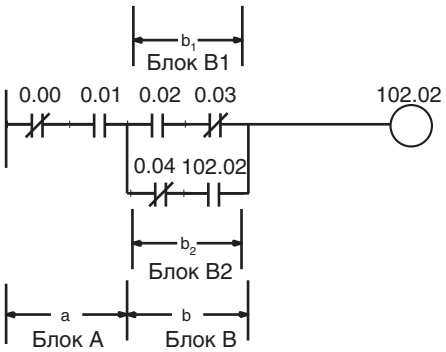
Сначала создайте блок A с параллельным соединением команд, затем создайте блок B.

Цепи с последовательно-параллельным соединением



Команда	Операнды	
LD	0.00	a
AND NOT	0.01	
LD	0.02	b
AND	0.03	
OR	102.01	
OR	0.04	
AND LD	---	
OUT	102.01	

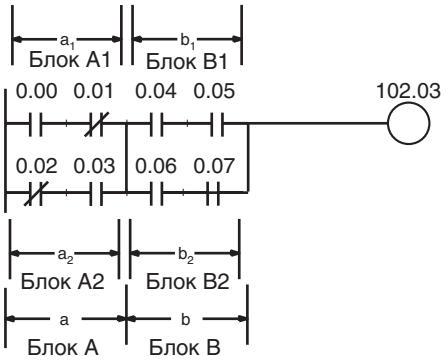
- Разбейте цепь на блоки A и B и создайте каждый из блоков отдельно.
- Соедините блоки A и B с помощью команды AND LD.
- Создайте программу блока A.



Команда	Операнды	
LD NOT	0.00	a
AND	0.01	
LD	0.02	b_1
AND NOT	0.03	
LD NOT	0.04	b_2
AND	102.02	
OR LD	---	$b_1 + b_2$
AND LD	---	$a \cdot b$
OUT	102.02	

- Составьте программу блока B_1, а затем — программу блока B_2.
- Соедините блоки B_1 и B_2 с помощью OR LD, а затем соедините блоки A и B с помощью AND LD.

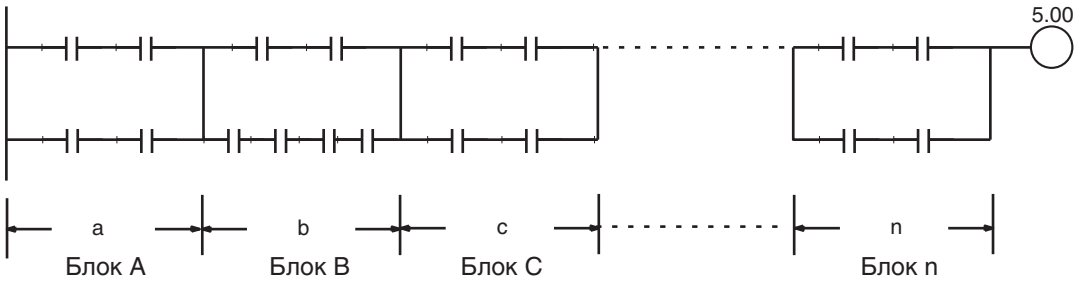
Пример последовательного соединения внутри цепи



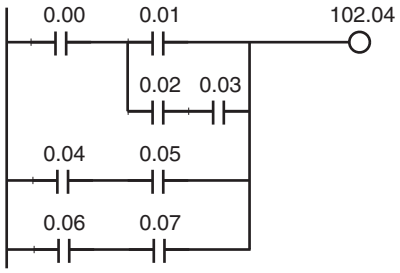
Команда	Операнды	
LD	0.00	a_1
AND NOT	0.01	
LD NOT	0.02	a_2
AND	0.03	
OR LD	---	$a_1 + a_2$
LD	0.04	b_1
AND	0.05	
LD	0.06	b_2
AND	0.07	
OR LD	---	$b_1 + b_2$
AND LD	---	$a \cdot b$
OUT	102.03	

- Создайте программу блока A_1, программу блока A_2, после чего соедините блоки A_1 и A_2 с помощью OR LD.
- Аналогичным образом создайте программы для B_1 и B_2.
- Соедините блок A и блок B с помощью AND LD.

• Выполните эти действия столько раз, сколько имеется таких блоков.

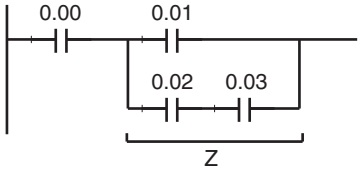


Сложные цепи

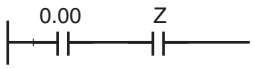


Команда	Операнд
LD	0.00
LD	0.01
LD	0.02
AND	0.03
OR LD	---
AND LD	---
LD	0.04
AND	0.05
OR LD	---
LD	0.06
AND	0.07
OR LD	---
OUT	102.04

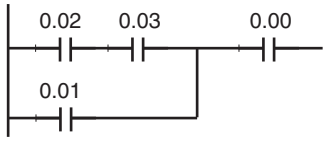
Z

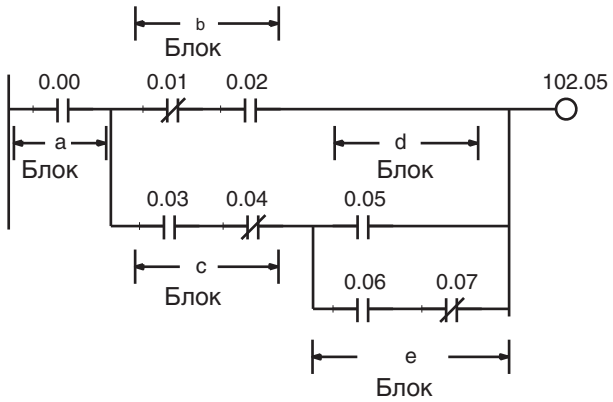


В основе верхней схемы лежит следующая схема.



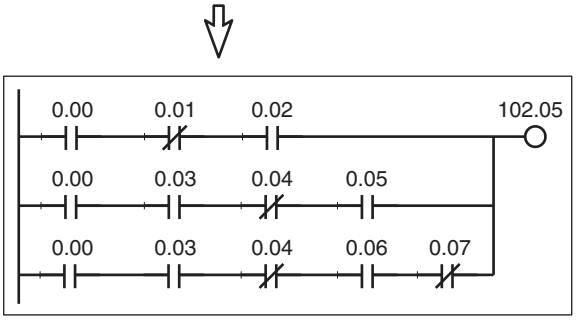
Если переписать программу, как показано ниже, можно получить более простую программу.

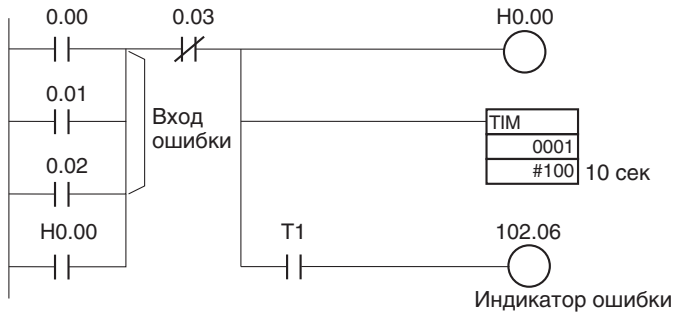




Команда	Операнд	
LD	0.00	a
LD NOT	0.01	b
AND	0.02	b
LD	0.03	c
AND NOT	0.04	c
LD	0.05	d
LD	0.06	d
AND NOT	0.07	e
OR LD	---	e
AND LD	---	d + e
OR LD	---	(d + e) · c
AND LD	---	(d + e) · c + b
OUT	102.05	((d + e) · c + b) · a

Эту цепь можно переписать следующим образом:





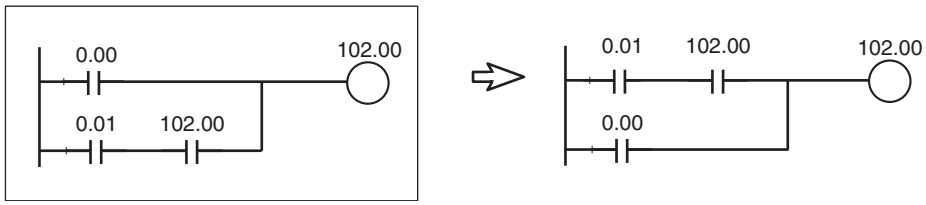
Команда	Операнд
LD	0.00
OR	0.01
OR	0.02
OR	H0.00
AND NOT	0.03
OUT	H0.00
TIM	0001
	#100
AND	T1
OUT	102.06

Если используется бит сохранения IOM, состояние (0 или 1) будет сохраняться в памяти даже после выключения питания, а сигнал ошибки будет действовать даже после повторного включения питания.

Цепи, требующие внимательности или реорганизации

Команды OR и OL LD

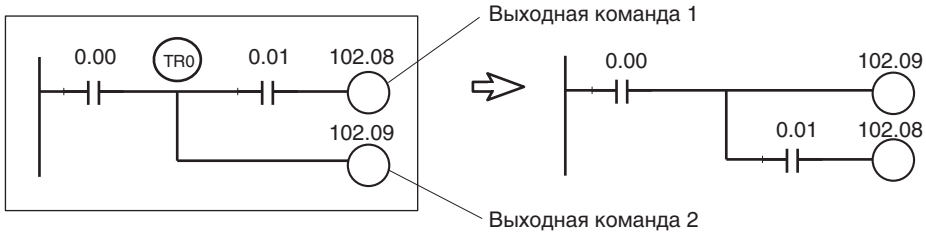
В случае команд OR или OR NOT логически складываются (ИЛИ) результаты выполнения команд лестничной диаграммы, расположенных между командой LD или LD NOT и командой OR или OR NOT, поэтому цепи можно реорганизовать таким образом, чтобы необходимость в команде OR LD отпала.



Пример: если показанные выше цепи оставить в том виде, как они есть, потребуется команда OR LD. Однако, изменив структуру цепей показанным выше способом, можно избавиться от нескольких шагов.

Ветвление выходных команд

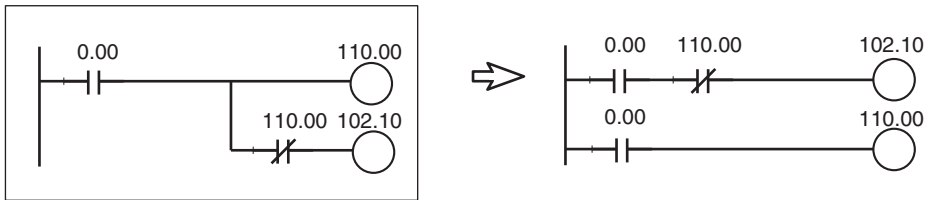
Если перед командой AND или AND NOT есть ветвление, требуется бит TR. Однако бит TR не потребуется, если ветвление происходит в точке, непосредственно подключенной к выходным командам и к команде AND или AND NOT, либо когда выходные команды могут быть продолжены в своем первоначальном виде.



Пример: без видоизменения показанных выше цепей в точке ветвления необходимо наличие выходной команды с битом временного хранения TR0 и входной (LD) команды. Перестроив цепи, можно избавиться от нескольких шагов.

Порядок выполнения мнемонических команд

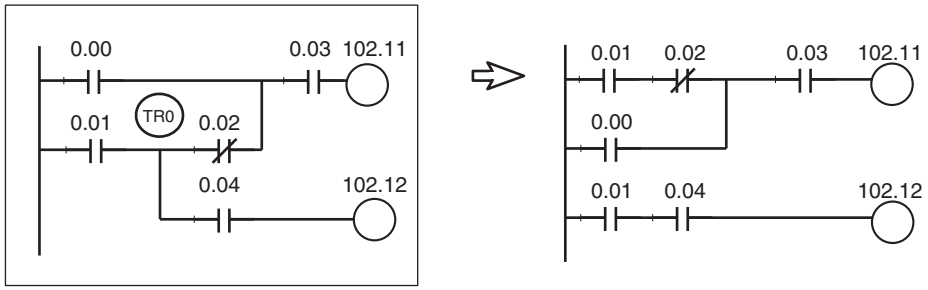
ПЛК выполняет РКС-программу (лестничную диаграмму) в том порядке, в котором вводились мнемонические обозначения, поэтому команды могут работать не так, как это предполагалось, в зависимости от способа записи цепей. При создании лестничных диаграмм всегда следует учитывать порядок выполнения мнемонических обозначений.



Пример: CIO 102.10 на представленной выше диаграмме не может быть передан на выход. Изменив структуру цепи, как показано выше, CIO 102.10 можно включить (ВКЛ) на один цикл.

Цепи, требующие видоизменения

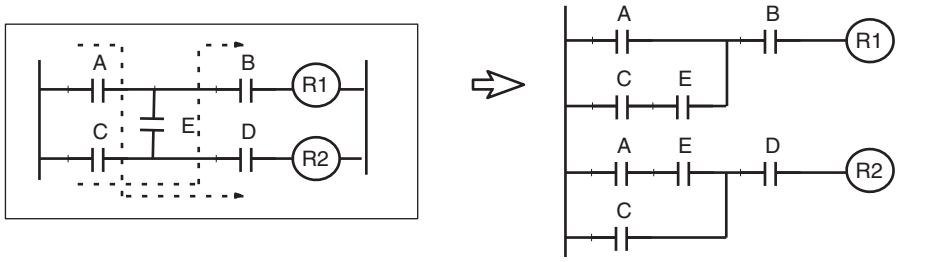
ПЛК выполняет команды в том порядке, в котором были введены мнемонические обозначения, поэтому ток (сигнал) в лестничной диаграмме направлен слева направо. Нельзя создавать программу, в которой ток протекает справа налево.



Пример: можно написать такую программу (см. верхнюю диаграмму слева), в которой в точку TR0 приходит ветвь. Однако такой же результат можно получить и с помощью цепей, показанных на рисунке справа. Такая схема более проста и понятна. Поэтому рекомендуется изменить структуру цепей, показанных слева, и использовать схему, показанную справа.

Измените структуру цепей, показанных на нижнем рисунке слева. В первоначальном виде такая программа выполнена быть не может.

Стрелками показано направление движения тока (сигнала), когда цепи состоят из управляющих реле.



1-2 Меры предосторожности

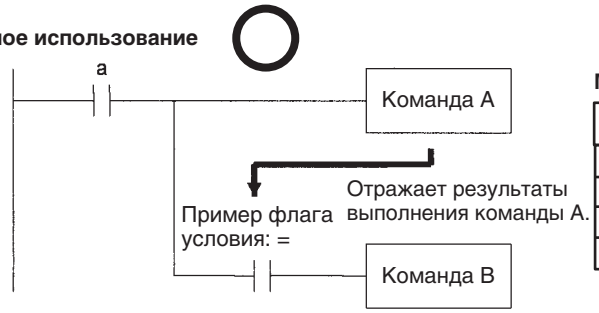
1-2-1 Флаги условий

Использование флагов условий

Флаги условий используются совместно всеми командами и изменяют свое состояние в пределах цикла в зависимости от результатов выполнения отдельных команд. Следовательно, на разветвленном выходе с одним и тем же условием выполнения флаги условий следует располагать сразу же после команды, чтобы они отражали результаты ее выполнения. Не следует подключать флаг условия непосредственно к шине, поскольку в этом случае он будет отражать результаты выполнения других команд.

Пример: использование результатов выполнения команды A.

Правильное использование

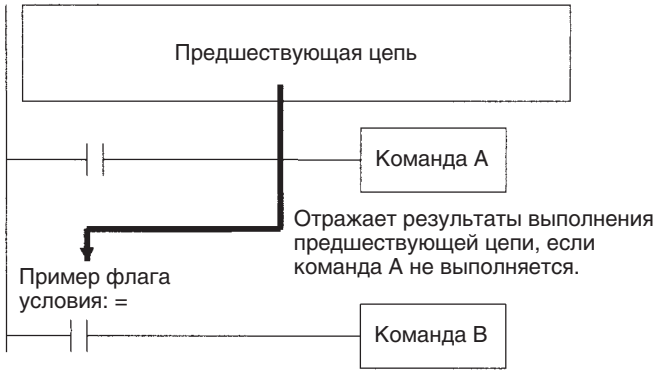


Мнемоническая форма

Команда	Операнд
LD	a
Команда A	
AND	=
Команда B	

Выполнение команды B зависит от результатов выполнения команды A, при этом для команд A и B используется общее условие выполнения (a). В этом случае команда B выполняется в соответствии с состоянием флага условия, только если была выполнена команда A.

Неправильное использование



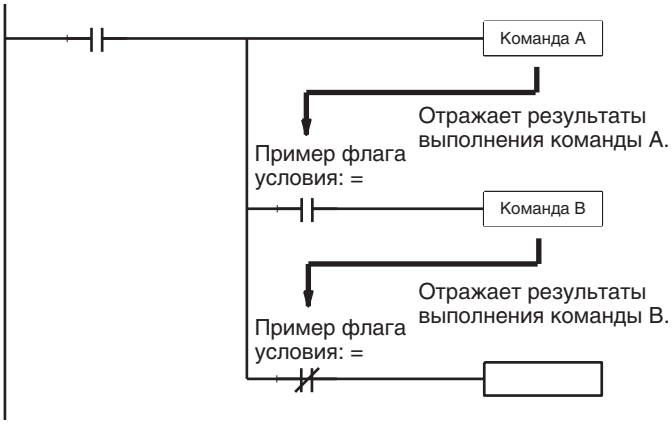
Если флаг условия подсоединен непосредственно к левой шине, команда B будет выполнена в соответствии с результатами выполнения предшествующей цепи, если не выполнена команда A.

Примечание. Флаги условий используются всеми командами в пределах одной программы (задачи), но они обнуляются при переключении задач. Следовательно, результаты выполнения предшествующей задачи не будут доступны в последующих задачах. Поскольку флаги условий используются совместно всеми командами, следует тщательно проверить, не мешают ли они работе друг друга в пределах всей программы. Пример работы с флагами приведен ниже.

Использование результатов выполнения на НЗ и НО входах

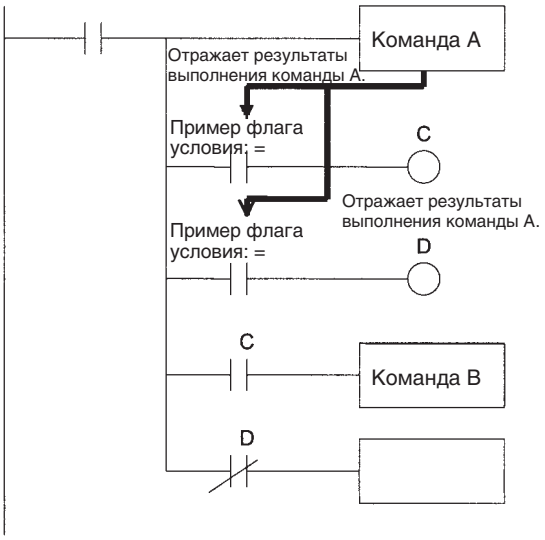
В приведенном ниже примере флаги условий будут отражать результаты выполнения команды B, хотя и нормально разомкнутый (НО), и нормально замкнутый (НЗ) биты входов используются в пределах одного разветвленного выхода.

Неправильное использование



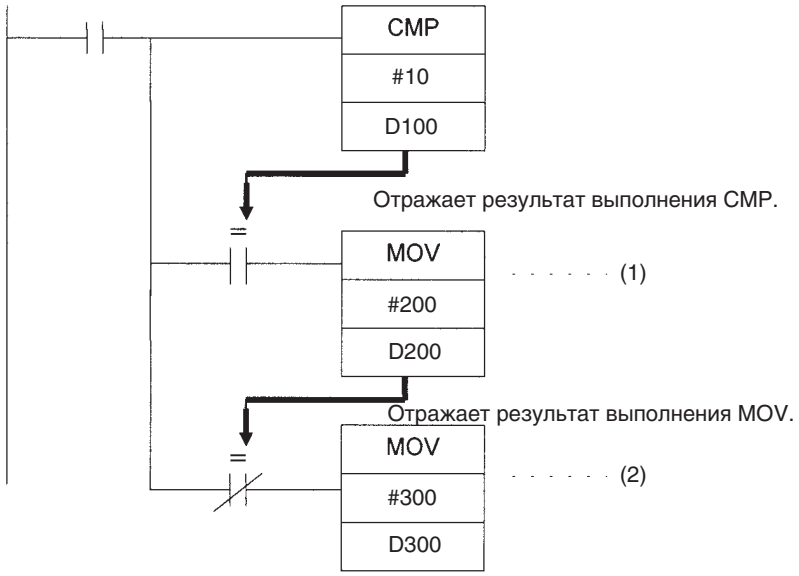
Позаботьтесь, чтобы каждый из результатов использовался командой вывода (OUTPUT) один раз, чтобы на состояния флагов не повлияли результаты выполнения команды B.

Правильное использование



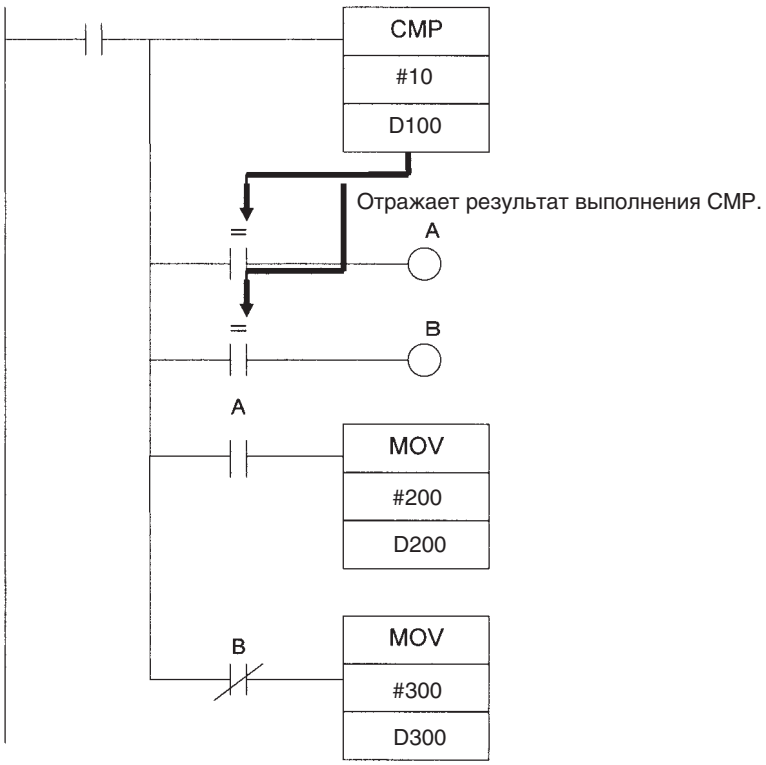
Пример: в D200 будет записано значение #200, если D100 содержит #10; если D100 не содержит значение #10, в D300 будет записано значение #300.

Неправильное использование



Если слово D100 в приведенной выше цепи содержит #10, установится флаг равенства. Команда (1) запишет в слово D200 значение #200, но после этого флаг равенства сбросится, поскольку входное значение (#200) не равно 0000 Hex. После этого будет выполнена команда MOV (2) и в слово D300 будет записано значение #300. Следовательно, чтобы предотвратить считывание результатов выполнения первой команды MOV, необходимо вставить дополнительную цепь, как показано на рисунке ниже.

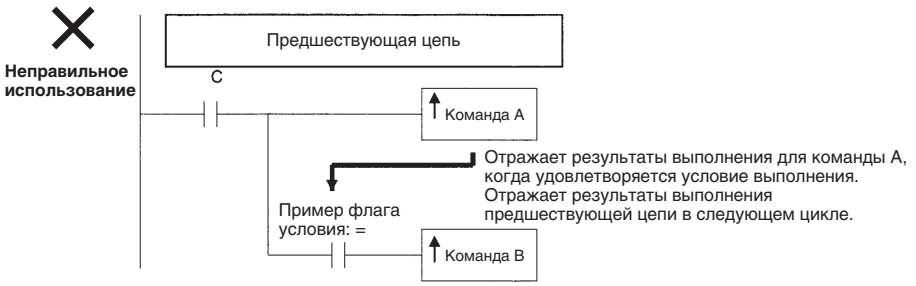
Правильное использование



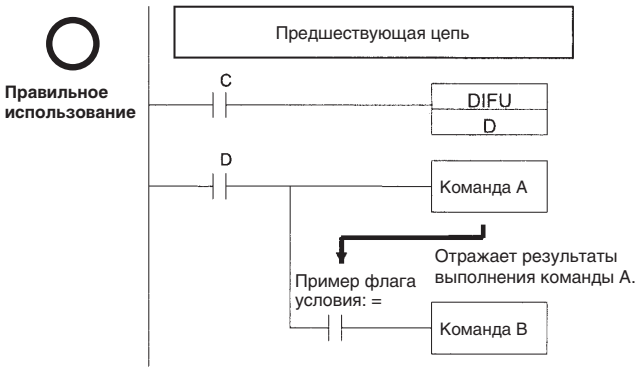
Использование результатов выполнения команд с различением фронтов

Результаты выполнения команд, действующих однократно по переключению условия выполнения, отражаются во флагах условий, только если удовлетворяется условие выполнения, и в следующем цикле во флагах условий отражаются результаты выполнения предшествующей цепи (а не результаты выполнения однократной команды). Поэтому, если требуется использовать результаты выполнения однократных команд, необходимо учесть, что будет происходить с флагами условий в следующем цикле.

В приведенном ниже примере команды A и B выполняются, только когда выполняется условие C. Но при считывании командой B результатов выполнения команды A возникает следующая проблема. Если условие выполнения C будет оставаться включенным в следующих циклах после выполнения команды A, то команда B выполнится (хотя это и не предусмотрено), когда флаг условия вдруг переключится из «0» в «1», отражая результаты выполнения предшествующей цепи.



Следовательно, в данном случае сами команды A и B не должны быть однократными, вместо этого должна использоваться команда DIFU (или DIFD) (см. рисунок ниже), благодаря которой обе команды A и B будут выполняться только в одном цикле по положительному или отрицательному перепаду условия выполнения C.



Примечание. Модулями ЦПУ CP1H поддерживаются команды, предназначенные для сохранения и загрузки состояний флагов условий. Это команды CCS(282) и CCL(283). Они могут использоваться для чтения/записи состояний флагов условий в других местах задачи или в другой задаче.

Основные условия, приводящие к включению флагов условий

Флаг ошибки

Флаг ER устанавливается при особых обстоятельствах, например, когда для команды указан ошибочный операнд. При установленном флаге ER команда не выполняется.

Когда флаг ER установлен, состояние других флагов условий, например флагов «<», «>», «OF» и «UF», не изменяется, а состояние флагов «=» и «N» изменяется от команды к команде.

Перечень условий, которые приводят к включению флага ER, можно найти в описаниях отдельных команд в настоящем руководстве. Следует быть внимательным, поскольку некоторые команды сбрасывают флаг ER независимо от условий и обстоятельств.

Примечание. С помощью параметра в области настроек ПЛК можно указать, должна ли прекращаться работа при включении флага ER, то есть при возникновении ошибки команды. По умолчанию включение флага ER не останавливает работу. Если же выбрано прекращение работы после включения флага ER, работа прекращается (воспринимается как ошибка программы), а в слова A298...A299 записывается адрес точки программы, в которой прервалась работа. Одновременно включается бит A295.08.

Флаг равенства

Флаг равенства — это временный флаг для всех команд (кроме случая равенства операндов в командах сравнения). Он автоматически устанавливается системой и изменяется в процессе работы. Флаг равенства может быть сброшен (установлен) следующей командой после того, как он был установлен (сброшен) предшествующей командой. Например, флаг равенства устанавливается, когда MOV или другая команда перемещения данных записывает в адресуемое слово значение 0000 Hex, и сбрасывается во всех других случаях. Даже если некоторая команда сбрасывает флаг равенства, выполняемая вслед за ней команда перемещения данных может сразу же установить этот флаг, если входное значение команды перемещения будет содержать 0000 Hex.

Флаг переноса

Флаг CY используется в командах сдвига данных, в командах сложения и вычитания со входом переноса, в командах сложения и вычитания со входом заема, а также в командах, предназначенных для специального модуля ввода/вывода, в командах ПИД-регулирования и в командах для FPD. Следует соблюдать следующие меры предосторожности.

Примечание.

(1) В результате выполнения некоторой команды флаг CY может остаться включенным (выключенным) и затем будет использоваться другой командой (командой сложения или вычитания с переносом или командой сдвига). Обязательно очищайте флаг переноса, когда это необходимо.

(2) Флаг CY, включенный (или выключенный) в результате выполнения определенной команды, может быть выключен (или включен) другой командой. Обязательно следите за тем, чтобы состояние флага переноса было актуально, когда он используется.

Флаг меньшего и большего значения

Флаги < и > используются в командах, выполняющих операции сравнения, а также в командах LMT, BAND, ZONE, PID и других командах.
Флаг < или > может быть сброшен (или установлен) другой командой, даже если он был установлен (или сброшен) в результате выполнения предшествующей команды.

Флаг отрицательного значения

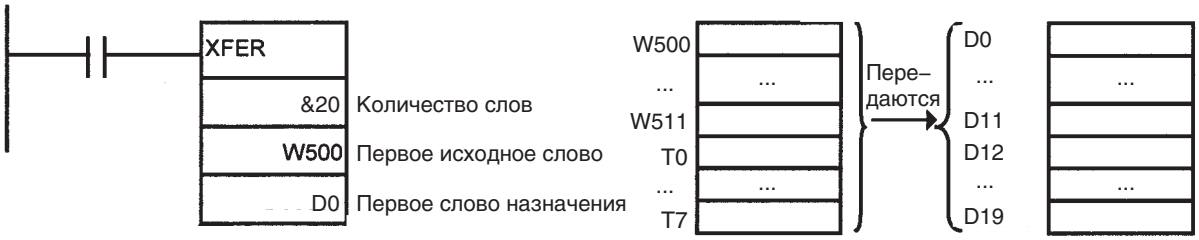
Флаг N сбрасывается, когда старший (крайний левый) бит слова результата выполнения определенных команд содержит «1», и сбрасывается без каких-либо условий для другой команды.

Операнды, занимающие несколько слов

В ПЛК серии CP команда будет выполнена в том виде, в котором она записана, даже если ее операнд, состоящий из нескольких слов, указан таким образом, что слова находятся в разных областях памяти. В этом случае обращение к словам производится в порядке возрастания адресов памяти ПЛК. Флаг ошибки при этом **не** включится.

В качестве примера рассмотрим результаты выполнения операции передачи блока с помощью команды XFER(070). Передается 20 слов, начиная с W500. В этом случае имеет место выход за рабочую область (Work Area), которая завершается словом W511, но команда будет выполнена без включения флага ошибки. В адресном пространстве ПЛК следом за рабочей областью располагаются текущие значения таймеров, поэтому в нашем случае в ячейки D0... D11 будут переданы слова W500...W511, а в ячейки D12...D19 будут записаны текущие значения таймеров T0...T7.

Примечание. Точное распределение адресов памяти ПЛК для модулей ЦПУ CP1H можно найти в *Приложении E: Распределение памяти* в руководстве *Серия CP — Модули ЦПУ CP1H. Руководство по эксплуатации (W450)*. Точное распределение адресов памяти ПЛК для модулей ЦПУ CP1L можно найти в *Приложении E: Распределение памяти* в руководстве *Серия CP — Модули ЦПУ CP1L. Руководство по эксплуатации (W462)*.



1-2-2 Специальные разделы программ

В программах ПЛК серии CP имеются особые разделы, осуществляющие управление условиями выполнения команд. Предусмотрены следующие специальные разделы программ.

Раздел программы	Команды	Условие выполнения команды	Состояние
Подпрограмма	Команды SBS, SBN и RET	Выполнение подпрограммы.	Выполняется подпрограмма, т. е. сегмент программы, размещенный между командами SBN и RET.
Сегмент IL – ILC	Команды IL и ILC	Блокировка сегмента программы.	Биты выходов выключаются, таймеры сбрасываются. Остальные команды не выполняются, сохраняя свои предшествующие состояния.
Сегмент пошаговой программы	Команды STEP S и команды STEP		
Цикл FOR – NEXT	Команды FOR и команды NEXT	Выдерживание паузы.	Зацикливание.
Сегмент JMP0 – JME0	Команды JMP0 и команды JME0		Переход.
Сегмент программного блока	Команды BPRG и команды BEND	Выполнение программного блока.	Выполняется программный блок, мнемонические команды которого размещены между командами BPRG и BEND.

Комбинации команд

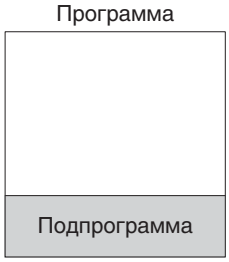
В следующей таблице показано, какие специальные команды могут использоваться внутри других особых разделов программ.

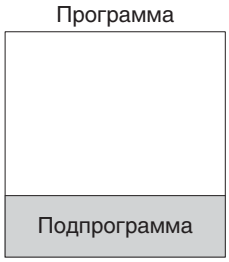
	Подпрограмма	Сегмент IL – ILC	Сегмент пошаговой программы	Цикл FOR – NEXT	Сегмент JMP0 – JME0	Сегмент програм-много блока
Подпрограмма	Невозможно	Невозможно	Невозможно	Невозможно	Невозможно	Невозможно
IL – ILC	**OK**	Невозможно	Невозможно	**OK**	**OK**	Невозможно
Сегмент пошаговой программы	Невозможно	**OK**	Невозможно	Невозможно	**OK**	Невозможно
Цикл FOR – NEXT	**OK**	**OK**	Невозможно	**OK**	**OK**	Невозможно
JMP0 – JME0	**OK**	**OK**	Невозможно	Невозможно	Невозможно	Невозможно
Сегмент программного блока	**OK**	**OK**	**OK**	Невозможно	**OK**	Невозможно

Примечание. Команды, которые служат для определения областей программы, нельзя использовать для программ в других задачах. Подробную информацию смотрите в разделе *2-2-2 Ограничения на применение команд в задачах* .

Подпрограммы

Во всех программах сегменты подпрограмм должны располагаться все вместе в самом конце программы (то есть после всех остальных сегментов) непосредственно перед командой END(001) (поэтому подпрограмму нельзя разместить в сегменте пошаговой программы, в программном блоке, в цикле FOR – NEXT или в разделе JMP0 – JME0). Никакая программа (кроме подпрограммы), расположенная после сегмента подпрограммы (после блока SBN – RET), выполняться не будет.





Команды, недоступные для подпрограмм

Следующие команды невозможно использовать внутри подпрограммы.

Функция	Мнемоническое обозначение	Команда
Пошаговое управление процессом	STEP(008)	Определяет начало сегмента пошаговой программы.
	SNXT(009)	Переход на один шаг вперед.

Примечание. **Сегменты программных блоков**
Подпрограмма может включать программные блоки. Но если программный блок находится в состоянии ожидания (WAIT), когда происходит возврат из подпрограммы к основной программе, программный блок останется в состоянии ожидания при следующем вызове.

Команды, недоступные для сегментов пошаговых программ

Функция	Мнемоническое обозначение	Команда
Управление последовательностью выполнения	FOR(512), NEXT(513) и BREAK(514)	FOR, NEXT и ВЫХОД ИЗ ЦИКЛА
	END(001)	КОНЕЦ
	IL(002) и ILC(003)	БЛОКИРОВКА и ОТМЕНА БЛОКИРОВКИ
	JMP(004) и JME(005)	ПЕРЕХОД и КОНЕЦ ПЕРЕХОДА
	CJP(510) и CJPN(511)	УСЛОВНЫЙ ПЕРЕХОД и УСЛОВНЫЙ ПЕРЕХОД НЕ
	JMP0(515) и JME0(516)	МНОЖЕСТВЕННЫЙ ПЕРЕХОД и КОНЕЦ ПЕРЕХОДА
Подпрограммы	SBN(092) и RET(093)	ВХОД В ПОДПРОГРАММУ и ВЫХОД ИЗ ПОДПРОГРАММЫ
Программные блоки	IF(802) (NOT), ELSE(803) и IEND(804)	Команды ветвления
	BPRG(096) и BEND(801)	НАЧАЛО/КОНЕЦ ПРОГРАММНОГО БЛОКА
	EXIT(806) (NOT)	ВЫХОД ИЗ БЛОКА ПО УСЛОВИЮ (НЕ)
	LOOP(809) и LEND(810) (NOT)	Управление циклами
	WAIT(805) (NOT)	ОДИН ЦИКЛ И ОЖИДАНИЕ (НЕ)
	TIMW(813) и TIMWX(816)	ТАЙМЕР ОЖИДАНИЯ
	TMHW(815) и TMHWX(817)	СКОРОСТНОЙ ТАЙМЕР ОЖИДАНИЯ
	CNTW(814) и CNTWX(818)	СЧЕТЧИК ОЖИДАНИЯ
	BPPS(811) и BPRS(812)	ПРИОСТАНОВКА и ПЕРЕЗАПУСК ПРОГРАММНОГО БЛОКА

Примечание.
(1) Сегмент пошагового выполнения программы можно использовать внутри блокируемого сегмента (т. е. между IL и ILC). При включении блокировки пошаговая программа полностью сбрасывается.
(2) Сегмент пошагового выполнения программы можно размещать между командами JMP0 (МНОЖЕСТВЕННЫЙ ПЕРЕХОД) и JME0 (КОНЕЦ МНОЖЕСТВЕННОГО ПЕРЕХОДА).

Команды, недоступные для программных блоков

Следующие команды невозможно использовать внутри сегментов программных блоков.

Функция	Мнемоническое обозначение	Команда
Управление последовательностью выполнения	FOR(512), NEXT(513) и BREAK(514)	FOR, NEXT и ВЫХОД ИЗ ЦИКЛА
	END(001)	КОНЕЦ
	IL(002) и ILC(003)	БЛОКИРОВКА и ОТМЕНА БЛОКИРОВКИ
	JMP0(515) и JME0(516)	МНОЖЕСТВЕННЫЙ ПЕРЕХОД и КОНЕЦ ПЕРЕХОДА
Входные битовые команды	UP(521)	УСЛОВИЕ ВКЛ
	DOWN(522)	УСЛОВИЕ ВЫКЛ
Выходные битовые команды	DIFU	ПОЛОЖИТЕЛЬНЫЙ ФРОНТ
	DIFD	ОТРИЦАТЕЛЬНЫЙ ФРОНТ
	KEEP	ЗАЩЕЛКА
	OUT	ВЫВОД
	OUT NOT	ВЫВОД НЕ
Таймер/Счетчик	TIM и TIMX(550)	ТАЙМЕР
	TIMH(015) и TIMHX(551)	СКОРОСТНОЙ ТАЙМЕР
	TMHH(540) и TMHHX(552)	1 мс ТАЙМЕР
	TTIM(087) и TTIMX(555)	НАКАПЛИВАЮЩИЙ ТАЙМЕР
	TIML(542) и TIMLX(553)	ДОЛГОСРОЧНЫЙ ТАЙМЕР
	MTIM(543) и MTIMX(554)	МНОГОВЫХОДНОЙ ТАЙМЕР
	CNT и CNTX(546)	СЧЕТЧИК
	CNTR(012) и CNTRX(548)	РЕВЕРСИВНЫЙ СЧЕТЧИК
Подпрограммы	SBN(092) и RET(093)	ВХОД В ПОДПРОГРАММУ и ВЫХОД ИЗ ПОДПРОГРАММЫ
Сдвиг данных	SFT	РЕГИСТР СДВИГА
Пошаговое выполнение программы	STEP(008) и SNXT(009)	ОПРЕДЕЛИТЬ ШАГ и ЗАПУСТИТЬ ШАГ
Регулирование	ПИД	ПИД-РЕГУЛЯТОР
Программный блок	BPRG(096)	НАЧАЛО ПРОГРАММНОГО БЛОКА
Диагностика неисправностей	FPD(269)	ОБНАРУЖЕНИЕ НЕИСПРАВНОГО КАНАЛА
Команды с префиксами вариантов выполнения	@XXX	Выполнение команды по положительному фронту
	%XXX	Выполнение команды по отрицательному фронту

Примечание. (1) Программные блоки можно использовать внутри сегмента пошагового выполнения программы.

(2) Программный блок можно использовать внутри блокируемого сегмента (т. е. между IL и ILC). При включенной блокировке программный блок не исполняется.

(3) Программный блок можно размещать между командами JMP0 (МНОЖЕСТВЕННЫЙ ПЕРЕХОД) и JME0 (КОНЕЦ МНОЖЕСТВЕННОГО ПЕРЕХОДА).

(4) Внутри программного блока можно использовать команды JMP (ПЕРЕХОД) и CJP/CJPN (УСЛОВНЫЙ ПЕРЕХОД). Вместе с командой JMP (ПЕРЕХОД) или CJP/CJPN (УСЛОВНЫЙ ПЕРЕХОД) в программном блоке также обязательно должна присутствовать парная команда JME (КОНЕЦ ПЕРЕХОДА). В противном случае программа будет работать неправильно.

1-3 Проверка программ

Проверку программ ПЛК серии CP можно выполнить на следующих этапах:

- проверка при вводе программы или выполнении других операций в CX-Programmer;
- проверка программы с помощью CX-Programmer;
- проверка команд в процессе выполнения программы;
- проверка критических ошибок (ошибок программы) в процессе выполнения программы.

1-3-1 CX-Programmer

CX-Programmer автоматически проверяет прикладную программу при выполнении следующих операций.

Момент проверки	Предмет проверки
В процессе создания программы	Вводимые команды, вводимые операнды, сегменты программы
При загрузке файлов	Все операнды во всех командах и все программные сегменты
При считывании файлов	Модели, поддерживаемые серией CP, и все операнды во всех командах
Во время онлайн-редактирования	Объем программы и т. п.

Результаты проверки выводятся в текстовой строке окна вывода информации (Output Window). Кроме того, левая шина сегментов программы, в которых имеются ошибки, в окне представления программы в виде «лестничной диаграммы» отображается красным цветом.

1-3-2 Проверка программы с помощью CX-Programmer

В следующей таблице перечислены ошибки, которые могут быть обнаружены при проверке программы в пакете CX-Programmer.

CX-Programmer не проверяет наличие ошибок диапазонов в случае косвенной адресации к операндам в командах. Ошибки косвенной адресации будут обнаружены во время проверки выполнения программы, что приведет к включению флага ER (см. описание в следующем разделе). См. подробное описание интересующей вас команды в соответствующем разделе.

При проверке программы с помощью CX-Programmer пользователь может выбрать один из уровней проверки программы: A, B или C (в порядке возрастания «серьезности» ошибки), а также уровень проверки, конфигурируемый самим пользователем.

Область	Проверка
Недопустимые данные: лестничная диаграмма	Размещение команд
	Каналы ввода/вывода
	Соединения
	Полнота (завершенность) команд и операций
Поддержка команд программируемым контроллером (ПЛК)	Команды и операнды, поддерживаемые ПЛК
	Варианты выполнения команд (НЕ, !, @, и %)
	Целостность объектного кода
Диапазоны значений операндов	Диапазоны значений операндов
	Типы данных операндов
	Проверка вида доступа к словам, защищенным от записи
	Проверка диапазонов значений операндов, в том числе: • проверка констант (#, &, +, –); • проверка управляющих кодов; • проверка границ областей для операндов, занимающих несколько слов; • проверка соответствия размеров для операндов, занимающих несколько слов; • обнаружение наложений диапазонов значений операндов; • проверка областей, отведенных для группы слов; • проверка операндов двойной длины; • проверка границ областей для смещений.
Размер программы ПЛК	Количество шагов
	Общий размер программы
	Количество задач
Синтаксис	Проверка вызовов для парных команд: • IL–ILC; • JMP–JME, CJP/CJPN–JME; • SBS–SBN–RET, MCRO–SBN–RET; • STEP–SNXT; • BPRG–BEND; • IF–IEND; • LOOP–LEND.
	Области, в которых запрещено размещать BPRG–BEND.
	Области, в которых запрещено размещать SBN–RET.
	Области, в которых запрещено размещать STEP–SNXT.
	Области, в которых запрещено размещать FOR–NEXT.
	Области, в которых запрещено размещать задачи обработки прерываний.
	Недопустимые многократные вложения.
	Команда END(001)
	Согласованность нумерации
Структура лестничной диаграммы	Переполнение стека

Область	Проверка
Дублирование выходов (см. примечание)	Проверка дублированных выходов: • по битам; • по словам; • команды управления таймерами/счетчиками; • длинные операнды (по 2 слова и по 4 слова); • резервирование нескольких слов; • диапазоны начала/завершения; • номера FAL; • команды с несколькими выходными операндами.
Задачи	Проверка задач, для которых выбран запуск в начале работы.
	Назначение программ задачам.

Примечание. Дублирование выходов проверяется только в пределах каждой отдельной задачи.

Операнды длиной в несколько слов

При проверке программы проверяются границы областей памяти для операндов, занимающих несколько слов (см. таблицу ниже).

Предмет проверки	Если операнды длиной в несколько слов выходят за границу области памяти, CX-Programmer работает следующим образом. • Программа не может быть записана в модуль ЦПУ. • Программа не может быть прочитана из модуля ЦПУ. • При проверке программы сигнализируются ошибки компиляции. • При автономном программировании на экране отображаются предупреждения. • При онлайн-редактировании в режимах «Программирование» или «Мониторинг» на экране отображаются предупреждения.

1-3-3 Проверка выполнения программы

Во время ввода программы и во время проверки программы в CX-Programmer проверяется правильность размещения операндов и команд. Но такая проверка не является окончательной.

В процессе выполнения команд проверяется наличие следующих ошибок.

Тип ошибки	Флаг, включающийся при ошибке	Прекращение/продолжение работы
1. Ошибка выполнения команды	Флаг ER **Примечание.** Если выбрано прекращение работы при возникновении ошибки, также установится флаг ошибки выполнения команды (A295.08).	В области настроек ПЛК предусмотрен параметр, с помощью которого можно указать, должна ли работа прекращаться или продолжаться в случае возникновения ошибок при выполнении команд. По умолчанию работа продолжается. При обнаружении ошибки программы работа прекращается только в том случае, если выбрано прекращение работы.
2. Ошибка доступа.	Флаг AER **Примечание.** Если выбрано прекращение работы при возникновении ошибки, также установится флаг ошибки доступа (A295.10).	В области настроек ПЛК предусмотрен параметр, с помощью которого можно указать, должна ли работа прекращаться или продолжаться в случае возникновения ошибок при выполнении команд. По умолчанию работа продолжается. При обнаружении ошибки программы работа прекращается только в том случае, если выбрано прекращение работы.
3. Ошибка недопустимой команды	Флаг «Ошибка недопустимой команды» (A295.14)	Критическая (ошибка программы)
4. Ошибка переполнения памяти пользователя (UM)	Флаг «Ошибка переполнения UM» (A295.15)	Критическая (ошибка программы)

Ошибки выполнения команд

Ошибка выполнения команды возникает, если для выполнения команды были предоставлены некорректные данные или была осуществлена попытка выполнения команды за пределами задачи. Необходимые данные проверяются в начале выполнения команды. Если в этих данных имеется ошибка, команда не выполняется, устанавливается флаг ошибки ER, а флаги EQ и N могут остаться включенными или сбросятся, в зависимости от типа команды.

Флаг ошибки ER сбрасывается, если команда (за исключением входных команд) завершается без ошибки. Условия, при которых устанавливается флаг ошибки, разные для каждой команды. См. подробное описание интересующей вас команды в соответствующем разделе.

Если в настройках ПЛК выбрано прекращение работы при ошибке команды, то при возникновении ошибки выполнения команды работа прекращается (критическая ошибка) и устанавливается флаг ошибки выполнения команды (A295.08), а также флаг ER.

Ошибки доступа

Ошибки, связанные с некорректным обращением к данным (ошибки доступа), информируют о том, что при обращении к адресу, который был указан для операнда команды, произошло обращение к недопустимой области, а именно:

1,2,3...

1. Произведено чтение или запись в область параметров.
2. Произведена запись в отсутствующую область памяти (см. примечание).

3. Произведена запись в область, предназначенную только для чтения.
4. Значение, указанное косвенным адресом DM в режиме BCD, оказалось не в двоично-десятичном формате (например, *D1 содержит #A000).

Регистр IR, содержащий адрес внутренней памяти для бита, используется как адрес слова; либо регистр IR, содержащий адрес внутренней памяти для слова, используется как адрес бита.

Когда происходит ошибка доступа, включается флаг ошибки доступа (флаг AER), но выполнение команды продолжается и флаг ошибки (флаг ER) не устанавливается.

Если в настройках ПЛК выбрано прекращение работы при ошибке команды, то при возникновении ошибки доступа работа прекращается (критическая ошибка) и устанавливается флаг ошибки доступа (A295.10).

Примечание. Флаг ошибки доступа (флаг AER) не очищается после завершения задачи. Если в настройках ПЛК выбрано продолжение работы при ошибке команды, состояние этого флага сохраняется вплоть до достижения команды END(001), благодаря чему можно проверить, не произошла ли при выполнении задачи ошибка доступа (при наблюдении за флагом AER в CX-Programmer отображается конечное состояние флага AER после выполнения целиком всей программы пользователя).

Прочие ошибки

Ошибка недопустимой команды

Ошибка недопустимой команды возникает при попытке выполнения команды, не предназначенной для данной системы. Как правило, если программа создается в CX-Programmer, то эта ошибка не наблюдается.

В тех исключительно редких случаях, когда такая ошибка происходит, она воспринимается как ошибка программы, работа прекращается (критическая ошибка) и устанавливается флаг недопустимой команды (A295.14).

Ошибки переполнения памяти пользователя (UM)

Ошибки переполнения UM возникают при попытке выполнения команды, данные которой выходят за последний адрес памяти пользователя (UM), выбранной в качестве области хранения программы. Как правило, если программа создается в CX-Programmer, то эта ошибка не наблюдается.

В тех исключительно редких случаях, когда такая ошибка происходит, она воспринимается как ошибка программы, работа прекращается (критическая ошибка) и устанавливается флаг переполнения памяти пользователя (A295.15).

1-3-4 Проверка на наличие критических ошибок

Ниже перечислены критические ошибки программы, возникновение которых прекращает работу модуля ЦПУ. Когда работа прекращается из-за ошибки программы, по адресу A294 записывается номер задачи, вызвавшей ошибку, а в слова A298 и A299 записывается адрес программы. Причину ошибки программы можно определить, руководствуясь следующей таблицей.

Адрес	Описание	Сохраняемая информация
A294	Если работа прекращается из-за ошибки программы, по этому адресу записывается тип задачи и номер задачи, на которой прекратилась работа. **Примечание.** Если в данном цикле активные циклические задачи отсутствуют, сюда будет записано значение FFFF Hex.	Циклическая задача: 0000...001F hex (циклические задачи 0...31) Задача обработки прерывания: 8000...80FF hex (задачи обработки прерываний 0...255)
A298/A299	Если работа прекратилась из-за ошибки программы, по этому адресу в двоичном коде записывается адрес точки программы, в которой прекратилась работа. **Примечание.** Если отсутствует команда END(001) (A295.11 = 1), записывается адрес, по которому предполагалось наличие END(001). **Примечание.** В случае ошибки выполнения задачи (A295.12 = 1) в A298/A299 записывается значение FFFFFFFF Hex.	A298: младшая часть адреса программы A299: старшая часть адреса программы

Примечание. Если устанавливается флаг ошибки (ER) или флаг ошибки доступа (AER), он воспринимается как ошибка программы, и его можно использовать для прекращения работы ЦПУ. Действия ЦПУ при возникновении ошибок программы определяются специальным параметром в настройках ПЛК.

Ошибка программы	Описание	Соответствующие флаги
Отсутствует команда END.	В программе отсутствует команда END.	Устанавливается флаг ошибки отсутствия END (A295.11).
Ошибка во время выполнения задачи	В цикле нет ни одной готовой к выполнению задачи. Задаче не назначена ни одна программа. Соблюдено условие выполнения для задачи обработки прерывания, но задача с соответствующим номером отсутствует.	Устанавливается флаг ошибки задачи (295.12).
Ошибка выполнения команды (флаг ER = 1) и в настройках ПЛК выбрано прекращение работы при ошибке команды.	При попытке выполнения команды обнаружен операнд, содержащий некорректные данные.	Если в настройках ПЛК выбрано прекращение работы при ошибке команды, устанавливаются флаг ER и флаг ошибки выполнения команды (A295.08).

Ошибка программы	Описание	Соответствующие флаги
Ошибка доступа (флаг AER = 1) и в настройках ПЛК выбрано прекращение работы при ошибке команды.	Произведено чтение или запись в область параметров. Произведена запись в отсутствующую область памяти. Произведена запись в область, предназначенную только для чтения. Значение, указанное косвенным адресом DM в режиме BCD, оказалось не в двоично-десятичном формате.	Если в настройках ПЛК выбрано прекращение работы при ошибке команды, устанавливаются флаг AER и флаг ошибки доступа (A295.10).
Ошибка косвенного адреса DM в режиме BCD и в настройках ПЛК выбрано прекращение работы при ошибке команды.	Значение, указанное косвенным адресом DM в режиме BCD, оказалось не в двоично-десятичном формате.	Если в настройках ПЛК выбрано прекращение работы при ошибке команды, устанавливаются флаг AER и флаг ошибки косвенного BCD-адреса DM (A295.09).
Ошибка переполнения адресов команд с различением фронтов	Во время онлайн-редактирования было вставлено или удалено более 131 072 команд с различением фронтов.	Устанавливается флаг ошибки переполнения команд с различением фронтов (A295.13).
Ошибка переполнения памяти пользователя (UM)	Предпринята попытка выполнения команды, данные которой выходят за последний адрес памяти пользователя (UM), выделенной под хранение программы.	Устанавливается флаг переполнения памяти пользователя (UM) (A295.15).
Ошибка недопустимой команды	Попытка выполнения невыполнимой команды.	Устанавливается флаг недопустимой команды (A295.14).

1-4 Функциональные блоки

Модули ЦПУ серии CP поддерживают применение функциональных блоков. Более подробную информацию о применении функциональных блоков можно найти в руководстве *CX-Programmer, версия 7.0 — Руководство по работе. Функциональные блоки* (W447).

1-4-1 Общие сведения и основные свойства

Программа CX-Programmer (начиная с версии 5.0) позволяет использовать функциональные блоки, соответствующие стандарту МЭК 61131-3. Эти функциональные блоки поддерживаются модулями ЦПУ серии CS/CJ (начиная с версии 3.0), а также модулями ЦПУ серии CP. Поддерживаются следующие функции.

- Программируя некоторые технологические процессы, пользователь может представлять их в виде схемы, состоящей из функциональных блоков.
- Программы (алгоритмы) функциональных блоков могут создаваться на языке релейно-контактных схем (LD) или на языке структурированного текста (ST) (см. примечание).
  - Применение языка РКС позволяет повторно использовать (копировать и вставлять) программы, созданные в более ранних версиях CX-Programmer (4.0 или ниже).
  - В то же время, на языке структурированного текста намного проще программировать различные математические операции, для которых использование операторов языка РКС затруднено.

Примечание. Язык структурированного текста (ST) — это язык высокого уровня, предназначенный для промышленных средств автоматизации (преимущественно, для ПЛК) и описанный в стандарте МЭК 61131-3. Язык структурированного текста, поддерживаемый в CX-Programmer, соответствует стандарту МЭК 61131-3.

- Переменные не требуется объявлять в тексте программы, что значительно упрощает создание блоков. Достаточно зарегистрировать переменные в таблице переменных.
  Более того, при вводе переменных в программу на языке LD или ST переменные могут регистрироваться автоматически. В создаваемую прикладную программу также можно вводить переменные, которые уже были ранее зарегистрированы в таблице переменных.
- Отдельный функциональный блок можно преобразовать в библиотечную функцию в виде отдельного файла, что упростит его последующее многократное применение при программировании стандартных операций.
- Проверку программы можно выполнить отдельно для одного функционального блока, чтобы убедиться в его работоспособности и пригодности для применения в качестве библиотечной функции.
- Программы, содержащие функциональные блоки (написанные на языке LD или ST), можно загружать и считывать точно так же, как стандартные программы, не имеющие в своем составе функциональных блоков. В то же время, если некоторая задача содержит функциональные блоки, ее невозможно загрузить в ПЛК отдельно от других задач (считывание возможно).
- Поддержка одномерных массивов переменных упрощает обработку данных во многих случаях применения.

Примечание. Стандарт МЭК 61131 был разработан Международной электротехнической комиссией (МЭК) в качестве международного стандарта для программируемых логических контроллеров (ПЛК). Стандарт состоит из семи отдельных частей. Языки программирования ПЛК описаны и детализированы в *Части 3. Языки программирования (МЭК 61131-3).*

- Функциональный блок (созданный на языке LD или ST) может быть вызван из другого функционального блока (созданного на языке LD или ST). Функциональные блоки можно вкладывать друг в друга (до 8 уровней вложения), при создании функциональных блоков языки LD и ST можно комбинировать произвольным образом.

1-4-2 Технические характеристики функциональных блоков

Характеристики, не нашедшие отражения в следующей таблице, см. в руководстве *CX-Programmer, версия 7.0 — Руководство по работе. Функциональные блоки* (W447).

Параметр	Характеристики
Номер модели	WS02-CXPC1-E-V6
Установочный диск	Компакт-диск

Параметр		Характеристики
Поддерживающие модули ЦПУ		Модули ЦПУ серии CP версии 1.0 и выше. Модули ЦПУ серии CS/CJ: CS1-H, CJ1-H и CJ1M версии 3.0 и выше. Тип устройстваТип ЦПУ • CS1G-H CS1G-CPU42H/43H/44H/45H • CS1H-H CS1H-CPU63H/64H/65H/66H/67H • CJ1G-H CJ1G-CPU42H/43H/44H/45H • CJ1H-H CJ1H-CPU65H/66H/67H • CJ1M CJ1M-CPU11/12/13/21/22/23
		Функциональные ограничения для серий CS/CJ/CP • В программах (определениях) функц. блоков не поддерживаются следующие команды: команды программных блоков (BPRG и BEND), команды подпрограмм (SBS, GSBS, RET, MCRO и SBN), команды перехода (JMP, CJP и CJPN), команды пошаговых программ (STEP и SNXT), команды с мгновенным обновлением (!), ОБНОВИТЬ ВХОДЫ/ВЫХОДЫ (IORF), 1 мс ТАЙМЕР (TMHH).
Совмести-мость с ПК	Компьютер	IBM PC/AT или совместимый
	ЦП	Pentium 133 МГц или выше для Windows 98, 98SE или NT 4.0 (SP6 или выше)
	Операционная система	Microsoft Windows 95, 98, 98SE, Me, 2000, XP или NT 4.0 (SP6 или выше)
	Память	Минимум 64 Мбайт для Windows 98, 98SE или NT 4.0 (SP6 или выше) Более подробно см. в руководстве *CX-Programmer, версия 7.0 — Руководство по работе* (W437).
	Жесткий диск	Минимум 100 Мбайт свободного места
	Монитор	Минимум SVGA (800 × 600 пиксел) **Примечание.** Следует использовать «мелкий шрифт» для отображения.
	Привод CD-ROM	Требуется как минимум 1 привод CD-ROM
	COM-порт	Требуется как минимум 1 порт RS-232C

Примечание. Язык структурированного текста (ST) соответствует стандарту МЭК 61131-3, но в программе CX-Programmer версии 5.0 поддерживаются только операторы присвоения, операторы выбора (операторы CASE и IF), операторы циклов (операторы FOR, WHILE, REPEAT и EXIT), операторы RETURN, операторы арифметических и логических операций, функции сравнения, числовые функции и комментарии.

1-4-3 Выходные файлы программы CX-Programmer

Файлы проекта (*.cxp)

И проекты, содержащие определения функциональных блоков, и проекты с экземплярами функциональных блоков сохраняются программой CX-Programmer в стандартные файлы проектов одного типа (*.cxp).

Содержание проекта показано на следующем рисунке. Создаваемые определения функциональных блоков сохраняются в папку, которая, вместе с папкой файлов программ, находится внутри папки соответствующего ПЛК.



Файлы библиотеки функциональных блоков (*.cxf)

Определение функционального блока, созданное в ходе разработки проекта в CX-Programmer, может быть сохранено в отдельный файл (1 определение = 1 файл), чтобы его можно было загружать и повторно использовать в других программах.

Примечание. Если некоторый функциональный блок содержит вложенные функциональные блоки, файл библиотеки этого функционального блока (.cxf) также будет содержать определения всех вложенных функциональных блоков.

Текстовые файлы проекта с функциональными блоками (*.cxt)

Данные, сохраняемые программой CX-Programmer в файлы проекта (*.cxp), также могут сохраняться в текстовые файлы CXT (*.cxt).

РАЗДЕЛ 2
Задачи

В данном разделе описаны основные принципы работы задач и поясняется применение задач при создании программ.

2-1 Программирование с использованием задач 56
2-1-1 Обзор 56
2-1-2 Задачи и программы 58
2-1-3 Основные принципы работы модуля ЦПУ 59
2-1-4 Типы задач 60
2-1-5 Условия выполнения задач и настройка 63
2-1-6 Возможные состояния циклической задачи 63
2-1-7 Переключение состояний 64
2-2 Применение задач 65
2-2-1 Команды ВКЛЮЧИТЬ ЗАДАЧУ и ВЫКЛЮЧИТЬ ЗАДАЧУ 65
2-2-2 Ограничения на применение команд в задачах 70
2-2-3 Специальные флаги задач 70
2-2-4 Примеры задач 73
2-2-5 Создание задач 74
2-2-6 Глобальные подпрограммы 76
2-3 Задачи обработки прерываний 77
2-3-1 Типы задач обработки прерываний 77
2-3-2 Флаги и слова задач обработки прерываний 82
2-3-3 Меры предосторожности при использовании 83
2-4 Использование CX-Programmer для задач 85

Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

2-1 Программирование с использованием задач

2-1-1 Обзор

Операции управления, реализуемые с помощью ПЛК серии CP, можно классифицировать по выполняемым функциям, по типу оборудования, по технологическим процессам, по разработчикам или по любому другому критерию. Каждую такую операцию можно запрограммировать в виде отдельного модуля, называемого «задачей». Создание программы из отдельных задач обладает следующими преимуществами:

1,2,3...

1. В создании программ могут участвовать одновременно несколько человек.

   Созданные по отдельности части программы можно очень легко объединить в единую программу.

2. Можно создавать стандартные программные модули.

   Ниже перечислены функции программы CX-Programmer для создания программ как автономных стандартных модулей, а не как узкоспециализированных программ, применимых только для определенных систем (машин, устройств). Это означает, что программы, разработанные по отдельности несколькими программистами, можно затем произвольно комбинировать, не изменяя тексты программ.

   - Программирование с использованием символов.
   - Определение глобальных и локальных символов.
   - Автоматическое назначение адресов локальным символам.

3. Повышается общее быстродействие.

   Общее быстродействие повышается, поскольку система состоит из общей управляющей программы и отдельных программ, причем выполняются только те программы, которые требуются в данный момент времени.

4. Упрощается проверка и отладка программ.

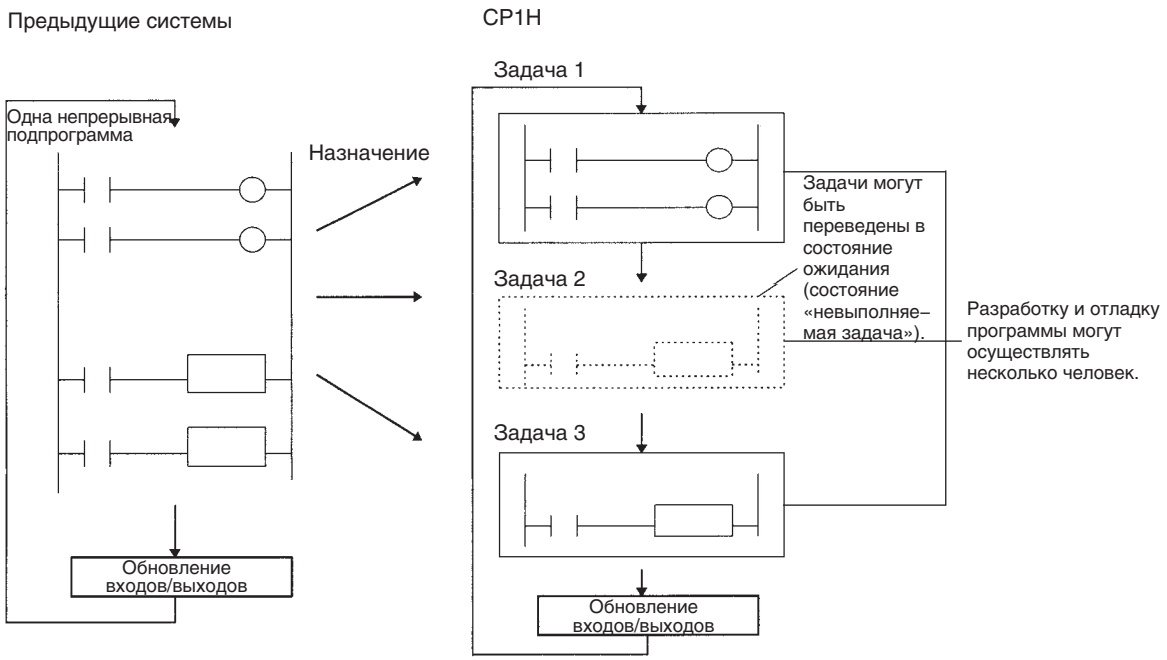
   - Скорость отладки программ значительно повышается, поскольку задачи разрабатываются отдельными программистами, соответственно проверка и отладка осуществляются отдельно для каждой задачи.
   - Дальнейшая модернизация также упрощается, поскольку изменяться будет только та задача, которая требует пересмотра.
   - Отладка упрощается и ускоряется еще и потому, что можно легко выяснить, является адрес локальным или глобальным. При этом адресное пространство, используемое программами, во время отладки требуется проверять только один раз, поскольку символы являются либо глобальными либо локальными и назначение адресов локальным символам производится автоматически с помощью CX-Programmer.

5. Легко переключаться между программами.

   Если в процессе управления технологическим процессом требуется перейти от одной операции к другой, для переключения между задачами (программами) можно использовать команду управления задачами.

6. Повышается наглядность прикладных программ.

   Программы состоят из отдельных блоков, благодаря чему они более наглядны и понятны. В частности, это упрощает применение команд перехода и других аналогичных команд.



Примечание. В отличие от прежнего подхода к выполнению программы, который можно сравнить со чтением непрерывной «ленты», выполнение программы в виде задач можно сравнить со чтением последовательности отдельных «карт».

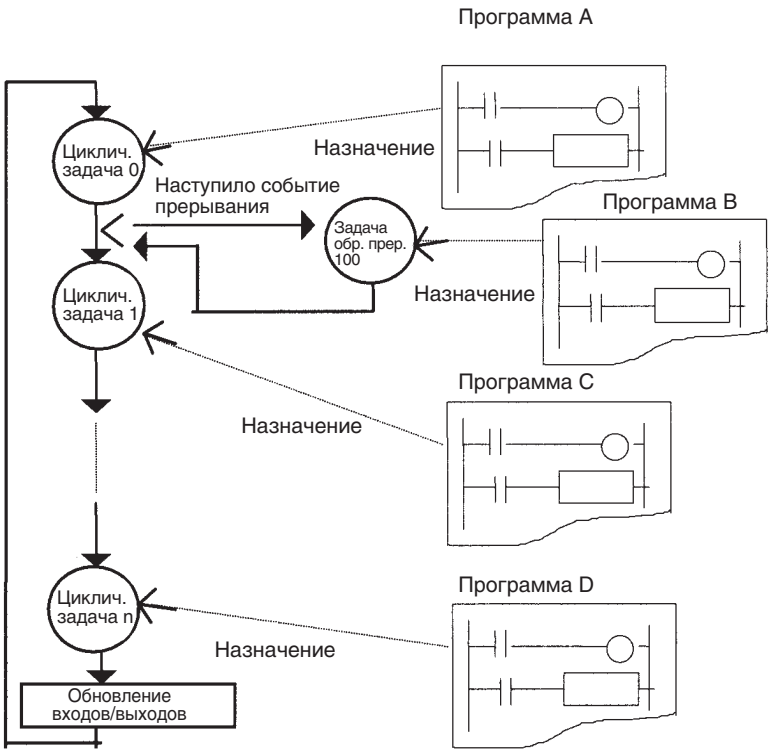
- Все «карты» считываются в заданной последовательности, начиная с «карты» с наименьшим номером.
- «Карты» делятся на активные и неактивные. Неактивные «карты» пропускаются («карты» активизируются и деактивизируются с помощью команд управления задачами).
- Активизированная «карта» остается активной и считывается в последующих циклах (последовательностях). Деактивизированная «карта» остается неактивной и пропускается до тех пор, пока она не активизируется повторно другой «картой».

2-1-2 Задачи и программы

Можно управлять максимум 288 программами (задачами). Одной задаче назначается одна программа. В широком смысле различают задачи следующих типов:

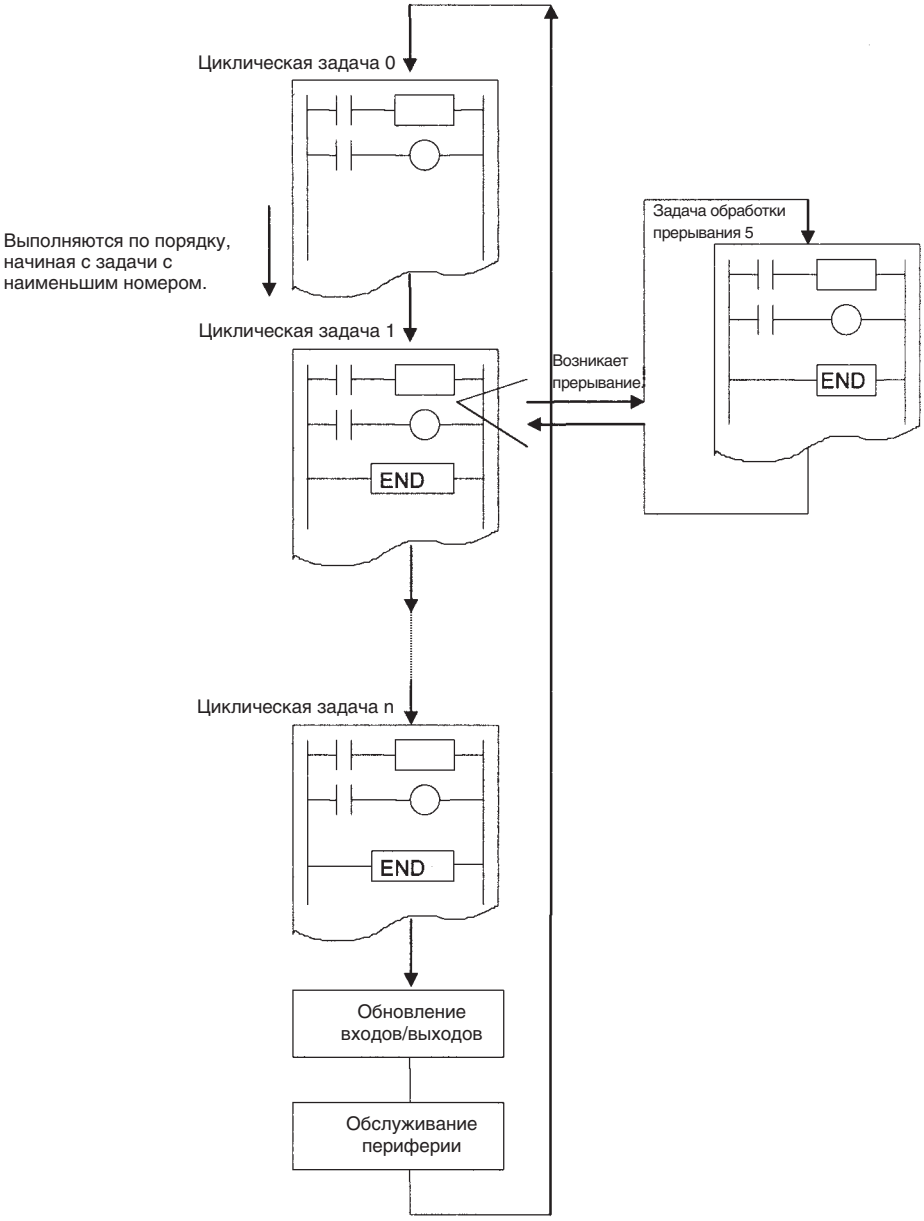
- задачи, выполняемые циклически (циклические задачи);
- задачи, выполняемые по прерыванию (задачи обработки прерываний).

Каждая программа, назначаемая какой-либо задаче, выполняется независимо от других программ, поэтому она должна завершаться командой END(001). Обновление входов/выходов производится только после того, как выполнены программы всех задач текущего цикла.



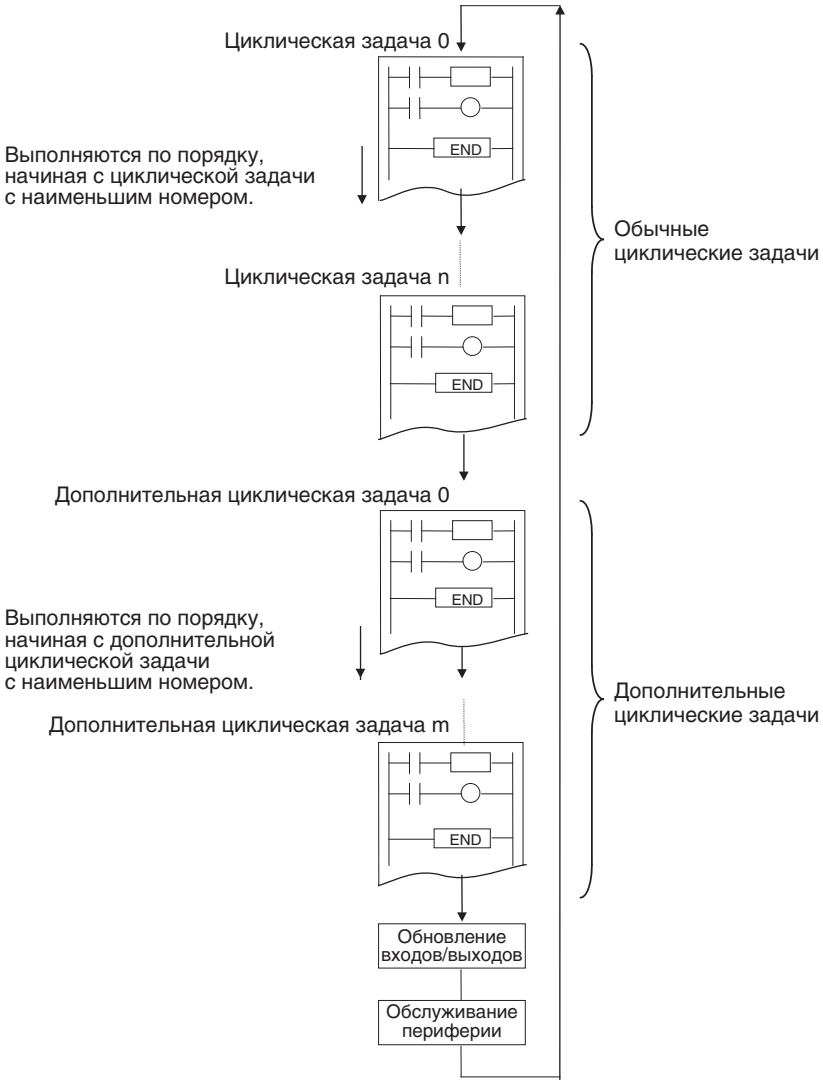
2-1-3 Основные принципы работы модуля ЦПУ

Модуль ЦПУ выполняет циклические задачи, начиная с задачи с наименьшим номером. Он также выполняет задачи обработки прерываний, если формируются соответствующие прерывания.



Примечание. В начале выполнения задачи обнуляются все флаги условий (ER, CY, «равенство», AER и т. п.), а также условия выполнения команд (включенные блокировки и т. п.). Поэтому, во-первых, флаги условий не могут использоваться совместно разными задачами, а во-вторых, пары команд БЛОКИРОВКА/ОТМЕНА БЛОКИРОВКИ (IL/ILC), ПЕРЕХОД/КОНЕЦ ПЕРЕХОДА (JMP/JME) или ВЫЗОВ ПОДПРОГРАММЫ/ВХОД В ПОДПРОГРАММУ (SBS/SBN) нельзя разносить между двумя разными задачами.

Задачи обработки прерываний могут выполняться в качестве циклических задач, если они запускаются командой TKON. В этом случае они называются «дополнительными циклическими задачами». Дополнительные циклические задачи (выполняемые по прерыванию задачи с номерами от 0 до 255) выполняются после того, как завершается выполнение обычных циклических задач (циклических задач с номерами от 0 до 31). Выполнение начинается с задачи с наименьшим номером.



2-1-4 Типы задач

В широком смысле задачи подразделяются на циклические задачи и задачи, выполняемые по прерыванию. Выполняемые по прерыванию задачи, в свою очередь, подразделяются на задачи обработки запланированных прерываний, прерываний от входов, прерываний от скоростных счетчиков и внешних прерываний. Кроме того, выполняемые по прерыванию задачи могут выполняться как дополнительные циклические задачи.

Циклические задачи

Циклические задачи, имеющие статус READY (готовность), выполняются один раз в каждом цикле (начиная с вершины программы до команды END(001)) в порядке возрастания номеров задач, начиная с задачи с наименьшим номером. Максимальное количество циклических задач: 32 (номера циклических задач: 00...31).

Задачи, выполняемые по прерыванию

Задача обработки прерывания выполняется, если возникает соответствующее прерывание, даже если в настоящий момент выполняется циклическая задача (в том числе дополнительная циклическая задача). Если условие формирования прерывания выполнено, задача обработки прерывания может быть выполнена в цикле в любое время, в том числе во время выполнения программы пользователя, во время обновления входов/выходов или обслуживания периферии.

Кроме того, выполняемые по прерыванию задачи, могут выполняться как дополнительные циклические задачи.

Прерывания от входов (прямой режим и режим счетчика)

Задача обработки прерывания вызывается по положительному или отрицательному фронту сигнала на соответствующем встроенном входе модуля ЦПУ (прямой режим) или по достижении заданного числа фронтов сигнала на соответствующем входе (режим счетчика).

Модель модуля ЦПУ		Количество задач	Номера задач обр. прерываний
CP1H	X или XA	8 задач максимум	140...147
	Y	6 задач максимум	140...145
CP1L	M (30 или 40 вх./вых.)	6 задач максимум	140...145
	L (20 вх./вых.)	6 задач максимум	140...145
	L (14 вх./вых.)	4 задачи максимум	140...143

Запланированные прерывания

Задача обработки запланированного прерывания выполняется через фиксированные промежутки времени, длительность которых определяется внутренним таймером модуля ЦПУ. В программе может использоваться только одна задача обработки запланированного прерывания (номер задачи обработки прерывания: 2).

Прерывания от скоростных счетчиков

Встроенный скоростной счетчик модуля ЦПУ ведет счет скоростных импульсов. По достижении заданного количества импульсов формируется прерывание.

Внешние прерывания (модули ЦПУ CP1H)

При возникновении внешнего прерывания может быть выполнена задача обработки прерывания, указанная пользователем (задачи с номерами от 0 до 255).

Задача обработки внешнего прерывания выполняется по запросу пользовательской программы специального модуля ввода/вывода серии CJ или модуля шины ЦПУ серии CJ.

В программе может быть использовано до 256 задач обработки внешних прерываний (номера задач обработки прерываний: от 0 до 255). Если номер задачи обработки внешнего прерывания совпадает с номером задачи обработки прерывания другого типа (запланированного, по входу или от скоростного счетчика), задача обработки прерывания будет выполнена по любому из этих прерываний (два условия объединяются по правилу ИЛИ), но в общем случае номера задач обработки прерываний дублировать не следует.

Дополнительные циклические задачи

Задачи обработки прерываний могут выполняться в каждом цикле так же, как обычные циклические задачи. Дополнительные циклические задачи (задачи обработки прерываний с номерами 0...255) выполняются, начиная с задачи с наименьшим номером, после завершения выполнения обычных циклических задач (циклических задач с номерами 0...31). Максимальное количество дополнительных циклических задач: 256 (номера задач обработки прерываний: 0...255). Однако циклические задачи обработки прерываний отличаются от обычных циклических задач в том, что они запускаются командой TKON (820). Они не могут запускаться автоматически в начале работы программы.

Если номер дополнительной циклической задачи совпадает с номером задачи обработки прерывания другого типа (запланированного, по входу или от скоростного счетчика), задача обработки прерывания будет выполнена по любому из этих прерываний (два условия объединяются по правилу ИЛИ). Однако не следует использовать задачи обработки прерываний одновременно в качестве обычных задач обработки прерываний и в качестве дополнительных циклических задач.

Примечание. (1) В пределах дополнительных циклических задач нельзя применять команды TKON (820) и TKOF(821), то есть из дополнительной циклической задачи нельзя запускать обычные циклические задачи и другие дополнительные циклической задачи.

(2) Различия между обычными циклическими задачами и дополнительными циклическими задачами отражены в следующей таблице.

Параметр	Дополнительные циклические задачи	Обычные циклические задачи
Вызов при запуске	Не поддерживается	Поддерживается (настраивается в CX-Programmer)
Использование TKON(820) и TKOF(821) внутри задачи	Не поддерживается	Поддерживается
Флаги задач	Не поддерживается	Поддерживается (циклические задачи 00...31 соответствуют флагам задач TK00...TK31).
Флаг первого выполнения задачи (A200.15) и флаг запуска задачи (A200.14)	Не поддерживается	Поддерживается
Значения регистров указателей (IR) и регистров данных (DR)	Не определены в начале задачи (как и у обычных задач обработки прерываний). Значения в начале каждого цикла не определены. Перед использованием регистров в них всегда следует записывать значения. Значения, установленные в предыдущем цикле, прочитать невозможно.	Не определены в начале работы. Значения, установленные в предшествующем цикле, могут быть прочитаны.

Предварительная версия
Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

2-1-5 Условия выполнения задач и настройка

В следующей таблице описаны условия выполнения задач, сопутствующие параметры и состояния.

Задача		Номер задачи	Условие выполнения	Сопутствующие параметры
Циклические задачи		0...31	Выполняются один раз в каждом цикле, если имеют статус READY (в зависимости от настройки, получают его автоматически в начале работы или из программы командой TKON (820)), когда получают право на выполнение.	Нет
Задачи обработки прерываний	Задача обработки запланированного прерывания 0	Задача обработки прерывания 2	Выполняется через фиксированные промежутки времени, отсчитываемые внутренним таймером модуля ЦПУ.	• С помощью команды MSKS(690) установлено время формирования запланированного прерывания (0...9999). • В настройках ПЛК выбраны единицы установки интервала формирования запланированного прерывания (10 мс, 1,0 мс или 0,1 мс).
	Задачи обработки прерываний от входов 0...7	Задачи обработки прерываний 140...147	Выполняются, когда включается или выключается соответствующий встроенный вход модуля ЦПУ.	• С помощью команды MSKS(690) отменены маски для назначенных входов.
	Задачи обработки прерываний от скоростных счетчиков	Задачи обработки прерываний 0...255	Выполняются, когда оказывается соблюдено условие сравнения с заданным значением или диапазоном, установленное для встроенного скоростного счетчика модуля ЦПУ.	
	Задачи обработки внешних прерываний (только CP1H)	Задачи обработки прерываний 0...255	Выполняются по запросу пользовательской программы специального модуля ввода/вывода или модуля шины ЦПУ.	Нет (всегда включены)
Дополнительные циклические задачи 0...255		Задачи обработки прерываний 0...255	Выполняются один раз в каждом цикле, если имеют статус READY (запущены командой TKON (820)), когда получают право на выполнение.	Нет (всегда включены)

2-1-6 Возможные состояния циклической задачи

В настоящем разделе описано понятие состояния (статуса) циклической задачи (в том числе дополнительной циклической задачи).

Любая циклическая задача всегда находится в одном из следующих четырех состояний: задача отключена (INI), задача готова к выполнению (READY), выполняемая задача (RUN) и ожидание (WAIT).

Отключенное состояние (INI)

В отключенном состоянии задача не выполняется. В режиме программирования (PROGRAM) все циклические задачи находятся в отключенном состоянии. Любая циклическая задача, переведенная из этого состояния в другое, не может возвратиться к этому состоянию без перехода в режим программирования.

Состояние готовности (READY)

С помощью атрибута задачи можно выбрать нужный момент переключения задачи в состояние готовности. В зависимости от настройки атрибута, задача будет активизироваться автоматически вместе с запуском режима выполнения (RUN), либо должна активизироваться из программы командой ВКЛЮЧИТЬ ЗАДАЧУ.

Активизация задач командой

Для перевода циклической задачи, для которой выбрана активизация командой, из отключенного состояния или состояния ожидания в состояние готовности служит команда ВКЛЮЧИТЬ ЗАДАЧУ (TKON(820)).

Автоматическая активизация задач

Циклическая задача, для которой выбрана автоматическая активизация при запуске программы, переходит из отключенного состояния в состояние готовности, когда ПЛК переключается из режима «Программирование» (PROGRAM) в режим «Выполнение» (RUN) или «Мониторинг» (MONITOR). Это относится только к обычным циклическим задачам.

Примечание. С помощью CX-Programmer можно выбрать одну или несколько задач, которые будут переходить в состояние готовности, когда начинается работа. Это можно сделать для задач с номерами 0...31. Для дополнительных циклических задач эта возможность недоступна.

Состояние выполнения (RUN)

Циклическая задача, находящаяся в состоянии готовности, переключится в состояние «выполняемая задача» (RUN) и будет выполнена, когда она получит право на выполнение.

Состояние ожидания (WAIT)

Для переключения циклической задачи из отключенного состояния в состояние ожидания предназначена команда ВЫКЛЮЧИТЬ ЗАДАЧУ (TKOF(821)).

Примечание. За текущим состоянием программ задач (т. е. выполняется задача или она остановлена) ПЛК серии CP можно наблюдать в режиме онлайн с помощью CX-Programmer. Состояния задач индицируются в CX-Programmer следующим образом.

- Running (выполнение): задача в состоянии READY или RUN (различить эти состояния невозможно).
- Stopped (выполнение остановлено): задача в состоянии INI или WAIT (различить эти состояния невозможно).

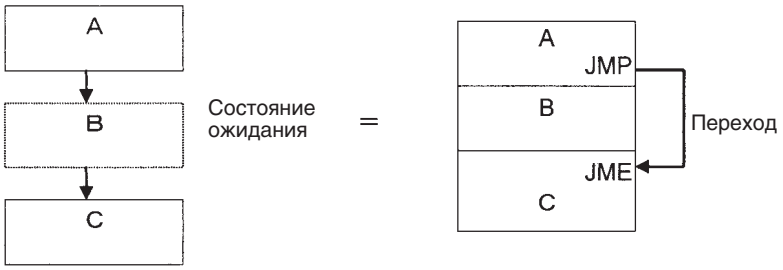
2-1-7 Переключение состояний



Примечание. (1) Активизация в начале работы возможна только для обычных циклических задач. Для дополнительных циклических задач эта возможность недоступна.

(2) Задача, находящаяся в состоянии выполнения, будет переведена в состояние ожидания командой TKOF(821), даже если команда TKOF(821) выполняется в пределах этой задачи.

Состояние «ожидание» эквивалентно действию команд перехода (JMP-JME). Задача в состоянии ожидания сохраняет состояния своих выходов.



В состоянии ожидания команды задачи не выполняются, поэтому общее время выполнения команд не возрастает. Если какие-либо сегменты программы требуется выполнять лишь время от времени, их можно создать в виде задач и переключать их в состояние ожидания с целью снижения длительности цикла.



Примечание. Состояние ожидания всего лишь означает, что эта задача будет пропущена при выполнении программы. Переход в состояние ожидания не приводит к завершению программы.

2-2 Применение задач

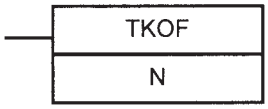
2-2-1 Команды ВКЛЮЧИТЬ ЗАДАЧУ и ВЫКЛЮЧИТЬ ЗАДАЧУ

Команды ВКЛЮЧИТЬ ЗАДАЧУ (TKON(820)) и ВЫКЛЮЧИТЬ ЗАДАЧУ (TKOF(821)) используются в программе для переключения циклических задач (в том числе дополнительных циклических задач) из состояния ожидания в состояние готовности и наоборот.



N: Номер задачи

Задача переходит в состояние READY (готовность), когда при включенном условии выполнения устанавливается соответствующий флаг задачи.



N: Номер задачи

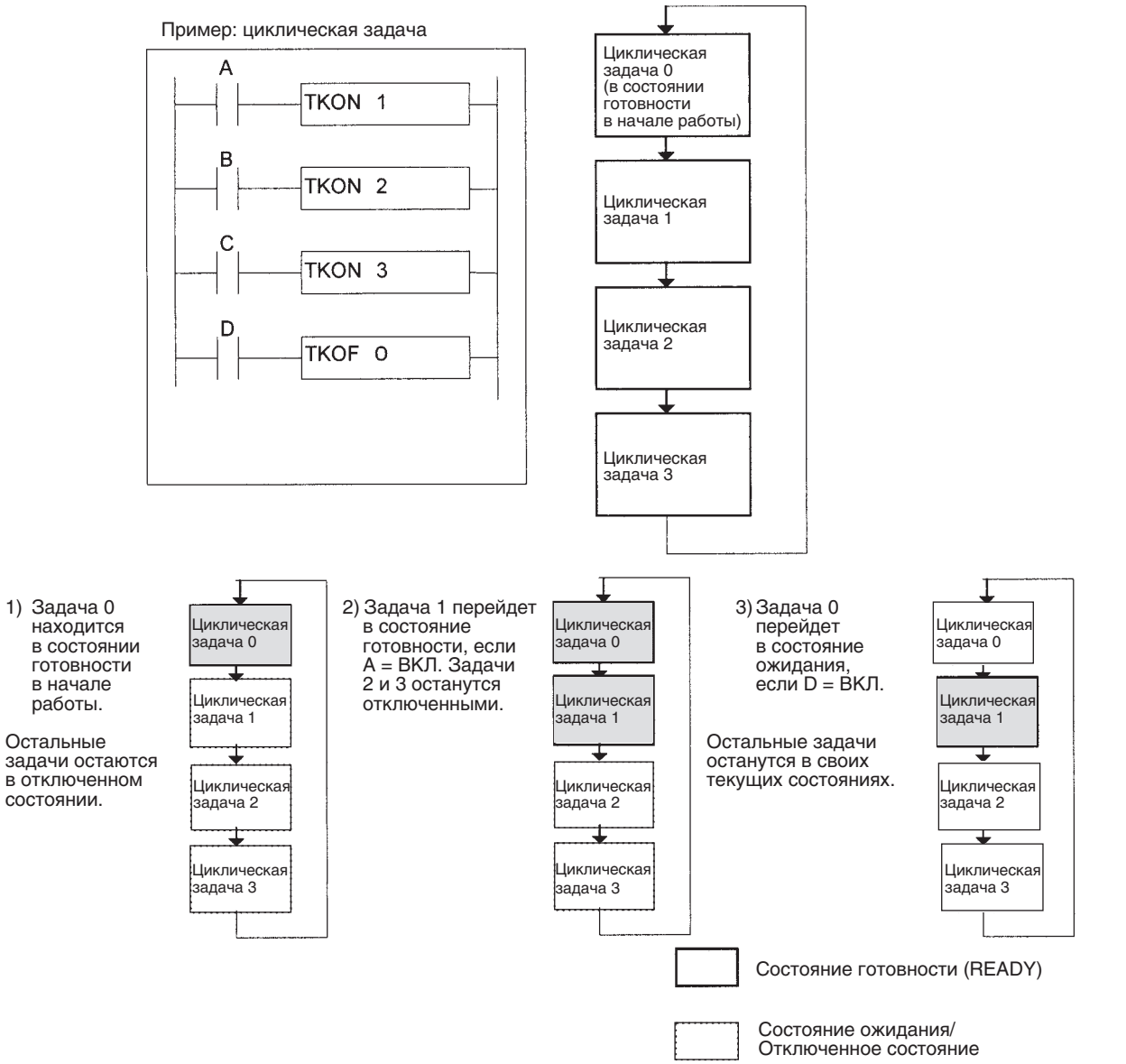
Задача переходит в состояние ожидания, когда при включенном условии выполнения сбрасывается соответствующий флаг задачи.

Команды ВКЛЮЧИТЬ ЗАДАЧУ и ВЫКЛЮЧИТЬ ЗАДАЧУ можно использовать в любое время для переключения любой циклической задачи (в том числе дополнительной) из состояния ожидания в состояние готовности и наоборот.

Циклическая задача, находящаяся в состоянии готовности, в последующих циклах остается в состоянии готовности, а циклическая задача, находящаяся в состоянии ожидания, в последующих циклах остается в состоянии ожидания.

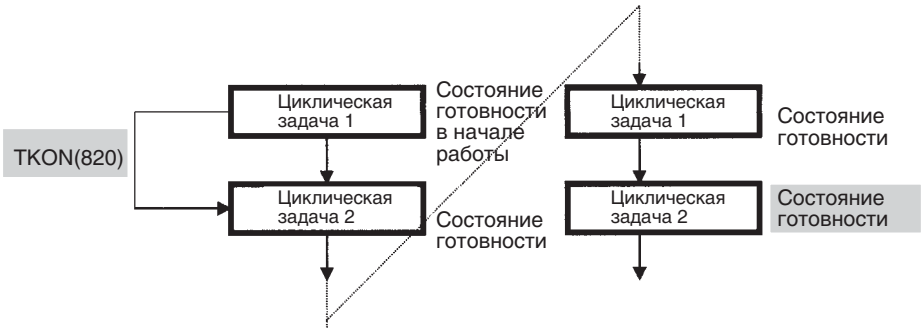
Команды ВКЛЮЧИТЬ ЗАДАЧУ и ВЫКЛЮЧИТЬ ЗАДАЧУ можно использовать только в циклических задачах. В задачах обработки прерываний их использовать нельзя.

Примечание. В каждом цикле хотя бы одна циклическая задача должна находиться в состоянии готовности. В противном случае установится флаг ошибки задачи (A295.12) и модуль ЦПУ прекратит выполнение программы.

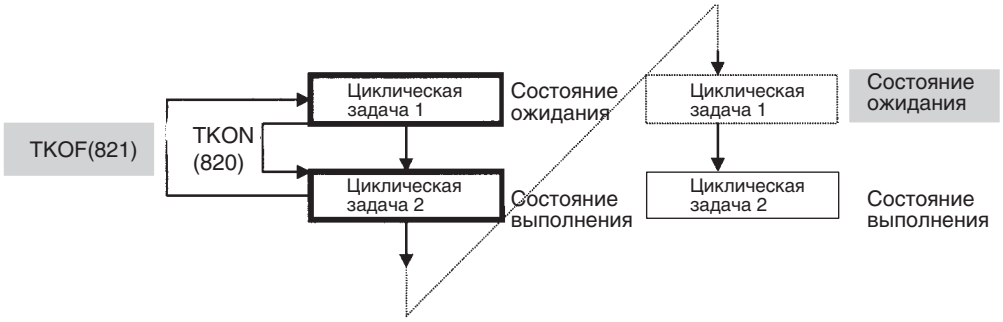


Задачи и цикл выполнения

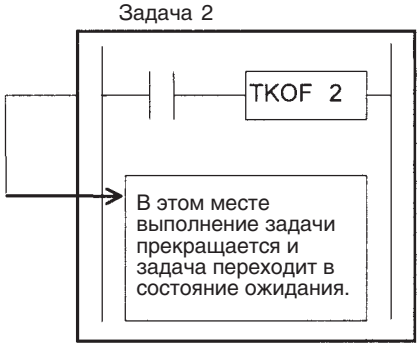
Циклическая задача (в том числе дополнительная циклическая задача), находящаяся в состоянии готовности, остается в этом состоянии в последующих циклах.



Циклическая задача, находящаяся в состоянии ожидания, остается в этом состоянии в последующих циклах. Для перевода задачи из состояния ожидания в состояние готовности задачу следует активизировать с помощью команды TKON(820).



Задача, внутри которой выполняется команда TKOF(821), прекращает выполняться в том месте, в котором была выполнена эта команда, и переходит в состояние ожидания.



Номера циклических задач и цикл выполнения (включая дополнительные циклические задачи)

Если задача m включает задачу n и m > n, задача n переходит в состояние готовности в следующем цикле.

Пример:если задача 5 включает задачу 2, задача 2 переходит в состояние готовности в следующем цикле.

Если задача m включает задачу n и m < n, задача n переходит в состояние готовности в том же цикле.

Пример:если задача 2 включает задачу 5, задача 5 переходит в состояние готовности в том же цикле.

Если задача m переводит задачу n в состояние ожидания и m > n, задача перейдет в ожидание в следующем цикле.

Пример:если задача 5 переводит задачу 2 в состояние ожидания, задача 2 переходит в состояние ожидания в следующем цикле.

Если задача m переводит задачу n в состояние ожидания и m < n, задача n переходит в ожидание в том же цикле.

Пример:если задача 2 переводит задачу 5 в состояние ожидания, задача 5 переходит в ожидание в том же цикле.

Взаимосвязь задач с памятью ввода/вывода

Существуют два разных способа применения регистров указателей (IR) и регистров данных (DR): 1) отдельно каждой задачей или 2) совместно всеми задачами.

В случае независимого использования регистров: регистр IR0, используемый, например, циклической задачей 1, не совпадает с регистром IR0, используемым циклической задачей 2. В случае совместного использования регистров: регистр IR0, используемый циклической задачей 1, совпадает с регистром IR0, используемым циклической задачей 2.

Раздельное или совместное использование регистров выбирается с помощью соответствующего параметра в CX-Programmer.

- Остальные слова и биты в памяти ввода/вывода используются совместно всеми задачами. Например, бит CIO 10.00 — это один и тот же бит как для циклической задачи 1, так и для циклической задачи 2. Следовательно, всякий раз, когда при программировании используется какая-либо из областей памяти ввода/вывода, кроме областей IR и DR, следует быть очень внимательным, поскольку изменение значений в одной задаче влияет и на другие задачи.

Память ввода/вывода	Взаимосвязь с задачами
CIO, вспомогательная область, память данных и все остальные области памяти, за исключением областей IR и DR.	Используются совместно всеми задачами.
Регистры указателей (IR) и регистры данных (DR) (см. примеч.)	Используются раздельно каждой задачей.

Примечание. В начале выполнения задач обработки прерываний (в том числе дополнительных циклических задач) значения регистров IR и DR не определены. Если в задачах обработки прерываний используются регистры IR и DR, в них должны записываться значения с помощью команд MOVR/MOVRW (ПЕРЕДАТЬ В РЕГИСТР и ПЕРЕДАТЬ ТЕКУЩЕЕ ЗНАЧЕНИЕ ТАЙМЕРА/СЧЕТЧИКА В РЕГИСТР) внутри самих задач обработки прерываний. После выполнения задачи обработки прерывания в регистры IR и DR возвращаются значения, которые находились в них до вызова задачи обработки прерывания. Это происходит автоматически.

Взаимосвязь задач с работой таймеров

Текущие значения таймеров TIM, TIMX, TIMH, TIMHX, TMHH, TMHHX, TIMW, TIMWX, TMHW и TMHWX с номерами T0000...T0015 продолжают обновляться, даже если происходит переключение задачи или задача, содержащая таймер, переходит в состояние ожидания или назад в состояние готовности.

Если задача, содержащая таймер TIM, переходит в состояние ожидания, а затем возвращается в состояние готовности, выполнение команды TIM при нулевом текущем значении приведет к установке флага завершения (флаги завершения таймеров обновляются только при выполнении команд). Если команда TIM была выполнена при ненулевом текущем значении, после переключения задачи в состояние ожидания текущее значение продолжает обновляться, как если бы задача находилась в состоянии готовности.

- Текущие значения таймеров с номерами T0016...T4095 не изменяются, пока задача находится в состоянии ожидания.

Взаимосвязь задач с флагами условий

Перед выполнением каждой задачи все флаги условий обнуляются. Следовательно, состояние флага условия, наблюдавшееся в конце задачи 1, невозможно прочитать в задаче 2. Однако для чтения состояний флагов условий из другого участка программы, например из другой задачи, можно использовать команды CCS(282) и CCL(283).

2-2-2 Ограничения на применение команд в задачах

Парные команды в пределах одной задачи

Ниже перечислены парные команды, которые всегда должны находиться в пределах одной задачи. Если такие команды оказываются в двух разных задачах, устанавливается флаг ER и команды не выполняются.

Мнемоническая форма	Команда
JMP/JME	ПЕРЕХОД/КОНЕЦ ПЕРЕХОДА
CJP/JME	УСЛОВНЫЙ ПЕРЕХОД/КОНЕЦ ПЕРЕХОДА
CJPN/JME	УСЛОВНЫЙ ПЕРЕХОД НЕ/КОНЕЦ УСЛОВНОГО ПЕРЕХОДА
JMP0/JME0	МНОЖЕСТВЕННЫЙ ПЕРЕХОД/КОНЕЦ ПЕРЕХОДА
FOR/NEXT	FOR/NEXT
IL/ILC	БЛОКИРОВКА/ОТМЕНА БЛОКИРОВКИ
SBS/SBN/RET	ВЫЗОВ ПОДПРОГРАММЫ/ВХОД В ПОДПРОГРАММУ/ВЫХОД ИЗ ПОДПРОГРАММЫ
MCRO/SBN/RET	МАКРОС/ВХОД В ПОДПРОГРАММУ/ВЫХОД ИЗ ПОДПРОГРАММЫ
BPRG/BEND	НАЧАЛО ПРОГРАММНОГО БЛОКА/КОНЕЦ ПРОГРАММНОГО БЛОКА
STEP /SNXT	ОПРЕДЕЛИТЬ ШАГ

Команды, недоступные для задач обработки прерываний

Ниже перечислены команды, которые нельзя размещать в задачах обработки прерываний. Любая попытка выполнения такой команды в задаче обработки прерывания приведет к включению флага ER, а команда выполнена не будет. Эти команды можно использовать в задаче обработки прерывания, только если последняя используется в качестве дополнительной задачи.

Мнемоническая форма	Команда
TKON(820)	ВКЛЮЧИТЬ ЗАДАЧУ
TKOF(821)	ВЫКЛЮЧИТЬ ЗАДАЧУ
STEP	ОПРЕДЕЛИТЬ ШАГ
SNXT	СЛЕДУЮЩИЙ ШАГ
STUP	ИЗМЕНИТЬ ПАРАМЕТРЫ ПОСЛЕД. ПОРТА
DI	ЗАПРЕТИТЬ ПРЕРЫВАНИЯ
EI	РАЗРЕШИТЬ ПРЕРЫВАНИЯ

Выполнение следующих команд непредсказуемо в случае их выполнения внутри задач обработки прерываний: ТАЙМЕР: TIM и TIMX(550); СКОРОСТНОЙ ТАЙМЕР: TIMH(015) и TIMHX(551); 1 мс ТАЙМЕР: TMHH(540) и TMHHX(552); НАКАПЛИВАЮЩИЙ ТАЙМЕР: TTIM(087) и TTIMX(555); МНОГОВЫХОДНОЙ ТАЙМЕР: MTIM(543) и MTIMX(554); ДОЛГОСРОЧНЫЙ ТАЙМЕР: TIML(542) и TIMLX(553); ТАЙМЕР ОЖИДАНИЯ: TIMW(813) и TIMWX(816); СКОРОСТНОЙ ТАЙМЕР ОЖИДАНИЯ: TMHW(815) и TMHWX(817); ПИД-РЕГУЛЯТОР: PID(190); ОБНАРУЖЕНИЕ НЕИСПРАВНОГО КАНАЛА: FPD(269); и ИЗМЕНИТЬ ПАРАМЕТРЫ ПОСЛЕД. ПОРТА: STUP(237).

2-2-3 Специальные флаги задач

Флаги циклических задач

Описанные ниже флаги работают только с обычными циклическими задачами. С дополнительными циклическими задачами они не работают.

Флаги задач (TK00...TK31)

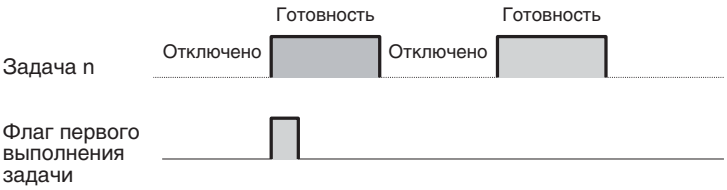
Флаг задачи включен, пока циклическая задача находится в состоянии готовности (READY), и выключен, когда задача находится в отключенном состоянии (INI) или в состояние ожидания (WAIT). Номера задач 00...31 соответствуют флагам задач TK00...TK31.



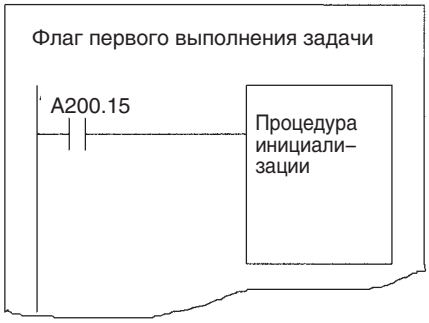
Примечание. Флаги задач используются только с циклическими задачами и не используются с задачами обработки прерываний. Если после начала работы выполняется задача обработки прерывания, то устанавливается бит A441.15, а в биты A441.00...A441.07 (2 разряда, hex) записывается номер задачи обработки прерывания, выполнение которой заняло наибольшее время.

Флаг первого выполнения задачи (A200.15)

Флаг первого выполнения задачи устанавливается, когда циклические задачи переходят из отключенного состояния в состояние готовности, получают право на выполнение и выполняются впервые. Флаг сбрасывается, когда первое выполнение задач завершено.



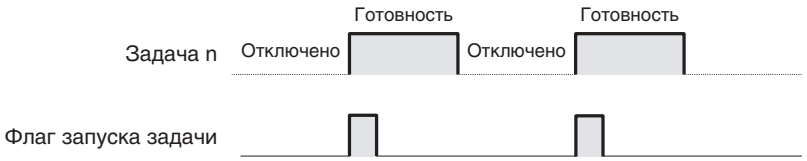
Флаг первого выполнения задачи информирует о том, что циклические задачи выполняются впервые. Следовательно, этот флаг можно использовать для выполнения процедуры инициализации в задачах.



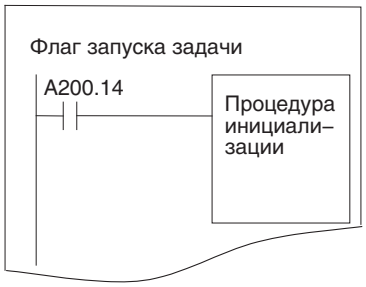
Примечание. Даже если циклическая задача возвращается из режима ожидания в состояние готовности с помощью команды TKON(820), это не считается первым выполнением и флаг первого выполнения задачи (A200.15) не установится. Флаг первого выполнения задачи (A200.15) также не устанавливается, если циклическая задача переводится из отключенного состояния в состояние выполнения, или если она переводится в состояние ожидания другой задачей посредством команды TKOF(821), прежде чем она получает право на выполнение.

Флаг запуска задачи (A200.14)

Флаг запуска задачи можно использовать для выполнения процедуры инициализации при запуске циклической задачи. Флаг запуска задачи сбрасывается всякий раз, когда циклическая задача переходит из отключенного состояния (INI) или состояния ожидания (WAIT) в состояние готовности (READY) (в отличие от флага первого выполнения задачи, который включается только тогда, когда отключенное состояние (INI) сменяется состоянием готовности (READY)).



Флаг запуска задачи можно использовать для выполнения процедуры инициализации всякий раз, когда задача переходит из состояния ожидания в состояние выполнения, то есть когда задача, находящаяся в режиме ожидания, активизируется командой TRON(820).



Флаги, относящиеся ко всем задачам

Флаг ошибки задачи (A295.12)

Флаг ошибки задачи включается, если происходит одна из следующих ошибок задачи.

- В пределах цикла отсутствуют циклические задачи (включая дополнительные циклические задачи) в состоянии готовности.
- Программа, назначенная циклической задаче (в том числе дополнительной циклической задаче), не существует (в случае использования CX-Programmer этой ситуации не произойдет).
- Вызванной задаче обработки прерывания не назначена программа.

Номер задачи при остановке программы (A294)

Если выполнение задачи прекращается из-за ошибки программы, в указанные ниже слова записываются тип задачи и номер текущей задачи.

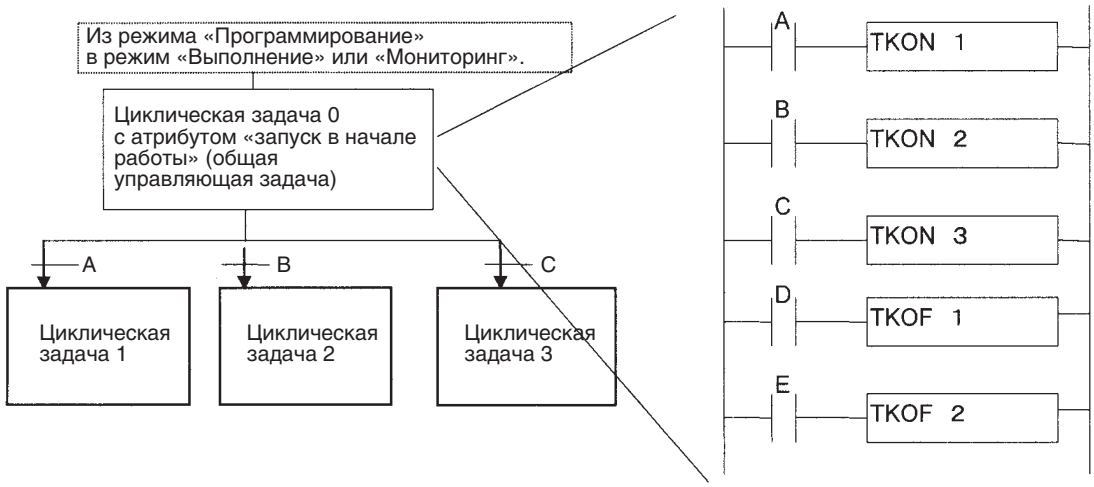
Тип	A294
Циклическая задача	0000...001F hex (соответствуют номерам задач 0...31)
Задача обработки прерывания	8000...80FF hex (соответствуют номерам задач обработки прерываний 0...255)

С помощью этой информации упрощается поиск места в программе, вызвавшего критическую ошибку. После сброса критической ошибки эта информация обнуляется. Адрес программы, по которому прекратилось выполнение задачи, сохраняется в слова A298 (младшие биты адреса программы) и A299 (старшие биты адреса программы).

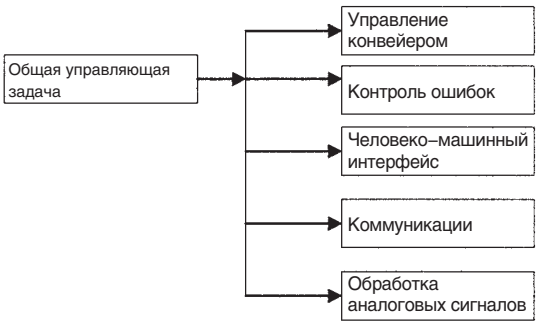
Предварительная версия
Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

2-2-4 Примеры задач

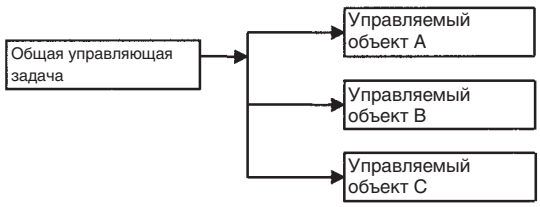
Общая управляющая задача, которая переходит в состояние готовности в начале работы, в общем случае служит для управления состояниями готовности/ожидания всех остальных циклических задач (включая дополнительные циклические задачи). Разумеется, любая циклическая задача может управлять состояниями готовности/ожидания любой другой циклической задачи, если это требуется в конкретном случае.



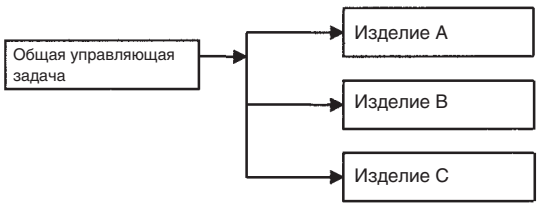
Классификация задач по функциям



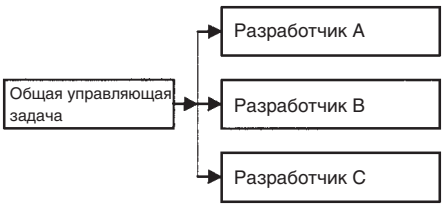
Классификация задач по объектам управления



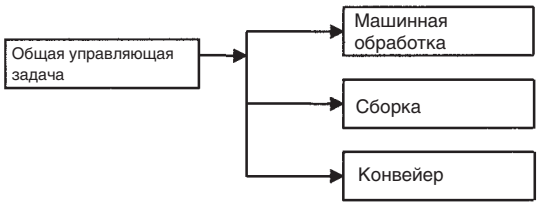
Классификация задач по изделиям



Классификация задач по разработчикам



Классификация задач по процессам



Можно также комбинировать перечисленные выше способы классификации, например, классифицировать задачи по функциям и технологическим процессам.

2-2-5 Создание задач

Ниже мы приводим свои рекомендации разработчикам задач.

1,2,3... 1. При разработке отдельных задач рекомендуется использовать следующие стандартные подходы.

a. Проанализируйте отдельные условия выполнения и невыполнения задач.

b. Проанализируйте условия присутствия или отсутствия сигналов на внешних входах/выходах.

c. Проанализируйте функции.

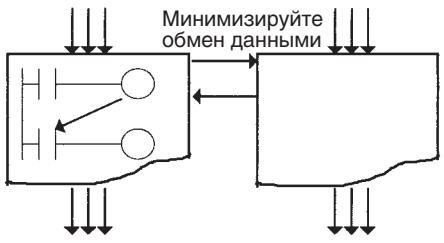
Старайтесь сводить к минимуму обмен данными между задачами дискретного управления, аналогового управления, человеко-машинного интерфейса, обработки ошибок и других процессов, сохраняя автономность задач, насколько это возможно.

d. Оцените порядок выполнения задач с точки зрения их приоритетности.

Разделите задачи на циклические задачи и задачи обработки прерываний.



2. Старайтесь распределять функции и разрабатывать программы таким образом, чтобы обеспечивалась автономность отдельных задач, а обмен данными между отдельными задачами (программами) был минимальным.



3. В общем случае для управления состояниями готовности/ожидания других задач следует использовать общую управляющую задачу.

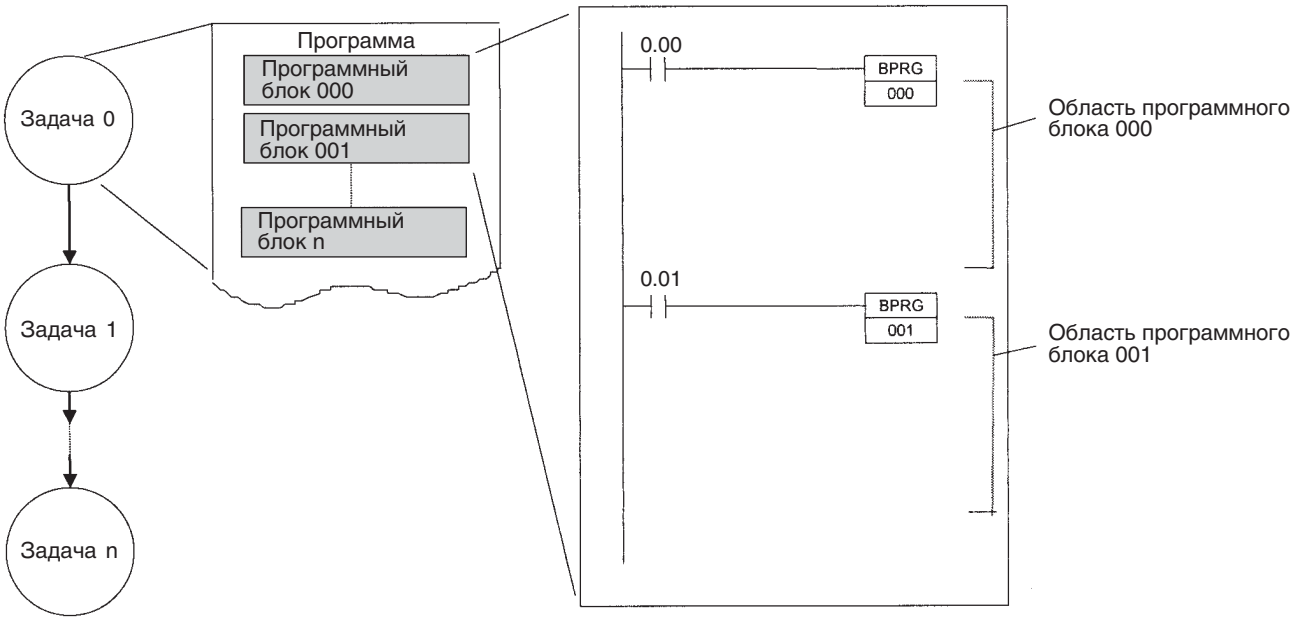
4. Назначайте наименьшие номера задачам, обладающим наибольшей важностью.
Пример: присвойте управляющей задаче наименьший номер по сравнению с прочими задачами обработки.

5. Назначайте наименьшие номера задачам обработки прерываний, обладающим наибольшей важностью.
6. Задача, находящаяся в состоянии готовности, будет выполняться в последующих циклах, пока она не будет переведена в состояние ожидания другой задачей или не переведет себя сама. Если предусматривается ветвление между задачами, в других задачах обязательно следует предусматривать команду TKOF(821) (ВЫКЛЮЧИТЬ ЗАДАЧУ).
7. Для инициализации задач в условиях выполнения команд следует использовать флаг первого выполнения задачи (A200.15) или флаг запуска задачи (A200.14). Флаг первого выполнения задачи будет включен во время первого выполнения каждой задачи. Флаг запуска задачи включается каждый раз, когда задача переходит в состояние готовности.
8. Распределяя память ввода/вывода, выделите в ней область, используемую совместно всеми задачами, и область, которая будет использоваться отдельными задачами. Последнюю разбейте на отдельные области для каждой задачи.

Взаимосвязь между задачами и программными блоками

В сумме во всех задачах может быть создано до 128 программных блоков. Управление выполнением каждого отдельного программного блока производится из лестничной диаграммы (РКС), но команды внутри программного блока записываются в мнемонической форме. Другими словами, программный блок является комбинацией РКС-символов и мнемонических обозначений.

Применение программных блоков облегчает создание лестничной диаграммы в таких случаях, как ветвление по условиям или шаговое выполнение процессов, которое довольно сложно программировать на языке релейно-контактных схем. Программные блоки располагаются в нижней части иерархии программ, при этом объемные программные модули, представляемые задачей, могут быть разбиты на отдельные программные блоки небольшого размера, запускаемые одним общим условием выполнения (в состоянии «1»).



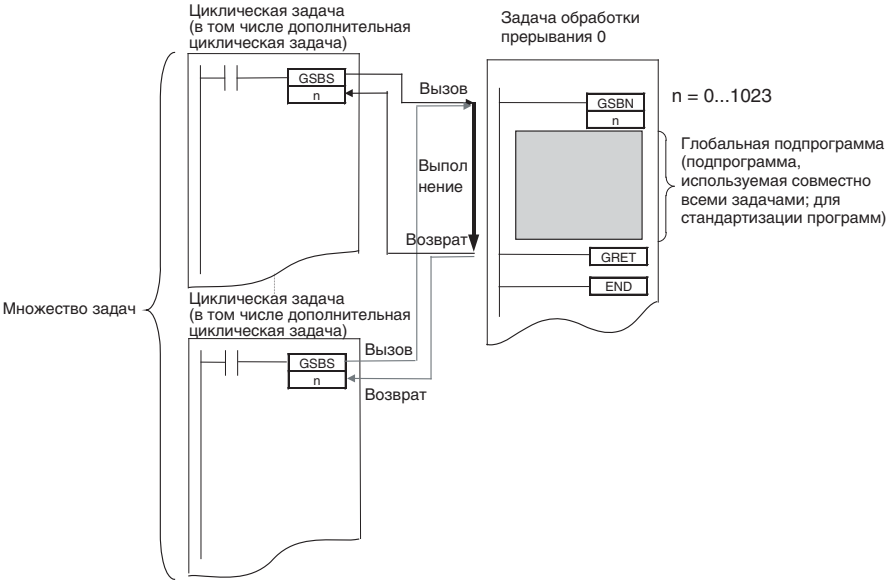
Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

2-2-6 Глобальные подпрограммы

Подпрограмма, используемая в одной задаче, не может быть вызвана из других задач. Однако в задаче обработки прерывания 0 можно создавать глобальные подпрограммы. Такие подпрограммы можно вызывать из циклических задач (в том числе из дополнительных циклических задач).

Для вызова глобальной подпрограммы служит команда GSBS(750). Номер подпрограммы должен находиться в пределах от 0 до 255. Глобальная подпрограмма записывается в конце задачи обработки прерывания с номером 0 (непосредственно перед END(001)) и размещается между командами GSBN(751) и GRET(752).

Глобальные подпрограммы можно использовать для создания библиотеки стандартных программных сегментов, которые можно вызывать по мере необходимости.



2-3 Задачи обработки прерываний

2-3-1 Типы задач обработки прерываний

Список задач обработки прерываний

Тип		Номер задачи	Условие выполнения	Способ настройки	Количество прерываний	Примеры применения
Прерывания от входов	CP1H X/XA: 0...7	140...147	Прерывание формируется по включению или выключению встроенного входа прерывания модуля ЦПУ (прямой режим) или по достижении заданного количества положительных или отрицательных фронтов сигнала на входе прерывания (режим счетчика).	С помощью команды УСТАНОВИТЬ МАСКУ ПРЕРЫВАНИЙ (MSKS(690)) следует разрешить работу требуемых входов прерывания.	8 каналов	Ускорение реакции системы на изменение состояний на определенных входах.
	CP1H Y: 0...5	140...145			6 каналов	
	CP1L M или L (20 вх./вых.): 0...5	140...145			6 каналов	
	CP1L L (14 вх./вых.): 0...3	140...143			4 канала	
Прерывания от скоростного счетчика		0...255	Прерывание формируется по выполнению условия сравнения текущего значения скоростного счетчика с заданным значением или диапазоном.	С помощью команды ЗАГРУЗКА ТАБЛИЦЫ СРАВНЕНИЯ (CTBL(882)) следует указать условие выполнения и номер прерывания.	256 каналов	Подсчет импульсов энкодера для выполнения операций позиционирования.
Запланированное прерывание 0		2	Прерывания формируются через фиксированные интервалы времени.	С помощью команды УСТАНОВИТЬ МАСКУ ПРЕРЫВАНИЙ (MSKS(690)) следует задать интервал формирования прерываний. См. параметр *Интервал запланированных прерываний* в области настроек ПЛК.	1 канал	Контроль рабочего состояния с фиксированной периодичностью.
Внешние прерывания (не поддерживаются модулями ЦПУ CP1L)		0...255	Прерывания вызываются специальным модулем ввода/вывода или модулем шины ЦПУ.	Нет (всегда включены)	256 каналов	Выполнение операций, запрашиваемых специальными модулями ввода/вывода серии CJ.

Задачи обработки прерываний от входов

В начале выполнения циклических задач выполнение задач обработки прерываний от входов по умолчанию выключено. Чтобы разрешить прерывания от входов, в пределах циклической задачи следует выполнить команду УСТАНОВИТЬ МАСКУ ПРЕРЫВАНИЙ (MSKS(690)), указав номер требуемого прерывания.

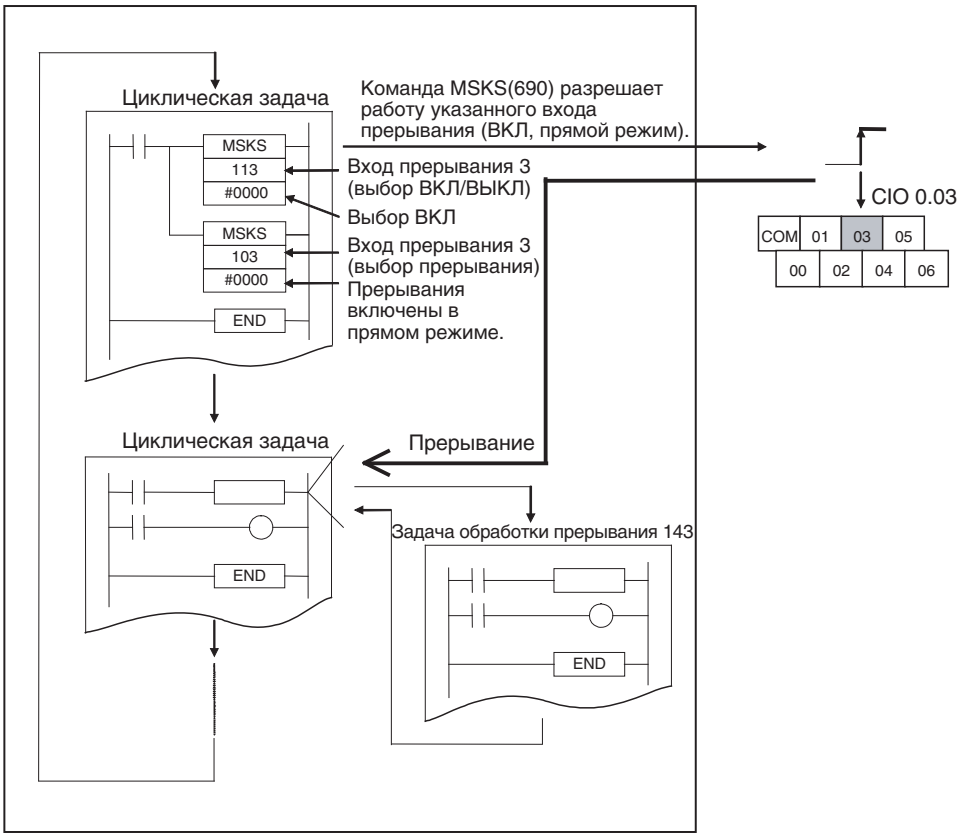
В настройках ПЛК следует заранее разрешить использование входов в качестве входов прерывания.

Примечание. Не следует разрешать ненужные задачи обработки прерываний от входов. Если вход прерывания сработает из-за помехи, а задача

Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

обработки прерывания с соответствующим номером будет отсутствовать, произойдет критическая ошибка (ошибка задачи), которая остановит выполнение программы.

Пример: в приведенном ниже примере задача обработки прерывания номер 143 выполняется, когда включается бит CIO 0.03 (вход прерывания номер 3).



• Модули ЦПУ CP1H X и XA

Вход прерывания	Номер входа прерывания	Номер задачи обработки прерывания
CIO 0.00	0	140
CIO 0.01	1	141
CIO 0.02	2	142
CIO 0.03	3	143
CIO 1.00	4	144
CIO 1.01	5	145
CIO 1.02	6	146
CIO 1.03	7	147

• Модули ЦПУ CP1H Y

Вход прерывания	Номер входа прерывания	Номер задачи обработки прерывания
CIO 0.00	0	140
CIO 0.01	1	141
CIO 1.00	2	142
CIO 1.01	3	143
CIO 1.02	4	144
CIO 1.03	5	145

• Модули ЦПУ CP1L M и L (20 вх./вых.)

Вход прерывания	Номер входа прерывания	Номер задачи обработки прерывания
CIO 0.04	0	140
CIO 0.05	1	141
CIO 0.06	2	142
CIO 0.07	3	143
CIO 0.08	4	144
CIO 0.09	5	145

• Модули ЦПУ CP1L L (14 вх./вых.)

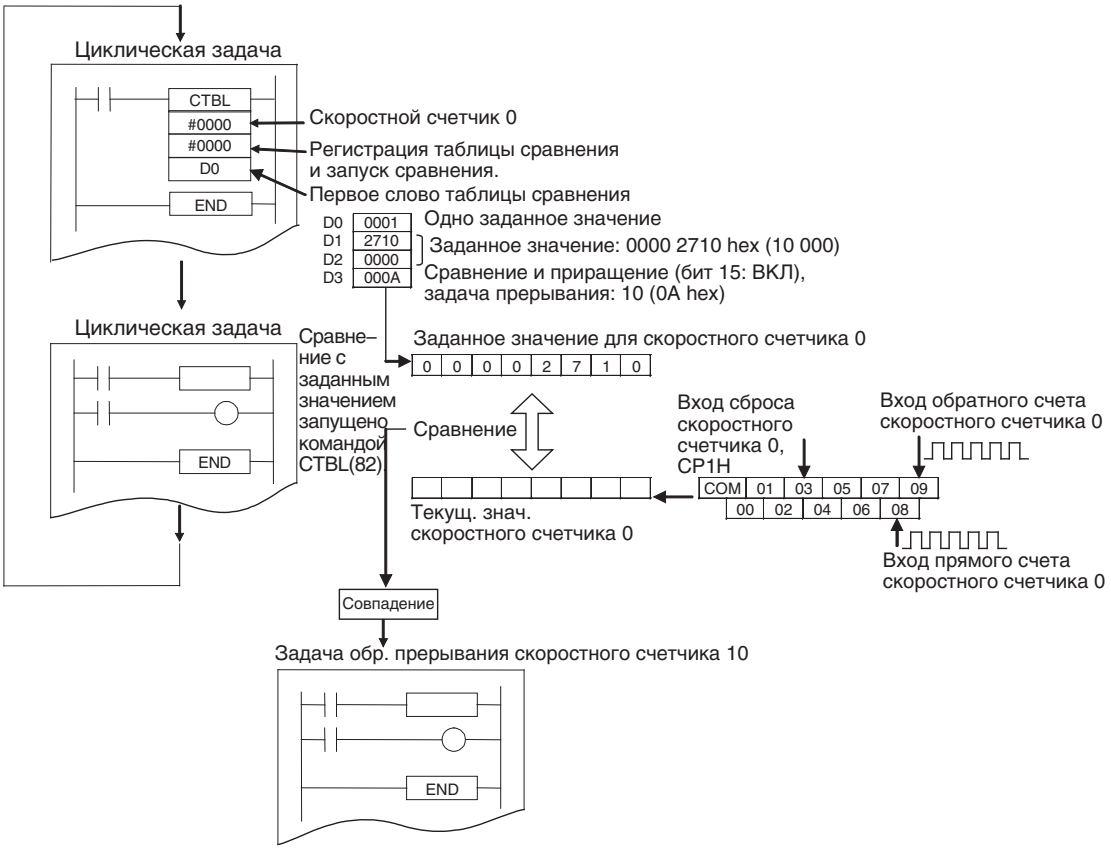
Вход прерывания	Номер входа прерывания	Номер задачи обработки прерывания
CIO 0.04	0	140
CIO 0.05	1	141
CIO 0.06	2	142
CIO 0.07	3	143

Задачи обработки прерываний от скоростных счетчиков: задачи 0...255

Для того чтобы разрешить выполнение задач обработки прерываний от скоростных счетчиков, следует выполнить команду ЗАГРУЗКА ТАБЛИЦЫ СРАВНЕНИЯ (CTBL(882)), указав в ней условие выполнения и номер прерывания. В качестве условия выполнения можно указать совпадение с заданным значением или попадание в заданный диапазон значений.

Пример

В приведенном ниже примере задача обработки прерывания от скоростного счетчика (задача номер 10) выполняется, когда текущее значение скоростного счетчика 0 достигает заданного значения при счете в прямом направлении.



Задача обработки запланированного прерывания: задача 2

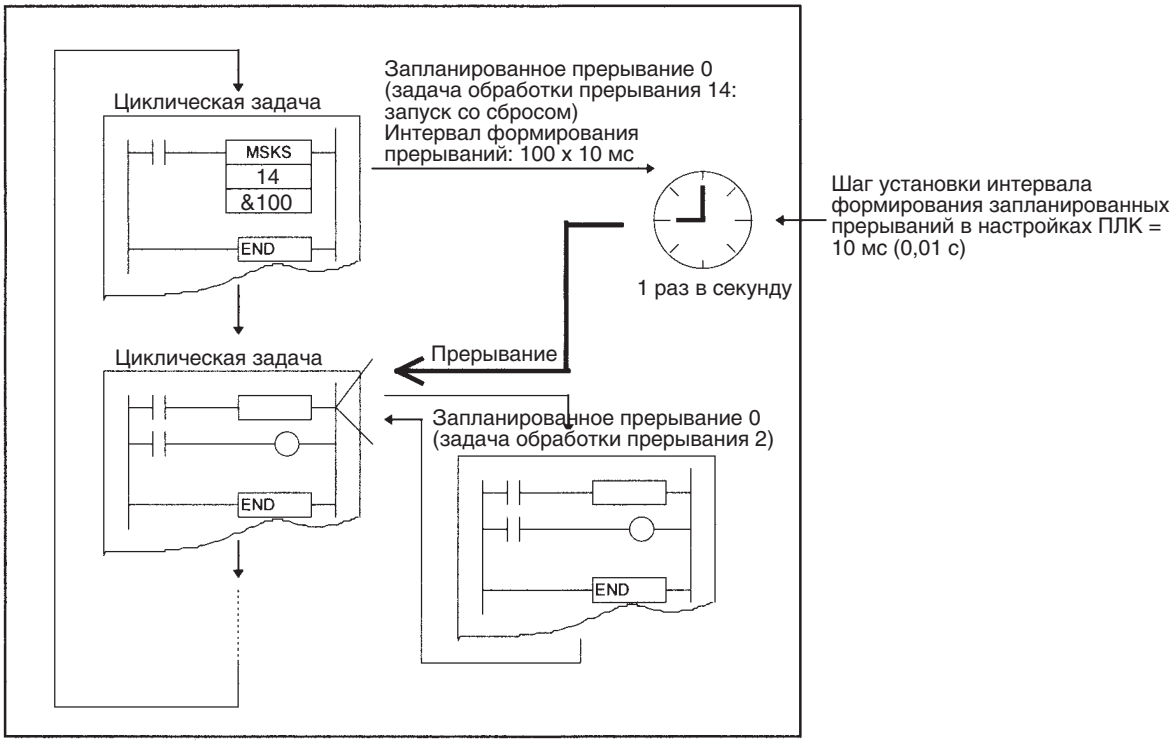
В начале выполнения циклических задач выполнение задачи обработки запланированного прерывания в настройках ПЛК по умолчанию выключено. Чтобы разрешить выполнение задачи обработки запланированного прерывания, необходимо выполнить следующие действия.

1,2,3...

1. Выполните в циклической задаче команду УСТАНОВИТЬ МАСКУ ПРЕРЫВАНИЙ (MSKS(690)) и задайте время (период) для указанного запланированного прерывания.
2. Задайте параметр *Интервал запланированных прерываний* в области настроек ПЛК.

Примечание. Значение интервала формирования прерываний влияет на выполнение циклических задач: чем короче интервал формирования прерываний, тем чаще выполняются задачи и тем больше длительность цикла.

Пример: в приведенном ноже примере задача обработки запланированного прерывания выполняется с периодичностью 1 раз в секунду.



Номера запланированных прерываний и номера задач обработки прерываний

Номер запланированного прерывания	Задачи обработки прерывания
0	2

Параметры в настройках ПЛК

Выберите для параметра *Scheduled interrupt interval (Интервал запланированных прерываний)* на вкладке Timings (Параметры времени) окна PLC Setup (Настройки ПЛК) одно из возможных значений: 0,1 / 1,0 / 10 мс.

Название	Значение
Интервал запланированных прерываний	10 мс (по умолч.)
	1,0 мс
	0,1 мс

Задачи обработки внешних прерываний: задачи 0...255

Задачи обработки внешних прерываний могут выполняться в любое время. Обработка прерывания выполняется в модуле ЦПУ в ПЛК, в состав которого входит специальный модуль ввода/вывода серии CJ или модуль шины ЦПУ серии CJ. Никакие параметры в модуле ЦПУ настраивать не требуется. Требуется лишь создать программу для указанной задачи обработки прерывания и загрузить ее в модуль ЦПУ.

Примечание. Если номер задачи обработки внешнего прерывания (0...255) совпадает с номером задачи обработки прерывания другого типа (запланированного (2), по входу (140...147) или от скоростного счетчика (0...255)), задача обработки прерывания будет выполнена как по внешнему прерыванию, так и по прерыванию другого типа. В общем случае номера задач обработки прерываний дублировать не следует.

Приоритетность и порядок выполнения задач обработки прерываний

Все задачи обработки прерываний обладают равным приоритетом. Это значит, что, если какая-либо из задач обработки прерывания запустилась, то она выполняется до конца и не прерывается, даже если во время ее выполнения поступает другое прерывание. К примеру, выполнение задачи обработки прерывания от входа не будет прервано с целью выполнения задачи обработки запланированного прерывания, т. е. задача обработки запланированного прерывания будет выполнена только после того, как завершится выполнение задачи обработки прерывания от входа.

При поступлении одновременно нескольких прерываний задачи обработки прерываний выполняются в следующей последовательности: задачи прерываний от входов (прямой режим или режим счетчика), задачи прерываний скоростных счетчиков, задачи внешних прерываний, задача запланированного прерывания.

При поступлении одновременно нескольких прерываний одного типа первой выполняется задача с наименьшим номером.

При разработке программы следует учитывать, что описанный выше порядок выполнения задач обработки прерываний означает, что, если программа пользователя допускает возможность одновременного возникновения нескольких прерываний, то запрограммированная задача может быть выполнена некоторое время спустя после возникновения прерывания. Например, из-за поступления другого прерывания задача обработки запланированного прерывания может быть выполнена в незапланированное время.

2-3-2 Флаги и слова задач обработки прерываний

Максимальное время выполнения задачи обработки прерывания (A440)

Максимальное время выполнения задачи обработки прерывания хранится в двоичном коде (1 бит = 0,1 мс) и обнуляется в начале работы.

Задача обработки прерывания с максимальным временем выполнения (A441)

Номер задачи обработки прерывания, потребовавшей максимальное время выполнения, хранится в двоичном формате. Значения 8000...80FF Hex соответствуют номерам задач 00...FF Hex.

После того как сформировано первое прерывание после начала работы, устанавливается бит A441.15. Максимальное время выполнения для последующих задач обработки прерываний сохраняется в двух старших разрядах в шестнадцатеричном формате и обнуляется в начале работы.

Флаг ошибки задачи обработки прерывания (некритическая ошибка) (A402.13)

Если в области настроек ПЛК установлен бит «Обнаруживать ошибку задачи прерывания», то в случае возникновения ошибок при выполнении задач обработки прерываний будет устанавливаться флаг ошибки задачи обработки прерывания.

Флаг ошибки задачи прерывания (A426.15)/Номер задачи обработки прерывания с ошибкой (A426.00...426.11)

Если включается бит A402.13, в слова A426.15 и A426.00...A426.11 записываются следующие данные.

A402.13	Описание ошибки задачи обработки прерывания	A426.15	A426.00...A426.11
Ошибка выполнения задачи обработки прерывания (если в настройках ПЛК установлен бит «Обнаруживать ошибку задачи прерывания»)	При попытке обновления входов/выходов для большого количества слов с помощью команды IORF(097) из задачи обработки прерывания во время обновления входов/выходов специального модуля ввода/вывода серии CJ в конце цикла программы.	ВКЛ	Будет записан 12-битовый двоичный номер специального модуля ввода/вывода, во время обновления входных/выходных состояний которого произошла ошибка (номер модуля 0...95: 000...05F hex).

Номер задачи при остановке программы (A294)

В следующей таблице указаны слова, в которые будут записаны тип и номер текущей задачи, в момент выполнения которой произошла остановка программы из-за возникновения ошибки.

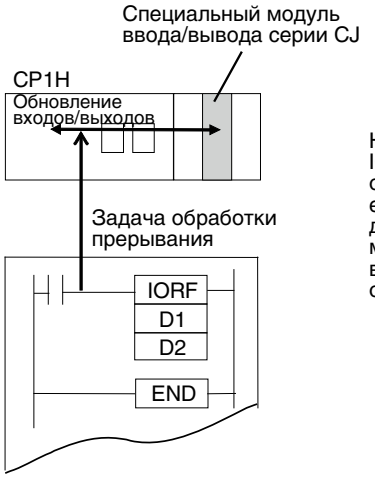
Тип	A294
Задача обработки прерывания	8000...80FF Hex (соответствуют задачам обработки прерываний 0...255)
Циклические задачи	0000...001F Hex (соответствуют задачам 0...31)

2-3-3 Меры предосторожности при использовании

Выполнение IORF(097) для специального модуля ввода/вывода

Если из задачи обработки прерывания должна быть выполнена команда IORF(097) для специального модуля ввода/вывода серии CJ, в настройках ПЛК следует обязательно отключить циклическое обновление для специального модуля ввода/вывода (с соответствующим номером).

При попытке выполнить обновление входов/выходов специального модуля ввода/вывода с помощью команды IORF(097) из задачи обработки прерывания, когда этот модуль уже обновляется в рамках циклического обновления входов/выходов или в результате выполнения команд обновления входов/выходов (IORF(097) или команд мгновенного обновления (!)), произойдет ошибка выполнения задачи обработки прерывания. Если в настройках ПЛК установлен бит «Обнаруживать ошибку задачи прерывания», то при возникновении ошибок во время выполнения задач обработки прерываний устанавливается флаг A402.13 (Флаг ошибки задачи обработки прерывания) и в слово A426 (Ошибка задачи обработки прерывания, номер задачи) записывается номер специального модуля ввода/вывода, с которым связано дублирование операции обновления входов/выходов. Модуль ЦПУ при этом продолжает работать.



Неправильное использование

Не выполняйте IORF(097) из задачи обработки прерывания, если в настройках ПЛК для специальных модулей ввода/вывода включено циклическое обновление.

Правильное использование

Запретите циклическое обновление для специальных модулей ввода/вывода в настройках ПЛК, прежде чем выполнять команду IORF(097) из задачи обработки прерывания.

Параметры в настройках ПЛК

Установите или снимите флажок *Detect Interrupt Task Error (Обнаруживать ошибку задачи прерывания)* в области *Execute Process (Процедура выполнения)* на вкладке Settings (Параметры) в окне PLC Setup (Настройки ПЛК).

Название	Значение	Описание
Обнаруживать ошибку задачи прерывания	Сброшен	Ошибки при выполнении задач обработки прерываний не обнаруживаются.
	Установлен	При обнаружении ошибки задачи обработки прерывания устанавливается флаг ошибки задачи обработки прерывания (A402.13).

Сопутствующие флаги и слова вспомогательной области

Название	Адрес		Описание
Флаг ошибки задачи прерывания	A402.13		Устанавливается при попытке выполнить обновление для специального модуля ввода/вывода серии CJ с помощью команды IORF(097) из задачи обработки прерывания, когда этот модуль уже обновляется в рамках процедуры циклического обновления входов/выходов.
Номер модуля, вызвавшего ошибку задачи обработки прерывания	A426	Биты 00...11	При включении бита A402.13 сюда записывается номер специального модуля ввода/вывода, с которым связана проблема дублированного обновления.
Флаг причины ошибки задачи обработки прерывания		Бит 15	При включении бита A402.13 данный флаг устанавливается с целью указания причины ошибки.

Отключение прерываний

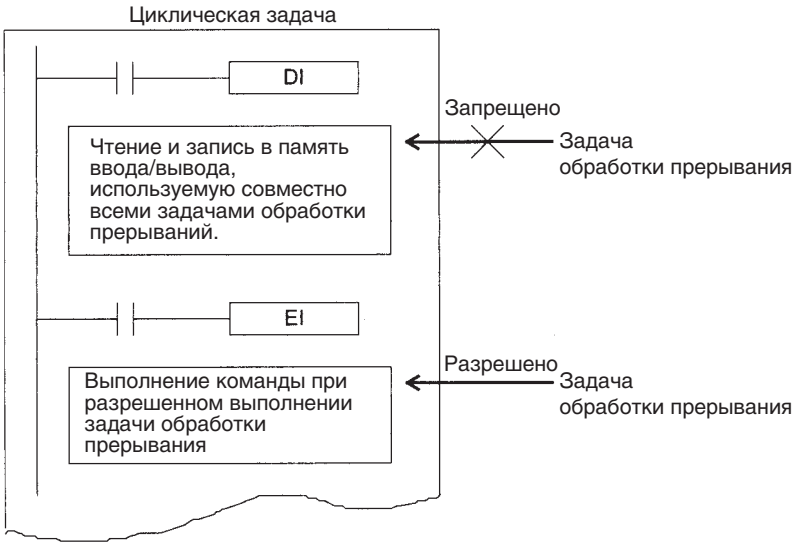
Ниже перечислены операции модуля ЦПУ серии CP, выполнение которых прерывается с целью выполнения задачи обработки прерывания.

- Выполнение команды.
- Обновление встроенных входов/выходов модуля ЦПУ, модулей расширения CPM1A, модулей расширения входов/выходов CPM1A или специальных модулей ввода/вывода серии CJ.
- Обслуживание периферийных устройств.

Совместное использование данных циклическими задачами и задачами обработки прерываний

Если задача обработки прерывания и циклическая задача (в том числе дополнительная циклическая задача) считывают и записывают данные по одним и тем же адресам памяти ввода/вывода, необходимо позаботиться о согласованности этих операций, то есть о согласованном использовании данных. Для запрета обработки прерываний во время обращения к памяти из циклических задач следует выполнить следующие действия.

- Непосредственно перед выполнением команды в циклической задаче, осуществляющей чтение или запись, следует выполнить команду ЗАПРЕТИТЬ ПРЕРЫВАНИЯ (DI(693)), чтобы запретить выполнение задач обработки прерываний.
- Сразу же после выполнения команды следует вновь разрешить выполнение задач обработки прерываний с помощью команды РАЗРЕШИТЬ ПРЕРЫВАНИЯ (EI(694)).



В случае команд, для которых требуется прием отклика и его обработка (например, команд для сетевого обмена данными или связи по последовательному интерфейсу), проблемы, связанные с несогласованностью данных, могут возникать, даже если для отключения задач обработки прерываний применяются команды DI(693) и EI(694).

2-4 Использование CX-Programmer для задач

Примечание. Для создания циклических задач (в том числе дополнительных циклических задач) следует использовать CX-Programmer. Обязательно используйте CX-Programmer для назначения типа и номера задачи создаваемым программам.

CX-Programmer

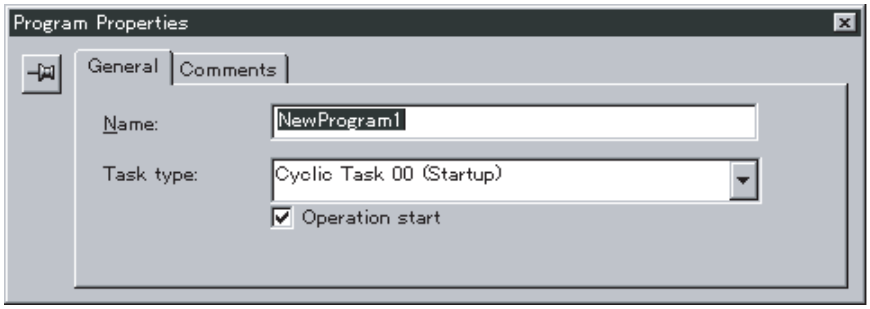
Для каждой программы в качестве атрибутов следует указать тип задачи и номер задачи.

1,2,3...

1. Выберите ***View/Properties (Вид/Свойства)*** или щелкните правой кнопкой мыши и выберите пункт ***Properties (Свойства)*** из всплывающего меню, чтобы отобразить программу, которой будет назначена задача.

Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

2. Выберите вкладку **General (Общие свойства)** и выберите тип задачи в поле *Task Type* и номер задачи в поле *Task No.* В случае циклической задачи также следует установить флажок ***Operation start (Запуск в начале работы)***, если требуется, чтобы задача активизировалась вместе с запуском программы.



РАЗДЕЛ 3
Команды

Данный раздел содержит подробное описание каждой команды, которая может быть использована для программирования ПЛК серии CP. Описываемые команды сгруппированы по выполняемым функциям.

3-1 Условные обозначения и структура информации при описании команд . . . 95
3-2 Входные битовые команды . . . 99
3-2-1 ЗАГРУЗКА: LD . . . 99
3-2-2 ЗАГРУЗКА НЕ: LD NOT . . . 101
3-2-3 И: AND . . . 104
3-2-4 И НЕ: AND NOT . . . 105
3-2-5 ИЛИ: OR . . . 107
3-2-6 ИЛИ НЕ: OR NOT . . . 109
3-2-7 И ЗАГРУЗКА: AND LD . . . 111
3-2-8 ИЛИ ЗАГРУЗКА: OR LD . . . 113
3-2-9 Однократное выполнение по фронту и мгновенное обновление . . . 116
3-2-10 Временные диаграммы работы входных и выходных команд . . . 118
3-2-11 Биты временного хранения (TR) . . . 119
3-2-12 НЕ: NOT(520) . . . 120
3-2-13 УСЛОВИЕ ВКЛ/ВЫКЛ: UP(521) и DOWN(522) . . . 121
3-2-14 ПРОВЕРКА БИТА: TST(350) и TSTN(351) . . . 123
3-3 Выходные битовые команды . . . 126
3-3-1 ВЫВОД: OUT . . . 126
3-3-2 ВЫВОД НЕ: OUT NOT . . . 127
3-3-3 ЗАЩЕЛКА: KEEP(011) . . . 128
3-3-4 ПОЛОЖИТЕЛЬНЫЙ/ОТРИЦАТЕЛЬНЫЙ ФРОНТ: DIFU(013) и DIFD(014) . . . 133
3-3-5 УСТАНОВКА и СБРОС: SET и RSET . . . 135
3-3-6 УСТАНОВИТЬ/СБРОСИТЬ НЕСКОЛЬКО БИТОВ: SETA(530)/RSTA(531) . . . 138
3-3-7 УСТАНОВИТЬ/СБРОСИТЬ ОДИН БИТ: SETB(532)/RSTB(533) . . . 141
3-3-8 ВЫВОД В ОДИН БИТ: OUTB(534) . . . 144
3-4 Команды управления последовательностью выполнения . . . 146
3-4-1 КОНЕЦ: END(001) . . . 146
3-4-2 ПУСТАЯ ОПЕРАЦИЯ: NOP(000) . . . 147
3-4-3 Обзор команд блокировки . . . 147
3-4-4 БЛОКИРОВКА и ОТМЕНА БЛОКИРОВКИ: IL(002) и ILC(003) . . . 150
3-4-5 ВЛОЖЕННАЯ БЛОКИРОВКА С ЗАПОМИНАНИЕМ ФРОНТА, ВЛОЖЕННАЯ БЛОКИРОВКА БЕЗ ЗАПОМИНАНИЯ ФРОНТА и ОТМЕНА ВЛОЖЕННОЙ БЛОКИРОВКИ: MILH(517), MILR(518) и MILC(519) . . . 155
3-4-6 ПЕРЕХОД и КОНЕЦ ПЕРЕХОДА: JMP(004) и JME(005) . . . 168
3-4-7 УСЛОВНЫЙ ПЕРЕХОД: CJP(510)/CJPN(511) . . . 172
3-4-8 МНОЖЕСТВЕННЫЙ ПЕРЕХОД и КОНЕЦ ПЕРЕХОДА: JMP0(515) и JME0(516) . . . 176
3-4-9 ЦИКЛЫ FOR-NEXT: FOR(512)/NEXT(513) . . . 178
3-4-10 ВЫХОД ИЗ ЦИКЛА: BREAK(514) . . . 182
3-5 Команды управления таймерами и счетчиками . . . 184
3-5-1 ТАЙМЕР: TIM/TIMX(550) . . . 186
3-5-2 СКОРОСТНОЙ ТАЙМЕР: TIMH(015)/TIMHX(551) . . . 191
3-5-3 1 мс ТАЙМЕР: TMHH(540)/TMHHX(552) . . . 195
3-5-4 НАКАПЛИВАЮЩИЙ ТАЙМЕР: TTIM(087)/TTIMX(555) . . . 199
3-5-5 ДОЛГОСРОЧНЫЙ ТАЙМЕР: TIML(542)/TIMLX(553) . . . 203

Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

3-5-6 МНОГОВЫХОДНОЙ ТАЙМЕР: MTIM(543)/MTIMX(554) . . . 206
3-5-7 СЧЕТЧИК: CNT/CNTX(546) . . . 212
3-5-8 РЕВЕРСИВНЫЙ СЧЕТЧИК: CNTR(012)/CNTRX(548) . . . 215
3-5-9 СБРОС ТАЙМЕРА/СЧЕТЧИКА: CNR(545)/CNRX(547) . . . 220
3-5-10 Примеры применения таймеров и счетчиков . . . 223
3-5-11 Косвенное обращение к таймерам/счетчикам . . . 227
3-6 Команды сравнения . . . 230
3-6-1 Входные команды сравнения (300...328) . . . 230
3-6-2 Команды сравнения значений времени (341...346) . . . 236
3-6-3 СРАВНИТЬ: CMP(020) . . . 241
3-6-4 СРАВНИТЬ ДВОЙНЫЕ СЛОВА: CMPL(060) . . . 244
3-6-5 СРАВНИТЬ ДВОИЧНЫЕ ЗНАЧЕНИЯ СО ЗНАКОМ: CPS(114) . . . 247
3-6-6 СРАВНИТЬ ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ: CPSL(115) . . . 250
3-6-7 СРАВНИТЬ НЕСКОЛЬКО ЗНАЧЕНИЙ: MCMP(019) . . . 253
3-6-8 СРАВНИТЬ С ТАБЛИЦЕЙ: TCMP(085) . . . 256
3-6-9 СРАВНИТЬ С ДИАПАЗОНАМИ: BCMP(068) . . . 258
3-6-10 РАСШИРЕННОЕ СРАВНЕНИЕ С ДИАПАЗОНАМИ: BCMP2(502) . . . 261
3-6-11 СРАВНИТЬ СЛОВО С ДИАПАЗОНОМ: ZCP(088) . . . 265
3-6-12 СРАВНИТЬ ДВОЙНОЕ СЛОВО С ДИАПАЗОНОМ: ZCPL(116) . . . 268
3-7 Команды передачи данных . . . 271
3-7-1 ПЕРЕДАТЬ: MOV(021) . . . 271
3-7-2 ПЕРЕДАТЬ ДОПОЛНЕНИЕ: MVN(022) . . . 273
3-7-3 ПЕРЕДАТЬ ДВОЙНОЕ СЛОВО: MOVL(498) . . . 274
3-7-4 ПЕРЕДАТЬ ДОПОЛНЕНИЕ К ДВОЙНОМУ СЛОВУ: MVNL(499) . . . 276
3-7-5 ПЕРЕДАТЬ БИТ: MOVB(082) . . . 277
3-7-6 ПЕРЕДАТЬ РАЗРЯД: MOVD(083) . . . 279
3-7-7 ПЕРЕДАТЬ НЕСКОЛЬКО БИТОВ: XFRB(062) . . . 282
3-7-8 ПЕРЕДАЧА БЛОКА: XFER(070) . . . 285
3-7-9 ЗАПОЛНЕНИЕ БЛОКА: BSET(071) . . . 287
3-7-10 ОБМЕН ДАННЫМИ: XCHG(073) . . . 289
3-7-11 ОБМЕН ДВОЙНЫМИ СЛОВАМИ: XCGL(562) . . . 290
3-7-12 РАЗМЕЩЕНИЕ ОДИНОЧНОГО СЛОВА: DIST(080) . . . 292
3-7-13 СБОР ДАННЫХ: COLL(081) . . . 294
3-7-14 ПЕРЕДАТЬ В РЕГИСТР: MOVR(560) . . . 296
3-7-15 ПЕРЕДАТЬ ТЕКУЩЕЕ ЗНАЧЕНИЕ ТАЙМЕРА/СЧЕТЧИКА В РЕГИСТР: MOVRW(561) . . . 298
3-8 Команды сдвига данных . . . 301
3-8-1 РЕГИСТР СДВИГА: SFT(010) . . . 302
3-8-2 РЕВЕРСИВНЫЙ РЕГИСТР СДВИГА: SFTR(084) . . . 304
3-8-3 АСИНХРОННЫЙ РЕГИСТР СДВИГА: ASFT(017) . . . 306
3-8-4 СДВИГ ПО СЛОВАМ: WSFT(016) . . . 309
3-8-5 АРИФМЕТИЧЕСКИЙ СДВИГ ВЛЕВО: ASL(025) . . . 311
3-8-6 СДВИГ ДВОЙНОГО СЛОВА ВЛЕВО: ASLL(570) . . . 312
3-8-7 АРИФМЕТИЧЕСКИЙ СДВИГ ВПРАВО: ASR(026) . . . 314
3-8-8 СДВИГ ДВОЙНОГО СЛОВА ВПРАВО: ASRL(571) . . . 316
3-8-9 ЦИКЛИЧЕСКИЙ СДВИГ ВЛЕВО: ROL(027) . . . 317
3-8-10 ЦИКЛИЧЕСКИЙ СДВИГ ДВОЙНОГО СЛОВА ВЛЕВО: ROLL(572) . . . 319
3-8-11 ЦИКЛИЧЕСКИЙ СДВИГ ВПРАВО: ROR(028) . . . 321
3-8-12 ЦИКЛИЧЕСКИЙ СДВИГ ДВОЙНОГО СЛОВА ВПРАВО: RORL(573) . . . 322
3-8-13 ЦИКЛИЧЕСКИЙ СДВИГ ВЛЕВО БЕЗ ПЕРЕНОСА: RLNC(574) . . . 324
3-8-14 ЦИКЛИЧЕСКИЙ СДВИГ ДВОЙНОГО СЛОВА ВЛЕВО БЕЗ ПЕРЕНОСА: RLNL(576) . . . 326

Предварительная версия
Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

3-8-15 ЦИКЛИЧЕСКИЙ СДВИГ ВПРАВО БЕЗ ПЕРЕНОСА: RRNC(575) 327
3-8-16 ЦИКЛИЧЕСКИЙ СДВИГ ДВОЙНОГО СЛОВА ВПРАВО БЕЗ ПЕРЕНОСА: RRNL(577). 329
3-8-17 СДВИГ НА ОДИН РАЗРЯД ВЛЕВО: SLD(074) 331
3-8-18 СДВИГ НА ОДИН РАЗРЯД ВПРАВО: SRD(075). 332
3-8-19 СДВИГ N БИТОВ ДАННЫХ ВЛЕВО: NSFL(578) 334
3-8-20 СДВИГ N БИТОВ ДАННЫХ ВПРАВО: NSFR(579) 336
3-8-21 СДВИГ НА N БИТОВ ВЛЕВО: NASL(580) 338
3-8-22 СДВИГ ДВОЙНОГО СЛОВА НА N БИТОВ ВЛЕВО: NSLL(582) 341
3-8-23 СДВИГ НА N БИТОВ ВПРАВО: NASR(581). 344
3-8-24 СДВИГ ДВОЙНОГО СЛОВА НА N БИТОВ ВПРАВО: NSRL(583) 346
3-9 Команды увеличения/уменьшения 349
3-9-1 УВЕЛИЧИТЬ ДВОИЧНОЕ СЛОВО: ++(590) 349
3-9-2 УВЕЛИЧИТЬ ДВОЙНОЕ ДВОИЧНОЕ СЛОВО: ++L(591) 351
3-9-3 УМЕНЬШИТЬ ДВОИЧНОЕ СЛОВО: – –(592) 353
3-9-4 УМЕНЬШИТЬ ДВОЙНОЕ ДВОИЧНОЕ СЛОВО: – –L(593). 355
3-9-5 УВЕЛИЧИТЬ ДВОИЧНО-ДЕСЯТИЧНОЕ СЛОВО: ++B(594) 358
3-9-6 УВЕЛИЧИТЬ ДВОЙНОЕ ДВОИЧНО-ДЕСЯТИЧНОЕ СЛОВО: ++BL(595). 360
3-9-7 УМЕНЬШИТЬ ДВОИЧНО-ДЕСЯТИЧНОЕ СЛОВО: – –B(596) 362
3-9-8 УМЕНЬШИТЬ ДВОЙНОЕ ДВОИЧНО-ДЕСЯТИЧНОЕ СЛОВО: – –BL(597). 364
3-10 Символьные математические команды 366
3-10-1 СЛОЖИТЬ БЕЗ ПЕРЕНОСА ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ: +(400) 367
3-10-2 СЛОЖИТЬ БЕЗ ПЕРЕНОСА ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ: +L(401) 370
3-10-3 СЛОЖИТЬ С ПЕРЕНОСОМ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ: +C(402) 371
3-10-4 СЛОЖИТЬ С ПЕРЕНОСОМ ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ: +CL(403). 374
3-10-5 СЛОЖИТЬ БЕЗ ПЕРЕНОСА ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА: +B(404) 376
3-10-6 СЛОЖИТЬ БЕЗ ПЕРЕНОСА ДВОЙНЫЕ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА: +BL(405). 377
3-10-7 СЛОЖИТЬ С ПЕРЕНОСОМ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА: +BC(406) 379
3-10-8 СЛОЖИТЬ С ПЕРЕНОСОМ ДВОЙНЫЕ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА: +BCL(407) 381
3-10-9 ВЫЧЕСТЬ БЕЗ ПЕРЕНОСА ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ: –(410) 382
3-10-10 ВЫЧЕСТЬ БЕЗ ПЕРЕНОСА ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ: –L(411) 385
3-10-11 ВЫЧЕСТЬ С ПЕРЕНОСОМ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ: –C(412). 389
3-10-12 ВЫЧЕСТЬ С ПЕРЕНОСОМ ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ: –CL(413) 391
3-10-13 ВЫЧЕСТЬ БЕЗ ПЕРЕНОСА ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА: –B(414) 394
3-10-14 ВЫЧЕСТЬ БЕЗ ПЕРЕНОСА ДВОЙНЫЕ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА: –BL(415) 396
3-10-15 ВЫЧЕСТЬ С ПЕРЕНОСОМ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА: –BC(416). 399
3-10-16 ВЫЧЕСТЬ С ПЕРЕНОСОМ ДВОЙНЫЕ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА: –BCL(417) 401
3-10-17 УМНОЖИТЬ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ: *(420) 403
3-10-18 УМНОЖИТЬ ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ: *L(421) 405
3-10-19 УМНОЖИТЬ ДВОИЧНЫЕ СЛОВА БЕЗ ЗНАКА: *U(422) 407
3-10-20 УМНОЖИТЬ ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА БЕЗ ЗНАКА: *UL(423) 408
3-10-21 УМНОЖИТЬ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА: *B(424). 410
3-10-22 УМНОЖИТЬ ДВОЙНЫЕ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА: *BL(425) 412
3-10-23 ДЕЛИТЬ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ: /(430) 413
3-10-24 ДЕЛИТЬ ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ: /L(431) 415

3-10-25 ДЕЛИТЬ ДВОИЧНЫЕ СЛОВА БЕЗ ЗНАКА: /U(432) 417
3-10-26 ДЕЛИТЬ ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА БЕЗ ЗНАКА: /UL(433) 419
3-10-27 ДЕЛИТЬ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА: /B(434) 421
3-10-28 ДЕЛИТЬ ДВОЙНЫЕ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА: /BL(435) 422
3-11 Команды преобразования 425
3-11-1 BCD В ДВОИЧНОЕ: BIN(023) 425
3-11-2 ДВОЙНОЕ BCD В ДВОЙНОЕ ДВОИЧНОЕ: BINL(058) 427
3-11-3 ДВОИЧНОЕ В BCD: BCD(024) 429
3-11-4 ДВОЙНОЕ ДВОИЧНОЕ В ДВОЙНОЕ BCD: BCDL(059) 430
3-11-5 ДОПОЛНЕНИЕ ДО 2: NEG(160) 432
3-11-6 ДОПОЛНЕНИЕ ДВОЙНОГО СЛОВА ДО 2: NEGL(161) 434
3-11-7 16-БИТОВОЕ ДВОИЧНОЕ СО ЗНАКОМ В 32-БИТОВОЕ: SIGN(600) 435
3-11-8 ДЕКОДИРОВАНИЕ ДАННЫХ: MLPX(076) 437
3-11-9 КОДИРОВАНИЕ ДАННЫХ: DMPX(077) 442
3-11-10 ПРЕОБРАЗОВАНИЕ В ASCII: ASC(086) 446
3-11-11 ASCII В HEX: HEX(162) 450
3-11-12 СТОЛБЕЦ В СТРОКУ: LINE(063) 454
3-11-13 СТРОКУ В СТОЛБЕЦ: COLM(064) 456
3-11-14 BCD СО ЗНАКОМ В ДВОИЧНОЕ: BINS(470) 459
3-11-15 ДВОЙНОЕ BCD СО ЗНАКОМ В ДВОИЧНОЕ: BISL(472) 462
3-11-16 ДВОИЧНОЕ СО ЗНАКОМ В BCD: BCDS(471) 465
3-11-17 ДВОЙНОЕ ДВОИЧНОЕ СО ЗНАКОМ В ДВОЙНОЕ BCD: BDSL(473) 468
3-11-18 ПРЕОБРАЗОВАНИЕ КОДА ГРЕЯ: GRY(474) 472
3-12 Логические команды 477
3-12-1 ЛОГИЧЕСКОЕ И: ANDW(034) 477
3-12-2 ДВОЙНОЕ ЛОГИЧЕСКОЕ И: ANDL(610) 478
3-12-3 ЛОГИЧЕСКОЕ ИЛИ: ORW(035) 480
3-12-4 ДВОЙНОЕ ЛОГИЧЕСКОЕ ИЛИ: ORWL(611) 482
3-12-5 ИСКЛЮЧАЮЩЕЕ ИЛИ: XORW(036) 483
3-12-6 ДВОЙНОЕ ИСКЛЮЧАЮЩЕЕ ИЛИ: XORL(612) 485
3-12-7 ИСКЛЮЧАЮЩЕЕ ИЛИ-НЕ: XNRW(037) 487
3-12-8 ДВОЙНОЕ ИСКЛЮЧАЮЩЕЕ ИЛИ-НЕ: XNRL(613) 488
3-12-9 ДОПОЛНЕНИЕ: COM(029) 490
3-12-10 ДВОЙНОЕ ДОПОЛНЕНИЕ: COML(614) 492
3-13 Специальные математические команды 493
3-13-1 КВАДРАТНЫЙ КОРЕНЬ ИЗ ДВОИЧНОГО ЧИСЛА: ROTB(620) 493
3-13-2 КВАДРАТНЫЙ КОРЕНЬ ИЗ ДВОИЧНО-ДЕСЯТИЧНОГО ЧИСЛА: ROOT(072) . . 495
3-13-3 МАТЕМАТИЧЕСКИЕ ФУНКЦИИ: APR(069) 499
3-13-4 ДЕЛЕНИЕ ЧИСЕЛ С ПЛАВАЮЩЕЙ ЗАПЯТОЙ: FDIV(079) 510
3-13-5 ПОДСЧЕТ БИТОВ: BCNT(067) 514
3-14 Команды математических операций над числами с плавающей запятой 516
3-14-1 ПЛАВ. ЗАПЯТАЯ В 16-БИТ: FIX(450) 522
3-14-2 ПЛАВ. ЗАПЯТАЯ В 32-БИТ: FIXL(451) 524
3-14-3 16-БИТ В ПЛАВ. ЗАПЯТАЯ: FLT(452) 525
3-14-4 32-БИТ В ПЛАВ. ЗАПЯТАЯ: FLTL(453) 527
3-14-5 СЛОЖЕНИЕ С ПЛАВАЮЩЕЙ ЗАПЯТОЙ: +F(454) 528
3-14-6 ВЫЧИТАНИЕ С ПЛАВАЮЩЕЙ ЗАПЯТОЙ: –F(455) 530
3-14-7 УМНОЖЕНИЕ С ПЛАВАЮЩЕЙ ЗАПЯТОЙ: *F(456) 532
3-14-8 ДЕЛЕНИЕ С ПЛАВАЮЩЕЙ ЗАПЯТОЙ: /F(457) 534
3-14-9 ГРАДУСЫ В РАДИАНЫ: RAD(458) 536
3-14-10 РАДИАНЫ В ГРАДУСЫ: DEG(459) 538

3-14-11 СИНУС: SIN(460) . . . 540
3-14-12 КОСИНУС: COS(461) . . . 542
3-14-13 ТАНГЕНС: TAN(462) . . . 543
3-14-14 АРКСИНУС: ASIN(463) . . . 545
3-14-15 АРККОСИНУС: ACOS(464) . . . 547
3-14-16 АРКТАНГЕНС: ATAN(465) . . . 549
3-14-17 КВАДРАТНЫЙ КОРЕНЬ: SQRT(466) . . . 551
3-14-18 ЭКСПОНЕНТА: EXP(467) . . . 553
3-14-19 ЛОГАРИФМ: LOG(468) . . . 555
3-14-20 ВОЗВЕДЕНИЕ В СТЕПЕНЬ: PWR(840) . . . 557
3-14-21 Команды сравнения чисел с плавающей запятой одинарной точности . . . 559
3-14-22 ПЛАВАЮЩАЯ ЗАПЯТАЯ В ASCII: FSTR(448) . . . 563
3-14-23 ASCII В ПЛАВАЮЩАЯ ЗАПЯТАЯ: FVAL(449) . . . 568
3-15 Команды для чисел с плавающей запятой двойной точности . . . 573
3-15-1 ПЛАВ. ЗАПЯТАЯ ДВОЙНОЙ ТОЧНОСТИ В 16-БИТ: FIXD(841) . . . 579
3-15-2 ПЛАВ. ЗАПЯТАЯ ДВОЙНОЙ ТОЧНОСТИ В 32-БИТ: FIXLD(842) . . . 581
3-15-3 16-БИТ В ПЛАВ. ЗАПЯТАЯ ДВОЙНОЙ ТОЧНОСТИ: DBL(843) . . . 582
3-15-4 32-БИТ В ПЛАВ. ЗАПЯТАЯ ДВОЙНОЙ ТОЧНОСТИ: DBLL(844) . . . 584
3-15-5 СЛОЖЕНИЕ С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ: +D(845) . . . 585
3-15-6 ВЫЧИТАНИЕ С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ: –D(846) . . . 587
3-15-7 УМНОЖЕНИЕ С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ: *D(847) . . . 589
3-15-8 ДЕЛЕНИЕ С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ: /D(848) . . . 591
3-15-9 ГРАДУСЫ В РАДИАНЫ ДВОЙНОЙ ТОЧНОСТИ: RADD(849) . . . 593
3-15-10 РАДИАНЫ В ГРАДУСЫ ДВОЙНОЙ ТОЧНОСТИ: DEGD(850) . . . 595
3-15-11 СИНУС ДВОЙНОЙ ТОЧНОСТИ: SIND(851) . . . 597
3-15-12 КОСИНУС ДВОЙНОЙ ТОЧНОСТИ: COSD(852) . . . 598
3-15-13 ТАНГЕНС ДВОЙНОЙ ТОЧНОСТИ: TAND(853) . . . 600
3-15-14 АРКСИНУС ДВОЙНОЙ ТОЧНОСТИ: ASIND(854) . . . 602
3-15-15 АРККОСИНУС ДВОЙНОЙ ТОЧНОСТИ: ACOSD(855) . . . 604
3-15-16 АРКТАНГЕНС ДВОЙНОЙ ТОЧНОСТИ: ATAND(856) . . . 606
3-15-17 КВАДРАТНЫЙ КОРЕНЬ ДВОЙНОЙ ТОЧНОСТИ: SQRTD(857) . . . 608
3-15-18 ЭКСПОНЕНТА ДВОЙНОЙ ТОЧНОСТИ: EXPD(858) . . . 610
3-15-19 ЛОГАРИФМ ДВОЙНОЙ ТОЧНОСТИ: LOGD(859) . . . 612
3-15-20 ВОЗВЕДЕНИЕ В СТЕПЕНЬ ДВОЙНОЙ ТОЧНОСТИ: PWRD(860) . . . 614
3-15-21 Входные команды для чисел с плавающей запятой двойной точности . . . 616
3-16 Команды обработки табличных данных . . . 620
3-16-1 СОЗДАТЬ СТЕК: SSET(630) . . . 621
3-16-2 ЗАПИСАТЬ В СТЕК: PUSH(632) . . . 623
3-16-3 ПЕРВЫМ ПРИШЕЛ — ПЕРВЫМ ВЫШЕЛ: FIFO(633) . . . 626
3-16-4 ПОСЛЕДНИМ ПРИШЕЛ — ПЕРВЫМ ВЫШЕЛ: LIFO(634) . . . 629
3-16-5 СОЗДАТЬ ТАБЛИЦУ УКАЗАННЫХ РАЗМЕРОВ: DIM(631) . . . 632
3-16-6 ЗАПИСАТЬ РАСПОЛОЖЕНИЕ СТРОКИ: SETR(635) . . . 634
3-16-7 СЧИТАТЬ НОМЕР СТРОКИ: GETR(636) . . . 636
3-16-8 ПОИСК ЗНАЧЕНИЯ: SRCH(181) . . . 638
3-16-9 ПОМЕНЯТЬ МЕСТАМИ БАЙТЫ: SWAP(637) . . . 641
3-16-10 НАЙТИ МАКСИМУМ: MAX(182) . . . 643
3-16-11 НАЙТИ МИНИМУМ: MIN(183) . . . 646
3-16-12 СЛОЖЕНИЕ: SUM(184) . . . 649
3-16-13 КОНТРОЛЬНАЯ СУММА КАДРА: FCS(180) . . . 653
3-16-14 СЧИТАТЬ РАЗМЕР СТЕКА: SNUM(638) . . . 655
3-16-15 ПРОЧИТАТЬ ДАННЫЕ ИЗ СТЕКА: SREAD(639) . . . 658

3-16-16 ЗАПИСАТЬ ДАННЫЕ В СТЕК: SWRIT(640) 661
3-16-17 ВСТАВИТЬ ДАННЫЕ В СТЕК: SINS(641) 664
3-16-18 УДАЛИТЬ ДАННЫЕ ИЗ СТЕКА: SDEL(642) 667
3-17 Команды управления данными 671
3-17-1 ПИД-РЕГУЛЯТОР: PID(190) 671
3-17-2 ПИД-РЕГУЛЯТОР С АВТОНАСТРОЙКОЙ: PIDAT(191) 684
3-17-3 ОГРАНИЧЕНИЕ: LMT(680) 696
3-17-4 ЗОНА НЕЧУВСТВИТЕЛЬНОСТИ: BAND(681) 698
3-17-5 МЕРТВАЯ ЗОНА: ZONE(682) 701
3-17-6 ВЫХОД ШИМ: TPO(685) 704
3-17-7 ИЗМЕНЕНИЕ МАСШТАБА: SCL(194) 713
3-17-8 ИЗМЕНЕНИЕ МАСШТАБА 2: SCL2(486) 718
3-17-9 ИЗМЕНЕНИЕ МАСШТАБА 3: SCL3(487) 722
3-17-10 СРЕДНЕЕ ЗНАЧЕНИЕ: AVG(195) 726
3-18 Команды подпрограмм 730
3-18-1 ВЫЗОВ ПОДПРОГРАММЫ: SBS(091) 730
3-18-2 МАКРОС: MCRO(099) 736
3-18-3 ВХОД В ПОДПРОГРАММУ: SBN(092) 740
3-18-4 ВЫХОД ИЗ ПОДПРОГРАММЫ: RET(093) 743
3-18-5 ВЫЗОВ ГЛОБАЛЬНОЙ ПОДПРОГРАММЫ: GSBS(750) 743
3-18-6 ВХОД В ГЛОБАЛЬНУЮ ПОДПРОГРАММУ: GSBN(751) 751
3-18-7 ВЫХОД ИЗ ГЛОБАЛЬНОЙ ПОДПРОГРАММЫ: GRET(752) 754
3-19 Команды управления прерываниями 755
3-19-1 УСТАНОВИТЬ МАСКУ ПРЕРЫВАНИЙ: MSKS(690) 755
3-19-2 СЧИТАТЬ МАСКУ ПРЕРЫВАНИЙ: MSKR(692) 759
3-19-3 ОЧИСТИТЬ ПРЕРЫВАНИЕ: CLI(691) 763
3-19-4 ЗАПРЕТИТЬ ПРЕРЫВАНИЯ: DI(693) 766
3-19-5 РАЗРЕШИТЬ ПРЕРЫВАНИЯ: EI(694) 768
3-20 Команды управления скоростными счетчиками и импульсными выходами 770
3-20-1 УПРАВЛЕНИЕ РЕЖИМОМ: INI(880) 770
3-20-2 ЧТЕНИЕ ТЕКУЩ. ЗНАЧ. СКОРОСТНОГО СЧЕТЧИКА: PRV(881) 775
3-20-3 ПРЕОБРАЗОВАНИЕ ЧАСТОТЫ НА ВХОДЕ СЧЕТЧИКА: PRV2(883) 781
3-20-4 ЗАРЕГИСТРИРОВАТЬ ТАБЛИЦУ СРАВНЕНИЯ: CTBL(882) 785
3-20-5 ВЫДАЧА ИМПУЛЬСОВ СКОРОСТИ: SPED(885) 790
3-20-6 ЗАДАТЬ КОЛ-ВО ИМПУЛЬСОВ: PULS(886) 796
3-20-7 ВЫДАЧА ИМПУЛЬСОВ: PLS2(887) 799
3-20-8 УПРАВЛЕНИЕ РАЗГОНОМ: ACC(888) 808
3-20-9 ПОИСК ИСХОДНОГО ПОЛОЖЕНИЯ: ORG(889) 817
3-20-10 ИМПУЛЬСЫ С ПЕРЕМЕННОЙ СКВАЖНОСТЬЮ: PWM(891) 822
3-21 Команды для пошагового выполнения 824
3-21-1 ОПРЕДЕЛИТЬ ШАГ и ЗАПУСТИТЬ ШАГ: STEP(008)/SNXT(009) 825
3-22 Команды базовых модулей ввода/вывода 843
3-22-1 ОБНОВЛЕНИЕ СЛОВ ВВОДА/ВЫВОДА: IORF(097) 843
3-22-2 ПРЕОБРАЗОВАТЬ В 7-СЕГМЕНТНЫЙ КОД: SDEC(078) 846
3-22-3 ЧТЕНИЕ ЦИФРОВОГО ПЕРЕКЛЮЧАТЕЛЯ — DSW(210) 849
3-22-4 ЧТЕНИЕ ДЕСЯТИЧНОЙ КЛАВИАТУРЫ — TKY(211) 853
3-22-5 ЧТЕНИЕ ШЕСТНАДЦАТЕРИЧНОЙ КЛАВИАТУРЫ — HKY(212) 857
3-22-6 ЧТЕНИЕ МАТРИЦЫ: MTR(213) 861
3-22-7 ВЫВОД НА 7-СЕГМЕНТНЫЙ ИНДИКАТОР — 7SEG(214) 865
3-22-8 ЧТЕНИЕ ИНТЕЛЛЕКТУАЛЬНЫХ ВХОДОВ/ВЫХОДОВ: IORD(222) 870
3-22-9 ЗАПИСЬ ИНТЕЛЛЕКТУАЛЬНЫХ ВХОДОВ/ВЫХОДОВ: IOWR(223) 873

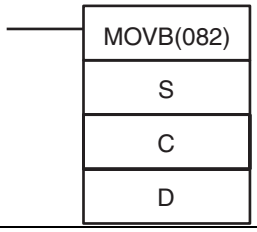
3-22-10 ОБНОВЛЕНИЕ ШИНЫ ВВОДА/ВЫВОДА ЦПУ: DLNK(226) 877
3-23 Команды последовательного интерфейса 882
3-23-1 Последовательный интерфейс 882
3-23-2 МАКРОС ПРОТОКОЛА: PMCR(260) 884
3-23-3 ОТПРАВИТЬ: TXD(236) 892
3-23-4 ПРИНЯТЬ: RXD(235) 898
3-23-5 ОТПРАВИТЬ ЧЕРЕЗ МОДУЛЬ ПОСЛЕД. ИНТЕРФЕЙСА: TXDU(256) 904
3-23-6 ПРИНЯТЬ ЧЕРЕЗ МОДУЛЬ ПОСЛЕД. ИНТЕРФЕЙСА: RXDU(255) 912
3-23-7 ИЗМЕНИТЬ ПАРАМЕТРЫ ПОСЛЕД. ПОРТА: STUP(237) 921
3-24 Сетевые команды 924
3-24-1 Общие сведения о сетевых командах 924
3-24-2 Сведения о командах передачи явных сообщений (только для CP1H) 940
3-24-3 ПЕРЕДАТЬ ПО СЕТИ: SEND(090) 945
3-24-4 ПРИНЯТЬ ПО СЕТИ: RECV(098) 952
3-24-5 ДОСТАВИТЬ КОМАНДУ: CMND(490) 958
3-24-6 ОТПРАВИТЬ ЯВНОЕ СООБЩЕНИЕ: EXPLT(720) 967
3-24-7 ЯВНЫЙ ЗАПРОС АТРИБУТА: EGATR(721) 975
3-24-8 ЯВНАЯ УСТАНОВКА АТРИБУТА: ESATR(722) 982
3-24-9 ЯВНОЕ ЧТЕНИЕ СЛОВА: ECHRD(723) 989
3-24-10 ЯВНАЯ ЗАПИСЬ СЛОВА: ECHWR(724) 993
3-25 Команды дисплея 997
3-25-1 ОТОБРАЗИТЬ СООБЩЕНИЕ: MSG(046) 997
3-25-2 ОТОБРАЗИТЬ ЗНАЧЕНИЕ СЛОВА НА 7-СЕГМЕНТНОМ ИНДИКАТОРЕ: SCH(047) 1000
3-25-3 УПРАВЛЕНИЕ 7-СЕГМЕНТНЫМ ИНДИКАТОРОМ: SCTRL(048) 1002
3-26 Команды для работы с часами 1006
3-26-1 ДОБАВИТЬ К КАЛЕНДАРНОЙ ДАТЕ: CADD(730) 1006
3-26-2 ВЫЧЕСТЬ ИЗ КАЛЕНДАРНОЙ ДАТЫ: CSUB(731) 1009
3-26-3 ЧАСЫ В СЕКУНДЫ: SEC(065) 1013
3-26-4 СЕКУНДЫ В ЧАСЫ: HMS(066) 1015
3-26-5 КОРРЕКТИРОВКА ЧАСОВ: DATE(735) 1017
3-27 Команды для отладки программы 1020
3-27-1 Отбор данных для памяти протокола данных: TRSM(045) 1020
3-28 Команды для диагностики неисправностей 1025
3-28-1 СИГНАЛИЗАЦИЯ НЕИСПРАВНОСТИ: FAL(006) 1025
3-28-2 СИГНАЛИЗАЦИЯ СЕРЬЕЗНОЙ НЕИСПРАВНОСТИ: FALS(007) 1033
3-28-3 ОБНАРУЖЕНИЕ НЕИСПРАВНОГО КАНАЛА: FPD(269) 1041
3-29 Прочие команды 1051
3-29-1 УСТАНОВИТЬ ПЕРЕНОС: STC(040) 1051
3-29-2 СБРОСИТЬ ПЕРЕНОС: CLC(041) 1052
3-29-3 УВЕЛИЧИТЬ МАКС. ВРЕМЯ ЦИКЛА: WDT(094) 1053
3-29-4 СОХРАНИТЬ ФЛАГИ УСЛОВИЙ: CCS(282) 1055
3-29-5 ЗАГРУЗИТЬ ФЛАГИ УСЛОВИЙ: CCL(283) 1057
3-29-6 ПРЕОБРАЗОВАТЬ АДРЕС СЕРИИ CV: FRMCV(284) 1059
3-29-7 ПРЕОБРАЗОВАТЬ В АДРЕС СЕРИИ CV: TOCV(285) 1063
3-30 Команды для создания программных блоков 1067
3-30-1 Введение 1068
3-30-2 НАЧАЛО/КОНЕЦ ПРОГРАММНОГО БЛОКА: BPRG(096)/BEND(801) 1072
3-30-3 ПРИОСТАНОВКА/ПЕРЕЗАПУСК ПРОГРАММНОГО БЛОКА: BPPS(811)/BPRS(812) 1075
3-30-4 Ветвление: IF(802), ELSE(803) и IEND(804) 1077

3-30-5 ВЫХОД ИЗ БЛОКА ПО УСЛОВИЮ (НЕ): EXIT (NOT)(806) 1081
3-30-6 ОДИН ЦИКЛ И ОЖИДАНИЕ (НЕ): WAIT(805)/WAIT(805) NOT 1084
3-30-7 ТАЙМЕР ОЖИДАНИЯ: TIMW(813) и TIMWX(816) 1088
3-30-8 СЧЕТЧИК ОЖИДАНИЯ: CNTW(814) и CNTWX(818) 1092
3-30-9 СКОРОСТНОЙ ТАЙМЕР ОЖИДАНИЯ: TMHW(815) и TMHWX(817) 1096
3-30-10 Создание цикла: LOOP(809)/LEND(810)/LEND(810) NOT 1099
3-31 Команды обработки текстовых строк 1103
3-31-1 Обзор операций над текстовыми строками 1103
3-31-2 КОПИРОВАТЬ СТРОКУ: MOV$(664) 1104
3-31-3 ОБЪЕДИНИТЬ СТРОКИ: +$(656) 1106
3-31-4 СКОПИРОВАТЬ СИМВОЛЫ СЛЕВА: LEFT$(652) 1109
3-31-5 СКОПИРОВАТЬ СИМВОЛЫ СПРАВА: RGHT$(653) 1111
3-31-6 СКОПИРОВАТЬ СИМВОЛЫ ВНУТРИ СТРОКИ: MID$(654) 1113
3-31-7 НАЙТИ В СТРОКЕ: FIND$(660) 1116
3-31-8 ДЛИНА СТРОКИ: LEN$(650) 1118
3-31-9 ЗАМЕНИТЬ В СТРОКЕ: RPLC$(661) 1120
3-31-10 УДАЛИТЬ СТРОКУ: DEL$(658) 1123
3-31-11 ОБМЕН СТРОКАМИ: XCHG$(665) 1125
3-31-12 ОЧИСТИТЬ СТРОКУ: CLR$(666) 1127
3-31-13 ВСТАВИТЬ В СТРОКУ: INS$(657) 1129
3-31-14 Команды сравнения строк (670..675) 1132
3-32 Команды управления задачами 1137
3-32-1 ВКЛЮЧИТЬ ЗАДАЧУ: TKON(820) 1137
3-32-2 ВЫКЛЮЧИТЬ ЗАДАЧУ: TKOF(821) 1141
3-33 Команды преобразования моделей 1146
3-33-1 ПЕРЕДАЧА БЛОКА: XFERC(565) 1148
3-33-2 РАЗМЕЩЕНИЕ ОДИНОЧНОГО СЛОВА: DISTC(566) 1150
3-33-3 СБОР ДАННЫХ: COLLC(567) 1154
3-33-4 ПЕРЕМЕЩЕНИЕ БИТА: MOVBC(568) 1158
3-33-5 ПОДСЧЕТ БИТОВ: BCNTC(621) 1160
3-33-6 СЧИТАТЬ ИДЕНТИФИКАТОР ПЕРЕМЕННОЙ: GETID(286) 1162

3-1 Условные обозначения и структура информации при описании команд

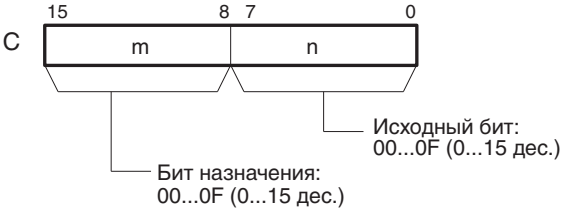
Описываемые команды сгруппированы по выполняемым ими функциям. *Приложение C Список команд в алфавитном порядке* содержит алфавитный перечень мнемонических обозначений команд с указанием номера страницы с описанием для каждой команды.

Все разделы с описанием команд программирования имеют общую структуру, которая описана в следующей таблице.

<table>
<tr><th colspan="2">Пункт</th><th>Содержание</th></tr>
<tr><td colspan="2">Наименование и мнемоническое обозначение</td><td>Заголовок каждого раздела содержит наименование команды, за которым следует ее мнемоническое обозначение и код функции в скобках. Пример: ПЕРЕДАТЬ БИТ: MOVB(082)</td></tr>
<tr><td colspan="2">Назначение</td><td>Сразу после заголовка раздела кратко формулируется основное назначение команды.</td></tr>
<tr><td colspan="2">Символ РКС и названия операндов</td><td>Символ РКС (РКС = релейно-контактная схема) служит для отображения команды в программе CX-Programmer. Ниже приведен пример символа РКС для команды ПЕРЕДАТЬ БИТ. Символ РКС также содержит буквенные обозначения всех операндов команды.
 

S: Исходное слово или данные
C: Управляющее слово
D: Адресуемое слово</td></tr>
<tr><td rowspan="2">Варианты выполнения</td><td>Варианты выполнения</td><td>Возможные варианты выполнения команды в зависимости от определенных условий и соответствующие им мнемонические обозначения. Если вариант выполнения командой не поддерживается, для него указывается «Не предусмотрено».
• Выполнение в каждом цикле при включенном условии выполнения: команда выполняется в каждом цикле, пока включено («1») ее условие выполнения.
• Однократное выполнение по положительному фронту: команда выполняется только в одном цикле после переключения условия выполнения из состояния «0» в состояние «1».
• Однократное выполнение по отрицательному фронту: команда выполняется только в одном цикле после переключения условия выполнения из состояния «1» в состояние «0».
• Постоянное выполнение: команда не требует условия выполнения и выполняется в каждом цикле.
• Включение условия выполнения....: команда выполняется в каждом цикле с целью включения условия выполнения для следующей команды.
<table><tr><td rowspan="3">Варианты выполнения</td><td>Выполнение в каждом цикле при включенном условии выполнения</td><td>MOVB(082)</td></tr><tr><td>Однократное выполнение по положительному фронту</td><td>@MOVB(082)</td></tr><tr><td>Однократное выполнение по отрицательному фронту</td><td>Не предусмотрено</td></tr></table></td></tr>
<tr><td>Варианты выполнения</td><td>Модификатор мгновенного обновления</td><td>Для некоторых команд может быть указано мгновенное обновление, означающее обновление данных в области ввода/вывода непосредственно при выполнении команды. Если команда поддерживает мгновенное обновление, для нее приводится соответствующее мнемоническое обозначение. Если мгновенное обновление командой не поддерживается, указывается «Не предусмотрено».
<table><tr><td>Модификатор мгновенного обновления</td><td>Не предусмотрено</td></tr></table></td></tr>
</table>

Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

<table>
<tr><th>Пункт</th><th>Содержание</th></tr>
<tr><td>Доступные области программы</td><td>Указываются области программы, в которых возможно использование данной команды. «OK» означает, что команда может использоваться в данной области.
<table>
<tr><th>Области программных блоков</th><th>Области пошаговых программ</th><th>Подпрограммы</th><th>Задачи обработки прерываний</th></tr>
<tr><td>OK</td><td>OK</td><td>OK</td><td>OK</td></tr>
</table></td></tr>
<tr><td>Операнды</td><td>При необходимости, указывается назначение слов и битов определенных операндов (например, управляющих слов).

</td></tr>
<tr><td>Характеристики операндов</td><td>В таблице, пример которой приведен ниже, для каждого операнда команды указываются доступные для использования адреса областей памяти. В заголовке таблицы используются те же буквенные обозначения операндов, что и в символе РКС. Области памяти, недоступные для данного операнда, обозначаются символом «---».
<table>
<tr><th>Область</th><th>S</th><th>C</th><th>D</th></tr>
<tr><td>Область CIO</td><td colspan="3">CIO 0...CIO 6143</td></tr>
<tr><td>Рабочая область</td><td colspan="3">W0...W511</td></tr>
<tr><td>Область битов хранения</td><td colspan="3">H0...H511</td></tr>
<tr><td>Область вспомогательных битов</td><td colspan="2">A0...A959</td><td>A448...A959</td></tr>
<tr><td>Область таймеров</td><td colspan="3">T0000...T4095</td></tr>
<tr><td>Область счетчиков</td><td colspan="3">C0000...C4095</td></tr>
<tr><td>Область DM</td><td colspan="3">D0...D32767</td></tr>
</table></td></tr>
<tr><td>Описание</td><td>Более развернуто описывается функционирование команды и ее операндов.</td></tr>
<tr><td>Флаги</td><td>В таблице флагов указываются состояния, в которых находятся флаги условий непосредственно после выполнения команды. Флаги, на состояние которых команда не влияет, в таблице отсутствуют. «ВЫКЛ» означает, что сразу после выполнения команды, независимо от результата ее выполнения, данный флаг находится в выключенном состоянии (сброшен).
<table>
<tr><th>Название</th><th>Обозначение</th><th>Описание работы</th></tr>
<tr><td>Флаг ошибки</td><td>ER</td><td>Включен, если управляющие данные и другие операнды не соответствуют требованиям (выходят за допустимые пределы и т. п.).
Выключен во всех остальных случаях.</td></tr>
<tr><td>Флаг равенства</td><td>=</td><td>ВЫКЛ</td></tr>
<tr><td>Флаг отрицательного значения</td><td>N</td><td>ВЫКЛ</td></tr>
</table></td></tr>
<tr><td>Меры предосторожности</td><td>Особые меры предосторожности и нюансы, которые необходимо учитывать при использовании данной команды. Обязательно ознакомьтесь и соблюдайте все указанные меры предосторожности.</td></tr>
<tr><td>Пример</td><td>Пример применения команды с конкретными значениями операндов, дополнительно поясняющий функционирование команды.</td></tr>
</table>

Постоянные

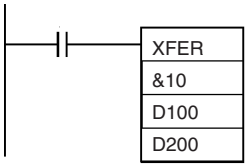
В качестве операндов команд допускается указывать константы с соблюдением следующих правил.

Описание операндов и характеристики операндов

- Операнды в формате битовых строк (обычно вводятся в шестнадцатеричном формате):
  Ввод операндов, указывающих битовые строки, возможен только в шестнадцатеричном формате. Например, для операнда S команды MOV(021) будет указан только один диапазон возможных значений: #0000...#FFFF. В то же время, в CX-Programmer битовые строки можно вводить в десятичном формате, используя префикс &.
- Операнды в формате числовых значений (обычно вводятся в десятичном формате, включая номера переходов):
  Ввод операндов, указывающих числовые значения, возможен как в десятичном, так и в шестнадцатеричном формате. Например, для операнда N команды XFER(070) будут указаны два диапазона возможных значений : #0000...#FFFF и &0...&65535.
- Операнды с числовыми значениями для управления (кроме номеров переходов):
  Ввод числовых значений, предназначенных для управления работой команды, возможен только в десятичном формате. Например, для операнда N команды SBS(091) будет указан диапазон возможных значений: 0...1023.

Примеры

В примерах используется форма записи констант, применяемая в CX-Programmer: для операндов, указывающих числовые значения, приводится десятичное значение с префиксом &, как показано на следующем рисунке.



В следующей таблице перечислены способы ввода констант в программе CX-Programmer.

Операнд	CX-Programmer
Операнды, указывающие битовые строки (обычно вводятся в шестнадцатеричном формате)	Вводятся в десятичном формате с префиксом & или в шестнадцатеричном формате с префиксом # (см. примечание).
Операнды, указывающие числовые значения (обычно вводятся в десятичном формате)	
Операнды, указывающие числовые значения для управления (кроме номеров переходов)	Вводятся в десятичном формате с префиксом # (см. примечание).

Примечание. Во время ввода операндов в программе CX-Programmer отображаются диапазоны возможных значений и соответствующие префиксы.

Флаги условий

В данном разделе для идентификации флагов используются названия флагов. В CX-Programmer для флагов условий используются зарезервированные глобальные символы.

Название флага (используется в данном разделе)	Обозначение в CX-Programmer
Флаг ошибки	P_ER
Флаг ошибки доступа	P_AER

Название флага (используется в данном разделе)	Обозначение в CX-Programmer
Флаг переноса	P_CY
Флаг большего значения («Больше»)	P_GT
Флаг равенства («Равно»)	P_EQ
Флаг меньшего значения («Меньше»)	P_LT
Флаг отрицательного значения («Меньше нуля»)	P_N
Флаг переполнения	P_OF
Флаг потери значимости	P_UF
Флаг большего значения или равенства («Больше или равно»)	P_GE
Флаг неравенства («Не равно»)	P_NE
Флаг меньшего значения или равенства («Меньше или равно»)	P_LE
Флаг постоянного включения («Всегда ВКЛ»)	P_On
Флаг постоянного выключения («Всегда ВЫКЛ»)	P_Off

Предостережения в отношении адресов области DM модулей ЦПУ CP1L L

В модулях ЦПУ CP1L L объем области DM меньше, чем в других модулях ЦПУ серии CP. В настоящем руководстве по программированию характеристики операндов приведены для модулей ЦПУ CP1H и CP1L M (на 30 или 40 входов/выходов). Для модулей ЦПУ серии CP1L L (на 14 или 20 входов/выходов) часть указанных адресов области DM может быть недоступна.

При создании программы в CX-Programmer ввод адресов в области DM вне допустимого диапазона невозможен. Если недопустимый адрес области DM все же будет введен в программу, при загрузке программы в ПЛК произойдет ошибка.

Следующая таблица содержит пример диапазонов адресов области DM, доступных для модулей ЦПУ CP1L L.

Модули ЦПУ CP1H и CP1L M

Область DM	D00000...D32767
Косвенные адреса DM в двоичном формате	@D00000...@D32767
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D00000...*D32767

⇨

Модули ЦПУ CP1L L

Область DM	D00000...D09999, D32000...D32767
Косвенные адреса DM в двоичном формате	@D00000...@D09999, @D30000...@D32767
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D00000...*D09999, *D30000...*D32767

3-2 Входные битовые команды

3-2-1 ЗАГРУЗКА: LD

Назначение

Указывает начало логической цепи (лестничной диаграммы) и формирует условие выполнения (ВКЛ/ВЫКЛ), которое определяется состоянием указанного битового операнда (ВКЛ/ВЫКЛ).

Символ РКС



Варианты выполнения

Варианты выполнения	**Начинает работу логической цепи, находится во включенном состоянии в каждом цикле, в котором включен бит операнда.**	LD
	Начинает работу логической цепи, включается однократно по положительному фронту.	@LD
	Начинает работу логической цепи, включается однократно по отрицательному фронту.	%LD
Модификатор мгновенного обновления		!LD
Комбинированная модификация	**Обновляет бит входа, начинает работу логической цепи, включается однократно по положительному фронту.**	!@LD
	Обновляет бит входа, начинает работу логической цепи, включается однократно по отрицательному фронту.	!%LD

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	Битовый операнд LD
Область CIO	CIO 0.00...CIO 6143.15
Рабочая область	W0.00...W511.15
Область битов хранения	H0.00...H511.15
Область вспомогательных битов	A0.00...A959.15
Область таймеров	T0000...T4095
Область счетчиков	C0000...C4095
Область флагов задач	TK00...TK31
Флаги условий	ER, CY, N, OF, UF, >, =, <, >=, <>, <=, A1, A0
Тактовые импульсы	0,0 / 2 с / 0,1 с / 0,2 с / 1 с / 1 мин
Область TR	TR0...TR15
Область DM	---

Область	Битовый операнд LD
Косвенные адреса DM в двоичном формате	---
Косвенные адреса DM в двоично-десятичном формате (BCD)	---
Постоянные	---
Регистры данных	---
Регистры указателей	---
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15

Описание

Команда LD используется в качестве самого первого нормально разомкнутого контакта (бита) логической цепи (присоединяемого непосредственно к левой шине) или самого первого нормально разомкнутого контакта (бита) логического блока. Если мгновенное обновление не указано, считывается состояние указанного бита из памяти ввода/вывода. Если указано мгновенное обновление, считывается и используется состояние физического входа базового модуля ввода.

Команда LD используется для обозначения начала логической цепи в следующих случаях.

- В случае прямого подключения к шине.
- В случае соединения логических блоков с помощью команд AND LD или OR LD, то есть в начале логического блока.

Команды AND LD (И ЗАГРУЗКА) и OR LD (ИЛИ ЗАГРУЗКА) служат для последовательного или параллельного соединения логических блоков, начинающихся с команд LD (ЗАГРУЗКА) или LD NOT (ЗАГРУЗКА НЕ).

Если команды, связанные с выводом, не могут быть присоединены непосредственно к левой шине, требуется как минимум одна команда LD или LD NOT в качестве условия выполнения. В противном случае при проверке программы в CX-Programmer будет обнаружена ошибка программы.

Если логические блоки соединяются с помощью команд AND LD или OR LD, общее количество команд AND LD/OR LD должно быть на 1 меньше, чем общее количество команд LD/LD NOT. В противном случае будет обнаружена ошибка программы. Подробные сведения смотрите в разделах *3-2-7 И ЗАГРУЗКА: AND LD* и *3-2-8 ИЛИ ЗАГРУЗКА: OR LD*.

Флаги

Данная команда не изменяет состояние какого-либо флага.

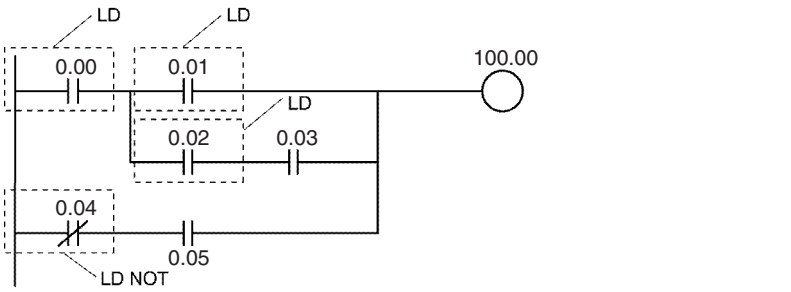
Меры предосторожности

Для команды LD может быть указано однократное выполнение по положительному (@) или отрицательному (%) фронту. Если указан положительный фронт (@), условие выполнения включается после переключения битового операнда из «0» в «1» и остается включенным только в течение одного цикла. Если указан отрицательный фронт (%), условие выполнения включается после переключения битового операнда из «1» в «0» и остается включенным только в течение одного цикла.

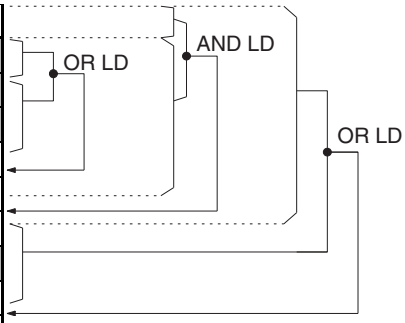
Для команды LD может быть указано мгновенное обновление (!). Команда, для которой указано мгновенное обновление, непосредственно перед выполнением обновляет состояние бита, соответствующего встроенному входу модуля ЦПУ.

Для команды LD также можно указать мгновенное обновление в комбинации с однократным выполнением по положительному или отрицательному фронту (!@ или !%). Если указана одна из этих комбинаций, состояние бита встроенного входа модуля ЦПУ обновляется непосредственно перед выполнением команды. Условие выполнения включается и остается включенным в течение одного цикла, если состояние бита переключилось из «0» в «1» или из «1» в «0».

Пример



Команда	Операнд
LD	0.00
LD	0.01
LD	0.02
AND	0.03
OR LD	---
AND LD	---
LD NOT	0.04
AND	0.05
OR LD	---
OUT	100.00

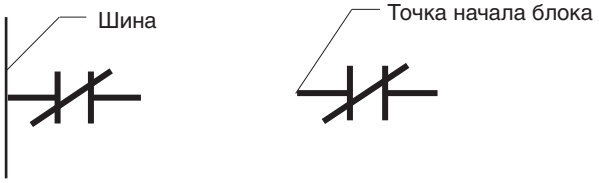


3-2-2 ЗАГРУЗКА НЕ: LD NOT

Назначение

Указывает начало логической цепи (лестничной диаграммы) и формирует условие выполнения (ВКЛ/ВЫКЛ), которое определяется инверсией состояния указанного битового операнда (ВЫКЛ/ВКЛ).

Символ РКС



Варианты выполнения

Варианты выполнения	**Начинает работу логической цепи, находится во включенном состоянии в каждом цикле, в котором выключен бит операнда.**	LD NOT
	Начинает работу логической цепи, включается однократно по положительному фронту.	@LD NOT
	Начинает работу логической цепи, включается однократно по отрицательному фронту.	%LD NOT
Модификатор мгновенного обновления		!LD NOT
Комбинированная модификация	**Обновляет бит входа, начинает работу логической цепи, включается однократно по положительному фронту.**	!@LD NOT
	Обновляет бит входа, начинает работу логической цепи, включается однократно по отрицательному фронту.	!%LD NOT

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	Битовый операнд LD NOT
Область CIO	CIO 0.00...CIO 6143.15
Рабочая область	W0.00...W511.15
Область битов хранения	H0.00...H511.15
Область вспомогательных битов	A0.00...A959.15
Область таймеров	T0000...T4095
Область счетчиков	C0000...C4095
Область флагов задач	TK00...TK31
Флаги условий	ER, CY, N, OF, UF, >, =, <, >=, <>, <=, ON, OFF, AER
Тактовые импульсы	0,02 с / 0,1 с / 0,2 с / 1 с / 1 мин
Область TR	---
Область DM	---
Косвенные адреса DM в двоичном формате	---
Косвенные адреса DM в двоично-десятичном формате (BCD)	---
Постоянные	---
Регистры данных	---
Регистры указателей	---
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15

Описание

Команда LD NOT используется в качестве самого первого нормально замкнутого контакта (бита) логической цепи (присоединяемого непосредственно к левой шине) или самого первого нормально замкнутого контакта (бита) логического блока. Если мгновенное обновление не указано, считывается и инвертируется состояние указанного бита из памяти ввода/вывода. Если указано мгновенное обновление, считывается, инвертируется и используется состояние физического входа базового модуля ввода.

Команда LD NOT используется для обозначения начала логической цепи в следующих случаях.

- В случае прямого подключения к шине.
- В случае соединения логических блоков с помощью команд AND LD или OR LD (используется в начале логического блока).

Команды AND LD (И ЗАГРУЗКА) и OR LD (ИЛИ ЗАГРУЗКА) служат для последовательного или параллельного соединения логических блоков, начинающихся с LD (ЗАГРУЗКА) или LD NOT (ЗАГРУЗКА НЕ).

Если команды, связанные с выводом, не могут быть присоединены непосредственно к левой шине, требуется как минимум одна команда LD или LD NOT в качестве условия выполнения. В противном случае при проверке программы в CX-Programmer будет обнаружена ошибка программы.

Если логические блоки соединяются с помощью команд AND LD или OR LD, общее количество команд AND LD/OR LD должно быть на 1 меньше, чем общее количество команд LD/LD NOT. В противном случае будет обнаружена ошибка программы.

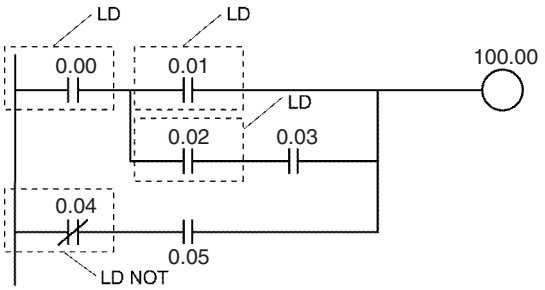
Флаги

Данная команда не изменяет состояние какого-либо флага.

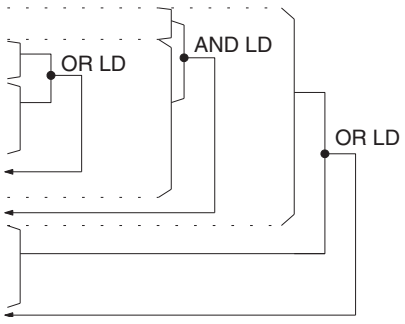
Меры предосторожности

Для команды LD NOT может быть указано мгновенное обновление (!). Если для команды указано мгновенное обновление, состояние входного бита, соответствующего встроенному входу модуля ЦПУ, обновляется непосредственно перед выполнением команды.

Пример



Команда	Операнд
LD	0.00
LD	0.01
LD	0.02
AND	0.03
OR LD	---
AND LD	---
LD NOT	0.04
AND	0.05
OR LD	---
OUT	100.00



3-2-3 И: AND

Назначение

Операция логического умножения («И») над состоянием указанного битового операнда и текущим условием выполнения.

Символ РКС



Варианты выполнения

Варианты выполнения	**Включен в каждом цикле, в котором результат AND = ВКЛ.**	AND
	Включается однократно по положительному фронту.	@AND
	Включается однократно по отрицательному фронту.	%AND
Модификатор мгновенного обновления		!AND
Комбинированная модификация	**Обновляет бит входа, включается однократно по положительному фронту.**	!@AND
	Обновляет бит входа, включается однократно по отрицательному фронту.	!%AND

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	Битовый операнд AND
Область CIO	CIO 0.00...CIO 6143.15
Рабочая область	W0.00...W511.15
Область битов хранения	H0.00...H511.15
Область вспомогательных битов	A0.00...A959.15
Область таймеров	T0000...T4095
Область счетчиков	C0000...C4095
Область флагов задач	TK00...TK31
Флаги условий	ER, CY, N, OF, UF, >, =, <, >=, <>, <=, ON, OFF, AER
Тактовые импульсы	0,02 с / 0,1 с / 0,2 с / 1 с / 1 мин
Область TR	---
Область DM	---
Косвенные адреса DM в двоичном формате	---
Косвенные адреса DM в двоично-десятичном формате (BCD)	---
Постоянные	---
Регистры данных	---
Регистры указателей	---
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15

Описание

Команда AND служит для последовательного подключения нормально разомкнутого контакта (бита). Команду AND нельзя подключать непосредственно к левой шине и невозможно использовать в начале логического блока. Если мгновенное обновление не указано, считывается состояние указанного бита из памяти ввода/вывода. Если указано мгновенное обновление, считывается состояние физического входа модуля ЦПУ.

Флаги

Данная команда не изменяет состояние какого-либо флага.

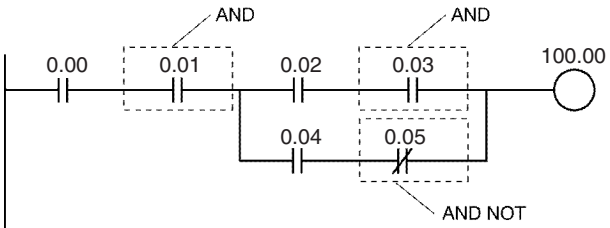
Меры предосторожности

Для команды AND может быть указано выполнение по положительному (@) или отрицательному (%) фронту. Если указано выполнение по положительному фронту (@), условие выполнения включается только на один цикл, после того как включается (0->1) битовый операнд. Если указано выполнение по отрицательному фронту (%), условие выполнения включается только на один цикл, после того как выключается (1->0) битовый операнд.

Для команды AND может быть указано мгновенное обновление (!). Если для команды указано мгновенное обновление, состояние входного бита, соответствующего встроенному входу модуля ЦПУ, обновляется непосредственно перед выполнением команды.

Для команды AND также можно указать мгновенное обновление в комбинации с однократным выполнением по положительному или отрицательному фронту (!@ или !%). Если указана одна из этих комбинаций, состояние бита входа обновляется модулем ЦПУ непосредственно перед выполнением команды. Условие выполнения включается и остается включенным в течение одного цикла, если состояние бита переключилось из «0» в «1» или из «1» в «0».

Пример



Команда	Операнд
LD	0.00
AND	0.01
LD	0.02
AND	0.03
LD	0.04
AND NOT	0.05
OR LD	---
AND LD	---
OUT	100.00

3-2-4 И НЕ: AND NOT

Назначение

Операция логического умножения («И») над инверсией состояния указанного битового операнда и текущим условием выполнения.

Символ РКС



Варианты выполнения

Варианты выполнения	**Включен в каждом цикле, в котором результат AND NOT = ВКЛ.**	AND NOT
	Включается однократно по положительному фронту.	@AND NOT
	Включается однократно по отрицательному фронту.	%AND NOT
Модификатор мгновенного обновления		!AND NOT
Комбинированная модификация	**Обновляет бит входа, включается однократно по положительному фронту.**	!@AND NOT
	Обновляет бит входа, включается однократно по отрицательному фронту.	!%AND NOT

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	Битовый операнд AND NOT
Область CIO	CIO 0.00...CIO 6143.15
Рабочая область	W0.00...W511.15
Область битов хранения	H0.00...H511.15
Область вспомогательных битов	A0.00...A959.15
Область таймеров	T0000...T4095
Область счетчиков	C0000...C4095
Область флагов задач	TK00...TK31
Флаги условий	ER, CY, N, OF, UF, >, =, <, >=, <>, <=, ON, OFF, AER
Тактовые импульсы	0,02 с / 0,1 с / 0,2 с / 1 с / 1 мин
Область TR	---
Область DM	---
Косвенные адреса DM в двоичном формате	---
Косвенные адреса DM в двоично-десятичном формате (BCD)	---
Постоянные	---
Регистры данных	---
Регистры указателей	---
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15

Описание

Команда AND NOT служит для последовательного подключения нормально замкнутого контакта (бита). Команду AND NOT нельзя подключать непосредственно к левой шине и невозможно использовать в начале логического блока. Если мгновенное обновление не указано, считывается состояние указанного бита из памяти ввода/вывода. Если мгновенное обновление указано, считывается состояние физического входа модуля ЦПУ.

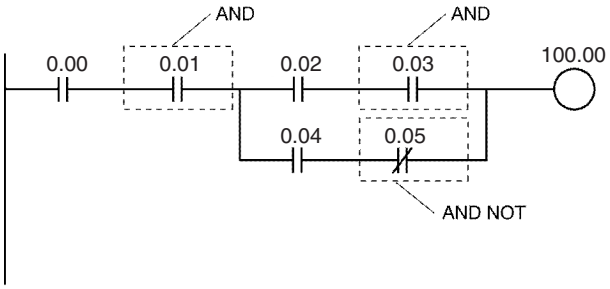
Флаги

Данная команда не изменяет состояние какого-либо флага.

Меры предосторожности

Для команды AND NOT может быть указано мгновенное обновление (!). Если для команды указано мгновенное обновление, состояние входного бита, соответствующего встроенному входу модуля ЦПУ, обновляется непосредственно перед выполнением команды.

Пример



Команда	Операнд
LD	0.00
AND	0.01
LD	0.02
AND	0.03
LD	0.04
AND NOT	0.05
OR LD	---
AND LD	---
OUT	100.00

3-2-5 ИЛИ: OR

Назначение

Операция логического сложения («ИЛИ») над состоянием (0 или 1) указанного битового операнда и текущим условием выполнения.

Символ РКС



Варианты выполнения

Варианты выполнения	**Включен в каждом цикле, в котором результат OR = ВКЛ.**	ИЛИ
	Включается однократно по положительному фронту.	@OR
	Включается однократно по отрицательному фронту.	%OR
Модификатор мгновенного обновления		!OR
Комбинированная модификация	**Обновляет бит входа, включается однократно по положительному фронту.**	!@OR
	Обновляет бит входа, включается однократно по отрицательному фронту.	!%OR

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	Битовый операнд OR
Область CIO	CIO 0.00...CIO 6143.15
Рабочая область	W0.00...W511.15
Область битов хранения	H0.00...H511.15
Область вспомогательных битов	A0.00...A959.15
Область таймеров	T0000...T4095
Область счетчиков	C0000...C4095
Область флагов задач	TK00...TK31
Флаги условий	ER, CY, N, OF, UF, >, =, <, >=, <>, <=, ON, OFF, AER
Тактовые импульсы	0,02 с / 0,1 с / 0,2 с / 1 с / 1 мин
Область DM	---
Косвенные адреса DM в двоичном формате	---
Косвенные адреса DM в двоично-десятичном формате (BCD)	---
Постоянные	---
Регистры данных	---
Регистры указателей	---
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15

Описание

Команда OR служит для параллельного подключения нормально разомкнутого контакта (бита). Нормально разомкнутый контакт подключается параллельно (т. е. по правилу «ИЛИ») логическому блоку, начинающемуся с команды LD или LD NOT (подключается к левой шине или к началу логического блока). Если мгновенное обновление не указано, считывается состояние указанного бита из памяти ввода/вывода. Если указано мгновенное обновление, считывается состояние физического входа модуля ЦПУ.

Флаги

Данная команда не изменяет состояние какого-либо флага.

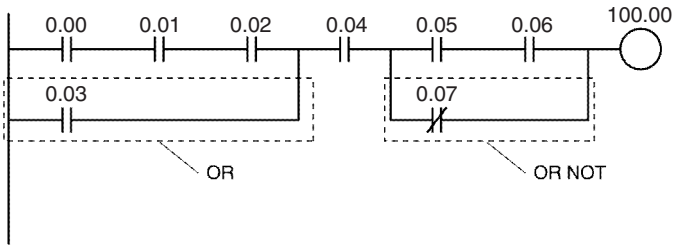
Меры предосторожности

Для команды OR может быть указано выполнение по положительному (@) или отрицательному (%) фронту. Если указано выполнение по положительному фронту (@), условие выполнения включается только на один цикл, после того как включается (0->1) битовый операнд. Если указано выполнение по отрицательному фронту (%), условие выполнения включается только на один цикл, после того как выключается (1->0) битовый операнд.

Для команды OR может быть указано мгновенное обновление (!). Если для команды указано мгновенное обновление, состояние входного бита, соответствующего встроенному входу модуля ЦПУ, обновляется непосредственно перед выполнением команды.

Для команды OR также можно указать мгновенное обновление в комбинации с однократным выполнением по положительному или отрицательному фронту (!@ или !%). Если указана одна из этих комбинаций, состояние бита входа обновляется модулем ЦПУ непосредственно перед выполнением команды. Условие выполнения включается и остается включенным в течение одного цикла, если состояние битового операнда переключилось из «0» в «1» или из «1» в «0».

Пример



Команда	Операнд
LD	0.00
AND	0.01
AND	0.02
OR	0.03
AND	0.04
LD	0.05
AND	0.06
OR NOT	0.07
AND LD	---
OUT	100.00

3-2-6 ИЛИ НЕ: OR NOT

Назначение

Операция логического сложения («ИЛИ») над инверсией состояния указанного бита и текущим условием выполнения.

Символ РКС



Варианты выполнения

Варианты выполнения	**Включен в каждом цикле, в котором результат OR NOT = ВКЛ.**	OR NOT
	Включается однократно по положительному фронту.	@OR NOT
	Включается однократно по отрицательному фронту.	%OR NOT
Модификатор мгновенного обновления		!OR NOT
Комбинированная модификация	**Обновляет бит входа, включается однократно по положительному фронту.**	!@OR NOT
	Обновляет бит входа, включается однократно по отрицательному фронту.	!%OR NOT

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограм-мы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	Битовый операнд OR NOT
Область CIO	CIO 0.00...CIO 6143.15
Рабочая область	W0.00...W511.15
Область битов хранения	H0.00...H511.15
Область вспомогательных битов	A0.00...A959.15
Область таймеров	T0000...T4095
Область счетчиков	C0000...C4095
Область флагов задач	TK00...TK31
Флаги условий	ER, CY, N, OF, UF, >, =, <, >=, <>, <=, A1, A0
Тактовые импульсы	0,02 с / 0,1 с / 0,2 с / 1 с / 1 мин
Область TR	---
Область DM	---
Косвенные адреса DM в двоичном формате	---
Косвенные адреса DM в двоично-десятичном формате (BCD)	---
Постоянные	---
Регистры данных	---
Регистры указателей	---
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15

Описание

Команда OR NOT служит для параллельного подключения нормально замкнутого контакта (бита). Нормально замкнутый контакт подключается параллельно (т. е. по правилу «ИЛИ») логическому блоку, начинающемуся с команды LD или LD NOT (подключается к левой шине или к началу логического блока). Если мгновенное обновление не указано, считывается состояние указанного бита из памяти ввода/вывода. Если указано мгновенное обновление, считывается состояние физического входа модуля ЦПУ.

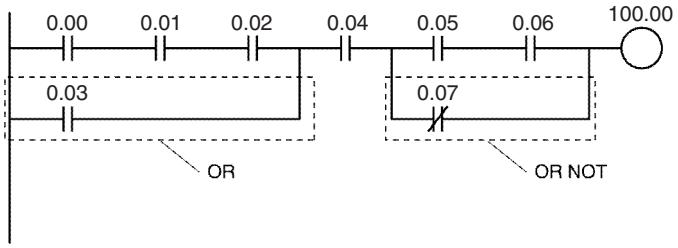
Флаги

Данная команда не изменяет состояние какого-либо флага.

Меры предосторожности

Для команды OR NOT может быть указано мгновенное обновление (!). Команда, для которой указано мгновенное обновление, непосредственно перед выполнением обновляет состояние бита, соответствующего дискретному входу модуля ЦПУ.

Пример



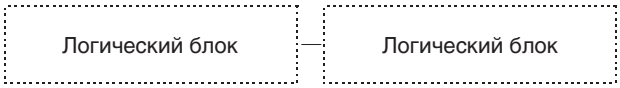
Команда	Операнд
LD	0.00
AND	0.01
AND	0.02
OR	0.03
AND	0.04
LD	0.05
AND	0.06
OR NOT	0.07
AND LD	---
OUT	100.00

3-2-7 И ЗАГРУЗКА: AND LD

Назначение

Операция логического умножения («И») над двумя логическими блоками.

Символ РКС



Варианты выполнения

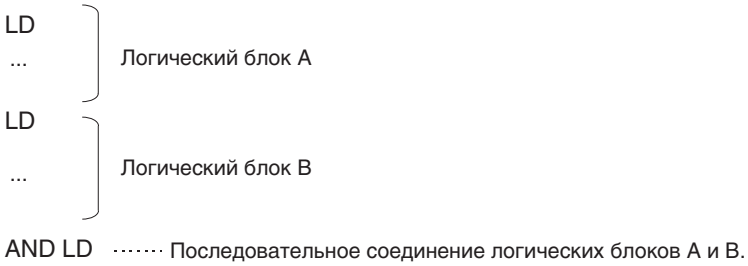
Варианты выполнения	**Включен в каждом цикле, в котором результат AND = ВКЛ.**	AND LD
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

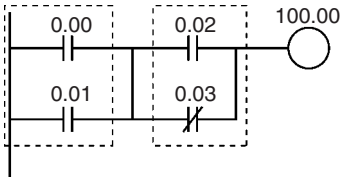
Описание

Команда AND LD последовательно соединяет логический блок, расположенный непосредственно перед этой командой, с другим логическим блоком.



Логический блок — это группа команд, начинающаяся с команды LD или LD NOT и заканчивающаяся выходом на следующую команду LD или LD NOT в пределах одной цепи.

На приведенном ниже рисунке два логических блока обведены пунктирными линиями. Проанализировав этот пример, можно видеть, что выходное условие выполнения включено, если включено любое из условий выполнения в логическом блоке слева (т. е., если CIO 0.00 или CIO 0.01 = «1») **И** включено любое из условий выполнения в логическом блоке справа (т. е., если CIO 0.02 = «1» или CIO 0.03 = «0»).



Флаги

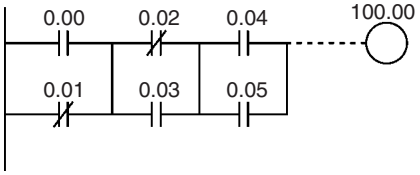
Данная команда не изменяет состояние какого-либо флага.

Меры предосторожности

Данная команда позволяет соединить последовательно три или большее количество логических блоков. Для этого следует сначала соединить два логических блока, а затем присоединить к ним следующий и все остальные блоки по порядку. Данную команду также можно разместить после трех или большего количества логических блоков, соединяя их последовательно.

Если логические блоки соединяются с помощью команд AND LD или OR LD, общее количество команд AND LD/OR LD должно быть на 1 меньше, чем общее количество команд LD/LD NOT. В противном случае будет обнаружена ошибка программы.

Пример



Пример программы (1)

Команда	Операнд
LD	0.00
OR NOT	0.01
LD NOT	0.02
OR	0.03
AND LD	---
LD	0.04
OR	0.05
AND LD	---
. .	. .
OUT	100.00

Пример программы (2)

Команда	Операнд
LD	0.00
OR NOT	0.01
LD NOT	0.02
OR	0.03
LD	0.04
OR	0.05
. .	. .
AND LD	---
AND LD	---
. .	. .
OUT	100.00

Команду AND LD можно использовать в программе несколько раз подряд. В приведенном выше варианте программы (2) команд AND LD меньше (на 1), чем предшествующих им команд LD и LD NOT.

Используя вариант (2), проследите, чтобы общее число команд LD и LD NOT перед командой AND LD не превышало восемь. Если нужно, чтобы их было девять или больше, используйте вариант программы (1). Если таких команд окажется девять или больше при использовании варианта программы (2), при проверке программы CX-Programmer обнаружит ошибку программы.

Код программы

Адрес	Команда	Операнд
000000	LD	0.00
000001	OR	0.01
000002	LD	0.02
000003	OR NOT	0.03
000004	AND LD	---
000005	OUT	100.00

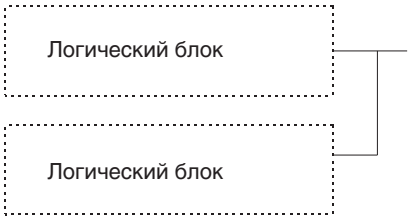
Второй оператор LD: первый контакт (бит) следующего блока, соединенного последовательно с предыдущим блоком.

3-2-8 ИЛИ ЗАГРУЗКА: OR LD

Назначение

Операция логического сложения («ИЛИ») над двумя логическими блоками.

Символ РКС



Варианты выполнения

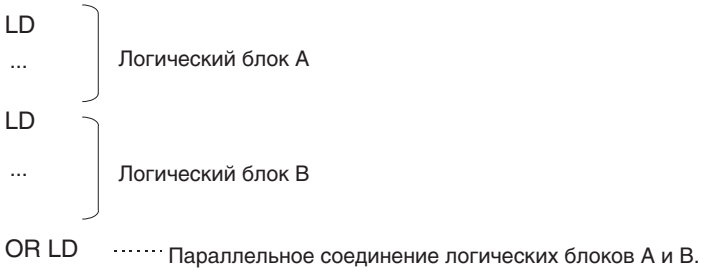
Варианты выполнения	**Включен в каждом цикле, в котором результат OR = ВКЛ.**	OR LD
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

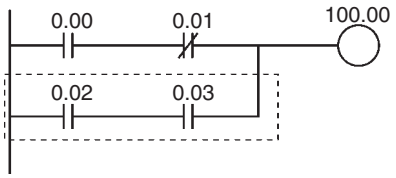
Описание

Команда OR LD служит для параллельного соединения логического блока, расположенного непосредственно перед этой командой, с другим логическим блоком.



Логический блок — это группа команд, начинающаяся с команды LD или LD NOT и заканчивающаяся выходом на следующую команду LD или LD NOT в пределах одной цепи.

На приведенном ниже рисунке между верхним и нижним логическими блоками должна присутствовать команда OR LD. Выходное условие выполнения будет включено, если либо CIO 0.00 = 1 и CIO 0.01 = 0, либо если CIO 0.02 = 1 и CIO 0.03 = 1. Команда OR LD работает точно так же, как команда AND LD, за исключением того, что текущее условие выполнения и предшествующее ему неиспользуемое (внутреннее) условие выполнения логически складываются (а не умножаются).



Флаги

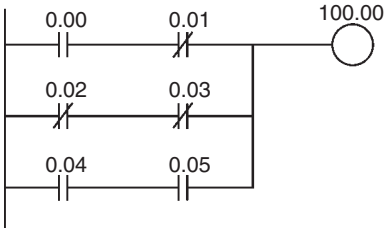
Данная команда не изменяет состояние какого-либо флага.

Меры предосторожности

Данная команда позволяет соединить параллельно три или большее количество логических блоков. Для этого следует сначала соединить два логических блока, а затем присоединить к ним следующий и все остальные блоки по порядку. Данную команду также можно разместить после трех или большего количества логических блоков, соединяя их параллельно.

Если логические блоки соединяются с помощью команд AND LD или OR LD, общее количество команд AND LD/OR LD должно быть на 1 меньше, чем общее количество команд LD/LD NOT. В противном случае будет обнаружена ошибка программы.

Пример



Пример программы (1)

Команда	Операнд
LD	0.00
AND NOT	0.01
LD NOT	0.02
AND NOT	0.03
OR LD	---
LD	0.04
AND	0.05
OR LD	---
. .	. .
OUT	100.00

Пример программы (2)

Команда	Операнд
LD	0.00
AND NOT	0.01
LD NOT	0.02
AND NOT	0.03
LD	0.04
AND	0.05
. .	. .
OR LD	---
OR LD	---
. .	. .
OUT	100.00

Команду OR LOAD можно использовать в программе несколько раз подряд. В приведенном выше варианте программы (2) команд OR LOAD меньше (на 1), чем предшествующих им команд LD и LD NOT.

Используя вариант (2), проследите, чтобы общее число команд LD и LD NOT перед командой OR LOAD не превышало восемь. Если нужно, чтобы их было девять или больше, используйте вариант программы (1). Если таких команд окажется девять или больше при использовании варианта программы (2), при проверке программы CX-Programmer обнаружит ошибку программы.

Код программы

Адрес	Команда	Операнд
000100	LD	0.00
000101	AND NOT	0.01
000102	LD	0.02
000103	AND	0.03
000104	OR LD	---
000105	OUT	100.00

Второй оператор LD: первый контакт (бит) следующего блока, соединенного последовательно с предыдущим блоком.

3-2-9 Однократное выполнение по фронту и мгновенное обновление

Команды LD, LD NOT, AND, AND NOT, OR и OR NOT, помимо обычной формы записи, имеют дополнительные формы записи для однократного выполнения по положительному или отрицательному фронту и для мгновенного обновления битов входов/выходов, а также две комбинированные формы записи.

Команды OUT и OUT NOT, помимо обычной формы записи, имеют форму записи для мгновенного обновления битов входов/выходов.

Обычные (циклические) команды, однократные (выполняемые по фронту) команды, команды с мгновенным обновлением и однократные команды с мгновенным обновлением отличаются между собой моментом времени обновления входов/выходов.

Циклические и однократные команды при выполнении используют входные данные, полученные во время предыдущей операции обновления входов/выходов, а результаты выполнения этих команд выводятся во время следующей операции обновления входов/выходов. Под «обновлением входов/выходов» подразумевается обмен данными между внутренней памятью модуля ЦПУ и встроенными входами/выходами модуля ЦПУ, модулями расширения CPM1A и модулями расширения входов/выходов CPM1A.

Дополнительно к описанному выше обновлению входов/выходов (осуществляемому циклически) команда с префиксом мгновенного обновления обменивается данными с модулем входов/выходов, адресному пространству которого принадлежат слова, обрабатываемые командой. Команда с префиксом мгновенного обновления обновляет состояния всех битов того слова, которое содержит указанный бит.

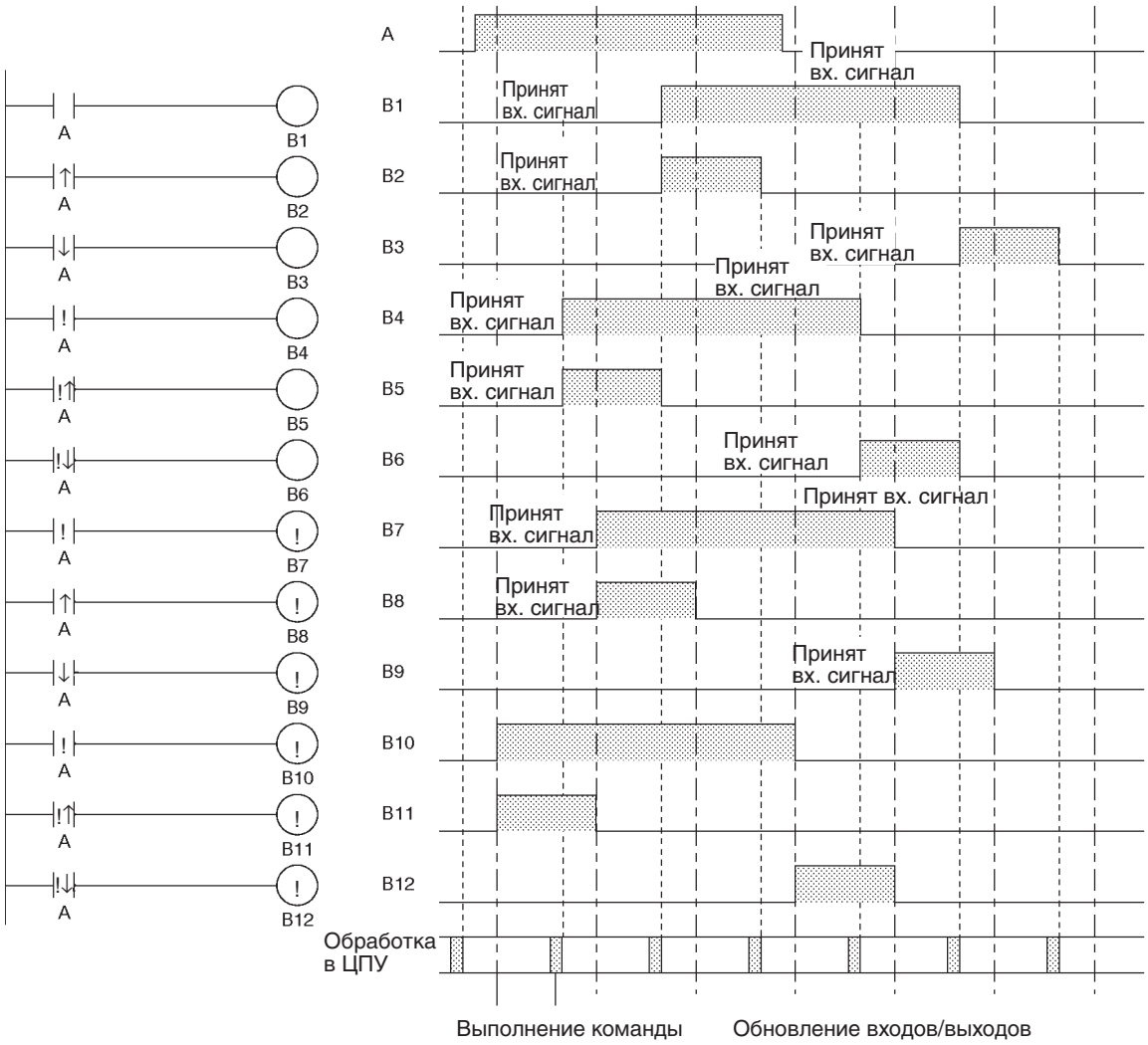
Вариант выполнения	Мнемоническое обозначение	Функция	Обновление входов/выходов
Обычное (циклическое) выполнение	LD, AND, OR, LD NOT, AND NOT, OR NOT	Во время выполнения команды используется состояние (1/0) указанного бита, считанное модулем ЦПУ при обновлении входов/выходов в предыдущем цикле.	Циклическое обновление
	OUT, OUT NOT	Состояние (1/0) указанного бита, полученное при выполнении команды, подается на выход при следующем обновлении входов/выходов.	
Выполнение по положительному фронту	@LD, @AND, @OR	Команда выполняется однократно после переключения указанного бита из состояния «0» в состояние «1». Полученное выходное состояние команды длится один цикл.	
Выполнение по отрицательному фронту	%LD, %AND, %OR	Команда выполняется однократно после переключения указанного бита из состояния «1» в состояние «0». Полученное выходное состояние команды длится один цикл.	
Мгновенное обновление	!LD, !AND, !OR, !LD NOT, !AND NOT, !OR NOT	Модуль ЦПУ считывает состояние указанного входа и обновляет состояние соответствующего бита входа непосредственно перед выполнением команды.	Перед выполнением команды
	!OUT, !OUT NOT	Состояние указанного бита выхода подается на выход сразу после выполнения команды.	После выполнения команды

Вариант выполнения	Мнемоническое обозначение	Функция	Обновление входов/ выходов
Положительный фронт/мгновенное обновление	!@LD, !@AND, !@OR	Модуль ЦПУ обновляет состояние указанного бита входа. Если произошло переключение бита из «0» в «1», команда выполняется однократно. Полученное выходное состояние команды сохраняется в течение одного цикла.	Перед выполнением команды
Отрицательный фронт/мгновенное обновление	!%LD, !%AND, !%OR	Модуль ЦПУ обновляет состояние указанного бита входа. Если произошло переключение бита из «1» в «0», команда выполняется однократно. Полученное выходное состояние команды сохраняется в течение одного цикла.	

Примечание. Команды с мгновенным обновлением (т. е. команды с символом «!») могут использоваться только для встроенных входов/выходов модуля ЦПУ. Их невозможно использовать для входов/выходов модулей расширения CPM1A или модулей расширения входов/выходов CPM1A. Для входов/выходов модулей расширения CPM1A или модулей расширения входов/выходов CPM1A следует использовать команду IORF(097).

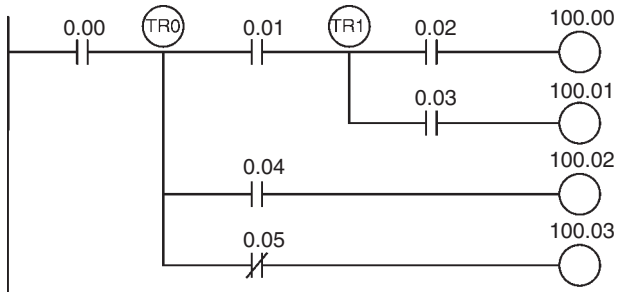
3-2-10 Временные диаграммы работы входных и выходных команд

Представленные ниже временные диаграммы иллюстрируют различия в работе входных (LD) и выходных (OUT) команд с разными вариантами выполнения и обновления.



3-2-11 Биты временного хранения (TR)

Биты TR используются для временного хранения состояний («1» и «0») условий выполнения в том случае, когда программа составляется на языке мнемонических кодов. В случае создания программы непосредственно в виде релейно-контактной схемы эти биты не используются, так как все необходимые операции автоматически выполняются программой CX-Programmer. Ниже показан пример простой программы, использующей два бита TR.

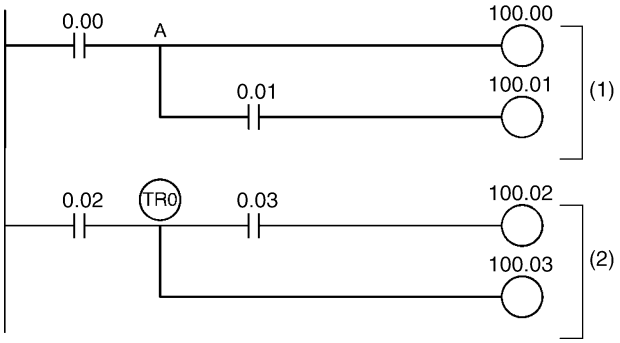


Адрес	Команда	Операнды
00200	LD	0.00
00201	OUT	TR0
00202	AND	0.01
00203	OUT	TR1
00204	AND	0.02
00205	OUT	100.00
00206	LD	TR1
00207	AND	0.03
00208	OUT	100.01
00209	LD	TR0
00210	AND	0.04
00211	OUT	100.02
00212	LD	TR0
00213	AND NOT	0.05
00214	OUT	100.03

Применение битов TR0...TR15

Биты TR0...TR15 используются только для команд LD (ЗАГРУЗКА) и OUT (ВЫВОД). Соблюдать очередность адресов при использовании этих битов не требуется.

В некоторых случаях программу можно упростить, переписав ее таким образом, чтобы биты TR не требовались. На следующем рисунке показано два варианта составления программы. В первом случае бит TR не требуется, а во втором он необходим.



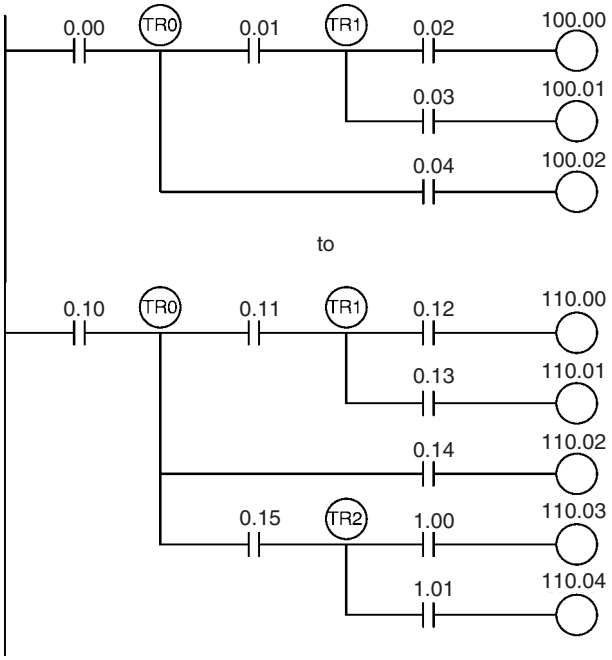
В первом блоке (1) состояние (1/0) в точке A всегда совпадает с состоянием выхода CIO 100.00, поэтому команды AND 0.01 и OUT 100.01 могут быть использованы в программе без бита TR. В блоке (2) состояние в точке ветвления может не совпадать с состоянием выхода CIO 100.02, поэтому необходимо использовать бит TR. Как видно из этого примера, количество шагов программы можно сократить, применив блок (1) вместо блока (2).

Особенности применения битов TR0...TR15

Биты TR предназначены исключительно для сохранения (в случае OUT TR0...TR15) и восстановления (в случае LD TR0...TR15) состояний точек ветвления, когда в программе имеется несколько ветвей вывода. Этим они отличаются от обычных битов, и их нельзя использовать с командами AND или OR, либо с командами, включающими NOT.

Дублирование битов TR0...TR15

В одном и том же блоке программы с несколькими ветвями вывода нельзя использовать биты TR с одинаковыми адресами (см. пример программы ниже). Если биты TR находятся в разных блоках программы, то они могут иметь одинаковый адрес.

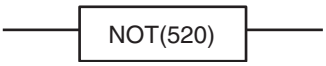


3-2-12 НЕ: NOT(520)

Назначение

Данная команда меняет состояние условия выполнения на противоположное.

Символ РКС



Варианты выполнения

Варианты выполнения	**Инвертирует условие выполнения в каждом цикле.**	NOT(520)
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	**Области пошаговых программ**	**Подпрограммы**	**Задачи обработки прерываний**
OK	OK	OK	OK

Описание

Команду NOT(520) вставляют между условием выполнения и другой командой с целью инверсии условия выполнения.

Флаги

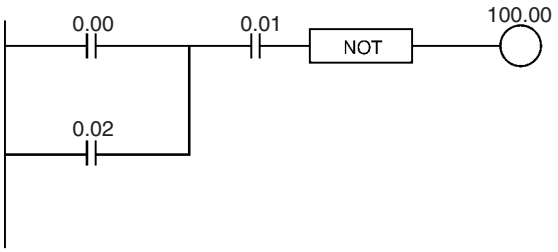
Команда NOT(520) не изменяет состояние какого-либо флага.

Меры предосторожности

NOT(520) — это промежуточная команда, ее нельзя использовать в качестве самой последней команды цепи. Справа от NOT(520) обязательно должна присутствовать какая-нибудь другая команда.

Пример

Ниже показан пример программы, в котором команда NOT(520) меняет состояние условия выполнения на противоположное.



Работу этого сегмента программы иллюстрирует следующая таблица.

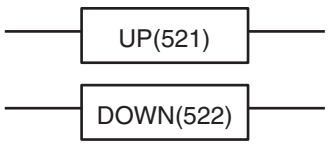
Состояние бита входа			Состояние бита выхода
CIO 0.00	**CIO 0,01**	**CIO 0,02**	**CIO 0,03**
1	1	1	0
1	1	0	0
1	0	1	1
0	1	1	0
1	0	0	1
0	1	0	1
0	0	1	1
0	0	0	1

3-2-13 УСЛОВИЕ ВКЛ/ВЫКЛ: UP(521) и DOWN(522)

Назначение

Команда UP(521) включает условие выполнения для следующей команды на один цикл, когда ее собственное условие выполнения переключается из состояния «0» в «1». Команда DOWN(522) включает условие выполнения для следующей команды на один цикл, когда ее собственное условие выполнения переключается из состояния «1» в «0».

Символы РКС



Варианты выполнения

Варианты выполнения	**Включается однократно по положительному фронту.**	UP(521)
Модификатор мгновенного обновления		Не предусмотрено

Варианты выполнения	**Включается однократно по отрицательному фронту.**	UP(522)
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Описание

Команду UP(521) вставляют между условием выполнения и другой командой с целью кратковременного включения условия выполнения по положительному перепаду. Команда, следующая за командой UP(521), выполняется только один раз, когда условие выполнения переключается из состояния «0» в состояние «1».

Команду DOWN(522) вставляют между условием выполнения и другой командой с целью кратковременного включения условия выполнения по отрицательному перепаду. Команда, следующая за командой DOWN(522), выполняется только один раз, когда условие выполнения переключается из состояния «1» в состояние «0».

Для этой же цели можно воспользоваться командами DIFU(013) и DIFD(014), но для этих команд требуются рабочие биты. Используя UP(521) и DOWN(522), можно упростить программу и сократить количество рабочих битов и адресов программы.

Флаги

Команды UP(521) и DOWN(522) не изменяют состояние какого-либо флага.

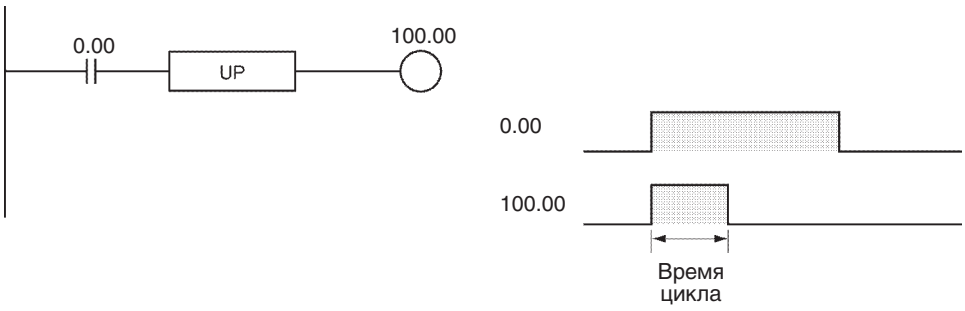
Меры предосторожности

UP(521) и DOWN(522) — это промежуточные команды, их нельзя использовать в качестве самых последних команд цепи. Справа от UP(521) или DOWN(522) обязательно должна присутствовать какая-нибудь другая команда.

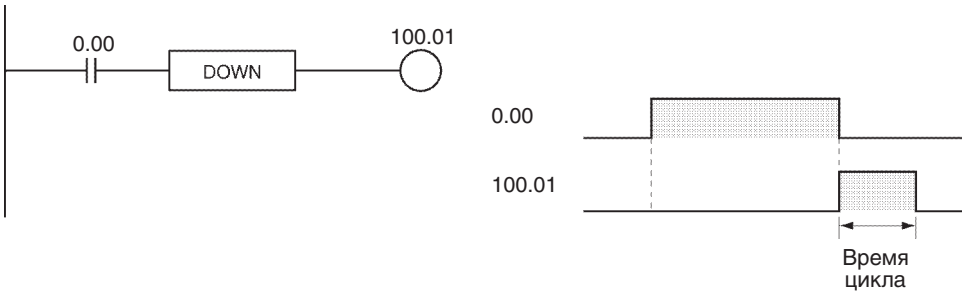
Работа команды UP(521)/DOWN(522) зависит от условия выполнения команды, а также от условия выполнения сегмента программы, если UP(521)/DOWN(522) используется в блокируемом сегменте программы, пропускаемом сегменте программы или в подпрограмме. Подробное описание смотрите в разделах *3-4-4 БЛОКИРОВКА и ОТМЕНА БЛОКИРОВКИ: IL(002) и ILC(003), 3-4-6 ПЕРЕХОД и КОНЕЦ ПЕРЕХОДА: JMP(004) и JME(005)* и *3-19 Команды управления прерываниями*.

Примеры

Если в следующем примере бит CIO 0.00 переключается из «0» в «1», бит CIO 100.00 включается и остается включенным в течение одного цикла.



Если в следующем примере бит CIO 0.00 переключается из «1» в «0», бит CIO 100.01 включается и остается включенным в течение одного цикла.



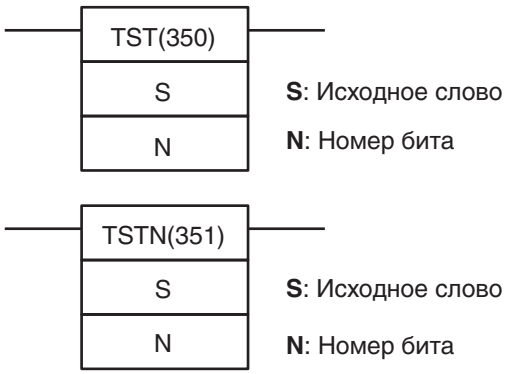
3-2-14 ПРОВЕРКА БИТА: TST(350) и TSTN(351)

Назначение

Команды LD TST(350), AND TST(350) и OR TST(350) используются в программе точно так же, как соответствующие команды LD, AND и OR: условие выполнения включено, когда включен указанный бит указанного слова, и выключено, когда этот бит выключен.

Команды LD TSTN(351), AND TSTN(351) и OR TSTN(351) используются в программе точно так же, как соответствующие команды LD NOT, AND NOT и OR NOT: условие выполнения выключено, когда включен указанный бит указанного слова, и включено, когда этот бит выключен.

Символы РКС



Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле**	TST(350)
Модификатор мгновенного обновления		Не предусмотрено

Варианты выполнения	**Выполнение в каждом цикле**	TSTN(351)
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	**Области пошаговых программ**	**Подпрограммы**	**Задачи обработки прерываний**
OK	OK	OK	OK

Операнды

N: Номер бита

Номер бита должен находиться в диапазоне от 0000 до 000F (hex) или от &0000 до &0015 (десятичн.). Если указывается адрес слова, то используется только самый младший бит (0...F hex) этого слова.

Характеристики операндов

Область	S	N
Область CIO	CIO 0...CIO 6143	
Рабочая область	W0...W511	
Область битов хранения	H0...H511	
Область вспомогательных битов	A0...A959	
Область таймеров	T0000...T4095	
Область счетчиков	C0000...C4095	
Область DM	D0...D32767	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767	
Постоянные	---	#0000...#000F (двоичный) или &0...&15
Регистры данных	DR0...DR15	
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

Описание

Команды LD TST(350), AND TST(350) и OR TST(350) можно использовать в программе таким же образом, как команды LD, AND и OR. Они включают условие выполнения, когда включен указанный бит указанного слова, и выключают его, когда этот бит выключен. В отличие от команд LD, AND и OR, в качестве операндов команды TST(350) могут использоваться биты области DM.

Команды LD TSTN(351), AND TSTN(351) и OR TSTN(351) можно использовать в программе таким же образом, как команды LD NOT, AND NOT и OR NOT. Они выключают условие выполнения, когда включен указанный бит указанного слова, и включают его, когда этот бит выключен. В отличие от команд LD NOT, AND NOT и OR NOT, в качестве операндов команды TSTN(351) могут использоваться биты области DM.

Флаги

Название	Обозначение	Операция
Флаг ошибки	ER	Выключен или не изменяется
Флаг равенства	=	Выключен или не изменяется
Флаг отрицательного значения	N	Выключен или не изменяется

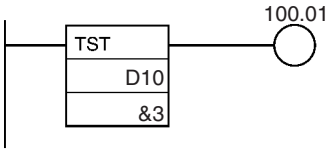
Меры предосторожности

TST(350) и TSTN(351) — это промежуточные команды, их нельзя использовать в качестве самых последних команд цепи. Справа от TST(350) или TSTN(351) обязательно должна присутствовать какая-нибудь другая команда.

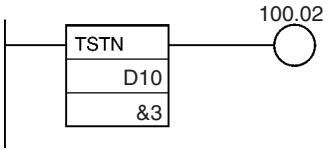
Примеры

LD TST(350) и LD TSTN(351)

В следующем примере бит CIO 100.01 включен, если включен бит 3 слова D10.

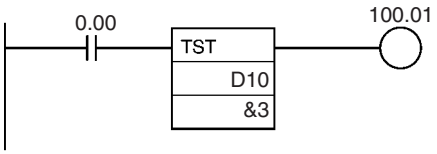


В следующем примере бит CIO 100.02 включен, если выключен бит 3 слова D10.

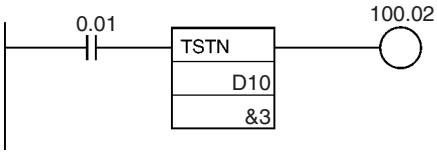


AND TST(350) и AND TSTN(351)

В следующем примере бит CIO 100.01 включен, если одновременно включены бит CIO 0.00 и бит 3 слова D10.

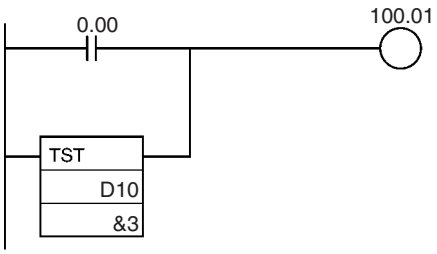


В следующем примере бит CIO 100.02 включен, если бит CIO 0.01 включен, а бит 3 слова D10 выключен.

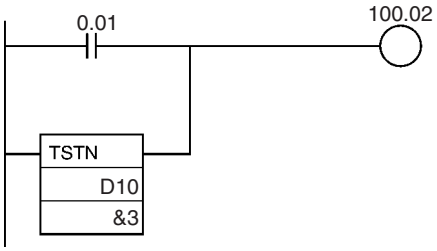


OR TST(350) и OR TSTN(351)

В следующем примере бит CIO 100.01 включен, если включен бит CIO 0.00 или бит 3 слова D10.



В следующем примере бит CIO 100.02 включен, если включен бит CIO 0.01 или выключен бит 3 слова D10.



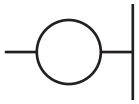
3-3 Выходные битовые команды

3-3-1 ВЫВОД: OUT

Назначение

Команда выдает результат (условие выполнения) логических вычислений в указанный бит.

Символ РКС



Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения.**	OUT
	Однократное выполнение по положительному фронту.	Не предусмотрено
	Однократное выполнение по отрицательному фронту.	Не предусмотрено
Модификатор мгновенного обновления		!OUT

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
Не допускается	OK	OK	OK

Характеристики операндов

Область	Битовый операнд команды OUT
Область CIO	CIO 0.00...CIO 6143.15
Рабочая область	W0.00...W511.15
Область битов хранения	H0.00...H511.15
Область вспомогательных битов	A448.00...A959.15
Область таймеров	---
Область счетчиков	---
Область TR	TR0...TR15
Область DM	---
Косвенные адреса DM в двоичном формате	---
Косвенные адреса DM в двоично-десятичном формате (BCD)	---
Постоянные	---
Регистры данных	---
Регистры указателей	---
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...,IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15

Описание

Если не указано мгновенное обновление, состояние условия выполнения (0 или 1) записывается в указанный бит памяти ввода/вывода. Если выбрано мгновенное обновление, состояние условия выполнения (0 или 1) записывается в бит выхода памяти ввода/вывода и одновременно передается на соответствующий физический выход модуля ЦПУ.

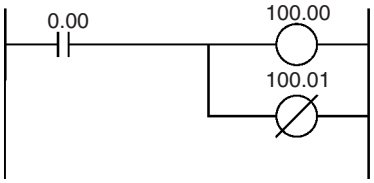
Флаги

Данная команда не изменяет состояние какого-либо флага.

Меры предосторожности

Для команд OUT и OUT NOT может быть указано мгновенное обновление (!). Команда, для которой выбрано мгновенное обновление, сразу после своего выполнения записывает состояние условия выполнения (0 или 1) в указанный бит выхода памяти ввода/вывода и одновременно обновляет состояние соответствующего физического выхода модуля ЦПУ.

Пример



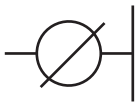
Команда	Операнд
LD	0.00
OUT	100.00
OUT NOT	100.01

3-3-2 ВЫВОД НЕ: OUT NOT

Назначение

Команда инвертирует результат (условие выполнения) логических вычислений и выводит его в указанный бит.

Символ РКС



Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения.**	OUT NOT
	Однократное выполнение по положительному фронту.	Не предусмотрено
	Однократное выполнение по отрицательному фронту.	Не предусмотрено
Модификатор мгновенного обновления		!OUT NOT

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
Не допускается	OK	OK	OK

Характеристики операндов

Область	Битовый операнд команды OUT
Область CIO	CIO 0.00...CIO 6143.15
Рабочая область	W0.00...W511.15
Область битов хранения	H0.00...H511.15

Область	Битовый операнд команды OUT
Область вспомогательных битов	A448.00...A959.15
Область таймеров	---
Область счетчиков	---
Область TR	TR0...TR15
Область DM	---
Косвенные адреса DM в двоичном формате	---
Косвенные адреса DM в двоично-десятичном формате (BCD)	---
Постоянные	---
Регистры данных	---
Регистры указателей	---
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...,IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15

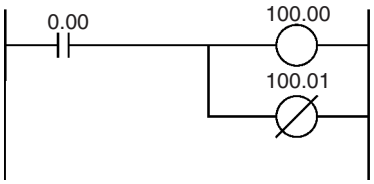
Описание

Если не указано мгновенное обновление, инвертированное состояние условия выполнения (0 или 1) записывается в указанный бит памяти ввода/вывода. Если выбрано мгновенное обновление, инвертированное состояние условия выполнения (0 или 1) записывается в бит выхода памяти ввода/вывода и одновременно передается на соответствующий физический выход модуля ЦПУ.

Флаги

Данная команда не изменяет состояние какого-либо флага.

Пример



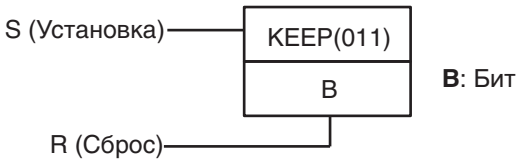
Команда	Операнд
LD	0.00
OUT	0.01
OUT NOT	0.02

3-3-3 ЗАЩЕЛКА: KEEP(011)

Назначение

Команда реализует реле с самоблокировкой.

Символ РКС



Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения.**	KEEP(011)
	Однократное выполнение по положительному фронту.	Не предусмотрено
	Однократное выполнение по отрицательному фронту.	Не предусмотрено
Модификатор мгновенного обновления		!KEEP(011)

Доступные области программы

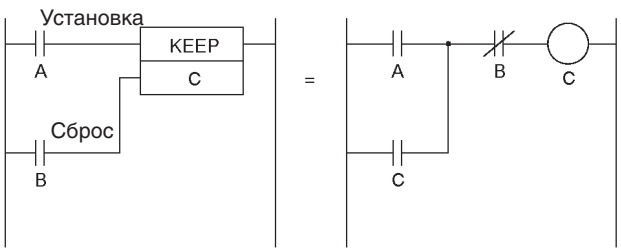
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
Не допускается	OK	OK	OK

Характеристики операндов

Область	B
Область CIO	CIO 0.00...CIO 6143.15
Рабочая область	W0.00...W511.15
Область битов хранения	H0.00...H511.15
Область вспомогательных битов	A448.00...A959.15
Область таймеров	---
Область счетчиков	---
Область DM	---
Косвенные адреса DM в двоичном формате	---
Косвенные адреса DM в двоично-десятичном формате (BCD)	---
Постоянные	---
Регистры данных	---
Регистры указателей	---
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15

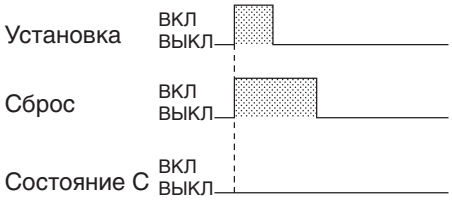
Описание

Включение входа S приводит к включению назначенного бита B. После этого бит B, независимо от состояния на входе S, остается включенным до тех пор, пока не производится сброс. Включение входа R приводит к выключению назначенного бита B. Взаимосвязь между состояниями условий выполнения и состоянием бита KEEP(011) показана ниже.

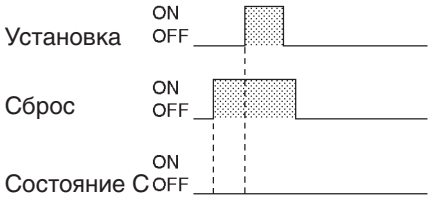




При одновременном включении входов S и R приоритетом обладает вход сброса.

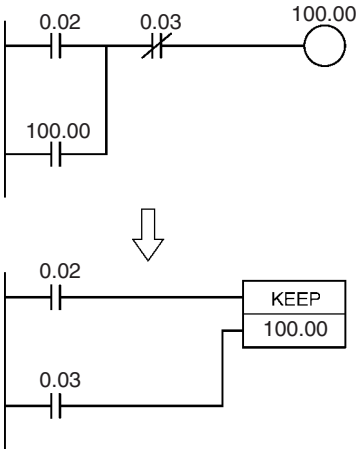


Пока включен вход сброса (R), вход установки (S) не действует (его включение не приводит к установке бита).

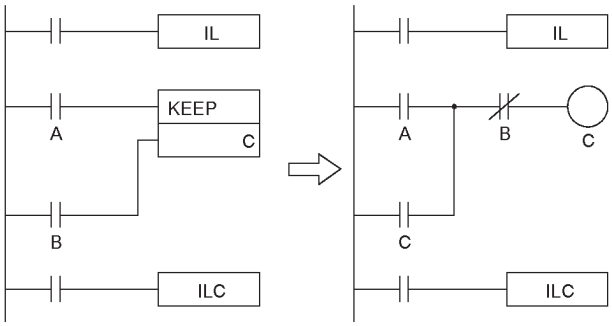


Команду KEEP(011) можно использовать с префиксом мгновенного обновления (!KEEP(011)). Если в качестве операнда B команды !KEEP(011) указан бит внешнего выхода, изменение состояния B сразу после выполнения команды !KEEP(011) немедленно передается в бит встроенного выхода модуля ЦПУ.

Команда KEEP(011) фактически работает как самоблокирующийся бит, но требует на одну команду меньше, чем самоблокирующийся бит, реализуемый программно (см. пример ниже).



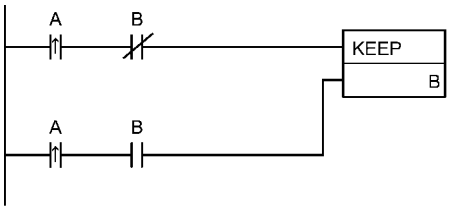
В отличие от самоблокирующегося бита, созданного без команды KEEP(011), самоблокирующийся бит, реализованный командой KEEP(011), сохраняет свое состояние, даже если он находится в блокированном сегменте программы.

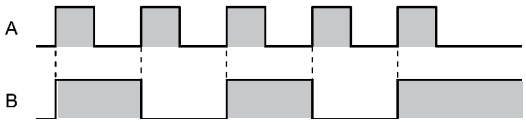


Выходной бит C сохраняет свое прежнее состояние в блокированном сегменте.

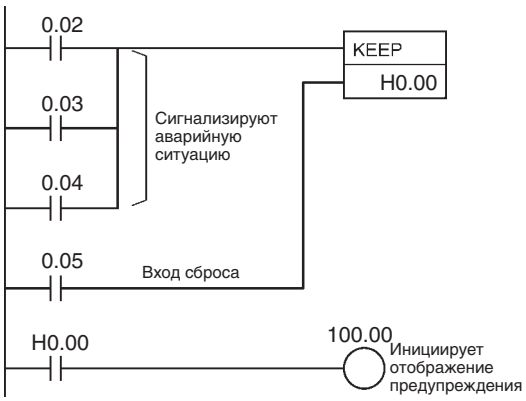
Выходной бит C сбрасывается в блокированном сегменте.

Ниже показан пример использования KEEP(011) для создания триггера.





В качестве операнда B может быть использован бит области хранения, состояние которого не утрачивается даже при прерывании питания. Таким образом, с помощью команды KEEP(011) можно создавать биты, сохраняющие свое прежнее состояние при перезапуске ПЛК после прерывания питания. В качестве примера ниже приведен фрагмент программы для выдачи сигнала предупреждения об аварийной ситуации после аварийного завершения работы системы.



Для того чтобы биты области ввода/вывода сохраняли свои прежние состояния при отключении питания, следует установить бит памяти ввода/вывода (IOM) и выбрать в настройках ПЛК сохранение состояния самого этого бита. В этом случае биты области ввода/вывода, используемые в команде KEEP(011), подобно битам хранения будут сохранять свои состояния при перезапуске ПЛК, происходящем после прерывания (отключения и повторного включения) питания ПЛК. Обязательно перезапустите ПЛК после внесения изменений в настройки ПЛК, иначе новые значения параметров не вступят в силу.

Флаги

Команда KEEP(011) не изменяет состояния флагов.

Меры предосторожности

Для входа сброса (R) команды KEEP(011) не следует использовать бит входа нормально замкнутого типа, если используется входное устройство с напряжением питания переменного тока. Из-за того что источник питания постоянного тока, используемый для ПЛК, может выключиться несколько позже источника питания переменного тока, используемого для входного устройства, битовый операнд команды KEEP(011) может быть сброшен. Описанную ситуацию иллюстрирует следующий рисунок.



Последовательность ввода операндов команды KEEP(011) зависит от того, какие символы используются для набора программы

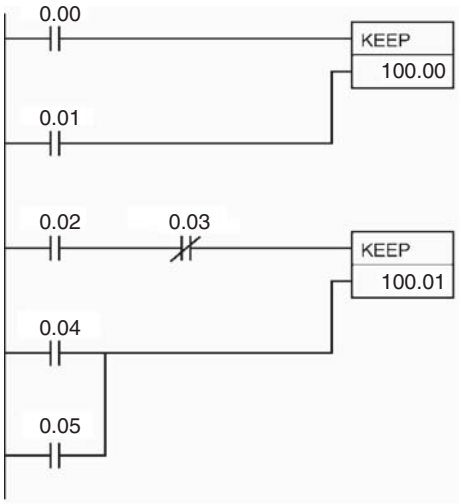
РКС: Вход установки → KEEP(011) → Вход сброса

Мнемонические коды: Вход установки→ Вход сброса → KEEP(011)

Пример

Включение бита CIO 0.00 приводит к включению бита CIO 100.00. Бит CIO 100.00 остается включенным, пока не включается бит CIO 0.01.

Включение бита CIO 0.02 и выключение бита CIO 0.03 приводит к включению бита CIO 100.01. После этого бит CIO 100.01 остается включенным, пока не включается бит CIO 0.04 или бит CIO 0.05.



Код программы

Адрес	Команда	Операнд
000100	LD	0.00
000101	LD	0.01
000102	KEEP(011)	100.00
000103	LD	0.02
000104	AND NOT	0.03
000105	LD	0.04
000106	OR	0.05
000107	KEEP(011)	100.01

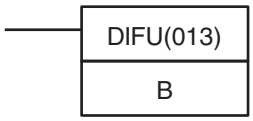
Примечание. Операнды KEEP(011) вводятся в разной последовательности при создании программы в виде лестничной диаграммы (РКС) и в мнемонической форме. При создании программы в виде лестничной диаграммы сначала вводится вход установки, затем символ KEEP(011), затем вход сброса. При создании программы в мнемонической форме сначала вводится вход установки, затем вход сброса, затем символ KEEP(011).

3-3-4 ПОЛОЖИТЕЛЬНЫЙ/ОТРИЦАТЕЛЬНЫЙ ФРОНТ: DIFU(013) и DIFD(014)

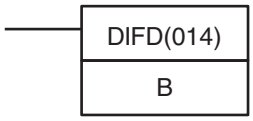
Назначение

Команда DIFU(013) устанавливает указанный бит на один цикл, когда условие выполнения переключается из состояния «0» в состояние «1» (положительный фронт).
Команда DIFD(014) устанавливает указанный бит на один цикл, когда условие выполнения переключается из состояния «1» в состояние «0» (отрицательный фронт).

Символы РКС



B: Бит



B: Бит

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения.**	Не предусмотрено
	Однократное выполнение по положительному фронту.	DIFU(013)
	Однократное выполнение по отрицательному фронту.	Не предусмотрено
Модификатор мгновенного обновления		!DIFU(013)

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения.**	Не предусмотрено
	Однократное выполнение по положительному фронту.	DIFD(014)
	Однократное выполнение по отрицательному фронту.	Не предусмотрено
Модификатор мгновенного обновления		!DIFD(014)

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
Не допускается	OK	OK	OK

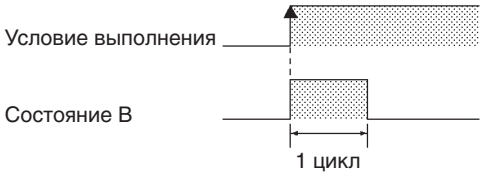
Характеристики операндов

Область	B
Область CIO	CIO 0.00...CIO 6143.15
Рабочая область	W0.00...W511.15

Область	B
Область битов хранения	H0.00...H511.15
Область вспомогательных битов	A448.00...A959.15
Область таймеров	---
Область счетчиков	---
Область DM	---
Косвенные адреса DM в двоичном формате	---
Косвенные адреса DM в двоично-десятичном формате (BCD)	---
Постоянные	---
Регистры данных	---
Регистры указателей	---
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –) IR0...,15–(– –) IR

Описание

По положительному перепаду условия выполнения (0 -> 1) команда DIFU(013) включает бит B. Когда выполнение программы доходит до команды DIFU(013) в следующем цикле, бит B выключается.



По отрицательному перепаду условия выполнения (1 -> 0) команда DIFD(014) включает бит B. Когда выполнение программы доходит до команды DIFD(014) в следующем цикле, бит B выключается.



Для команд DIFU(013) и DIFD(014) доступны формы записи с префиксом мгновенного обновления (!DIFU(013) и !DIFD(014)). Если в одной из этих команд в качестве операнда B указан бит внешнего выхода, изменение состояния B сразу после выполнения команды немедленно передается в выходной бит, соответствующий встроенному выходу модуля ЦПУ.

Для того чтобы некоторая команда выполнялась только в одном цикле после положительного ($0 \rightarrow 1$) или отрицательного ($1 \rightarrow 0$) перепада условия выполнения, можно использовать команды UP(521) или DOWN(522). Подробное описание смотрите в разделе *3-2-13 УСЛОВИЕ ВКЛ/ВЫКЛ: UP(521) и DOWN(522)* .

Флаги

Команды DIFU(013) и DIFD(014) не изменяют состояние какого-либо флага.

Меры предосторожности

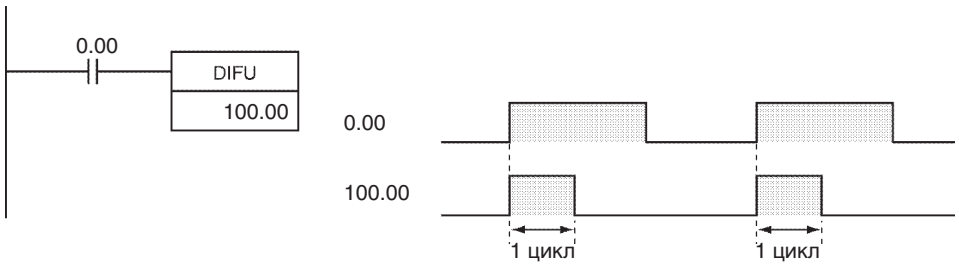
Работа команды DIFU(013)/DIFD(014) зависит от условия выполнения самой этой команды, а также от условия выполнения сегмента программы, если команда DIFU(013)/DIFD(014) используется в блокируемом или пропускаемом сегменте программы или в подпрограмме. Подробное описание смотрите в разделах *3-4-4 БЛОКИРОВКА и ОТМЕНА БЛОКИРОВКИ: IL(002) и ILC(003),* *3-4-6 ПЕРЕХОД и КОНЕЦ ПЕРЕХОДА: JMP(004) и JME(005)* и *3-19 Команды управления прерываниями* .

В том случае, если DIFU(013) применяется в цикле FOR-NEXT и цикл FOR-NEXT повторяется в пределах цикла программы, битовый операнд команды DIFU(013) будет все время включен или выключен внутри цикла FOR-NEXT.

Примеры

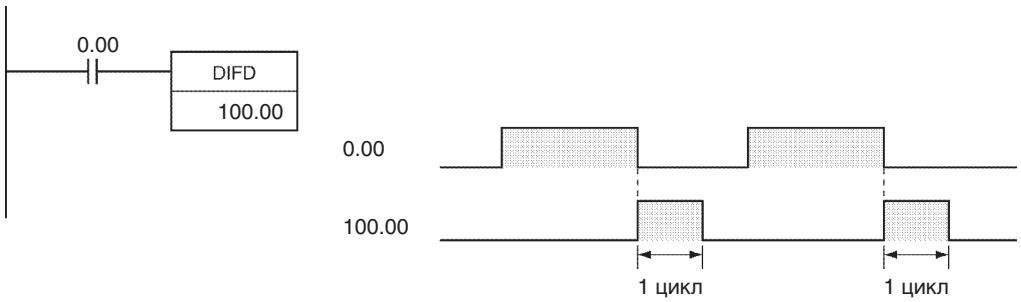
Работа команды DIFU(013)

Если в следующем примере бит CIO 0.00 переключается из «0» в «1», бит CIO 100.00 включается на один цикл.



Работа команды DIFD(014)

Если в следующем примере бит CIO 0.00 переключается из «1» в «0», бит CIO 100.00 включается на один цикл.



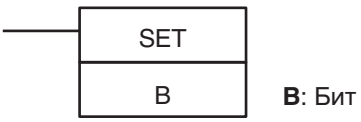
3-3-5 УСТАНОВКА и СБРОС: SET и RSET

Назначение

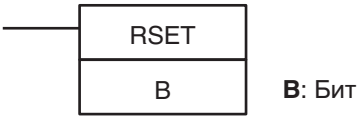
Команда SET включает битовый операнд, если включено ее условие выполнения.

Команда RSET выключает битовый операнд, если включено ее условие выполнения.

Символы РКС



B: Бит



B: Бит

Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения.**	SET
	Однократное выполнение по положительному фронту.	@SET
	Однократное выполнение по отрицательному фронту.	%SET
Модификатор мгновенного обновления		!SET
Комбинированные модификации	**Однократное выполнение по положительному фронту с мгновенным обновлением бита.**	!@SET
	Однократное выполнение по отрицательному фронту с мгновенным обновлением бита.	!%SET

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения.**	RSET
	Однократное выполнение по положительному фронту.	@RSET
	Однократное выполнение по отрицательному фронту.	%RSET
Модификатор мгновенного обновления		!RSET
Комбинированная модификация	**Однократное выполнение по положительному фронту и мгновенное обновление.**	!@RSET
	Однократное выполнение по отрицательному фронту и мгновенное обновление.	!%RSET

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

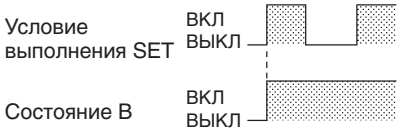
Характеристики операндов

Область	B
Область CIO	CIO 0.00...CIO 6143.15
Рабочая область	W0.00...W511.15
Область битов хранения	H0.00...H511.15
Область вспомогательных битов	A448.00...A959.15
Область таймеров	---
Область счетчиков	---
Область DM	---
Косвенные адреса DM в двоичном формате	---
Косвенные адреса DM в двоично-десятичном формате (BCD)	---
Постоянные	---
Регистры данных	---

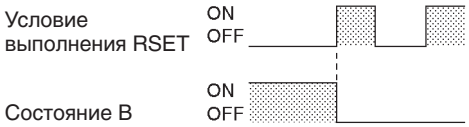
Область	B
Регистры указателей	---
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15

Описание

Если условие выполнения команды SET включено, она включает битовый операнд и не выключает его даже после выключения условия выполнения. Для сброса бита, установленного командой SET, следует использовать команду RSET.



Если условие выполнения команды RSET включено, она выключает битовый операнд и не включает его даже после выключения условия выполнения. Для установки бита, сброшенного командой RSET, следует использовать команду SET.



Для команд SET и RSET доступны варианты выполнения с мгновенным обновлением (!SET и !RSET). Если в одной из этих команд в качестве операнда B указан бит внешнего выхода, изменение состояния B сразу после выполнения команды немедленно передается в выходной бит, соответствующий встроенному выходу модуля ЦПУ.

В отличие от входов установки и сброса команды KEEP(011), команды SET и RSET могут использоваться в программе отдельно и независимо друг от друга. Кроме того, один и тот же бит может использоваться в качестве операнда для неограниченного числа команд SET или RSET.

Флаги

Команды SET и RSET не изменяют состояние какого-либо флага.

Меры предосторожности

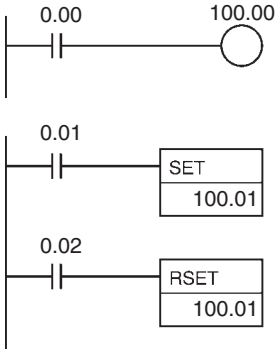
Команды SET и RSET невозможно использовать для установки и сброса таймеров и счетчиков.

Если команда SET или RSET размещается между командами IL(002) и ILC(003) или между командами JMP(004) и JME(005), состояние указанного бита не будет меняться, если этот сегмент программы будет заблокирован или будет пропускаться.

Пример

Различия между командами OUT/OUT NOT и SET/RSET

Команда SET отличается от команды OUT тем, что команда OUT сбрасывает битовый операнд, когда выключается ее условие выполнения. Аналогично, команда RSET отличается от команды OUT NOT тем, что команда OUT NOT устанавливает битовый операнд, когда выключается ее условие выполнения.



CCIO 100.00 включается/ выключается, когда включается/выключается CIO 0.00.

CIO 100.01 включается, когда включается CIO 0.01; он остается включенным, пока не включается CIO 0.02.

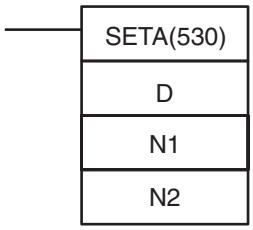
3-3-6 УСТАНОВИТЬ/СБРОСИТЬ НЕСКОЛЬКО БИТОВ: SETA(530)/RSTA(531)

Назначение

Команда SETA(530) включает указанное количество расположенных последовательно битов.

Команда RSTA(531) выключает указанное количество расположенных последовательно битов.

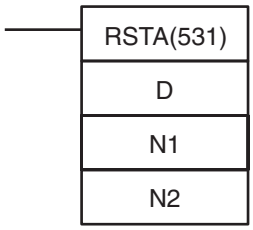
Символы РКС



D: Первое слово

N1: Первый бит

N2: Количество битов



D: Первое слово

N1: Первый бит

N2: Количество битов

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения.**	SETA(530)
	Однократное выполнение по положительному фронту.	@SETA(530)
	Однократное выполнение по отрицательному фронту.	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения.**	RSTA(531)
	Однократное выполнение по положительному фронту.	@RSTA(531)
	Однократное выполнение по отрицательному фронту.	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограм-мы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

D: Первое слово

Указывает первое слово, биты которого должны быть установлены или сброшены.

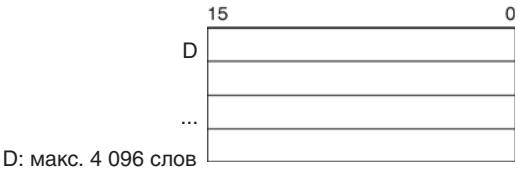
N1: Первый бит

Указывает первый бит, который будет установлен или сброшен. N1 должно находиться в диапазоне от #0000 до #000F (от &0 до &15).

N2: Количество битов

Указывает количество битов, которое будет установлено или сброшено. N2 должно находиться в диапазоне от #0000 до #FFFF (от &0 до &65535).

Примечание. Устанавливаемые или сбрасываемые биты должны находиться в одной области данных. (Приблизительное определение диапазона слов: от D до D+N2÷16.)



Характеристики операндов

Область	D	N1	N2
Область CIO	CIO 0...CIO 6143		
Рабочая область	W0...W511		
Область битов хранения	H0...H511		
Область вспомогательных битов	A448...A959	A0...A959	
Область таймеров	T0000...T4095		
Область счетчиков	C0000...C4095		
Область DM	D0...D32767		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767		
Постоянные	---	#0000...#000F (двоичный) или &0...&15	#0000...#FFFF (двоичный) или &0...&65535
Регистры данных	---	DR0...DR15	
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

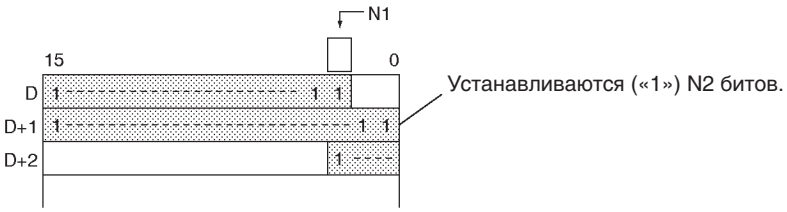
Описание

Функционирование команд SETA(530) и RSTA(531) описано ниже отдельно для каждой из этих команд.

Работа команды SETA(530)

Команда SETA(530) устанавливает бит N1 слова D и следующие за ним более старшие биты (всего N2 битов). Состояния всех остальных битов не меняются (если N2 задано равным «0», ни один бит не изменяется).

Биты, установленные командой SETA(530), могут быть сброшены с помощью команды RSTA(531) или любой другой команды.

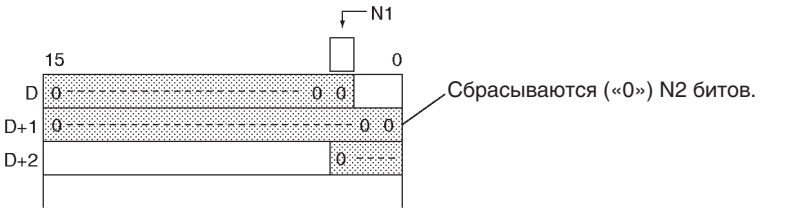


Командой SETA(530) можно воспользоваться для установки требуемых битов в областях данных, обычно адресуемых пословно (например, в области DM).

Работа команды RSTA(531)

Команда RSTA(531) сбрасывает бит N1 слова D и следующие за ним более старшие биты (всего N2 битов). Состояния всех остальных битов не меняются (если N2 задан равным «0», ни один бит не изменяется).

Биты, сброшенные командой RSTA(531), можно установить с помощью команды SETA(530) или любой другой команды.



Командой RSTA(531) можно воспользоваться для сброса требуемых битов в областях данных, обычно адресуемых пословно (например, в области DM).

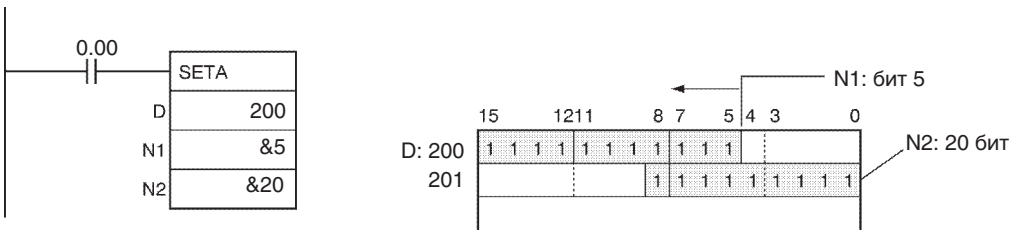
Флаги

Название	Обозначение	Операция
Флаг ошибки	ER	Включен, если N1 выходит за допустимый диапазон 0000...000F. Выключен во всех остальных случаях.

Примеры

Пример для команды SETA(530)

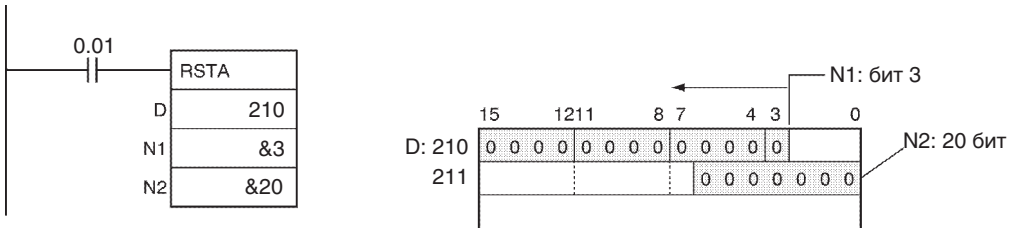
В следующем примере включение бита CIO 0.00 приводит к включению 20 битов (0014 hex) слова CIO 200, начиная с бита 5.



Пример для RSTA(531)

В следующем примере включение бита CIO 0.01 приводит к выключению 20 битов (0014 hex) слова CIO 210, начиная с бита 3.

Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

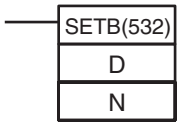


3-3-7 УСТАНОВИТЬ/СБРОСИТЬ ОДИН БИТ: SETB(532)/RSTB(533)

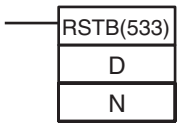
Назначение

Команда SETB(532) устанавливает указанный бит.
Команда RSTB(533) сбрасывает указанный бит.

Символы РКС



D: Адрес слова
N: Номер бита



D: Адрес слова
N: Номер бита

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения.**	SETB(532)
	Однократное выполнение по положительному фронту.	@SETB(532)
	Однократное выполнение по отрицательному фронту.	Не предусмотрено
Модификатор мгновенного обновления		!SETB(532)
Комбинированная модификация	**Однократное выполнение по положительному фронту с мгновенным обновлением бита.**	!@SETB(532)
	Однократное выполнение по отрицательному фронту с мгновенным обновлением бита.	Не предусмотрено

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения.**	RSTB(533)
	Однократное выполнение по положительному фронту.	@RSTB(533)
	Однократное выполнение по отрицательному фронту.	Не предусмотрено
Модификатор мгновенного обновления		!RSTB(533)
Комбинированная модификация	**Однократное выполнение по положительному фронту с мгновенным обновлением бита.**	!@RSTB(533)
	Однократное выполнение по отрицательному фронту с мгновенным обновлением бита.	Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

D: Адрес слова

Указывает слово, бит которого будет установлен или сброшен.

N: Первый бит

Указывает бит, который будет установлен или сброшен. Значение N должно быть в диапазоне от #0000 до #000F (&0...&15).

Характеристики операндов

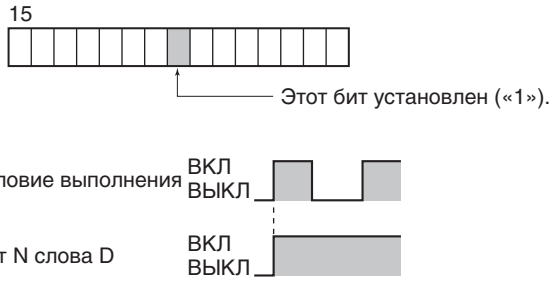
Область	D	N
Область CIO	CIO 0...CIO 6143	
Рабочая область	W0...W511	
Область битов хранения	H0...H511	
Область вспомогательных битов	A448...A959	A0...A959
Область таймеров	T0000...T4095	
Область счетчиков	C0000...C4095	
Область DM	D0...D32767	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767	
Постоянные	---	#0000...#000F (двоичный) или &0...&15
Регистры данных	DR0...DR15	
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

Описание

Функционирование команд SETB(532) и RSTB(533) описано ниже отдельно для каждой команды.

Работа команды SETB(532)

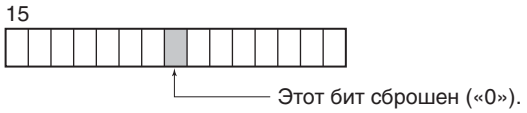
Команда SETB(532) устанавливает бит N слова D, когда включено условие выполнения. Если условие выполнения выключено, состояние бита не изменяется. В отличие от команды SET, команда SETB(532) может установить бит в области DM.

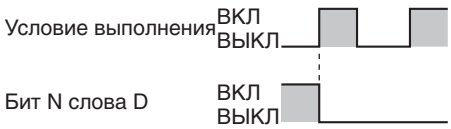


Бит, установленный командой SETB(532), можно сбросить с помощью команды RSTB(533) или любой другой команды.

Работа команды RSTB(533)

Команда RSTB(533) сбрасывает бит N слова D, когда включено условие выполнения. Если условие выполнения выключено, состояние бита не изменяется. (Для установки этого бита следует использовать команду SETB(532).) В отличие от команды RST, команда RSTB(533) может сбросить бит в области DM.





Бит, сброшенный командой RSTB(533), можно установить с помощью команды SETB(532) или любой другой команды.

Флаги

Название	Обозначение	Операция
Флаг ошибки	ER	Включен, если N выходит за допустимый диапазон 0000...000F (&0...&15). Выключен во всех остальных случаях.

Меры предосторожности

Команды SETB(532) и RSTB(533) невозможно использовать для установки и сброса таймеров и счетчиков.

Если команда SETB(532) или RSTB(533) располагается между командами IL(002) и ILC(003) или JMP(004) и JME(005), состояние указанного бита не будет меняться, если данный сегмент программы заблокирован или пропускается (т. е. условие блокировки или перехода выключено).

Команды SETB(532) и RSTB(533) могут работать с мгновенным обновлением (!SETB(532) и !RSTB(533)). Если в одной из этих команд в качестве операнда указан бит внешнего выхода, изменение состояния операнда сразу после выполнения команды немедленно передается в выходной бит, соответствующий встроенному выходу модуля ЦПУ.

Различия между командами SET/RSET и SETB(532)/RSTB(533)

Действие команд SET и RSET немного отличается от действия команд SETB(532) и RSTB(533).

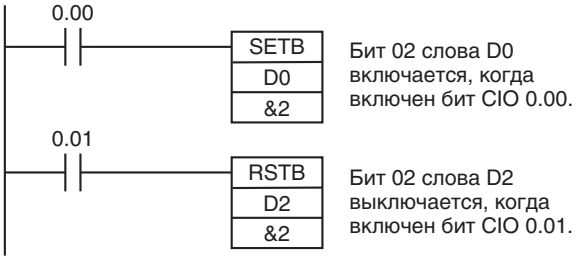
1. Команды работают одинаковым образом, если в качестве операнда указан бит области CIO, W, H или A.
2. В отличие от команд SET и RSET, команды SETB(532) и RSTB(533) также могут управлять состоянием битов области DM.

Различия между командами OUTB(534) и SETB(532)/RSTB(533)

Действие команды OUTB(534) немного отличается от действия команд SETB(532) и RSTB(533).

1. Команды SETB(532) и RSTB(533) изменяют состояние указанного бита, только если их условие выполнения включено. Когда условие выполнения этих команд выключено, они не влияют на состояние указанного бита.
2. Команда OUTB(534) включает указанный бит, когда ее условие выполнения включено, и выключает указанный бит, когда ее условие выполнения выключено.

3. В отличие от входов установки и сброса команды KEEP(011), команды SETB(532) и RSTB(533) могут использоваться в программе отдельно и независимо друг от друга. Кроме того, один и тот же бит может использоваться в качестве операнда для неограниченного числа команд SETB(532) и RSTB(533).

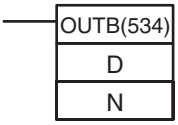


3-3-8 ВЫВОД В ОДИН БИТ: OUTB(534)

Назначение

Команда OUTB(534) записывает состояние своего условия выполнения в указанный бит. В отличие от команды OUT, команда OUTB(534) может управлять состоянием бита области DM.

Символы РКС



D: Адрес слова
N: Номер бита

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения.**	OUTB(534)
	Однократное выполнение по положительному фронту.	@OUTB(534)
	Однократное выполнение по отрицательному фронту.	Не предусмотрено
Модификатор мгновенного обновления		!OUTB(534)

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
Не допускается	OK	OK	OK

Операнды

D: Адрес слова

Указывает адрес слова, содержащего изменяемый бит.

N: Первый бит

Указывает изменяемый бит. Значение N должно быть в диапазоне от #0000 до #000F (&0...&15).

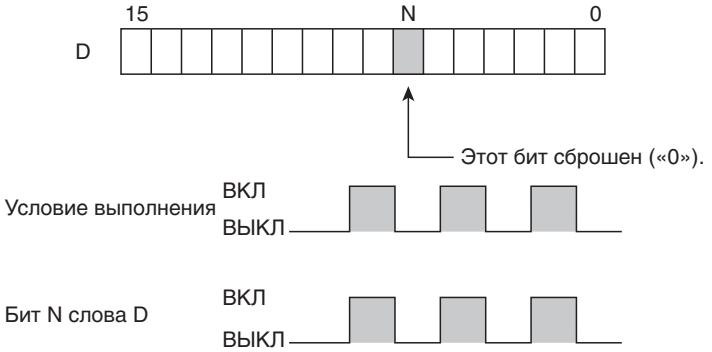
Характеристики операндов

Область	D	N
Область CIO	CIO 0...CIO 6143	
Рабочая область	W0...W511	
Область битов хранения	H0...H511	
Область вспомогательных битов	A448...A959	A0...A959
Область таймеров	T0000...T4095	
Область счетчиков	C0000...C4095	
Область DM	D0...D32767	

Область	D	N
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767	
Постоянные	---	#0000...#000F (двоичный) или &0...&15
Регистры данных	DR0...DR15	
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

Описание

Команда OUTB(534) устанавливает бит N слова D, если ее условие выполнения включено. Команда OUTB(534) сбрасывает бит N слова D, если ее условие выполнения выключено.



Если мгновенное обновление не указано, состояние условия выполнения (0 или 1) записывается в указанный бит памяти ввода/вывода. Если указано мгновенное обновление, состояние условия выполнения (0 или 1) записывается в бит выхода памяти ввода/вывода и одновременно передается на физический выход модуля ЦПУ.

Флаги

Данная команда не изменяет состояние какого-либо флага.

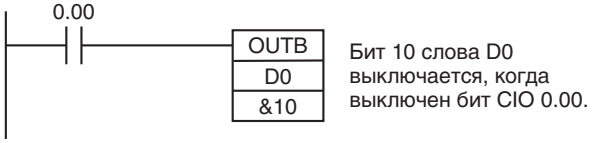
Меры предосторожности

Для данной команды может быть указано мгновенное обновление (!OUTB(534)). Команда, для которой выбрано мгновенное обновление, сразу после своего выполнения записывает состояние условия выполнения (0 или 1) в указанный бит выхода памяти ввода/вывода и одновременно обновляет состояние соответствующего встроенного выхода модуля ЦПУ.

Если команда OUTB(534) располагается между командами IL(002) и ILC(003), указанный бит будет выключен, если данный сегмент программы будет блокирован. (Точно так же, как в случае, когда в заблокированном сегменте программы находится команда OUT.)

Если в качестве номера бита (N) указано слово, используются только биты 00...03 слова N. Например, если N содержит значение FFFA hex, команда OUTB(534) управляет состоянием бита 10 слова D.

Пример



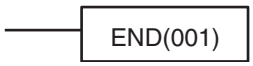
3-4 Команды управления последовательностью выполнения

3-4-1 КОНЕЦ: END(001)

Назначение

Указывает на завершение программы.

Символ РКС



Варианты выполнения

Варианты выполнения	Выполнение в каждом цикле при включенном условии выполнения	END(001)
Модификатор мгновенного обновления		Не предусмотрено

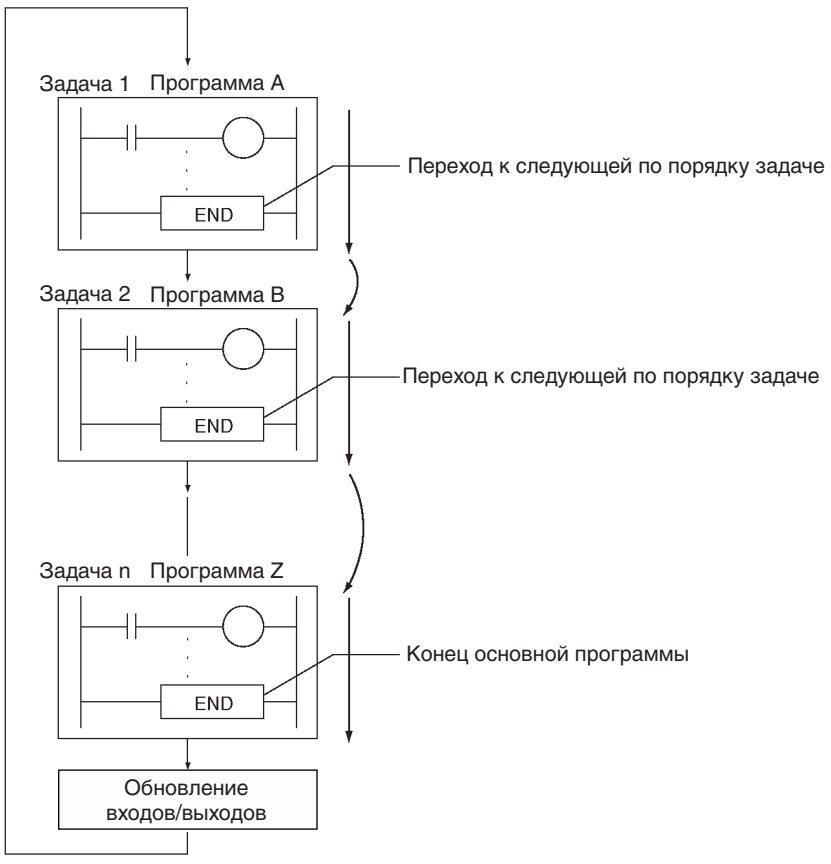
Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
Не допускается	Не допускается	Не допускается	OK

Описание

Команда END(001) завершает выполнение программы в данном цикле. Команды, размещенные после END(001), выполняться не будут.

Далее начинает выполняться программа задачи со следующим номером. Если выполняется программа задачи с наивысшим номером, команда END(001) означает завершение выполнения всей основной программы.



Предварительная версия
Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

Меры предосторожности Команду END(001) следует предусматривать в конце каждой программы. Если в программе отсутствует команда END(001), возникает ошибка программирования.

3-4-2 ПУСТАЯ ОПЕРАЦИЯ: NOP(000)

Назначение Эта команда не выполняет никаких функций (команда NOP(000) не предполагает выполнение каких-либо операций).

Символ РКС Для команды NOP(000) символов РКС не предусмотрено .

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	NOP(000)
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	**Области пошаговых программ**	**Подпрограммы**	**Задачи обработки прерываний**
OK	OK	OK	OK

Описание Команда NOP(000) не предполагает выполнения каких-либо операций, но ее можно использовать в программе, чтобы зарезервировать строки для последующей вставки команд. В таком случае последующая вставка команд не приводит к изменению адресов программы.

Флаги Команда NOP(000) не изменяет состояния флагов.

Меры предосторожности Команду NOP(000) можно использовать только в мнемоническом представлении, но не в лестничных диаграммах.

3-4-3 Обзор команд блокировки

Команды блокировки Следующие комбинации команд могут использоваться для блокировки выходов в сегменте программы.

- БЛОКИРОВКА и ОТМЕНА БЛОКИРОВКИ (IL(002) и IL(003))
- ВЛОЖЕННАЯ БЛОКИРОВКА С ЗАПОМИНАНИЕМ ФРОНТА и ОТМЕНА ВЛОЖЕННОЙ БЛОКИРОВКИ (MILH(517) и MILC(519))*

Примечание. Команда MILH(517) сохраняет состояние флага различения фронтов, так что заблокированные команды с различением фронтов выполняются после отмены блокировки.

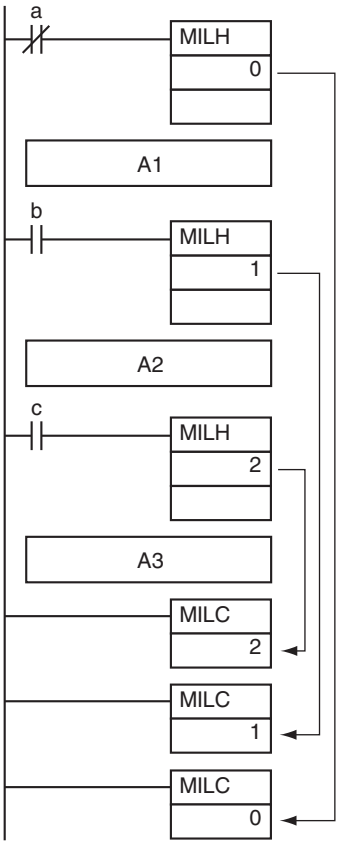
- ВЛОЖЕННАЯ БЛОКИРОВКА БЕЗ ЗАПОМИНАНИЯ ФРОНТА и ОТМЕНА ВЛОЖЕННОЙ БЛОКИРОВКИ (MILH(518) и MILC(519))*

Примечание. Команда MILR(518) не сохраняет состояние флага различения фронтов, так что заблокированные команды с различением фронтов не выполняются после отмены блокировки.

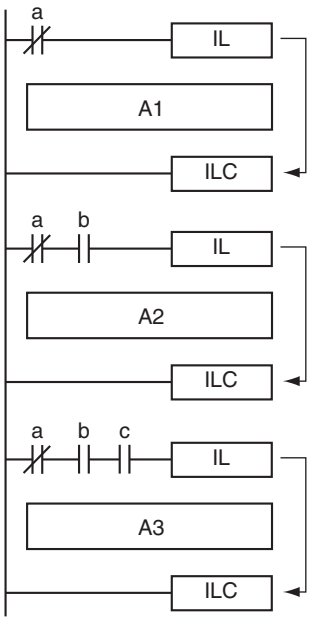
Различия между обычной и вложенной блокировкой.

Обычные команды блокировки (IL(002) и IL(003)) не допускают вложения, а команды вложенной блокировки (MILH(517), MILR(518) и MILC(519)) допускают вложение. Вложение команд блокировки, как показано на следующей схеме, может упростить разработку лестничной диаграммы.

Блокировка с помощью MILH и MILC



Блокировка с помощью IL и ILC



Различия между MILH(517) и MILR(518)

Команды MILH(517) и MILR(518) по-разному воздействуют на работу команд с различением фронтов (DIFU, DIFD или команд с префиксом @ или %).

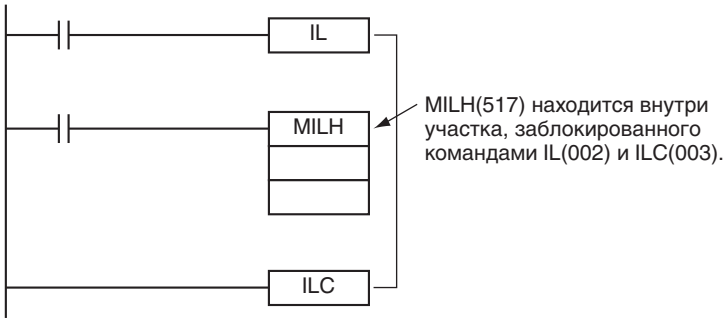
При блокировке командой MILH(517) команды с различением фронтов работают так же, как при использовании команды IL(002).

Подробные сведения смотрите в *3-4-5 ВЛОЖЕННАЯ БЛОКИРОВКА С ЗАПОМИНАНИЕМ ФРОНТА, ВЛОЖЕННАЯ БЛОКИРОВКА БЕЗ ЗАПОМИНАНИЯ ФРОНТА и ОТМЕНА ВЛОЖЕННОЙ БЛОКИРОВКИ: MILH(517), MILR(518) и MILC(519)*.

Меры предосторожности

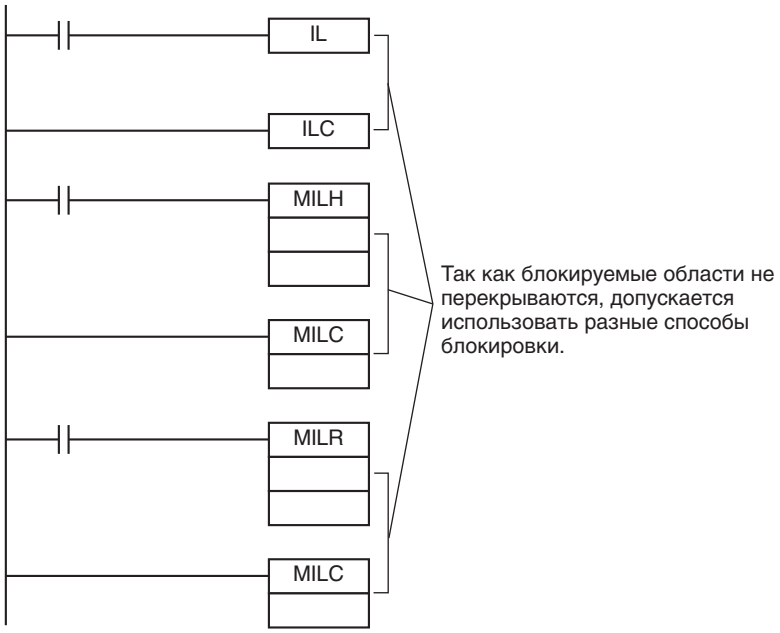
Не следует комбинировать разные команды блокировки (IL-ILC, MILH-MILC и MILR-MILC). При одновременном использовании разных методов блокировки программа может работать неверно. Подробные сведения о совместном использовании команд см. в разделе *3-4-5 ВЛОЖЕННАЯ БЛОКИРОВКА С ЗАПОМИНАНИЕМ ФРОНТА, ВЛОЖЕННАЯ БЛОКИРОВКА БЕЗ ЗАПОМИНАНИЯ ФРОНТА и ОТМЕНА ВЛОЖЕННОЙ БЛОКИРОВКИ: MILH(517), MILR(518) и MILC(519)*.

Например, не допускается располагать команду MILH(517) между командами IL(002) и IL(003).



Примечание. Совместное использование разных методов блокировки (IL-ILC, MILH-MILC и MILR-MILC) допускается только в том случае, если блокируемые сегменты программы не перекрываются.

На следующей схеме показан пример использования сразу трех методов блокировки для неперекрывающихся сегментов программы.



Различия между блокировкой и переходом

В следующей таблице показаны различия между командами блокировки (IL(002)/ILC(003), MILH(517)/MILC(519) или MILR(518)/MILC(519)) и командами перехода (JMP(004)/JME(005)).

Параметр	Действие в IL(002)/ILC(003), MILH(517)/ MILC(519) или MILR(518)/MILC(519))	Действие в JMP(004)/JME(005)
Выполнение команд	Не выполняются никакие команды, кроме OUT, OUT NOT, OUTB(534) и команд таймеров.	Не выполняются никакие команды.
Состояния выходов в командах	Все выходы сохраняют свои прежние состояния, за исключением выходов команд OUT, OUT NOT, OUTB(534) и команд таймеров.	Все выходы сохраняют свои прежние состояния.

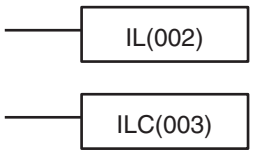
Параметр	Действие в IL(002)/ILC(003), MILH(517)/ MILC(519) или MILR(518)/MILC(519))	Действие в JMP(004)/JME(005)
Биты в OUT, OUT NOT, OUTB(534)	ВЫКЛ	Все выходы сохраняют свои прежние состояния.
Состояния команд таймеров (кроме TTIM(087), TTIMX(555), MTIM(543) и MTIMX(554))	Сброс	Работающие таймеры (только команды TIM, TIMX(550), TIMH(015), TIMHX(551), TMHH(540), TMHHX(552)) продолжают отсчет времени, поскольку текущие значения обновляются, даже если команды таймеров в это время не выполняются.

3-4-4 БЛОКИРОВКА и ОТМЕНА БЛОКИРОВКИ: IL(002) и ILC(003)

Назначение

Блокировка всех выходов между IL(002) и ILC(003), когда выключено условие выполнения для IL(002). IL(002) и ILC(003), как правило, используются в паре.

Символы РКС



Варианты выполнения

Варианты выполнения	Блокирует, если ВЫКЛ / Не блокирует, если ВКЛ	IL(002)
Модификатор мгновенного обновления		Не предусмотрено

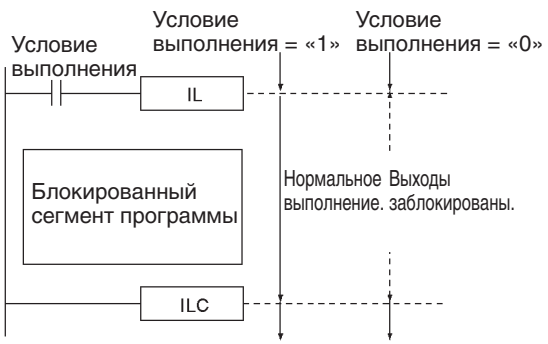
Варианты выполнения	Выполнение в каждом цикле при включенном условии выполнения	ILC(003)
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
Не допускается	Не допускается	OK	OK

Описание

Если условие выполнения для IL(002) выключено, выходы всех команд, расположенных между IL(002) и ILC(003), блокируются. Если условие выполнения для IL(002) включено, команды, расположенные между IL(002) и ILC(003), работают обычным образом.



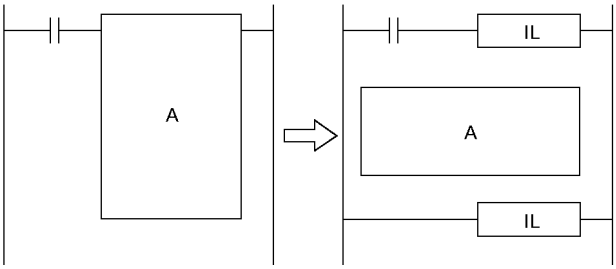
В следующей таблице показано состояние различных выходов в заблокированном сегменте программы между командами IL(002) и ILC(003).

Команда		Состояние
Биты, указанные в OUT, OUT NOT или OUTB(534)		ВЫКЛ
TIM, TIMX(550), TIMH(015), TIMHX(551), TMHH(540), TMHHX(552), TIML(542) и TIMXL(553)	Флаг завершения	ВЫКЛ (сброс)
	Текущее значение	Заданное время (сброс)
Биты/слова, указанные во всех остальных командах (см. прим.)		Сохраняют прежние состояния.

Примечание. Биты и слова во всех остальных командах, включая команды TTIM(087), TTIMX(555), MTIM(543), MTIMX(554), SET, RSET, CNT, CNTX(546), CNTR(012), CNTRX(548), SFT и KEEP(011), сохраняют свои прежние состояния.

Если есть необходимость оставить включенными какие-либо биты в заблокированном сегменте программы, установите их командой SET непосредственно перед командой IL(002).

Сегменты программы зачастую более эффективно переключать с помощью команд IL(002) и ILC(003). Если выполнение нескольких процессов зависит от одного условия, можно сократить число программных шагов, поместив эти процессы между командами IL(002) и ILC(003).



В следующей таблице показаны различия между командами IL(002)/ ILC(003) и JMP(004)/JME(005).

Параметр	Действие в IL(002)/ILC(003)	Действие в JMP(004)/JME(005)
Выполнение команд	Не выполняются никакие команды, кроме OUT, OUT NOT, OUTB(534) и команд таймеров.	Не выполняются никакие команды.
Состояния выходов в командах	Все выходы сохраняют свои прежние состояния, за исключением выходов команд OUT, OUT NOT, OUTB(534) и команд таймеров.	Все выходы сохраняют свои прежние состояния.
Биты в OUT, OUT NOT, OUTB(534)	ВЫКЛ	Все выходы сохраняют свои прежние состояния.
Состояния команд таймеров (кроме TTIM(087), TTIMX(555), MTIM(543) и MTIMX(554))	Сброс	Работающие таймеры (только команды TIM, TIMX(550), TIMH(015), TIMHX(551), TMHH(540), TMHHX(552)) продолжают отсчет времени, поскольку текущие значения обновляются, даже если команды таймеров в это время не выполняются.

Флаги

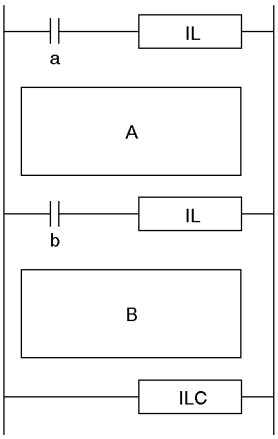
Название	Обозначение	Описание работы
Флаг ошибки	ER	ВЫКЛ
Флаг равенства	=	ВЫКЛ или не изменяется
Флаг отрицательного значения	N	ВЫКЛ или не изменяется

Меры предосторожности

При блокировке сегмента программы длительность цикла не уменьшается, так как заблокированные команды все равно обрабатываются.

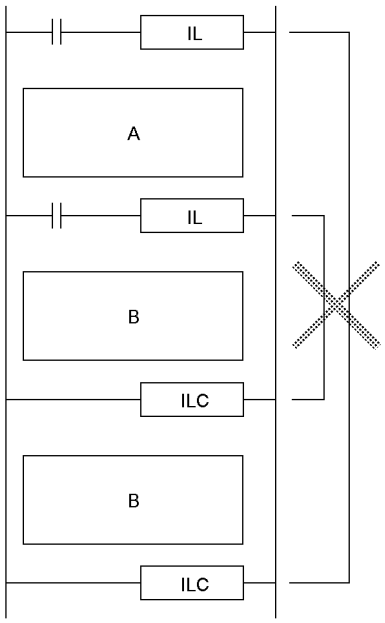
Если команда DIFU(013), DIFD(014) или команда с различением фронтов располагается между командами IL(002) и ILC(003), ее работа зависит не только от состояния условия выполнения. Если условие выполнения для IL(002) выключено, изменение состояния условия выполнения команды DIFU(013), DIFD(014) или команды с различением фронтов не регистрируется.

Обычно команды IL(002) и ILC(003) используются попарно, однако можно использовать две команды IL(002) и более с одной командой ILC(003), как показано на следующей схеме. Если IL(002) и ILC(003) используются не в паре, при проверке программы отображается сообщение об ошибке, но сама программа выполняется корректно.



Условие выполнения		Сегмент программы	
a	b	A	B
ВЫКЛ	ВКЛ	Блокирован	Блокирован
ВЫКЛ	ВЫКЛ	Блокирован	Блокирован
ВКЛ	ВЫКЛ	Не блокирован	Блокирован
ВКЛ	ВКЛ	Не блокирован	Не блокирован

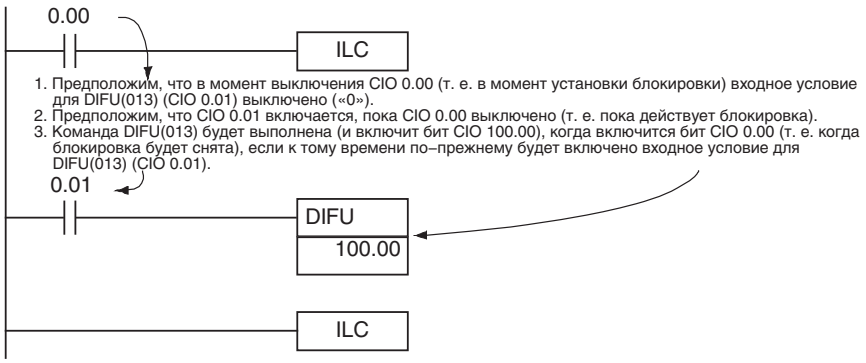
Не допускается вложение команд IL(002) и ILC(003), как показано на следующей схеме. (Для вложения блокировок следует использовать команды MILH(517)/MILR(518) и MILC(519).)



Команды с различением фронтов в блокировках

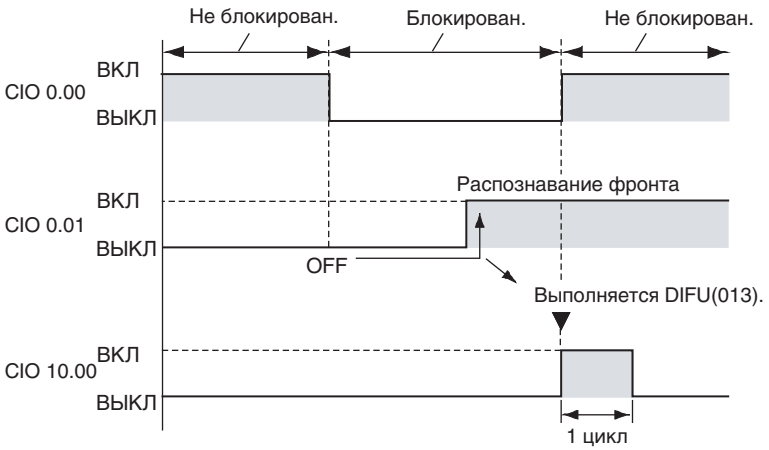
Команды с различением фронтов (DIFU(013), DIFD(014) или команды с префиксом @ или %), расположенные между IL(002) и ILC(003), выполняются или не выполняются в зависимости от того, изменилось ли состояние условия выполнения в памяти с момента установки блокировки до момента ее отмены. Если произошло переключение состояния, распознаваемое командой, эта команда будет выполнена после отмены блокировки.

Например, если в момент установки блокировки входное условие команды DIFU(013) выключено, а в момент снятия блокировки оно включено, вместе со снятием блокировки будет включен бит операнда команды DIFU(013).



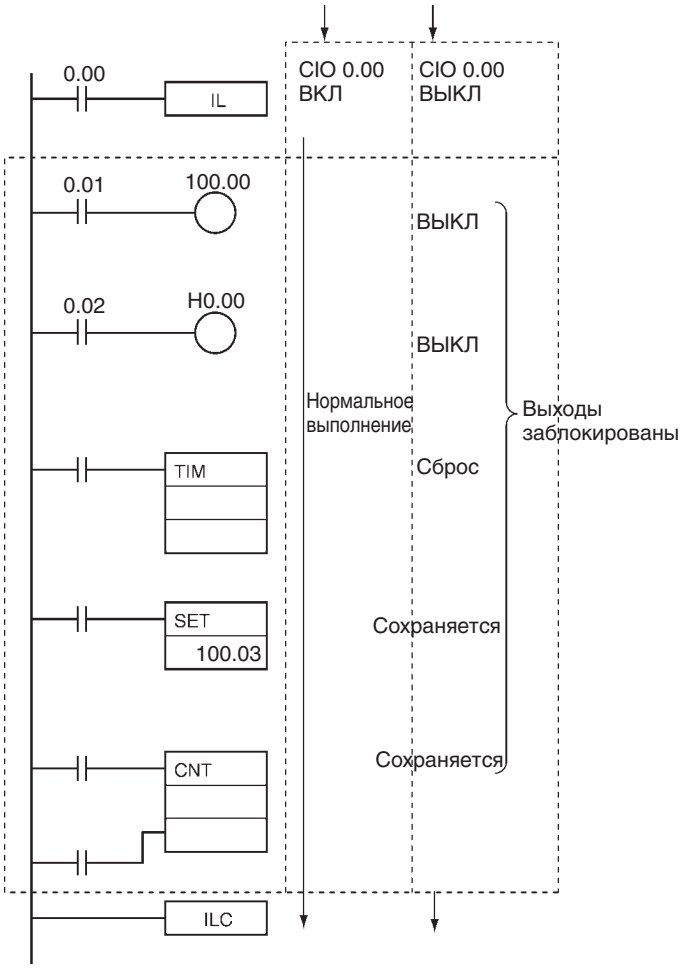
Команда IL(002) действует на команду с различением фронтов так же, как и команда MILH(517).

Временная диаграмма



Примеры

Если в следующем примере бит CIO 0.00 выключен, все выходы между командами IL(002) и ILC(003) будут заблокированы. Если бит CIO 0.00 включен, все команды между командами IL(002) и ILC(003) работают обычным образом.



3-4-5 ВЛОЖЕННАЯ БЛОКИРОВКА С ЗАПОМИНАНИЕМ ФРОНТА, ВЛОЖЕННАЯ БЛОКИРОВКА БЕЗ ЗАПОМИНАНИЯ ФРОНТА и ОТМЕНА ВЛОЖЕННОЙ БЛОКИРОВКИ: MILH(517), MILR(518) и MILC(519)

Назначение

Если условие выполнения для MILH(517) (или MILR(518)) выключено, блокируются выходы всех команд, расположенных между MILH(517)(или MILR(518)) и MILC(519). Команды MILH(517) (или MILR(518)) и MILC(519) обычно используются в паре.

В отличие от команд блокировки IL(002)/ILC(003), команды MILH(517)/ MILC(519) и MILR(518)/MILC(519) допускают вложение. Команды MILH(517) и MILR(518) по-разному влияют на выполнение команд с различением фронтов.

Символы PKC



Операнды

N: Номер блокировки

Номер блокировки должен находиться в интервале 0...15. Номер блокировки команды MILH(517) (или MILR(518)) должен совпадать с номером соответствующей команды MILC(519).

Блокировки можно нумеровать в любом порядке.

D: Бит состояния блокировки

- Включен, если сегмент программы не заблокирован.
- Выключен, если сегмент программы заблокирован.

Если блокировка включена, можно принудительно установить бит состояния блокировки, чтобы отменить блокировку. И наоборот, если блокировка не включена, можно принудительно сбросить бит состояния блокировки, чтобы установить блокировку.

Характеристики операндов

Область	N	D
Область CIO	---	CIO 0.00...CIO 6143.15
Рабочая область	---	W0.00...W511.15
Область битов хранения	---	H0.00...H511.15
Область вспомогательных битов	---	A0.00...A959.15
Область таймеров	---	---
Область счетчиков	---	---
Область DM	---	---
Косвенные адреса DM в двоичном формате	---	---

Область	N	D
Косвенные адреса DM в двоично-десятичном формате (BCD)	---	---
Постоянные	0...15	---
Регистры данных	---	---
Регистры указателей	---	---
Косвенная адресация с помощью регистров указателей	---	,IR0...,IR15 -2048...+2047 ,IR0... -2048...+2047 ,IR15 DR0...DR15, IR0...IR15

Варианты выполнения

Варианты выполнения	**Блокирует, если ВЫКЛ / Не блокирует, если ВКЛ**	MILH(517) и MILR(518)
Модификатор мгновенного обновления		Не предусмотрено

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	MILC(519)
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

В следующей таблице показаны доступные области программы для команд MILH(517), MILR(518) и MILC(519).

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
Не допускается	Не допускается	OK	OK

Описание

Если условие выполнения команды MILH(517) (или MILR(518)) с номером блокировки N выключено, будут заблокированы выходы всех команд, расположенных между этой командой MILH(517)/MILR(518) и следующей командой MILC(519) с номером блокировки N.

Если условие выполнения команды MILH(517) (или MILR(518)) с номером блокировки N включено, команды, расположенные между этой командой MILH(517)/MILR(518) и следующей командой MILC(519) с номером блокировки N, работают обычным образом.

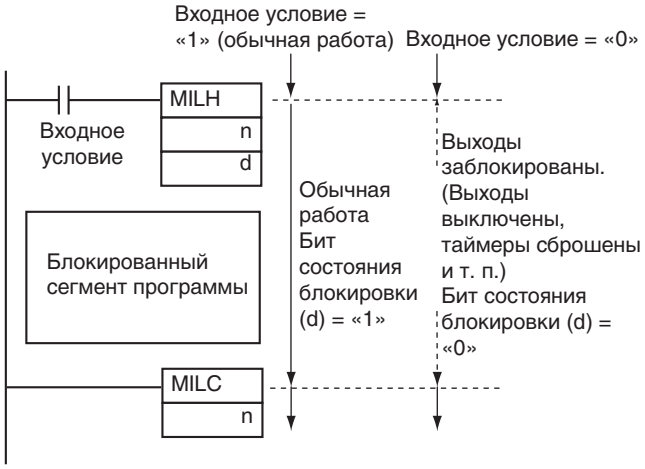
Состояние блокировки

В следующей таблице показано состояние различных выходов в заблокированном сегменте программы между командами MILH(517)/ MILR(518) и следующей командой MILC(519).

Команда		Состояние
Биты, указанные в OUT, OUT NOT или OUTB(534)		ВЫКЛ
TIM, TIMX(550), TIMH(015), TIMHX(551), TMHH(540), TMHHX(552), TIML(542) и TIMXL(553)	Флаг завершения	ВЫКЛ (сброс)
	Текущее значение	Заданное время (сброс)
Биты/слова, указанные во всех остальных командах (см. прим.)		Сохраняют прежние состояния.

Примечание. Биты и слова во всех остальных командах, включая команды TTIM(087), TTIMX(555), MTIM(543), MTIMX(554), SET, RSET, CNT, CNTX(546), CNTR(012), CNTRX(548), SFT и KEEP(011), сохраняют свои прежние состояния.

Команда MILH(517)/MILR(518) сбрасывает бит состояния блокировки (операнд D) при установке блокировки и устанавливает этот бит при выключении блокировки. Таким образом, по состоянию бита состояния блокировки можно определить, включена ли блокировка с заданным номером.



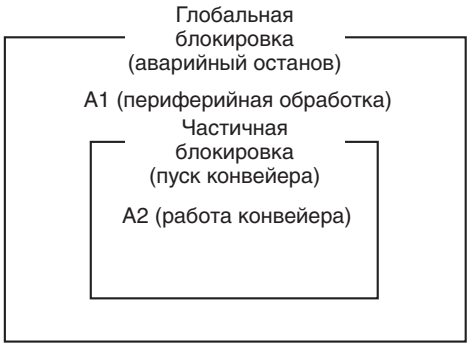
Вложение блокировок

Вложение блокировок означает, что один блокируемый сегмент программы (между командами MILH(517)/MILR(518) и MILC(519)) размещается в другом блокируемом сегменте программы (между командами MILH(517)/MILR(518) и MILC(519)). Может быть до 16 уровней вложения.

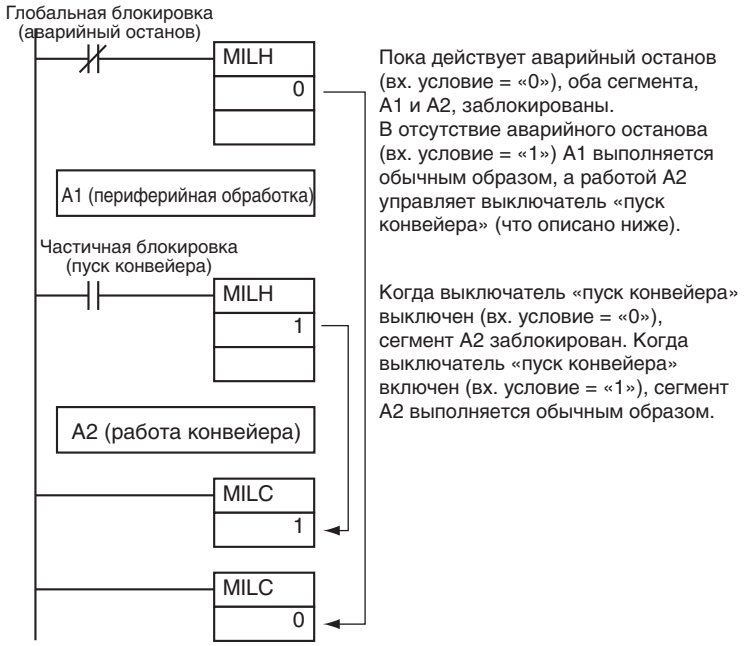
Вложение блокировок позволяет выполнять следующие задачи.

- Пример 1

  Блокировка всей программы при одном условии, блокировка только части программы — при другом условии (1 уровень вложения).

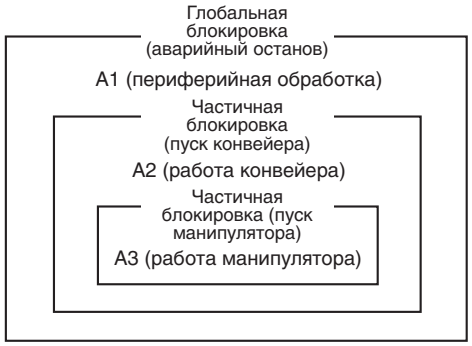


  - A1 и A2 блокируются, если нажата кнопка аварийного останова.
  - A2 блокируется, если выключен переключатель «Ход конвейера».



• Пример 2

Блокировка всей программы при одном условии и блокировка двух перекрывающихся частей программы при других условиях (2 уровня вложения).



- A1, A2 и A3 блокируются, если нажата кнопка аварийного останова.
- A2 и A3 блокируются, если выключен переключатель «Ход конвейера».
- A3 блокируется, если выключен переключатель «Ход манипулятора».



Пока действует аварийный останов (вх. условие = «0»), сегменты A1, A2 и A3 заблокированы.
В отсутствие аварийного останова (вх. условие = «1») A1 выполняется обычным образом, а работой A2 и A3 управляют выключатели «пуск конвейера» и «пуск манипулятора» (что описано ниже).

Когда выключатель «пуск конвейера» выключен (вх. условие = «0»), оба сегмента, A2 и A3, заблокированы.
Когда выключатель «пуск конвейера» включен (вх. условие = «1»), сегмент A2 выполняется обычным образом, а работой A3 управляет выключатель «пуск манипулятора» (что описано ниже).

Когда выключатель «пуск манипулятора» выключен (вх. условие = «0»), сегмент A3 заблокирован.
Когда выключатель «пуск манипулятора» включен (вх. условие = «1»), сегмент A3 выполняется обычным образом.

Различия между MILH(517) и MILR(518)

Команды блокировки MILH(517) и MILR(518) по-разному воздействуют на работу команд с различением фронтов (DIFU(013), DIFD(014) или команд с префиксом @ или %).

Если сегмент программы блокирован командой MILR(518), команда с различением фронтов **не выполняется** после отмены блокировки, даже если во время действия блокировки произошло переключение состояния, различаемое этой командой (что определяется путем сравнения состояний условия выполнения при включении и при отмене блокировки).

Если сегмент программы блокирован командой MILH(517), команда с различением фронтов **выполняется** после отмены блокировки, если во время действия блокировки произошло переключение состояния, различаемое этой командой (что определяется путем сравнения состояний условия выполнения при включении и при отмене блокировки).

Команда	Работа команд с различением фронтов
MILH(517) ВЛОЖЕННАЯ БЛОКИРОВКА С ЗАПОМИНАНИЕМ ФРОНТА	Команда с различением фронтов (DIFU, DIFD или команда с префиксом @ или %) **выполняется** после отмены блокировки, если во время блокировки произошло распознаваемое переключение состояния. (Состояние условия выполнения при включении блокировки сравнивается с его состоянием при отмене блокировки.)
MILR(518) ВЛОЖЕННАЯ БЛОКИРОВКА БЕЗ ЗАПОМИНАНИЯ ФРОНТА	Команда с различением фронтов (DIFU, DIFD или команда с префиксом @ или %) **не выполняется** после отмены блокировки, даже если во время блокировки произошло распознаваемое переключение состояния.

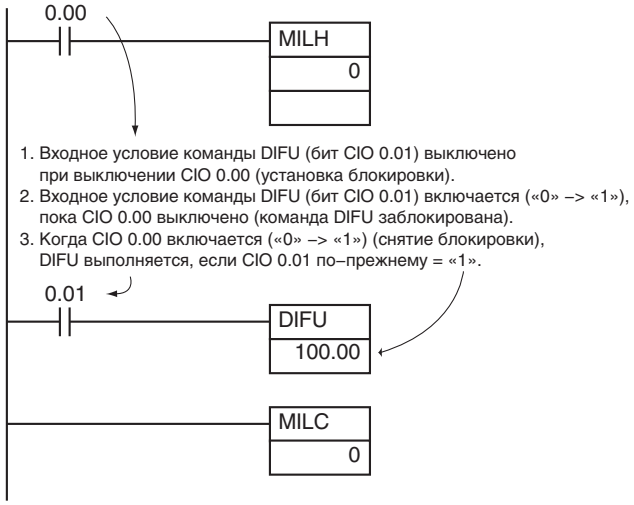
- Работа команд с различением фронтов при блокировке командой MILH(517)

  Если между командой MILH(517) и соответствующей командой MILC(519) расположена команда с различением фронтов (DIFU, DIFD или команда с префиксом @ или %), такая команда **выполняется** после отмены блокировки, если произошло переключение состояния, распознаваемое этой командой. (Система сравнивает состояние условия выполнения при включении блокировки с его состоянием при отмене блокировки.)

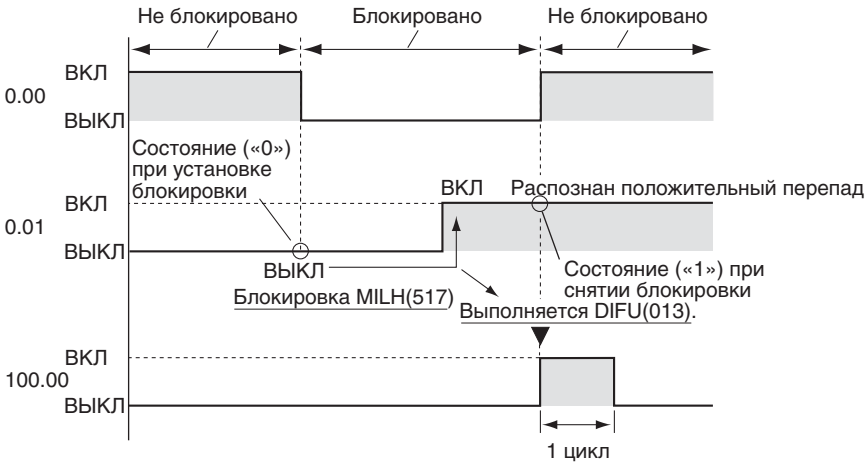
  Команда с различением фронтов будет выполнена и в том случае, если распознаваемое ею переключение состояния произойдет в момент установки или отмены блокировки.

  Многие другие события программы могут вызвать сброс условия выполнения команды с различением фронтов, даже если оно было установлено во время действия блокировки. В этом случае команда с различением фронтов не выполняется после отмены блокировки.

  - Пример
    Если используется команда различения положительного фронта (DIFU(013)) и входное условие выключено в момент установки блокировки и включено в момент отмены блокировки, команда DIFU(013) **выполняется** после отмены блокировки. (Команда блокировки MILH(517) действует на команды с различением фронтов так же, как команда блокировки IL(002).)



Временная диаграмма

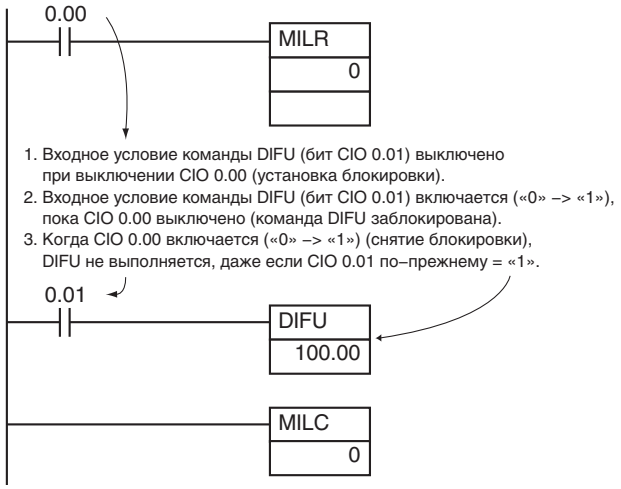


- Работа команд с различением фронтов при блокировке командой MILR(518)

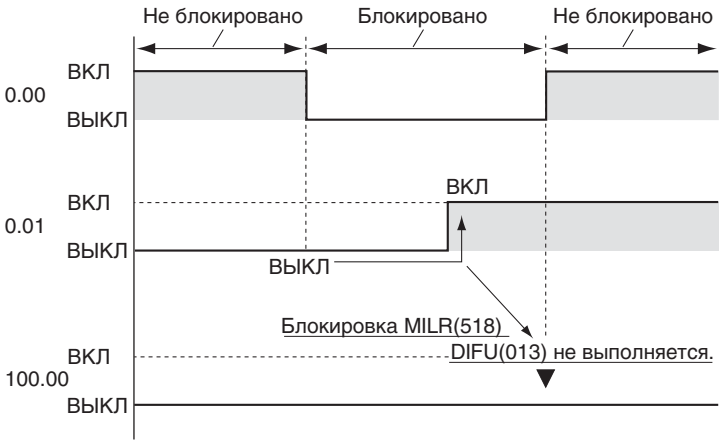
  Если между командой MILR(518) и соответствующей командой MILC(519) расположена команда с различением фронтов (DIFU, DIFD или команда с префиксом @ или %), такая команда **не выполняется** после отмены блокировки, даже если в промежутке между установкой и снятием блокировки произошло переключение состояния, распознаваемое этой командой. (Система сравнивает состояние условия выполнения в цикле, в котором была установлена блокировка, с состоянием условия выполнения в цикле, в котором блокировка была отменена.)

  Команда с распознаванием фронтов не выполняется и в том случае, если распознаваемое ею переключение состояния происходит в момент установки или отмены блокировки.

  - Пример
    Если используется команда различения положительного фронта (DIFU(013)) и входное условие выключено в момент установки блокировки и включено в момент отмены блокировки, команда DIFU(013) **не выполняется** после отмены блокировки.



Временная диаграмма



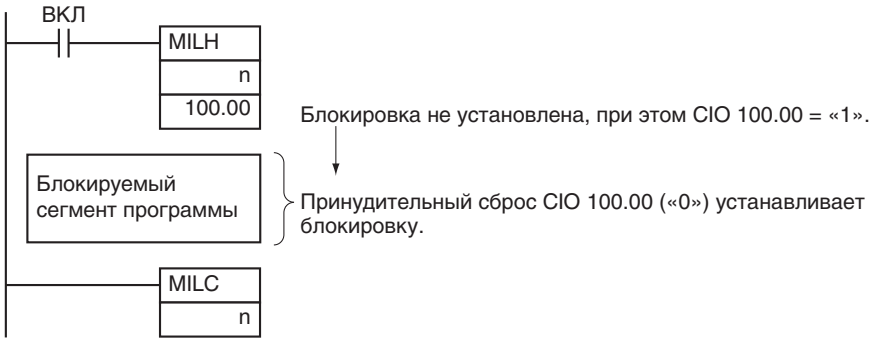
Управление состоянием блокировки из CX-Programmer

Блокировка может быть включена или отменена вручную путем принудительного сброса или установки бита состояния блокировки (который указывается в операнде D команд MILH(517) и MILR(518)) из CX-Programmer. Принудительное состояние бита состояния блокировки обладает приоритетом и отменяет любое состояние блокировки, полученное в результате выполнения программы.

Принудительная установка: отменяет блокировку.

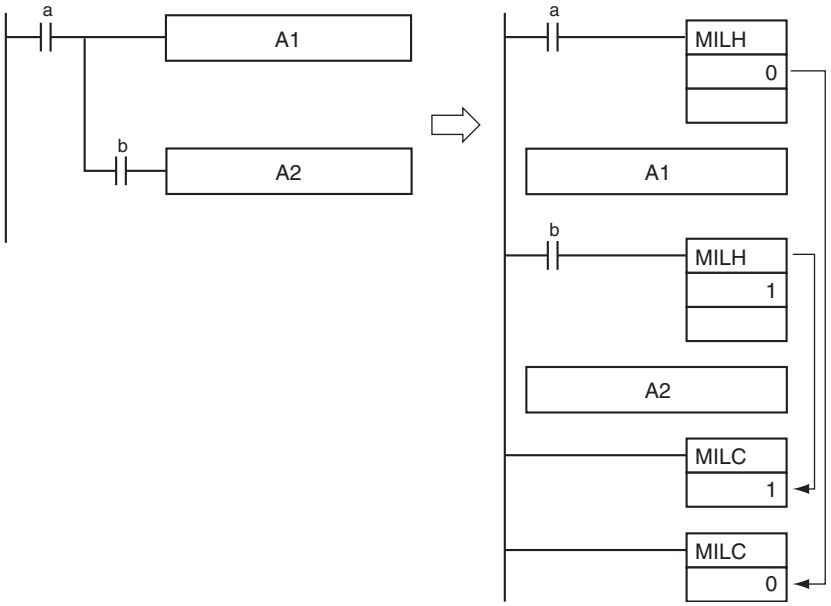


Принудительный сброс: устанавливает блокировку.



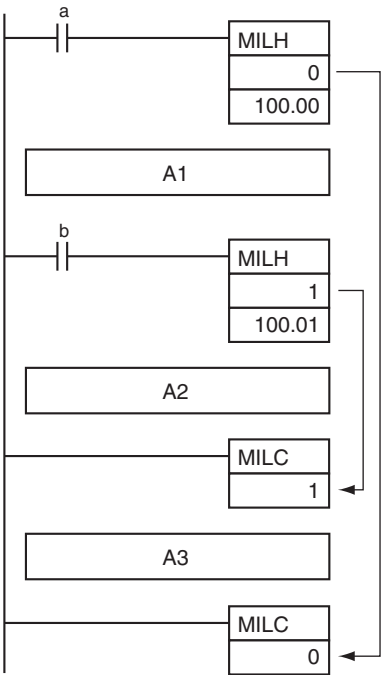
Примечание. Режимы работы программы можно более эффективно переключать с помощью команд блокировки MILH(517) или MILR(518).

Вместо того чтобы использовать для переключения между сегментами программы сложные условия, разместите перед каждым процессом команду MILH(517) или MILR(518), а команду MILC(519) — после каждого процесса.



В отличие от команды блокировки IL(002), команды блокировки MILH(517) и MILR(518) допускают вложение, поэтому та же программа будет работать по-другому, если вместо команды ILC(002) будет использована команда MILH(517) или MILR(518).

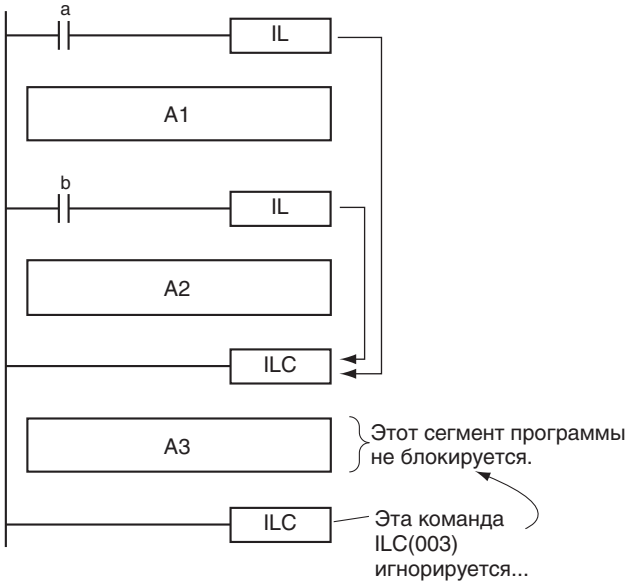
Программа с блокировками MILH(517)/MILC(519)



Условие выполнения		Сегмент программы		
a	**b**	**A1**	**A2**	**A3**
ВЫКЛ	ВКЛ	Блокирован	Блокирован	Блокирован
	ВЫКЛ			

Условие выполнения		Сегмент программы		
a	**b**	**A1**	**A2**	**A3**
ВКЛ	ВЫКЛ	Не блокирован	Блокирован	Не блокирован
ВКЛ	ВКЛ	Не блокирован	Не блокирован	Не блокирован

Программа с блокировками IL(002)/ILC(003)



Условие выполнения		Сегмент программы		
a	**b**	**A1**	**A2**	**A3**
ВЫКЛ	ВКЛ	Блокирован	Блокирован	Не блокирован (Блокировка IL(002)/ILC(003) не действует.)
	ВЫКЛ			
ВКЛ	ВЫКЛ	Не блокирован	Блокирован	
ВКЛ	ВКЛ	Не блокирован	Не блокирован	

Если необходимо оставить включенными какие-либо биты в сегменте программы, заблокированном командой MILH(517) или MILR(518), установите эти биты командой SET непосредственно перед командой MILH(517) или MILR(518).

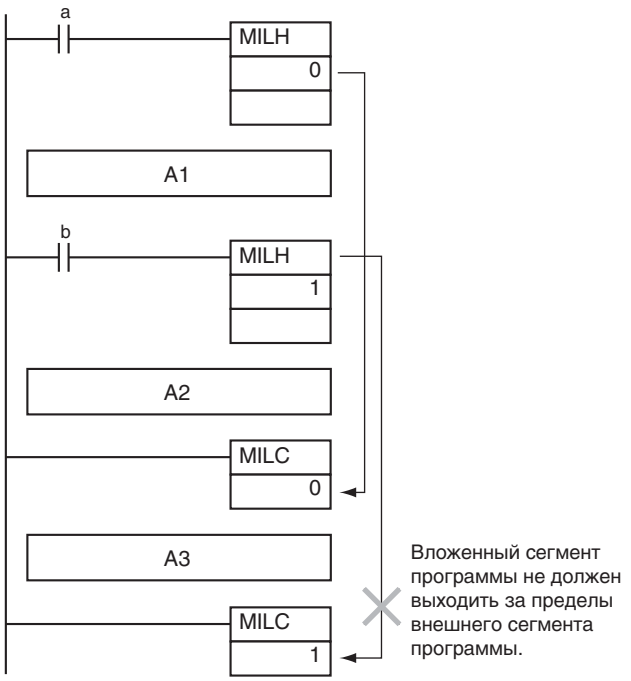
Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	ВЫКЛ

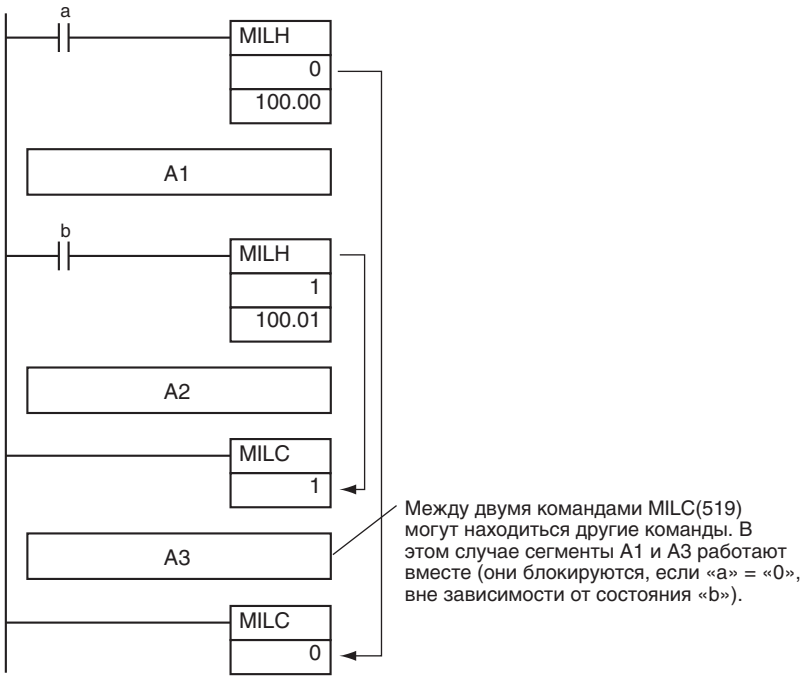
Меры предосторожности

При блокировке сегмента программы командой MILH(517) или MILR(518) длительность цикла не уменьшается, так как заблокированные команды все равно обрабатываются.

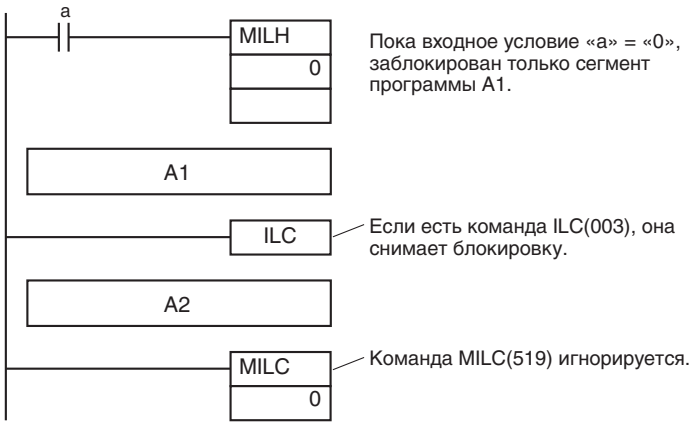
При вложении блокировок их следует нумеровать так, чтобы каждая вложенная блокировка не выходила за пределы блокировки более высокого уровня.



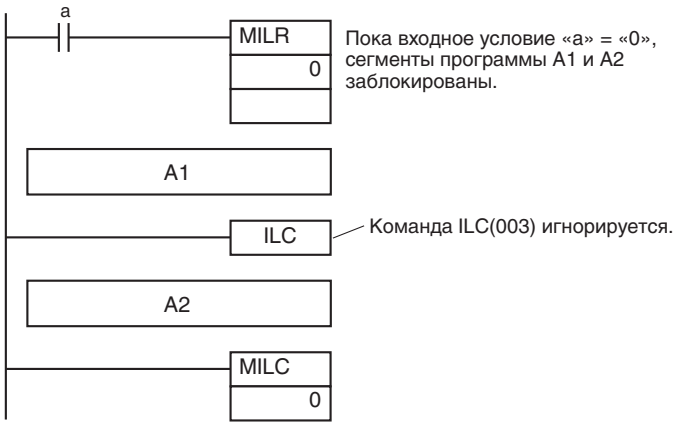
Другие команды можно вставлять между командами MILC(519), как показано на следующей схеме.



Если между парой команд MILH(517) и MILC(519) расположена команда ILC(003), будет заблокирован сегмент программы между MILH(517) и ILC(003).

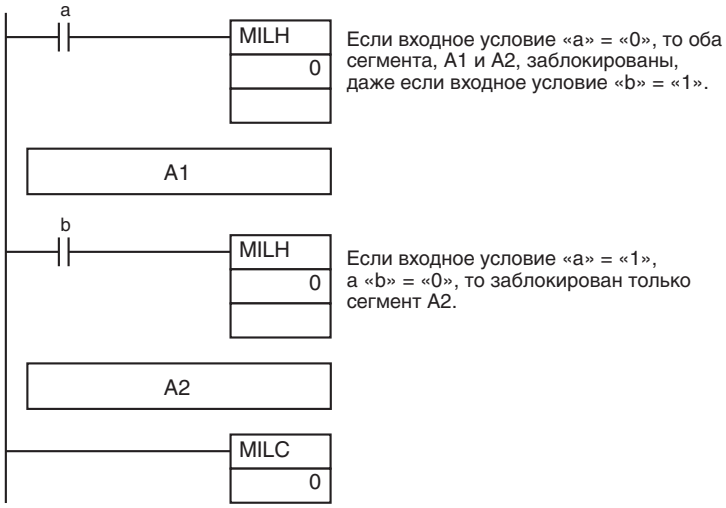


Если команда ILC(003) расположена между парой команд MILR(518) и MILC(519), она игнорируется и блокируется весь сегмент программы между MILR(518) и MILC(519).



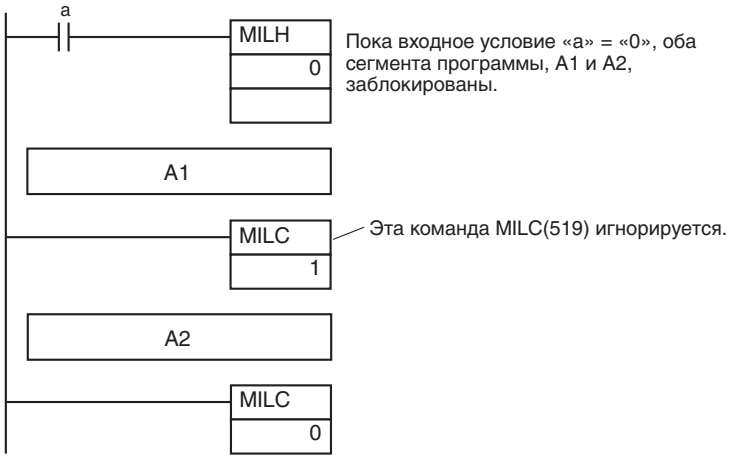
Если между парой команд MILH(517) и MILC(519) расположена еще одна команда MILH(517) или MILR(518) с тем же номером блокировки и первая команда MILH(517) приведена в действие, вторая команда MILH(517)/MILR(518) не действует.

Если между парой команд MILH(517) и MILC(519) расположена еще одна команда MILH(517) или MILR(518) с тем же номером блокировки и первая команда MILH(517) не приведена в действие, вторая команда MILH(517)/MILR(518) работает обычным образом.



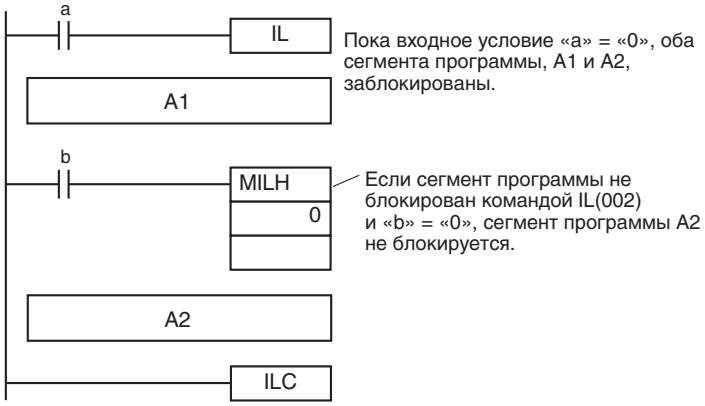
Примечание. Команда блокировки MILR(518) работает аналогичным образом, если между парой команд MILR(518) и MILC(519) расположена еще одна команда MILH(517) или MILR(518) с тем же номером блокировки.

Если между парой команд MILH(517)/MILR(518) и MILC(519) находится команда MILC(519) с другим номером блокировки, она игнорируется.

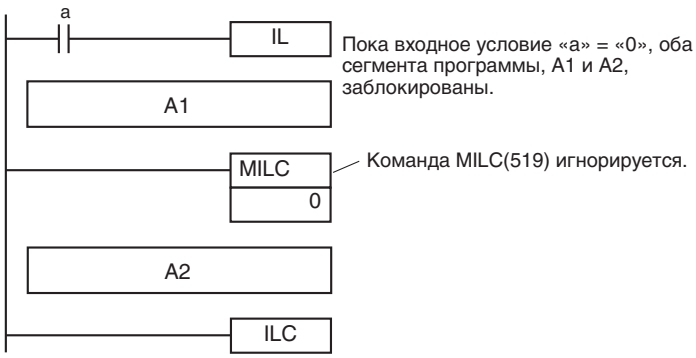


Если между парой команд IL(002) и ILC(003) расположена команда MILH(517) и команда IL(002) приводится в действие, команда MILH(517) не оказывает никакого влияния. В этом случае блокируется сегмент программы между IL(002) и ILC(003).

Если команда IL(002) не приведена в действие и выключено условие выполнения команды MILH(517) (в данном случае «b»), блокируется сегмент программы между MILH(517) и ILC(003).



Если между парой команд IL(002) и ILC(003) находится команда MILC(519), она игнорируется и блокируется весь сегмент программы между IL(002) и ILC(003).

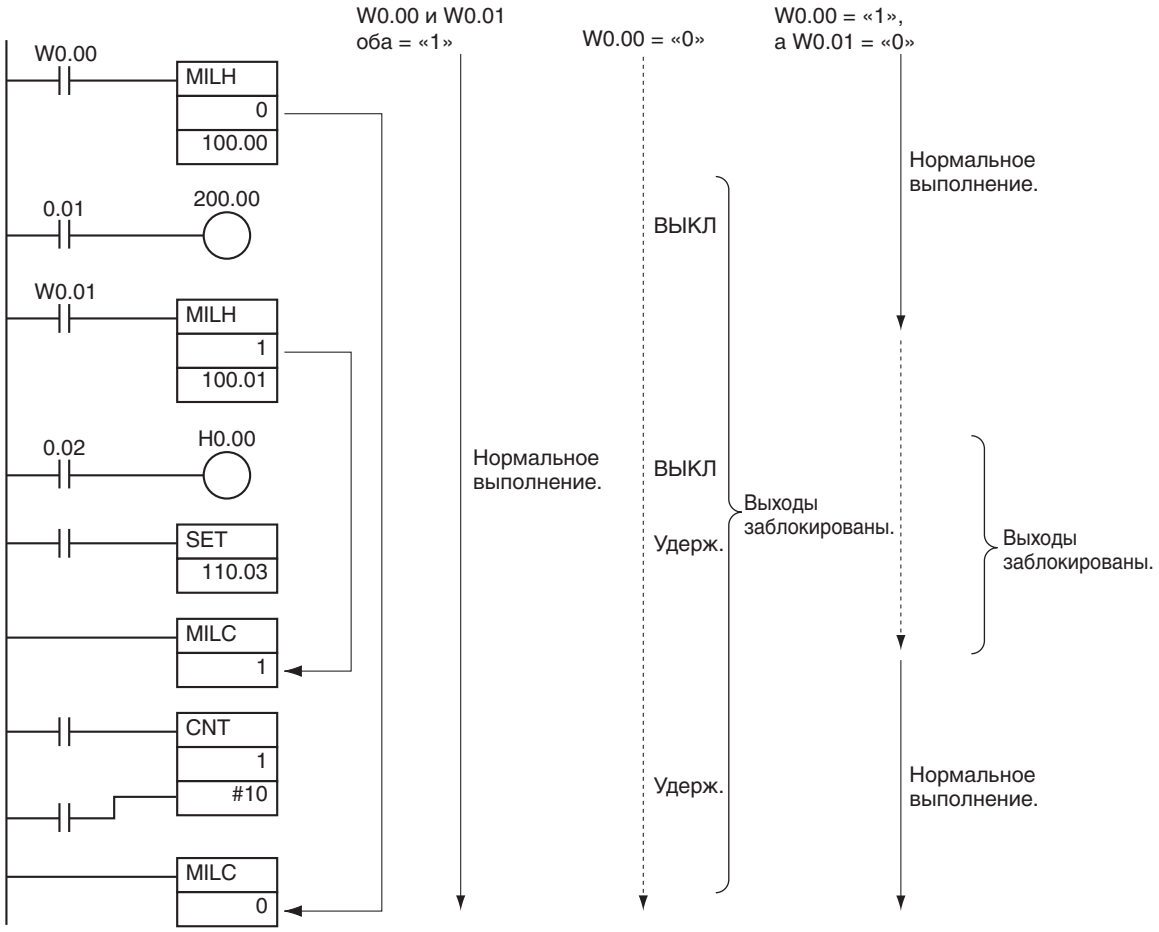


Примеры

Если одновременно включены биты W0.00 и W0.01, команды, расположенные между командой MILH(517) с номером блокировки 0 и MILC(519) с номером блокировки 0, выполняются обычным образом.

Если выключен бит W0.00, блокируются команды, расположенные между командой MILH(517) с номером блокировки 0 и MILC(519) с номером блокировки 0.

Если бит W0.00 включен, а бит W0.01 выключен, блокируются команды, расположенные между командой MILH(517) с номером блокировки 1 и MILC(519) с номером блокировки 1. Остальные команды работают обычным образом.

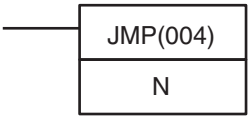


3-4-6 ПЕРЕХОД и КОНЕЦ ПЕРЕХОДА: JMP(004) и JME(005)

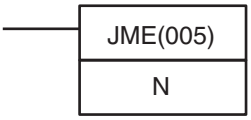
Назначение

Если условие выполнения для JMP(004) выключено, происходит переход к первой встретившейся в программе команде JME(005) с тем же номером и выполнение программы продолжается с того места. Команды JMP(004) и JME(005) используются в паре.

Символы РКС



N: Номер перехода



N: Номер перехода

Варианты выполнения

Варианты выполнения	Переходит, если ВЫКЛ / Не переходит, если ВКЛ	JMP(004)
Модификатор мгновенного обновления		Не предусмотрено

Варианты выполнения	Выполнение в каждом цикле при включенном условии выполнения	JME(005)
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	Не допускается	OK	OK

Операнды

N: Номер перехода

Номер перехода должен находиться в диапазоне 0000...00FF (&0...&255 десятичн.).

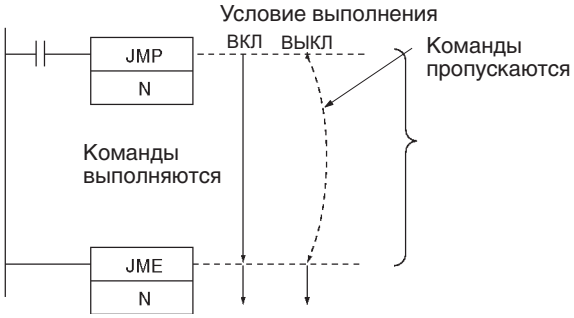
Характеристики операндов

Область	N	
	JMP(004)	JME(005)
Область CIO	CIO 0...CIO 6143	---
Рабочая область	W0...W511	---
Область битов хранения	H0...H511	---
Область вспомогательных битов	A0...A959	---
Область таймеров	T0000...T4095	---
Область счетчиков	C0000...C4095	---
Область DM	D0...D32767	---
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	---
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767	---
Постоянные	#0000...#00FF (двоичный) или &0...&255	#0000...#00FF (двоичный) или &0...&255
Регистры данных	DR0...DR15	---
Регистры указателей	---	---
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0... -2048...+2047 ,IR15 DR0...DR15, IR0...IR15	---

Описание

Если условие выполнения для JMP(004) включено, перехода не происходит и команды программы выполняются в порядке следования.

Если условие выполнения для JMP(004) выключено, происходит переход к первой встретившейся в программе команде JME(005) с тем же номером и выполнение программы продолжается с того места. Команды, расположенные между JMP(004) и JME(005), не выполняются, поэтому состояния выходов между JMP(004) и JME(005) остаются неизменными. В блочных программах команды, расположенные между JMP(004) и JME(005), пропускаются независимо от состояния условия выполнения.



Когда условие выполнения команды JMP(004) выключено, ни одна из команд между JMP(004) и JME(005) не выполняется, поэтому длительность цикла уменьшается на общее время выполнения пропущенных команд. Если же переход организован командами JMP0(515) и JME0(516), то вместо пропускаемых команд выполняются команды NOP(000), поэтому в этом случае время выполнения цикла сокращается не так существенно.

В следующей таблице приведено сравнение различных команд перехода.

Параметр	JMP(004) JME(005)	CJP(510) JME(005)	CJPN(511) JME(005)	JMP0(515) JME0(516)
Условие выполнения перехода	ВЫКЛ	ВКЛ	ВЫКЛ	ВЫКЛ
Допустимое количество	256 максимум			Не ограничено
Выполнение команд при переходе	Не выполняется.			Выполняется NOP(000)
Время выполнения команд при переходе	Нет			Время выполнения NOP(000)
Состояние выходов (битов и слов) при переходе	Биты и слова сохраняют свои прежние значения.			
Состояние работающих таймеров при переходе	Работающие таймеры продолжают отсчет времени.			
Выполнение в блочных программах	Переход выполняется всегда.	Переход, если ВКЛ.	Переход, если ВЫКЛ.	Не допустимо.

Флаги (JMP)

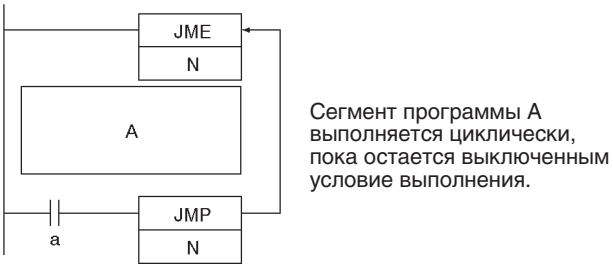
Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если значение N выходит за допустимый диапазон 0...255 (0000...00FF hex). Включен, если в программе есть команда JMP(004), но нет команды JME(005) с тем же номером перехода. Включен, если в задаче есть команда JMP(004), но нет команды JME(005) с тем же номером перехода. Выключен во всех остальных случаях.

Меры предосторожности

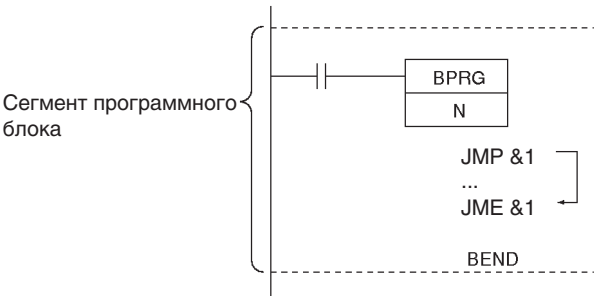
Все выходы (биты и слова) команд пропускаемого участка программы сохраняют свои прежние состояния. Работающие таймеры (TIM, TIMX(550), TIMH(015), TIMHX(551), TMHH(540) и TMHHX(552)) продолжают отсчет времени, поскольку текущие значения обновляются, даже если команды таймеров в это время не выполняются.

Если в программе присутствуют несколько (две и больше) команд JME(005) с одинаковым номером перехода, в работе участвует только команда с наименьшим адресом. Команды JME(005) с тем же номером перехода, расположенные в программе по более высоким адресам, игнорируются.

Если команда JME(005) находится в программе перед командой JMP(004), команды, расположенные между JME(005) и JMP(004), будут выполняться циклически до тех пор, пока не включится условие выполнения JMP(004). Если по истечении максимальной длительности цикла не включится условие выполнения или не будет выполнена команда END(001), будет сгенерирована ошибка превышения допустимой длительности цикла.



В блочных программах команды между JMP(004) и JME(005) пропускаются независимо от состояния условия выполнения JMP(004).



Парные команды JMP(004) и JME(005) должны находиться в пределах одной задачи, так как переход между задачами не допускается. Если в задаче присутствует команда JME(005), но отсутствует соответствующая ей команда JMP(004), будет сгенерирована ошибка.

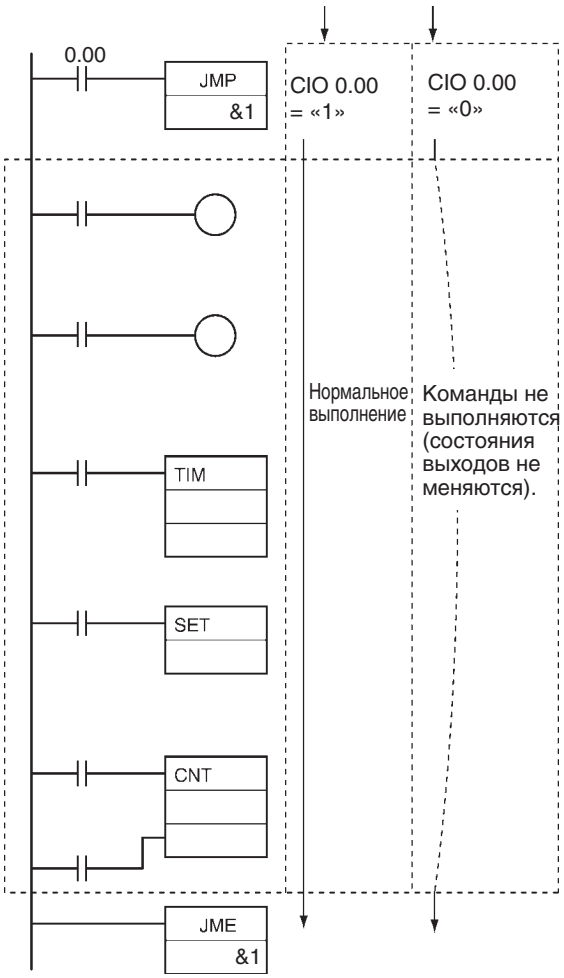
Работа команд DIFU(013), DIFD(014) и команд с различением фронтов, размещенных между командами JMP(004) и JME(005), зависит не только от состояния условия выполнения. Если команда DIFU(013), DIFD(014) или команда с различением фронтов в пропускаемом сегменте выполняется сразу после включения условия выполнения JMP(004), состояние условия выполнения команды DIFU(013), DIFD(014) или команды с различением фронтов сравнивается с состоянием условия выполнения до осуществления такого перехода (т. е. до выключения условия выполнения команды JMP(004)).

Примеры

Базовые операции

Если в следующем примере бит CIO 0.00 выключен, расположенные между JMP(004) и JME(005) команды не выполняются, состояние выходов остается неизменным.

Если бит CIO 0.00 включен, расположенные между JMP(004) и JME(005) команды выполняются обычным образом.



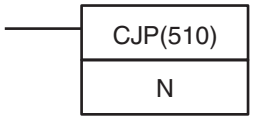
3-4-7 УСЛОВНЫЙ ПЕРЕХОД: CJP(510)/CJPN(511)

Назначение

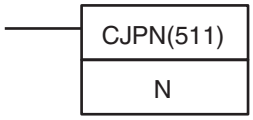
Действие команды CJP(510) противоположно действию команды JMP(004). Если условие выполнения для CJP(510) включено, происходит переход к первой встретившейся в программе команде JME(005) с тем же номером и выполнение программы продолжается с того места. Команды CJP(510) и JME(005) используются в паре.

Действие команды CJPN(511) практически полностью совпадает с действием JMP(004). Если условие выполнения для CJPN(511) выключено, происходит переход к первой встретившейся в программе команде JME(005) с тем же номером перехода. Команды CJPN(511) и JME(005) используются в паре.

Символы РКС



N: Номер перехода



N: Номер перехода

Варианты выполнения

Варианты выполнения	**Переходит, если ВКЛ / Не переходит, если ВЫКЛ**	CJP(510)
Модификатор мгновенного обновления		Не предусмотрено

Варианты выполнения	**Переходит, если ВЫКЛ / Не переходит, если ВКЛ**	CJPN(511)
Модификатор мгновенного обновления		Не предусмотрено

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	JME(005)
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	**Области пошаговых программ**	**Подпрограммы**	**Задачи обработки прерываний**
OK	Не допускается	OK	OK

Операнды

N: Номер перехода

Номер перехода должен находиться в диапазоне 0000...00FF (0...255 десятичн.).

Характеристики операндов

Область	**N**		
	CJP(510)	**CJPN(511)**	**JME(005)**
Область CIO	CIO 0...CIO 6143		---
Рабочая область	W0...W511		---
Область битов хранения	H0...H511		---
Область вспомогательных битов	A0...A959		---
Область таймеров	T0000...T4095		---
Область счетчиков	C0000...C4095		---
Область DM	D0...D32767		---
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		---
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767		---
Постоянные	#0000...#00FF (двоичный) или &0...&255		#0000...#00FF (двоичный) или &0...&255
Регистры данных	DR0...DR15		---
Регистры указателей	---		---
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15		---

Описание

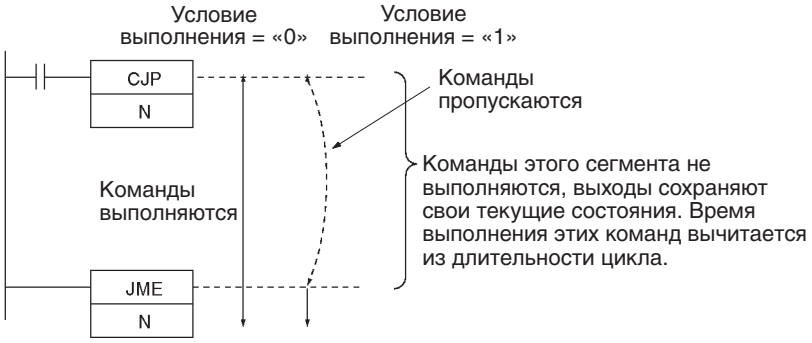
Действие команд CJP(510) CJPN(511) отличается только условием выполнения. Команда CJP(510) выполняет переход к первой команде JME(005), когда условие выполнения включено, а CJPN(511) выполняет переход к первой команде JME(005), когда условие выполнения выключено.

Так как пропускаемые при переходе команды не выполняются, длительность цикла сокращается на общее время выполнения пропускаемых команд.

Работа команды CJP(510)

Если условие выполнения для CJP(510) выключено, перехода не происходит и команды программы выполняются в порядке следования.

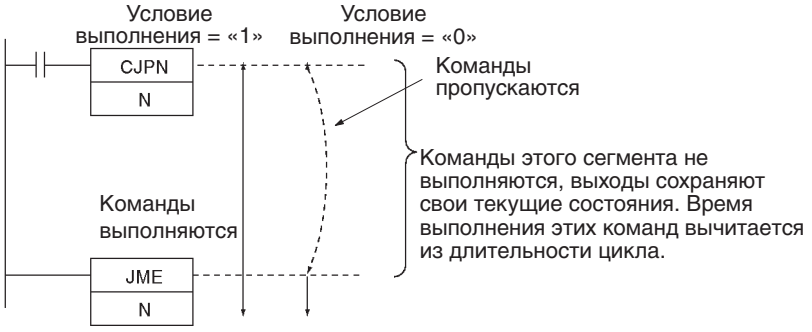
Если условие выполнения для CJP(510) включено, происходит переход к первой встретившейся в программе команде JME(005) с тем же номером и выполнение программы продолжается с того места.



Работа команды CJPN(511)

Если условие выполнения для CJPN(511) включено, перехода не происходит и команды программы выполняются в порядке следования.

Если условие выполнения для CJPN(511) выключено, происходит переход к первой встретившейся в программе команде JME(005) с тем же номером перехода и выполнение программы продолжается с этого места.



Флаги

В следующей таблице приведены флаги, на состояние которых влияют команды CJP(510) и CJPN(511).

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если отсутствует команда JME(005) с номером перехода, соответствующим номеру команды CJP(510) или CJPN(511). Включен, если значение N выходит за допустимый диапазон 0...255 (0000...00FF hex). Включен, если в задаче есть команда CJP(510) или CJPN(511), но нет команды JME(005) с таким же номером. Выключен во всех остальных случаях.

Меры предосторожности

Все выходы (биты и слова) команд пропускаемого участка программы сохраняют свои прежние состояния. Работающие таймеры (TIM, TIMX(550), TIMH(015), TIMHX(551), TMHH(540) и TMHHX(552)) продолжают отсчет времени, поскольку текущие значения обновляются, даже если команды таймеров в это время не выполняются.

Если в программе присутствуют несколько (две и больше) команд JME(005) с одинаковым номером перехода, в работе участвует только команда с наименьшим адресом. Команды JME(005) с тем же номером перехода, расположенные в программе по более высоким адресам, игнорируются.

Если команда JME(005) в программе предшествует команде CJP(510) или CJPN(511), расположенные между ними команды будут выполняться циклически до тех пор, пока условие выполнения не включится (для CJP(510)) или не выключится (для CJPN(511)). Если по истечении максимальной длительности цикла переход не будет завершен переключением условия выполнения или командой END(001), будет сгенерирована ошибка превышения допустимой длительности цикла.

Команды CJP(510) и CJPN(511) работают в блочных программах обычным образом.

Если установлено условие выполнения команды CJP(510) или сброшено условие выполнения команды CJPN(511), программа переходит непосредственно к команде JME, не выполняя команд, расположенных между CJP(510)/CJPN(511) и JME. Длительность цикла сокращается, так как пропускаемые команды не выполняются и не требуют затрат времени.

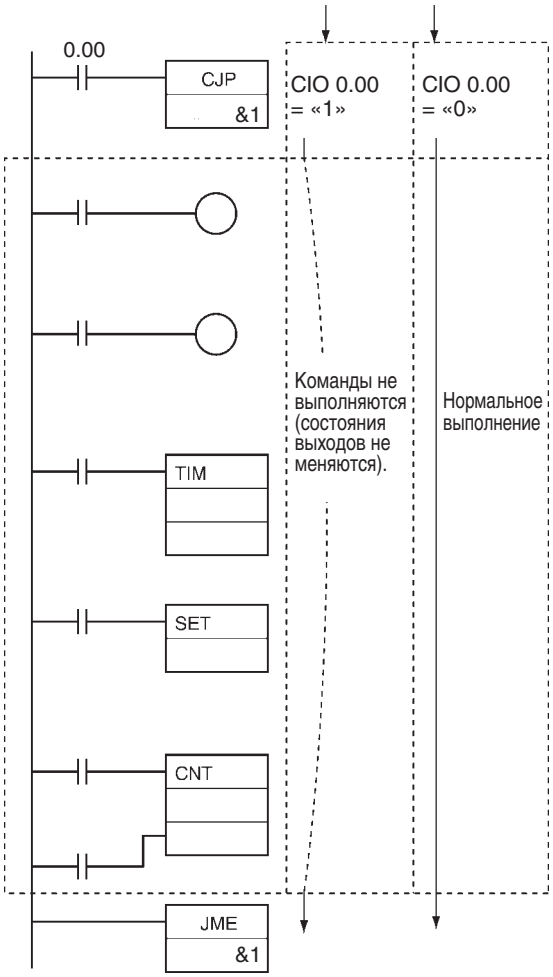
Если условие выполнения команды JMP0 выключено, вместо каждой из команд, расположенных между JMP0 и JME0, выполняется команда NOP, что требует времени. Поэтому длительность цикла в этом случае не сокращается.

Если в задаче используется команда CJP(510) или CJPN(511), в задаче обязательно должна присутствовать команда JME(005) с таким же номером перехода, так как переход между задачами невозможен. Если в задаче отсутствует соответствующая команда JME(005), генерируется ошибка.

Работа команд DIFU(013), DIFD(014) и команд с различением фронтов, расположенных в пропускаемом сегменте программы, зависит не только от состояния условия выполнения. Если команда DIFU(013), DIFD(014) или команда с различением фронтов в пропущенном сегменте программы выполняется сразу после выключения условия выполнения CJP(510) (включения для CJPN(511)), состояние условия выполнения команды DIFU(013), DIFD(014) или команды с различением фронтов сравнивается с состоянием условия выполнения до осуществления такого перехода.

Пример

Если в следующем примере бит CIO 0.00 включен, расположенные между CJP(510) и JME(005) команды не выполняются, состояние выходов остается неизменным.
Если бит CIO 0.00 выключен, расположенные между CJP(510) и JME(005) команды выполняются обычным образом.



Примечание. При выполнении CJPN(511) состояние бита CIO 0.00 («0» или «1») меняется на противоположное.

3-4-8 МНОЖЕСТВЕННЫЙ ПЕРЕХОД и КОНЕЦ ПЕРЕХОДА: JMP0(515) и JME0(516)

Назначение

Если условие выполнения для JMP0(515) выключено, все команды программы, заключенные между JMP0(515) и следующей командой JME0(516), обрабатываются как команда NOP(000). Команды JMP0(515) и JME0(516) должны использоваться в паре. Количество пар таких команд в программе не ограничено.

Символы РКС



Варианты выполнения

Варианты выполнения	**Переходит, если ВЫКЛ / Не переходит, если ВКЛ**	JMP0(515)
Модификатор мгновенного обновления		Не предусмотрено

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	JME0(516)
Модификатор мгновенного обновления		Не предусмотрено

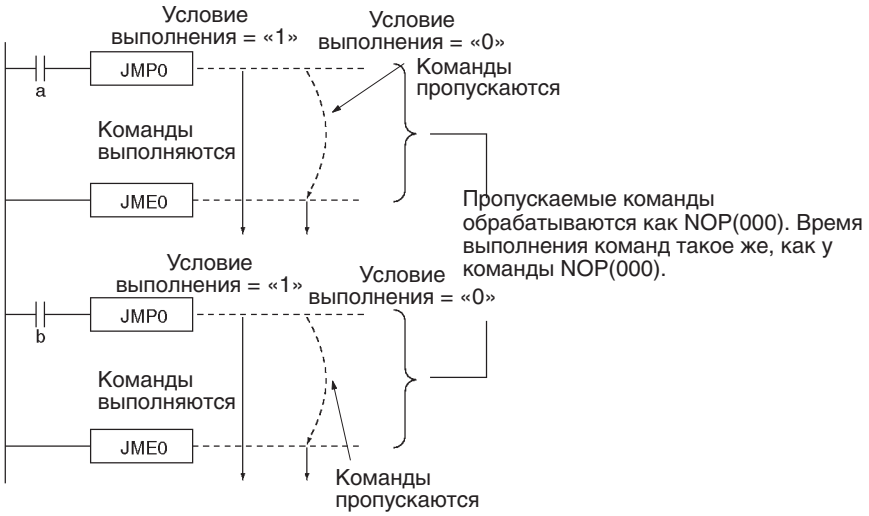
Доступные области программы

Области программных блоков	**Области пошаговых программ**	**Подпрограммы**	**Задачи обработки прерываний**
Не допускается	Не допускается	OK	OK

Описание

Если условие выполнения для JMP0(515) включено, перехода не происходит и команды программы выполняются в порядке следования.

Если условие выполнения для JMP0(515) выключено, все команды программы, заключенные между JMP0(515) и следующей командой JME0(516), обрабатываются как команда NOP(000). В отличие от команд JMP(004), CJP(510) и CJPN(511), для JMP0(515) не используются номера переходов, поэтому эти команды могут располагаться в любом месте программы.



В отличие от команд JMP(004), CJP(510) и CJPN(511), которые выполняют переход к первой встретившейся в программе команде JME(005), при использовании JMP0(515) все команды, расположенные между JMP0(515) и JME0(516), выполняются как команды NOP(000). Время выполнения пропускаемых команд сокращается, но не исключается полностью. Сами пропущенные команды не выполняются, а состояние их выходов (битов и слов) остается неизменным.

Меры предосторожности

В программе может использоваться множество пар команд JMP0(515) и JME0(516), однако вложение пар не допускается.

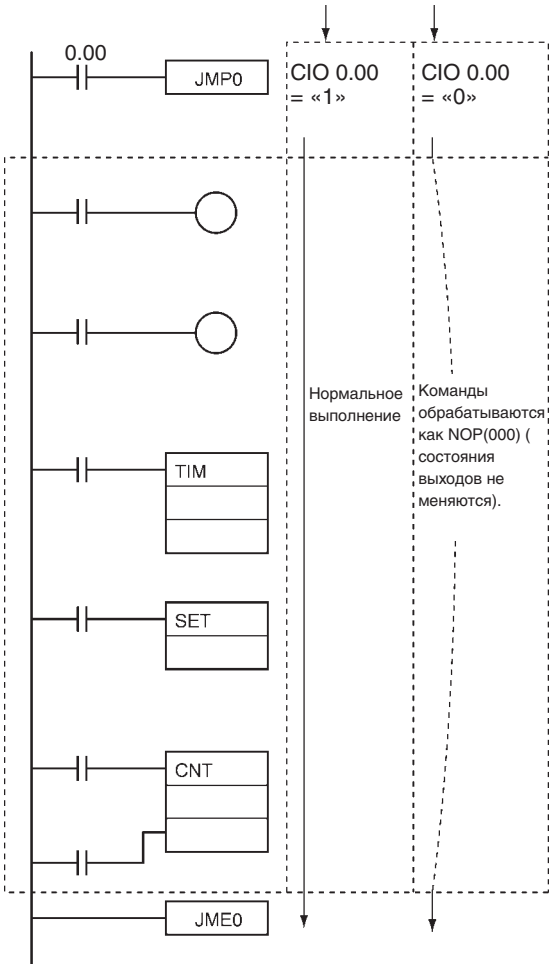
Команды JMP0(515) и JME0(516) не могут применяться в блочных программах.

Парные команды JMP0(515) и JME0(516) должны находиться в одной задаче, так как переходы между задачами не допускаются.

Работа команд DIFU(013), DIFD(014) и команд с различением фронтов, размещенных между командами JMP0(515) и JME0(516), зависит не только от состояния условия выполнения. Если команда DIFU(013), DIFD(014) или команда с различением фронтов в пропущенном сегменте программы выполняется сразу после включения условия выполнения JMP0(515), состояние условия выполнения команды DIFU(013), DIFD(014) или команды с различением фронтов сравнивается с состоянием условия выполнения до осуществления такого перехода (т. е. до переключения условия выполнения команды JMP0(515) в состояние ВЫКЛ).

Пример

Если в следующем примере бит CIO 0.00 выключен, расположенные между JMP0(515) и JME0(516) команды обрабатываются как команды NOP(000), состояние выходов остается неизменным.
Если бит CIO 0.00 включен, расположенные между JMP0(515) и JME0(516) команды выполняются обычным образом.

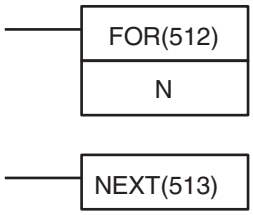


3-4-9 ЦИКЛЫ FOR-NEXT: FOR(512)/NEXT(513)

Назначение

Команды, расположенные между FOR(512) и NEXT(513), повторяются указанное количество раз. Команды FOR(512) и NEXT(513) используются в паре.

Символы РКС



N: Количество циклов

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	FOR(512)
	Выполнение в каждом цикле при включенном условии выполнения	NEXT(513)
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
Не допускается	OK	OK	OK

Операнды

N: Количество циклов

Количество циклов должно находиться в диапазоне 0000...FFFF (0...65 535 десятичн.).

Характеристики операндов

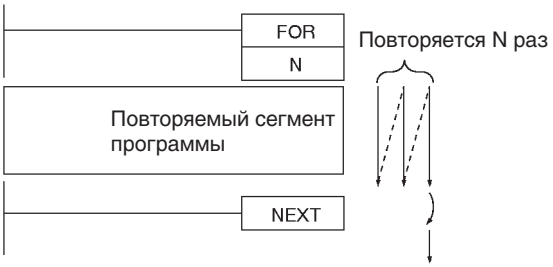
Область	N
Область CIO	CIO 0...CIO 6143
Рабочая область	W0...W511
Область битов хранения	H0...H511
Область вспомогательных битов	A0...A959
Область таймеров	T0000...T4095
Область счетчиков	C0000...C4095
Область DM	D0...D32767
Косвенные адреса DM в двоичном формате	@ D0...@ D32767
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767
Постоянные	#0000...#FFFF (двоичный) или &0...&65.535
Регистры данных	DR0...DR15
Регистры указателей	---
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15

Описание

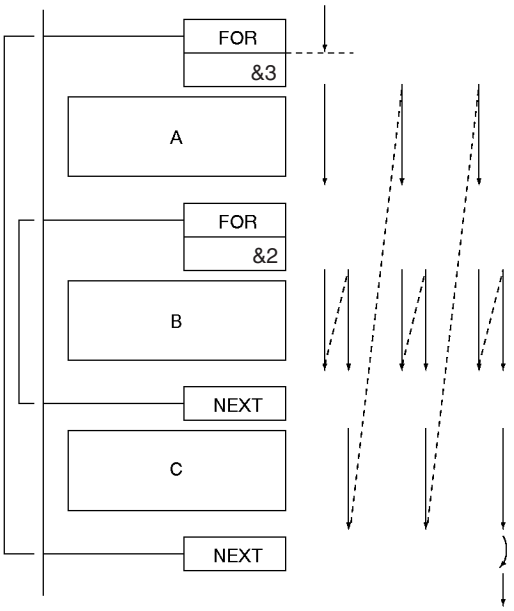
Команды, расположенные между FOR(512) и NEXT(513), выполняются N раз, затем выполнение программы продолжается с команды, следующей за командой NEXT(513). Выполнение цикла можно прервать командой BREAK(514).

Если N = 0, команды, расположенные между FOR(512) и NEXT(513), обрабатываются как команды NOP(000).

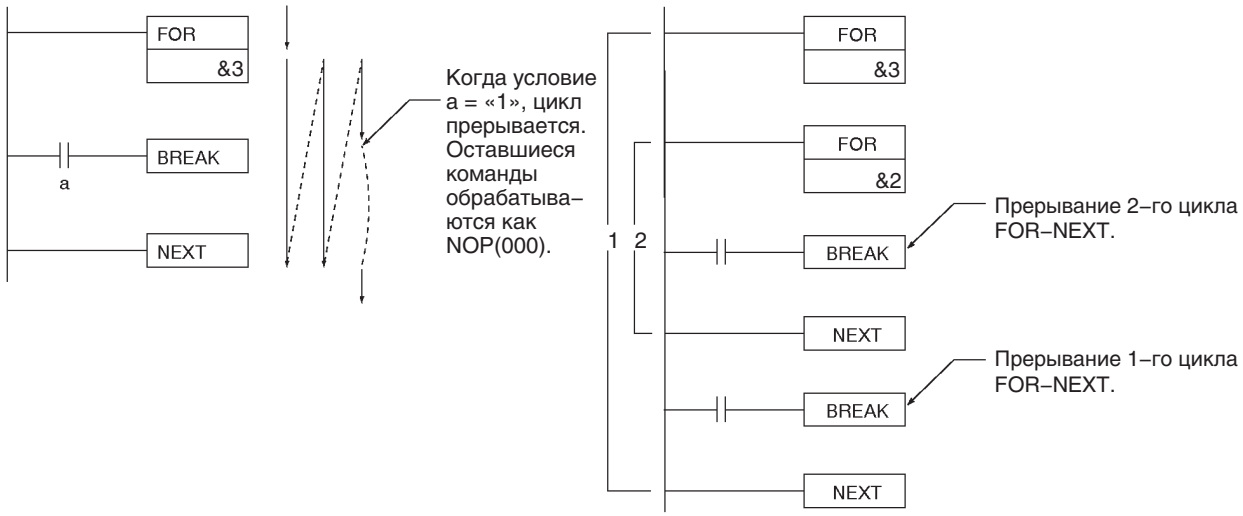
С помощью циклов можно быстро создавать программы для обработки массивов данных.



Допускается вложение циклов FOR-NEXT вплоть до 15 уровней. В приведенном ниже примере сегменты программы A, B и C выполняются следующим образом:
A → B → B → C, A → B → B → C и A → B → B → C



Для выхода из цикла FOR-NEXT используйте команду BREAK(514). Для выхода из вложенных циклов требуется несколько команд BREAK(514) (по числу уровней вложения). Все команды цикла после команды BREAK(514) обрабатываются как команды NOP(000).



Альтернативные методы создания циклов

Существует два способа зацикливания сегмента программы с выходом по заданному условию.

1,2,3...

1. Цикл FOR-NEXT с использованием команды BREAK

   Создайте цикл FOR-NEXT с максимальным количеством повторений N. Вставьте в цикл команду BREAK(514) с требуемым условием выполнения. При поступлении сигнала на вход этого условия цикл будет прерван, даже если еще не будет выполнено N повторений.

2. Цикл JME(005)-JMP(004)

   Создайте цикл, разместив команду JME(005) перед командой JMP(004). Команды, размещенные между JME(005) и JMP(004), будут выполняться циклически, пока будет оставаться выключенным условие выполнения команды JMP(004). (Однако если по истечении максимальной длительности цикла не включится условие выполнения или не будет выполнена команда END(001), будет сгенерирована ошибка превышения допустимой длительности цикла.)

Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если создано более 15 вложенных циклов. Выключен во всех остальных случаях.
Флаг равенства	=	ВЫКЛ
Флаг отрицательного значения	N	ВЫКЛ

Меры предосторожности

Команды FOR(512) и NEXT(513) следует использовать в пределах одной задачи. При расположении этих команд в разных задачах циклического выполнения не будет.

В цикле FOR-NEXT допускается выполнение команды перехода (такой как JMP(004)), но переход допускается только в пределах цикла FOR-NEXT.

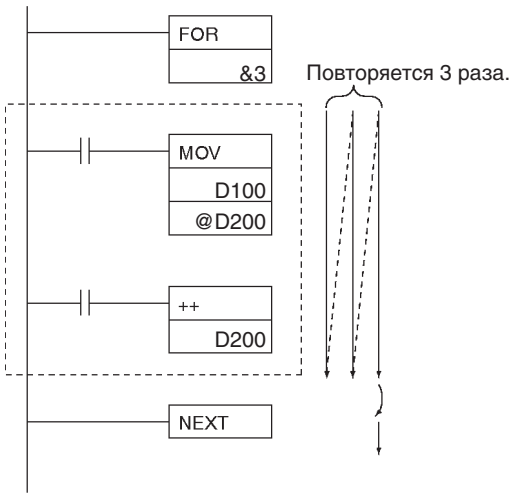
В циклах FOR-NEXT не допускается использование следующих команд:

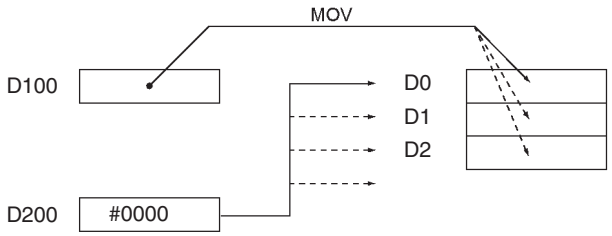
- команды программирования блоков;
- МНОЖЕСТВЕННЫЙ ПЕРЕХОД и КОНЕЦ ПЕРЕХОДА: JMP(515) и JME(516);
- ОПРЕДЕЛИТЬ ШАГ и ЗАПУСТИТЬ ШАГ: STEP(008)/SNXT(009).

Примечание. Если цикл FOR-NEXT, в котором используется бит различения фронтов, повторяется в одном цикле программы, этот бит будет все время включен или все время выключен в пределах этого цикла FOR-NEXT.

Пример

В следующем примере зацикленный сегмент программы записывает содержимое слова D100 по адресу, указанному в слове D200, постоянно увеличивая содержимое слова D200 на 1.





3-4-10 ВЫХОД ИЗ ЦИКЛА: BREAK(514)

Назначение

Используется в цикле FOR-NEXT для прекращения выполнения цикла в случае соблюдения условия выполнения. Все команды цикла после команды BREAK(514) обрабатываются как команды NOP(000).

Символ РКС



Варианты выполнения

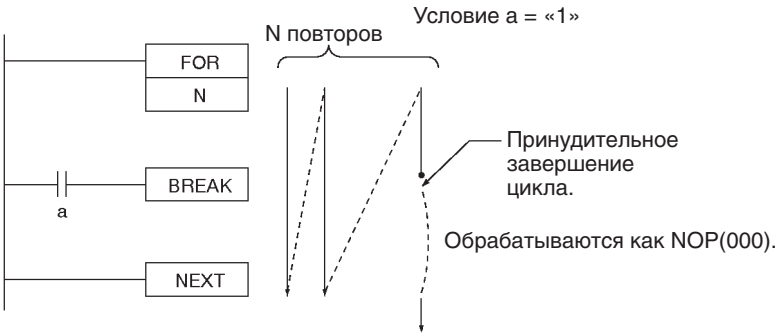
Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	BREAK(514)
	Однократное выполнение по положительному фронту	Не предусмотрено
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
Не допускается	OK	OK	OK

Описание

Для прекращения выполнения цикла FOR-NEXT вставьте команду BREAK(514) между командами FOR(512) и NEXT(513). Выполнение команды BREAK(514) будет прерывать выполнение цикла. При выполнении команды BREAK(514) все оставшиеся команды вплоть до NEXT(513) обрабатываются как команды NOP(000).



Флаги

Название	Обозна-чение	Описание работы
Флаг ошибки	ER	ВЫКЛ
Флаг равенства	=	ВЫКЛ
Флаг отрицательного значения	N	ВЫКЛ

Меры предосторожности

Команда BREAK(514) отменяет выполнение только одного цикла, поэтому для выхода из вложенных циклов требуется несколько команд BREAK(514) (по числу уровней вложения).

Команду BREAK(514) можно использовать только в циклах FOR-NEXT.

3-5 Команды управления таймерами и счетчиками

В данном разделе описаны команды, используемые для работы с таймерами и счетчиками.

Команда	Мнемоническое обозначение	Код функции	Стр.
ТАЙМЕР	TIM/TIMX	---/550	186
СКОРОСТНОЙ ТАЙМЕР	TIMH/TIMHX	015/551	191
1 мс ТАЙМЕР	TMHH/TIMHHX	540/552	195
НАКАПЛИВАЮЩИЙ ТАЙМЕР	TTIM/TTIMX	087/555	199
ДОЛГОСРОЧНЫЙ ТАЙМЕР	TIML/TIMLX	542/553	203
МНОГОВЫХОДНОЙ ТАЙМЕР	MTIM/MTIMX	543/554	206
СЧЕТЧИК	CNT/CNTX	---/546	212
РЕВЕРСИВНЫЙ СЧЕТЧИК	CNTR/CNTRX	012/548	215
СБРОС ТАЙМЕРА/СЧЕТЧИКА	CNR/CNRX	545/547	220

Формы представления текущего значения таймера/счетчика

■ Обзор

Для модулей ЦПУ серии CP можно указать один из двух форматов представления текущих значений в командах управления таймерами и счетчиками: двоично-десятичный (BCD) или двоичный.

Использование двоичного, а не двоично-десятичного формата позволяет расширить диапазон уставок таймеров и счетчиков с 0...9999 до 0...65535. Кроме того, в этом случае в качестве уставки таймера/счетчика можно использовать непосредственно двоичные значения, рассчитанные другими командами, без какого-либо дополнительного преобразования. Выбранный формат представления действует даже в случае косвенного указания уставки (посредством содержимого слова памяти). (Это значит, что содержимое адресованного слова воспринимается как двоично-десятичное или двоичное значение в зависимости от указанного формата представления.)

Подробную информацию о форматах представления смотрите в руководстве *Серия SYSMAC CP: Модули ЦПУ CP1H — Руководство по эксплуатации*.

■ Поддерживаемые команды

Категория	Команда	Мнемоническое обозначение	
		BCD	Двоичное
Команды управления таймерами/ счетчиками	ТАЙМЕР	TIM	TIMX(550)
	СКОРОСТНОЙ ТАЙМЕР	TIMH(015)	TIMHX(551)
	1 мс ТАЙМЕР	TMHH(540)	TMHHX(552)
	НАКАПЛИВАЮЩИЙ ТАЙМЕР	TTIM(087)	TTIMX(555)
	ДОЛГОСРОЧНЫЙ ТАЙМЕР	TIML(542)	TIMLX(553)
	МНОГОВЫХОДНОЙ ТАЙМЕР	MTIM(543)	MTIMX(554)
	СЧЕТЧИК	CNT	CNTX(546)
	РЕВЕРСИВНЫЙ СЧЕТЧИК	CNTR(012)	CNTRX(548)
	СБРОС ТАЙМЕРА/ СЧЕТЧИКА	CNR(545)	CNRX(547)
Команды программирования блоков	ТАЙМЕР ОЖИДАНИЯ	TIMW(813)	TIMWX(816)
	СКОРОСТНОЙ ТАЙМЕР ОЖИДАНИЯ	TMHW(815)	TMHWX(817)
	СЧЕТЧИК ОЖИДАНИЯ	CNTW(814)	CNTWX(818)

Предварительная версия
Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

Основные характеристики таймеров

В следующей таблице представлены основные характеристики таймеров.

Параметр		TIM/TIMX(550)	TIMH(015)/ TIMHX(551)	TMHH(540)/ TMHHX(552)	TTIM(087)/ TTIMX(555)	TIML(542)/ TIMLX(553)	MTIM(543)/ MTIMX(554)
Метод отсчета времени		Убывание	Убывание	Убывание	Приращение	Убывание	Приращение
Единица отсчета времени		0,1 с	0,01 с	0,001 с	0,1 с	0,1 с	0,1 с
Макс. значение уставки		TIM: 999,9 с TIMX: 6553,5 с	TIMH: 99,99 с TIMHX: 655,35 с	TMHH: 9,999 с TMHHX: 65,535 с	TTIM: 999,9 с TTIMX: 6553,5 с	TIML: 115 дней TIMLX: 49710 дней	MTIM: 999,9 с MTIMX: 6553,5 с
Выходы/ команда		1	1	1	1	1	8
Номера таймеров		Используются	Используются	Используются	Используются	Не используются	Не используются
Обновление флага завершения		При выполнении	При выполнении	По прерыванию каждую 1 мс	При выполнении	При выполнении	При выполнении
Обновление текущ. знач. таймера		См. примеч. 1	См. примеч. 2	Каждую 1 мс	При выполнении	При выполнении	При выполнении
Значение после сброса	Флаги заверш.	ВЫКЛ	ВЫКЛ	ВЫКЛ	ВЫКЛ	ВЫКЛ	ВЫКЛ
	Текущ. знач.	Уставка	Уставка	Уставка	0	Уставка	0

Примечание.

(1) Текущие значения таймеров, запущенных командой TIM, всегда обновляются при выполнении, а таймеры с номерами T0000...T0015 также обновляются каждые 100 мс.

(2) Текущие значения таймеров, запущенных командой TIMH(015)/ TIMHX(551), всегда обновляются при выполнении, а таймеры с номерами T0000...T0015 также обновляются каждые 10 мс.

Работа таймера

В следующей таблице показано влияние режимов и условий работы ПЛК и программы на работу таймеров.

Параметр	TIM/ TIMX(550)	TIMH(015)/ TIMHX(551)	TMHH(540)/ TMHHX(552)	TTIM(087)/ TTIMX(555)	TIML(542)/ TIMLX(553)	MTIM(543)/ MTIMX(554)
Изменение режима работы	Текущ. знач. = 0 Флаг завершения = ВЫКЛ				---	---
Прерывание питания/ сброс	Текущ. знач. = 0 Флаг завершения = ВЫКЛ				---	---
Выполнение CNR(545)/ CNRX(547)	Двоичный: текущ. знач. = FFFF, флаг завершения = ВЫКЛ BCD: текущ. знач. = FFFF или 9999, флаг завершения = ВЫКЛ				Не применимо	Не применимо
Работа в пропускаемом сегменте программы (JMP(004)–JME(005))	Работающие таймеры продолжают отсчет времени.			Состояние таймера сохраняется.		
Работа в блокированном сегменте программы (IL(002)–ILC(003))	Текущ. знач. = уставка Флаг завершения = ВЫКЛ			Состояние таймера сохраняется.	Текущ. знач. = уставка Флаг завершения = ВЫКЛ	Состояние таймера сохраняется.

Параметр		TIM/ TIMX(550)	TIMH(015)/ TIMHX(551)	TMHH(540)/ TMHHX(552)	TTIM(087)/ TTIMX(555)	TIML(542)/ TIMLX(553)	MTIM(543)/ MTIMX(554)
Принуд. установка	Флаги заверш.	ВКЛ				---	---
	Текущ. знач.	Сброс в 0				---	---
Принуд. сброс	Флаги заверш.	ВЫКЛ				---	---
	Текущ. знач.	Сброс к уставке			Сброс в 0	---	---

3-5-1 ТАЙМЕР: TIM/TIMX(550)

Назначение

Команда TIM или TIMX(550) запускает таймер обратного отсчета с шагом 0,1 с. Задание таймера (уставку) (SV) можно установить в пределах от 0 до 999,9 с для TIM и от 0 до 6553,5 с для TIMX(550). Погрешность работы таймера: 0...0,01 с.

Символ РКС

Форма предст. текущ. знач.	Символ	Операнды
BCD	TIM N S **N**: Номер таймера **S**: Уставка	N: 0000...4095 (десятичн.) S: #0000...#9999 (BCD)
Двоичное	TIMX(550) N S **N**: Номер таймера **S**: Уставка	N: 00000...4095 (десятичн.) S: &0...&65535 (десятичн.) #0000...#FFFF (hex)

Варианты выполнения

Варианты выполнения		TIM/TIMX(550)
	Выполнение в каждом цикле при включенном условии выполнения	TIM/TIMX(550)
	Однократное выполнение по положительному фронту	Не предусмотрено
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
Не допускается	OK	OK	Не допускается

Операнды

N: Номер таймера

Номер таймера должен быть указан в диапазоне от 0000 до 4095 (десятичн).

S: Уставка

Задание таймера должно лежать в диапазоне от #0000 до 9999 (BCD). (Если заданная уставка равна #0000, флаг завершения будет установлен сразу при выполнении команды TIM/TIMX(550).)

Характеристики операндов

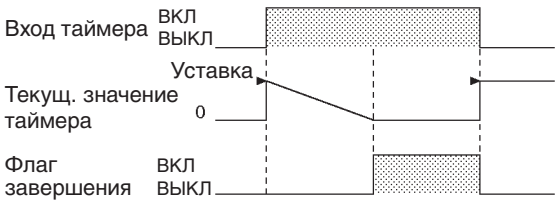
Область	N	S
Область CIO	---	CIO 0...CIO 6143
Рабочая область	---	W0...W511
Область битов хранения	---	H0...H511
Область вспомогательных битов	---	A0...A959
Область таймеров	0000...4095 (десятичн.)	T0000...T4095
Область счетчиков	---	C0000...C4095
Область DM	---	D0...D32767
Косвенные адреса DM в двоичном формате	---	@ D0...@ D32767
Косвенные адреса DM в двоично-десятичном формате (BCD)	---	*D0...*D32767
Постоянные	---	BCD: #0000...9999 (BCD) «&» использовать нельзя. Двоичный: &0...&65535 (десятичн.) #0000...#FFFF (hex)
Регистры данных	---	DR0...DR15
Регистры указателей	---	---
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15	

Описание

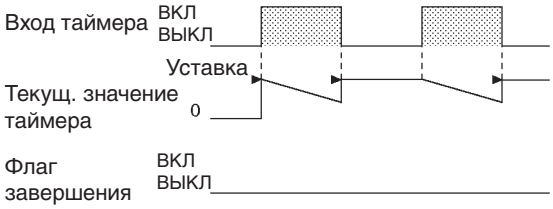
Когда вход команды таймера выключается, таймер с указанным номером N сбрасывается: его текущее значение становится равным уставке, сбрасывается флаг завершения работы.

При переключении входа таймера во включенное состояние команда TIM/TIMX(550) начинает уменьшать текущее значение. Обратный отсчет продолжается, пока вход команды таймера остается включенным. Когда текущее значение достигает 0000, устанавливается флаг завершения работы таймера.

По завершении работы таймера текущее значение и состояние флага завершения остаются неизменными. Для перезапуска таймера требуется выключить и вновь включить вход таймера либо изменить текущее значение таймера (ввести ненулевое значение командой MOV(021) или другой аналогичной командой).



Следующая временная диаграмма иллюстрирует поведение текущего значения и флага завершения таймера в случае, когда вход команды таймера выключается раньше окончания отсчета заданного времени.



Флаги

Название	Обозначение	Операция
Флаг ошибки	ER	Включен, если адрес N, косвенно указанный через регистр указателя, не является адресом текущего значения таймера. Включен, если выбран режим BCD, но значение S не представлено в формате BCD. Выключен во всех остальных случаях.
Флаг равенства	=	Выключен или не изменяется
Флаг отрицательного значения	N	Выключен или не изменяется

Меры предосторожности

Номера таймеров являются общими для команд TIM, TIMX(550), TIMH(015), TIMHX(551), TMHH(540), TMHHX(552), TTIM(087), TTIMX(555), TIMW(813), TIMWX(816), TMHW(815) и TMHWX(817). Если в программе используются две команды таймеров с одинаковым номером таймера, при проверке программы генерируется ошибка дублирования. При этом оба таймера будут работать нормально, если их работа будет происходить в разное время. Таймеры с одинаковым номером таймера будут работать неправильно, если они будут работать одновременно.

Таймеры с номерами от 16 до 4095 не могут работать надлежащим образом, если цикл ЦПУ длится дольше 100 мс. В этом случае следует использовать таймеры с номерами от 0 до 15.

Текущие значения таймеров с номерами от 0 до 15 продолжают обновляться, даже если таймеры находятся в режиме ожидания. Текущие значения таймеров с номерами от 16 до 4095 остаются неизменными, пока таймеры пребывают в режиме ожидания.

В указанных ниже ситуациях таймеры сбрасываются или приостанавливают отсчет времени. (При сбросе таймера его текущее значение возвращается к значению уставки, а флаг завершения сбрасывается.)

Условие	Текущее значение	Флаг завершения
Переключение из режима «Выполнение»/«Мониторинг» в режим «Программирование» или наоборот.[1]	0000	ВЫКЛ
Прерывание питания и сброс.[2]	0000	ВЫКЛ
Выполнение команды CNR(545) или CNRX(547) (СБРОС ТАЙМЕРА/ СЧЕТЧИКА).[3]	BCD: 9999 Двоичный: FFFF	ВЫКЛ
Работа в блокированном сегменте программы (IL(002)–ILC(003))	Сброс к уставке	ВЫКЛ
Работа в пропускаемом сегменте программы (JMP(004)–JME(005))	Текущее значение продолжает уменьшаться.	Сохраняется прежнее состояние.

Примечание.

(1) Если включен бит сохранения памяти ввода/вывода (A500.12), текущие значения таймеров и состояния их флагов завершения при смене режима работы ЦПУ не меняются.

(2) Если включен бит сохранения памяти ввода/вывода (A500.12) и в настройках ПЛК выбрано сохранение состояния самого этого бита, текущие значения таймеров и состояния их флагов завершения не меняются даже после выключения и последующего включения питания.

(3) При выполнении команды TIM/TIMX(550) текущее значение возвращается к значению уставки.

Если команда TIM/TIMX(550) используется в сегменте между командами IL(002) и ILC(003) и данный сегмент программы блокируется, текущее значение возвращается к значению уставки (SV), а флаг завершения сбрасывается.

Если работающий таймер с номером от 0 до 15, запущенный командой TIM/TIMX(550), расположен в пропускаемом сегменте программы (JMP(004), CJMP(510), CJPN(511), JME(005)), отсчет времени продолжается. Пропускаемая команда TIM/TIMX(550) не выполняется, но текущее значение обновляется в каждом цикле после выполнения всех задач.

В случае принудительной установки таймера TIM/TIMX(550) его флаг завершения будет установлен, а текущее значение будет сброшено в 0. В случае принудительного сброса таймера TIM/TIMX(550) его флаг завершения будет сброшен, а текущее значение будет возвращено к значению уставки.

Работа флагов «=» и «N» зависит от модели модуля ЦПУ. Подробнее см. в таблице *Флаги* выше.

Флаг завершения таймера обновляется только при выполнении команды TIM/TIMX(550), поэтому задержка установки флага завершения после завершения отсчета может составлять до одного цикла.

При использовании онлайн-редактирования для преобразования таймера одного типа в таймер другого типа с тем же номером таймера (например, TIM/TIMX(550) ↔ TIMH(015)/TIMHX(551) или TIM/TIMX(550) ↔ TMHH(540)/TMHHX(552)) обязательно сбросьте флаг завершения. Если флаг завершения не будет сброшен, таймер будет работать неправильно.

Текущее значение и флаг завершения таймера, запущенного командой TIM/TIMX(550), могут обновляться одним из следующих способов в зависимости от номера таймера.

Команды таймеров с номерами таймеров 0000...2047.

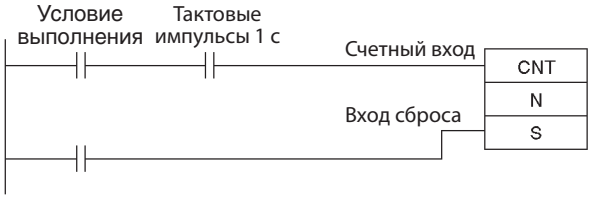
Выполнение TIM/ TIMX(550)	Текущее значение обновляется при каждом выполнении команды TIM/TIMX(550). Флаг завершения установлен, если текущ. знач. равно 0000. Флаг завершения сброшен, если текущ. знач. не равно 0000.
Обновление с интервалом 100 мс	Если длительность цикла превышает 100 мс, текущее значение таймера обновляется каждые 100 мс.

Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

Команды таймеров с номерами таймеров T0016...T4095

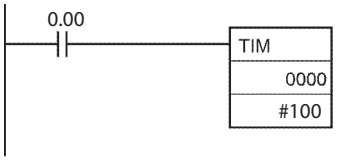
Выполнение TIM	Текущее значение обновляется при каждом выполнении команды TIM. Флаг завершения установлен, если текущ. знач. равно 0. Флаг завершения сброшен, если текущ. знач. не равно 0.

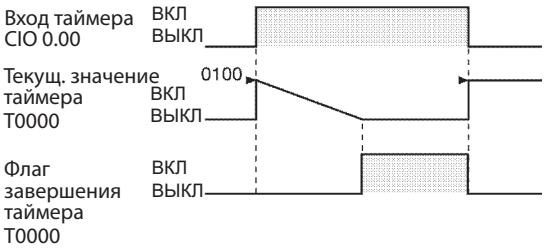
Сброс таймеров (текущее значение = уставка, флаг завершения сброшен) происходит при прерывании питания за исключением случая, когда установлен бит сохранения памяти ввода/вывода (A500.12) и в настройках ПЛК выбрано сохранение состояния этого бита. Для создания таймера, текущее значение которого сохраняется при прерывании питания, также можно использовать бит тактовой последовательности и команду счетчика, как показано на следующей схеме.



Пример

Если в следующем примере вход таймера CIO 0.00 переключается из «0» в «1», таймер начинает обратный отсчет от заданного значения. Флаг завершения таймера T0000 включится, когда текущее значение достигнет 0000.
При сбросе CIO 0.00 текущее значение таймера будет возвращено к значению уставки, флаг завершения будет сброшен.





3-5-2 СКОРОСТНОЙ ТАЙМЕР: TIMH(015)/TIMHX(551)

Назначение

Команда TIMH(015)/TIMHX(551) запускает таймер обратного отсчета с шагом 10 мс. Задание таймера (уставку) (SV) можно установить в пределах от 0 до 99,99 с для TIMH(015) и от 0 до 655,35 с для TIMHX(551). Погрешность работы таймера: 0...0,01 с.

Символ РКС

Форма предст. текущ. знач.	Символ	Операнды
BCD	TIMH(015) N S **N**: Номер таймера **S**: Уставка	N: 0000...4095 (десятичн.) S: #0000...#9999 (BCD)
Двоичное	TIMHX(551) N S **N**: Номер таймера **S**: Уставка	N: 00000...4095 (десятичн.) S: &0...&65535 (десятичн.) #0000...#FFFF (hex)

Варианты выполнения

Варианты выполнения		TIMH(015)/ TIMHX(551)
	Выполнение в каждом цикле при включенном условии выполнения	
	Однократное выполнение по положительному фронту	Не предусмотрено
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
Не допускается	OK	OK	Не допускается

Операнды

N: Номер таймера

Номер таймера должен быть указан в диапазоне от 0000 до 4095 (десятичн).

S: Уставка

Задание таймера должно лежать в диапазоне от #0000 до 9999 (BCD).

Характеристики операндов

Область	N	S
Область CIO	---	CIO 0...CIO 6143
Рабочая область	---	W0...W511
Область битов хранения	---	H0...H511
Область вспомогательных битов	---	A0...A959
Область таймеров	0000...4095 (десятичн.)	T0000...T4095
Область счетчиков	---	C0000...C4095

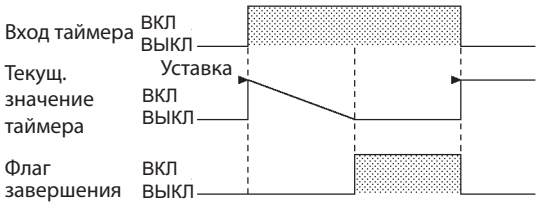
Область	N	S
Область DM	---	D0...D32767
Косвенные адреса DM в двоичном формате	---	@ D0...@ D32767
Косвенные адреса DM в двоично-десятичном формате (BCD)	---	*D0...*D32767
Постоянные	---	BCD: #0000...9999 (BCD) «&» использовать нельзя. Двоичный: &0...&65535 (десятичн.) #0000...#FFFF (hex)
Регистры данных	---	DR0...DR15
Регистры указателей	---	---
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15	

Описание

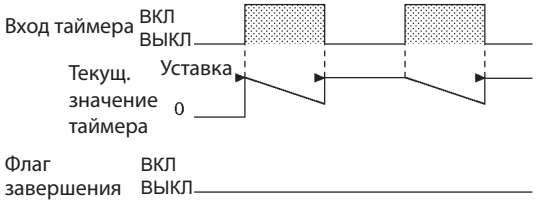
Когда вход команды таймера выключается, таймер с указанным номером N сбрасывается: его текущее значение становится равным уставке, сбрасывается флаг завершения работы.

При переключении входа таймера во включенное состояние команда TIMH(015)/TIMHX(551) начинает уменьшать текущее значение. Обратный отсчет продолжается, пока вход команды таймера остается включенным. Когда текущее значение достигает 0000, устанавливается флаг завершения работы таймера.

По завершении работы таймера текущее значение и состояние флага завершения остаются неизменными. Для перезапуска таймера требуется выключить и вновь включить вход таймера либо изменить текущее значение таймера (ввести ненулевое значение командой MOV(021) или другой аналогичной командой).



Следующая временная диаграмма иллюстрирует поведение текущего значения и флага завершения таймера в случае, когда вход команды таймера выключается раньше окончания отсчета заданного времени.



Флаги

Название	Обозначение	Операция
Флаг ошибки	ER	Включен, если адрес N, косвенно указанный через регистр указателя, не является адресом текущего значения таймера. Включен, если выбран режим BCD, но значение S не представлено в формате BCD. Выключен во всех остальных случаях.
Флаг равенства	=	Выключен или не изменяется
Флаг отрицательного значения	N	Выключен или не изменяется

Меры предосторожности

Номера таймеров являются общими для команд TIM, TIMX(550), TIMH(015), TIMHX(551), TMHH(540), TMHHX(552), TTIM(087), TTIMX(555), TIMW(813), TIMWX(816), TMHW(815) и TMHWX(817). Если в программе используются две команды таймеров с одинаковым номером таймера, при проверке программы генерируется ошибка дублирования. При этом оба таймера будут работать нормально, если их работа будет происходить в разное время. Таймеры с одинаковым номером таймера будут работать неправильно, если они будут работать одновременно.

Таймеры с номерами от 16 до 4095 не могут работать надлежащим образом, если цикл ЦПУ длится дольше 100 мс. В этом случае следует использовать таймеры с номерами от 0 до 15.

Таймеры TIMH(015)/TIMHX(551) с номерами от 0 до 15 обновляются каждые 10 мс. Используйте эти номера таймеров, если пользовательская программа обращается к текущему значению таймера.

Текущие значения таймеров с номерами от 0 до 15 продолжают обновляться, даже если таймеры находятся в режиме ожидания. Текущие значения таймеров с номерами от 16 до 4095 остаются неизменными, пока таймеры пребывают в режиме ожидания.

Работа флагов «=» и «N» зависит от модели модуля ЦПУ. Подробнее см. в таблице *Флаги* выше.

В указанных ниже ситуациях таймеры сбрасываются или приостанавливают отсчет времени. (При сбросе таймера его текущее значение возвращается к значению уставки, а флаг завершения сбрасывается.)

Условие	Текущее значение	Флаг завершения
Переключение из режима «Выполнение»/«Мониторинг» в режим «Программирование» или наоборот.[1]	0000	ВЫКЛ
Прерывание питания и сброс.[2]	0000	ВЫКЛ
Выполнение команды CNR(545) или CNRX(547) (СБРОС ТАЙМЕРА/ СЧЕТЧИКА).[3]	BCD: 9999 Двоичный: FFFF	ВЫКЛ
Работа в блокированном сегменте программы (IL(002)–ILC(003))	Сброс к уставке	ВЫКЛ
Работа в пропускаемом сегменте программы (JMP(004)–JME(005))	Текущее значение продолжает уменьшаться.	Сохраняется прежнее состояние.

Примечание.

(1) Если включен бит сохранения памяти ввода/вывода (A500.12), текущие значения таймеров и состояния их флагов завершения при смене режима работы ЦПУ не меняются.

(2) Если включен бит сохранения памяти ввода/вывода (A500.12) и в настройках ПЛК выбрано сохранение состояния самого этого бита, текущие значения таймеров и состояния их флагов завершения не меняются даже после выключения и последующего включения питания.

(3) При выполнении команды TIMH(015)/TIMHX(551) текущее значение возвращается к значению уставки.

Если работающий таймер с номером от 0 до 15, запущенный командой TIMH(015)/TIMHX(551), расположен в пропускаемом сегменте программы (JMP(004), CJMP(510), CJPN(511), JME(005)), отсчет времени продолжается. (Пропускаемая команда TIMH(015)/TIMHX(551) не выполняется, но текущее значение обновляется каждые 10 мс и в каждом цикле после выполнения всех задач.)

Если команда TIMH(015)/TIMHX(551) используется в сегменте между командами IL(002) и ILC(003) и данный сегмент программы блокируется, текущее значение возвращается к значению уставки (SV), флаг завершения сбрасывается.

При принудительной установке таймера, запущенного командой TIMH(015)/TIMHX(551), флаг завершения устанавливается, а текущее значение сбрасывается в 0. При принудительном сбросе таймера, запущенного командой TIMH(015)/TIMHX(551), флаг завершения сбрасывается, а текущее значение возвращается к значению уставки.

Работа флагов «=» и «N» зависит от модели модуля ЦПУ. Подробнее см. в таблице *Флаги*.

Флаг завершения таймера обновляется только при выполнении команды TIMH(015)/TIMHX(551), поэтому задержка установки флага завершения после завершения отсчета может составлять до одного цикла.

При использовании онлайн-редактирования для преобразования таймера одного типа в таймер другого типа с тем же номером таймера (например, TIMH(015)/TIMHX(551) ↔ TIM/TIMX(550) или TIMH(015)/ TIMHX(551) ↔ TMHH(540)/TMHHX(552)) обязательно сбросьте флаг завершения. Если флаг завершения не будет сброшен, таймер будет работать неправильно.

Текущее значение и флаг завершения таймера, запущенного командой TIMH(015)/TIMHX(551), могут обновляться одним из следующих способов в зависимости от номера таймера.

Команды таймеров с номерами таймеров T0000...T0015

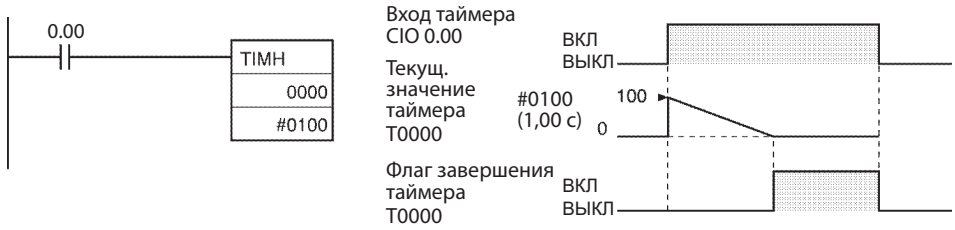
Выполнение TIMH(015)/ TIMHX(551)	Флаг завершения установлен, если текущ. знач. равно 0000. Флаг завершения сброшен, если текущ. знач. не равно 0000.
Обновление с интервалом 10 мс	Текущее значение таймера обновляется каждые 10 мс.

Команды таймеров с номерами таймеров T0016...T4095

Выполнение TIMH(015)/ TIMHX(551)	Текущее значение обновляется при каждом выполнении команды TIMH(015). Флаг завершения установлен, если текущ. знач. равно 0. Флаг завершения сброшен, если текущ. знач. не равно 0.

Пример

Если в следующем примере вход таймера CIO 0.00 переключается из «0» в «1», таймер начинает обратный отсчет от заданного значения (#0064 = 100 = 1,00 с). Флаг завершения таймера T0000 включится, когда текущее значение достигнет 0000.
При выключении CIO 0.00 текущее значение таймера будет возвращено к значению уставки, флаг завершения будет сброшен.



3-5-3 1 мс ТАЙМЕР: TMHH(540)/TMHHX(552)

Назначение

Команда TMHH(540)/TMHHX(552) запускает таймер обратного отсчета с шагом 1 мс. Задание таймера (уставку) (SV) можно установить в пределах от 0 до 9,999 с для TMHH(540) и от 0 до 65,535 с для TMHHX(552). Погрешность работы таймера: -0,001...0 с.

Символ РКС

Форма предст. текущ. знач.	Символ	Операнды
BCD	TMHH(540) N S **N**: Номер таймера **S**: Уставка	N: 0000...15 (десятичн.) S: #0000...#9999 (BCD)
Двоичное	TMHHX(552) N S **N**: Номер таймера **S**: Уставка	N: 00000...15 (десятичн.) S: &0...&65535 (десятичн.) #0000...#FFFF (hex)

Варианты выполнения

Варианты выполнения		TMHH(540)/ TMHHX(552)
	Выполнение в каждом цикле при включенном условии выполнения	
	Однократное выполнение по положительному фронту	Не предусмотрено
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
Не допускается	OK	OK	Не допускается

Операнды

N: Номер таймера

Номер таймера должен быть указан в диапазоне от 0000 до 0015 (десятичн).

S: Уставка

Задание таймера должно лежать в диапазоне от #0000 до 9999 (BCD).

Характеристики операндов

Область	N	S
Область CIO	---	CIO 0...CIO 6143
Рабочая область	---	W0...W511
Область битов хранения	---	H0...H511
Область вспомогательных битов	---	A0...A959
Область таймеров	0000...0015 (десятичн.)	T0000...T4095
Область счетчиков	---	C0000...C4095
Область DM	---	D0...D32767
Косвенные адреса DM в двоичном формате	---	@ D0...@ D32767
Косвенные адреса DM в двоично-десятичном формате (BCD)	---	*D0...*D32767
Постоянные	---	BCD: #0000...9999 (BCD) «&» использовать нельзя. Двоичный: &0...&65535 (десятичн.) #0000...#FFFF (hex)
Регистры данных	---	DR0...DR15
Регистры указателей	---	---
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15	

Описание

Когда вход команды таймера выключается, таймер с указанным номером N сбрасывается: его текущее значение становится равным уставке, сбрасывается флаг завершения работы.

При переключении входа таймера во включенное состояние команда TMHH(540)/TMHHX(552) начинает уменьшать текущее значение. Обратный отсчет продолжается, пока вход команды таймера остается включенным. Когда текущее значение достигает 0000, устанавливается флаг завершения работы таймера.

По завершении работы таймера текущее значение и состояние флага завершения остаются неизменными. Для перезапуска таймера требуется выключить и вновь включить вход таймера либо изменить текущее значение таймера (ввести ненулевое значение командой MOV(021) или другой аналогичной командой).

Флаги

Название	Обозначение	Операция
Флаг ошибки	ER	Включен, если адрес N, косвенно указанный через регистр указателя, не является адресом текущего значения таймера. Включен, если выбран режим BCD, но значение S не представлено в формате BCD. Выключен во всех остальных случаях.
Флаг равенства	=	Выключен или не изменяется
Флаг отрицательного значения	N	Выключен или не изменяется

Меры предосторожности

Номера таймеров являются общими для команд TIM, TIMX(550), TIMH(015), TIMHX(551), TMHH(540), TMHHX(552), TTIM(087), TTIMX(555), TIMW(813), TIMWX(816), TMHW(815) и TMHWX(817). Если в программе используются две команды таймеров с одинаковым номером таймера, при проверке программы генерируется ошибка дублирования. При этом оба таймера будут работать нормально, если их работа будет происходить в разное время. Таймеры с одинаковым номером таймера будут работать неправильно, если они будут работать одновременно.

Флаг завершения обновляется только при выполнении команды TMHH(540)/TMHHX(552). Поэтому задержка установки флага завершения после достижения заданного значения может составлять до одного цикла.

Текущие значения таймеров с номерами от 0000 до 2047 продолжают обновляться, даже если таймеры находятся в режиме ожидания. Текущие значения таймеров с номерами от 2048 до 4095 остаются неизменными, пока таймеры пребывают в режиме ожидания.

В указанных ниже ситуациях таймеры сбрасываются или приостанавливают отсчет времени. (При сбросе таймера его текущее значение возвращается к значению уставки, а флаг завершения сбрасывается.)

Условие	Текущее значение	Флаг завершения
Переключение из режима «Выполнение»/«Мониторинг» в режим «Программирование» или наоборот.[1]	0000	ВЫКЛ
Прерывание питания и сброс.[2]	0000	ВЫКЛ
Выполнение команды CNR(545) или CNRX(547) (СБРОС ТАЙМЕРА/ СЧЕТЧИКА).[3]	BCD: 9999 Двоичный: FFFF	ВЫКЛ
Работа в блокированном сегменте программы (IL(002)–ILC(003))	Сброс к уставке	ВЫКЛ
Работа в пропускаемом сегменте программы (JMP(004)–JME(005))	Текущее значение продолжает уменьшаться.	Сохраняется прежнее состояние.

Примечание.

(1) Если включен бит сохранения памяти ввода/вывода (A500.12), текущие значения таймеров и состояния их флагов завершения при смене режима работы ЦПУ не меняются.

(2) Если включен бит сохранения памяти ввода/вывода (A500.12) и в настройках ПЛК выбрано сохранение состояния самого этого бита, текущие значения таймеров и состояния их флагов завершения не

меняются даже после выключения и последующего включения питания.

(3) При выполнении команды TMHH(540)/TMHHX(552) текущее значение возвращается к значению уставки.

Если работающий таймер, запущенный командой TMHH(540)/ TMHHX(552), расположен в пропускаемом сегменте программы (JMP(004), CJMP(510), CJPN(511), JME(005)), отсчет времени продолжается. (Пропускаемая команда TMHH(540)/TMHHX(552) не выполняется, но текущее значение обновляется каждую 1 мс.)

Если команда TMHH(540)/TMHHX(552) используется в сегменте между командами IL(002) и ILC(003) и данный сегмент программы блокируется, текущее значение возвращается к значению уставки (SV), флаг завершения сбрасывается.

В случае принудительной установки таймера, запущенного командой TMHH(540)/TMHHX(552), его флаг завершения будет установлен, а текущее значение будет сброшено в 0. В случае принудительного сброса таймера, запущенного командой TMHH(540)/TMHHX(552), его флаг завершения будет сброшен, а текущее значение будет возвращено к значению уставки.

Работа флагов «=» и «N» зависит от модели модуля ЦПУ. Подробнее см. в таблице *Флаги* выше.

При использовании онлайн-редактирования для преобразования таймера одного типа в таймер другого типа с тем же номером таймера (например, TMHH(540)/TMHHX(552) ↔ TIM/TIMX(550) или TMHH(540)/ TMHHX(552) ↔ TIMH(015)/TIMHX(551)) обязательно сбросьте флаг завершения. Если флаг завершения не будет сброшен, таймер будет работать неправильно.

Текущее значение и флаг завершения таймера, запущенного командой TMHH(540)/TMHHX(552), обновляются так, как показано в следующей таблице.

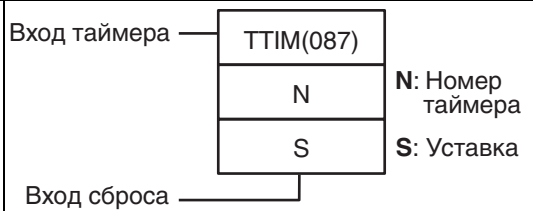
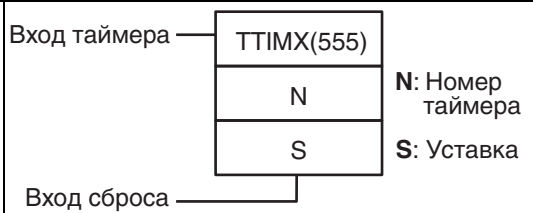
Выполнение команды TMHH(540)/ TMHHX(552)	Флаг завершения установлен, если текущ. знач. равно 0000. Флаг завершения сброшен, если текущ. знач. не равно 0000.
Обновление с интервалом 1 мс	Текущее значение таймера обновляется каждые 1 мс.

3-5-4 НАКАПЛИВАЮЩИЙ ТАЙМЕР: TTIM(087)/TTIMX(555)

Назначение

Команда TTIM(087)/TTIMX(555) запускает таймер прямого отсчета с шагом 0,1 с. Задание таймера (уставку) (SV) можно установить в пределах от 0 до 999,9 с для TTIM(087) и от 0 до 6553,5 с для TTIMX(555). Погрешность работы таймера: -0,01...0 с.

Символ РКС

Форма предст. текущ. знач.	Символ	Операнды
BCD		N: 0000...15 (десятичн.) S: #0000...#9999 (BCD)
Двоичное		N: 00000...15 (десятичн.) S: &0...&65535 (десятичн.) #0000...#FFFF (hex)

Варианты выполнения

Варианты выполнения		TTIM(087)/ TTIMX(555)
	Выполнение в каждом цикле при включенном условии выполнения	
	Однократное выполнение по положительному фронту	Не предусмотрено
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
Не допускается	OK	OK	Не допускается

Операнды

N: Номер таймера

Номер таймера должен быть указан в диапазоне от 0000 до 4095 (десятичн).

S: Уставка

Задание таймера должно лежать в диапазоне от #0000 до 9999 (BCD).

Характеристики операндов

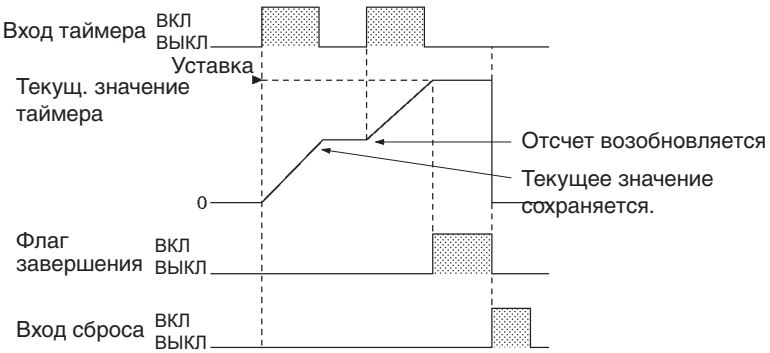
Область	N	S
Область CIO	---	CIO 0...CIO 6143
Рабочая область	---	W0...W511
Область битов хранения	---	H0...H511
Область вспомогательных битов	---	A0...A959

Область	N	S
Область таймеров	0000...4095 (десятичн.)	T0000...T4095
Область счетчиков	---	C0000...C4095
Область DM	---	D0...D32767
Косвенные адреса DM в двоичном формате	---	@ D0...@ D32767
Косвенные адреса DM в двоично-десятичном формате (BCD)	---	*D0...*D32767
Постоянные	---	BCD: #0000...9999 (BCD) «&» использовать нельзя. Двоичный: &0...&65535 (десятичн.) #0000...#FFFF (hex)
Регистры данных	---	DR0...DR15
Регистры указателей	---	---
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15	

Описание

Если вход таймера включен, команда TTIM(087)/TTIMX(555) увеличивает текущее значение. При выключении входа таймера отсчет прекращается, но текущее значение не сбрасывается. После того как вход таймера снова переходит во включенное состояние, отсчет возобновляется. Флаг завершения таймера включается, когда текущее значение достигает заданного значения.

По завершении работы таймера текущее значение и состояние флага завершения остаются неизменными. Предусмотрено три способа перезапуска таймера: путем изменения текущего значения таймера (ввода ненулевого значения командой MOV(021) или другой аналогичной командой), путем включения входа сброса или путем выполнения команды CNR(545)/CNRX(547).



Флаги

Название	Обозначение	Операция
Флаг ошибки	ER	Включен, если адрес N, косвенно указанный через регистр указателя, не является адресом текущего значения таймера. Включен, если выбран режим BCD, но значение S не представлено в формате BCD. Выключен во всех остальных случаях.

Меры предосторожности

Номера таймеров являются общими для команд TIM, TIMX(550), TIMH(015), TIMHX(551), TMHH(540), TMHHX(552), TTIM(087), TTIMX(555), TIMW(813), TIMWX(816), TMHW(815) и TMHWX(817). Если в программе используются две команды таймеров с одинаковым номером таймера, при проверке программы генерируется ошибка дублирования. При этом оба таймера будут работать нормально, если их работа будет происходить в разное время. Таймеры с одинаковым номером таймера будут работать неправильно, если они будут работать одновременно.

В указанных ниже ситуациях таймеры сбрасываются или приостанавливают отсчет времени. (При сбросе таймера, запущенного командой TTIM(087)/TTIMX(555), его текущее значение сбрасывается в 0000, флаг завершения сбрасывается.)

Условие	Текущее значение	Флаг завершения
Переключение из режима «Выполнение»/«Мониторинг» в режим «Программирование» или наоборот.[1]	0000	ВЫКЛ
Прерывание питания и сброс.[2]	0000	ВЫКЛ
Выполнение команды CNR(545) или CNRX(547) (СБРОС ТАЙМЕРА/ СЧЕТЧИКА).[3]	BCD: 9999 Двоичный: FFFF	ВЫКЛ
Работа в блокированном сегменте программы (IL(002)–ILC(003))	Сохраняется прежнее состояние.	Сохраняется прежнее состояние.
Работа в пропускаемом сегменте программы (JMP(004)–JME(005))	Сохраняется прежнее состояние.	Сохраняется прежнее состояние.

Примечание.

(1) Если включен бит сохранения памяти ввода/вывода (A500.12), текущие значения таймеров и состояния их флагов завершения при смене режима работы ЦПУ не меняются.

(2) Если включен бит сохранения памяти ввода/вывода (A500.12) и в настройках ПЛК выбрано сохранение состояния самого этого бита, текущие значения таймеров и состояния их флагов завершения не меняются даже после выключения и последующего включения питания.

(3) При выполнении команды TTIM(087)/TTIMX(555) текущее значение возвращается к значению уставки.

Если команда TTIM(087)/TTIMX(555) используется в сегменте между командами IL(002) и ILC(003) и данный сегмент программы блокируется, сохраняется предыдущее текущее значение (значение не сбрасывается). Обязательно принимайте во внимание этот факт при размещении команды TTIM(087)/TTIMX(555) между командами IL(002) и ILC(003).

Если работающий таймер, запущенный командой TTIM(087)/TTIMX(555), расположен в сегменте программы между командами JMP(004) и JME(005) и этот сегмент программы пропускается, сохраняется предыдущее текущее значение. Обязательно принимайте во внимание этот факт при размещении команды TTIM(087)/TTIMX(555) между командами JMP(004) и JME(005).

Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

При принудительной установке таймера, запущенного командой TTIM(087)/TTIMX(555), текущее значение сбрасывается в 0000 и устанавливается флаг завершения. При принудительном сбросе таймера, запущенного командой TTIM(087)/TTIMX(555), текущее значение сбрасывается в 0000 и сбрасывается флаг завершения. Операции принудительной установки и принудительного сброса обладают приоритетом по отношению к состояниям входа таймера и входа сброса.

Текущее значение таймера обновляется только при выполнении команды TTIM(087)/TTIMX(555), поэтому таймер будет работать неправильно, если время выполнения цикла превысит 100 мс (так как шаг отсчета времени данного таймера составляет 100 мс).

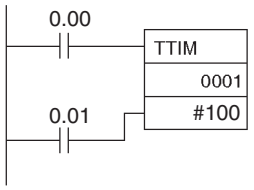
Флаг завершения таймера обновляется только при выполнении команды TTIM(087)/TTIMX(555), поэтому задержка установки флага завершения после завершения отсчета может составлять до одного цикла.

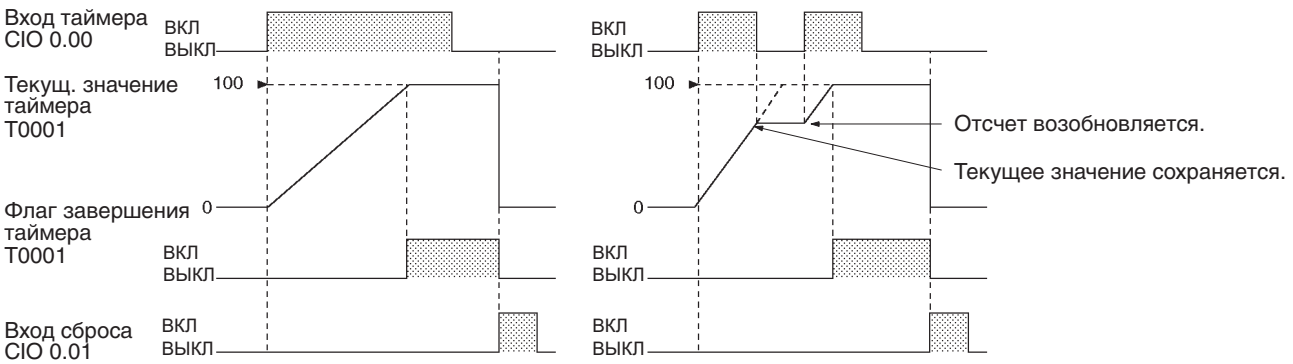
Обычные таймеры, такие как TIM/TIMX(550), производят обратный отсчет, поэтому текущее значение указывает время, остающееся до завершения отсчета. Текущее значение таймера, запущенного командой TTIM(087)/TTIMX(555), указывает истекшее время, поэтому его можно использовать для вычислений и отображения на экране без дополнительных преобразований.

Пример

Если в следующем примере вход CIO 0.00 переходит из «0» в «1», таймер начинает отсчет времени с 0. Флаг завершения таймера T0001 будет установлен, когда текущее значение достигнет уставки.
Если включается вход сброса, текущее значение таймера сбрасывается в 0000, флаг завершения (T0001) также сбрасывается. (Обычно таймер сначала сбрасывают путем включения входа сброса, а затем запускают отсчет включением входа таймера.)

Если вход таймера выключается до того, как достигнуто заданное значение, отсчет прекращается, но текущее значение сохраняется. При повторном включении входа таймера отсчет продолжается с предыдущего текущего значения.





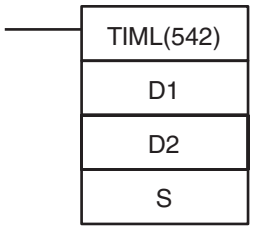
3-5-5 ДОЛГОСРОЧНЫЙ ТАЙМЕР: TIML(542)/TIMLX(553)

Назначение

Команда TIML(542)/TIMLX(553) запускает таймер обратного отсчета с шагом 0,1 с, способный отсчитывать временной интервал до 115 дней (TIML(542)) и до 4971 дней (TIMLX(543)). Погрешность работы таймера: 0...-0,01 с.

Символ РКС

BCD

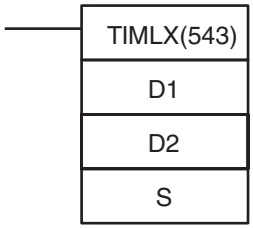


D1: Флаг завершения

D2: Слово текущ. значения (PV)

S: Слово уставки (SV)

Двоичное



D1: Флаг завершения

D2: Слово текущ. значения (PV)

S: Слово уставки (SV)

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	TIML(542)/ TIMLX(553)
	Однократное выполнение по положительному фронту	Не предусмотрено
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

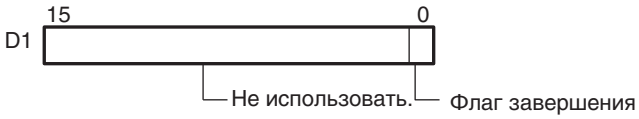
Доступные области программы

Области программных блоков	**Области пошаговых программ**	**Подпрограммы**	**Задачи обработки прерываний**
Не допускается	OK	OK	Не допускается

Операнды

D1: Флаг завершения

Бит 0 слова D1 служит флагом завершения таймера TIML(542)/ TIMLX(553).



D2: Слово текущего значения

Слова D2+1 и D2 содержат 8-разрядное двоичное или двоично-десятичное текущее значение. (Слова D2 и D2+1 должны принадлежать одной области.) Текущее значение может изменяться в пределах от #00000000 до #99999999 для TIML(542) и от &00000000 до &4294967295 (десятичн.) либо от #00000000 до #FFFFFFFF (hex) для TIMLX(553).



S: Слово уставки

Слова S+1 и S содержат 8-разрядное двоичное или двоично-десятичное заданное значение. (Слова S и S+1 должны принадлежать одной области.) Заданное значение должно лежать в диапазоне от #00000000 до #99999999 для TIML(542) и от &00000000 до &4294967294 (десятичн.) либо от #00000000 до #FFFFFFFF (hex) для TIMLX(553).



Характеристики операндов

Область	D1	D2	S
Область CIO	CIO 0...CIO 6143	CIO 0...CIO 6142	
Рабочая область	W0...W511	W0...W510	
Область битов хранения	H0...H511	H0...H510	
Область вспомогательных битов	A448...A959	A448...A958	A0...A958
Область таймеров	---	---	T0000...T4094
Область счетчиков	---	---	C0000...C4094
Область DM	D0...D32767	D0...D32766	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767		
Постоянные	---		BCD: #00000000...99999999 (BCD) «&» использовать нельзя. Двоичный: &00000000...&4294967295 (десятичн.) #00000000...#FFFFFFFF (hex)
Регистры данных	---		
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15		

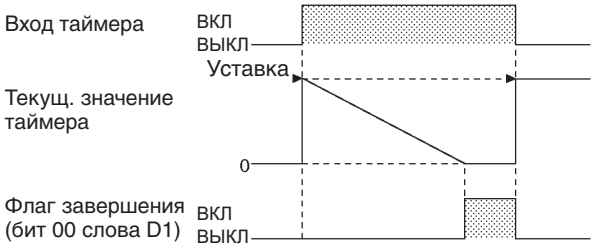
Описание

Команда TIML(542)/TIMLX(553) запускает таймер задержки включения обратного отсчета с шагом 0,1 с, использующий 8-разрядную уставку и 8-разрядное текущее значение.

Когда вход команды таймера выключается, таймер сбрасывается: его текущее значение возвращается к значению уставки, флаг завершения сбрасывается.

При переключении входа таймера во включенное состояние команда TIML(542)/TIMLX(553) начинает уменьшать текущее значение в словах D2+1 и D2. Обратный отсчет продолжается, пока вход команды таймера остается включенным. Когда текущее значение достигает 0000 0000, устанавливается флаг завершения работы таймера.

По завершении работы таймера текущее значение и состояние флага завершения остаются неизменными. Для перезапуска таймера требуется выключить и вновь включить вход таймера либо изменить текущее значение таймера (ввести ненулевое значение командой MOV(021) или другой аналогичной командой).



Флаги

Название	Обозначение	Операция
Флаг ошибки	ER	Включен, если указан формат BCD, а текущее значение в словах D2+1 и D2 представлено в другом формате. Включен, если заданное значение в словах S+1 и S не представлено в формате BCD. Выключен во всех остальных случаях.

Меры предосторожности

В отличие от большинства таймеров, команда TIML(542)/TIMLX(553) не использует номер таймера. (Обновление текущих значений в области таймеров для TIML(542)/TIMLX(553) не выполняется.)

Так как флаг завершения TIML(542)/TIMLX(553) находится в области данных, его можно принудительно установить или принудительно сбросить, как любые другие биты, но текущее значение при этом не изменится.

Текущее значение таймера обновляется только при выполнении команды TIML(542)/TIMLX(553), поэтому таймер будет работать неправильно, если время выполнения цикла превысит 100 мс (так как шаг отсчета времени данного таймера составляет 100 мс).

Флаг завершения таймера обновляется только при выполнении команды TIML(542)/TIMLX(553), поэтому задержка установки флага завершения после завершения отсчета может составлять до одного цикла.

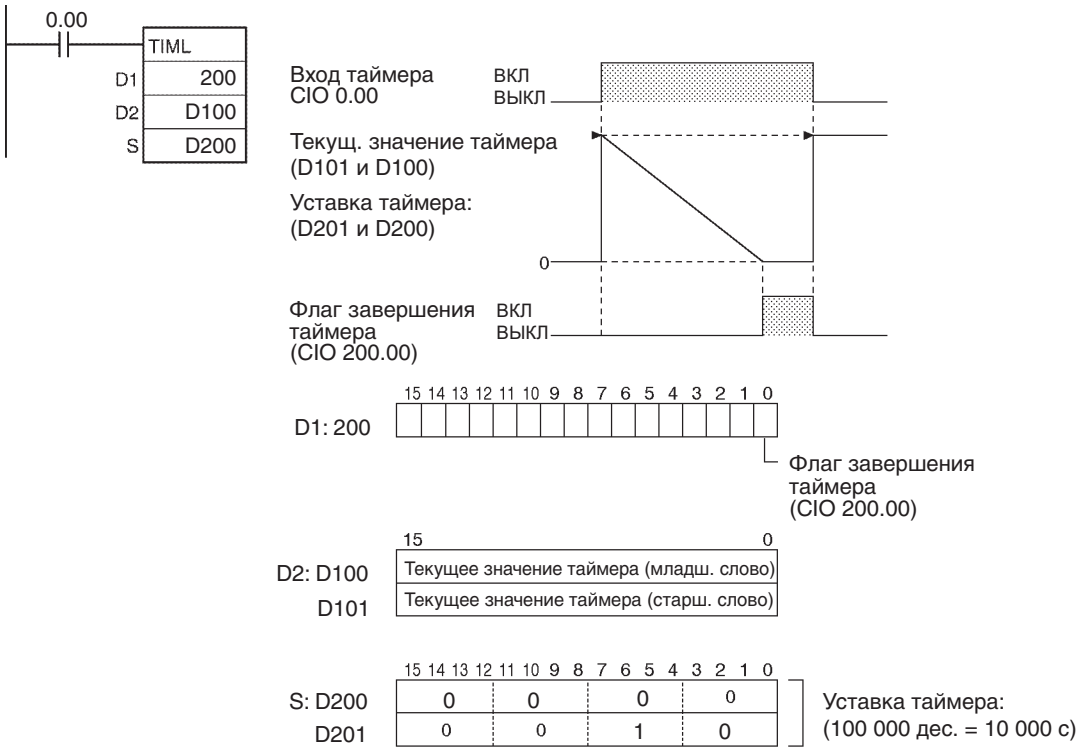
Если команда TIML(542)/TIMLX(553) используется в сегменте между командами IL(002) и ILC(003) и данный сегмент программы блокируется, текущее значение возвращается к значению уставки (SV), флаг завершения сбрасывается.

Если работающий таймер TIML(542)/TIMLX(553) расположен в сегменте программы между командами JMP(004) и JME(005) и этот сегмент программы пропускается, сохраняется предыдущее текущее значение. Обязательно принимайте во внимание этот факт при размещении команды TIML(542)/TIMLX(553) между командами JMP(004) и JME(005).

Позаботьтесь, чтобы слова, указанные для флага завершения и текущего значения (D1, D2 и D2+1), не использовались другими командами. Если другие команды будут оказывать влияние на содержимое этих слов, таймер может работать неправильно.

Пример

Если в следующем примере вход таймера CIO 0.00 перейдет из «0» в «1», текущее значение таймера (в словах D101 и D100) будет возвращено к значению уставки (в словах D201 и D200) и начнется обратный отсчет времени. Флаг завершения таймера (CIO 200.00) включится, когда текущее значение достигнет 0000 0000.
При выключении CIO 0.00 текущее значение таймера будет возвращено к значению уставки, флаг завершения будет сброшен.



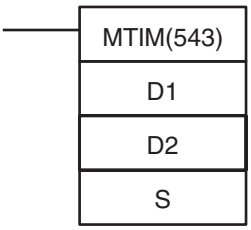
3-5-6 МНОГОВЫХОДНОЙ ТАЙМЕР: MTIM(543)/MTIMX(554)

Назначение

Команда MTIM(543)/MTIMX(554) запускает таймер прямого отсчета с шагом 0,1 с, имеющий восемь независимых уставок и флагов завершения. Задание таймера должно быть указано в диапазоне от 0 до 999,9 с (для MTIM(543)) и от 0 до 6553,5 с для MTIMX(554), погрешность отсчета времени составляет 0...0,01 с.

Символ РКС

BCD

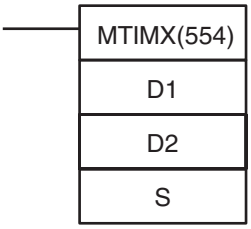


D1: Флаги завершения

D2: Слово текущ. значения (PV)

S: Первое слово уставки (SV)

Двоичное



D1: Флаги завершения

D2: Слово текущ. значения (PV)

S: Первое слово уставки (SV)

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	MTIM(543)/ MTIMX(554)
	Однократное выполнение по положительному фронту	Не предусмотрено
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

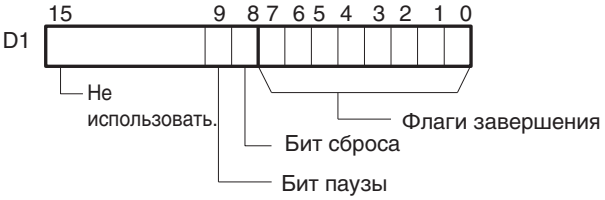
Доступные области программы

Области программных блоков	**Области пошаговых программ**	**Подпрограммы**	**Задачи обработки прерываний**
Не допускается	OK	OK	Не допускается

Операнды

D1: Флаги завершения

Слово D1 содержит восемь флагов завершения, а также биты паузы и сброса.



D2: Слово текущего значения

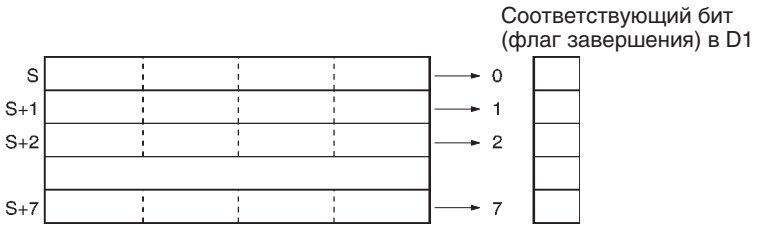
Слово D2 содержит 4-разрядное двоичное или двоично-десятичное текущее значение.

Данные	**Диапазон**
BCD	#0000...#9999
Двоичное	&0...&65535 (десятичн.) #0000...#FFFF (hex)

S: Первое слово уставки

Слова S...S+7 содержат восемь независимых уставок.
Каждая уставка должна соответствовать следующим условиям:

Данные	**Диапазон**
BCD	#0000...#9999
Двоичное	&0...&65535 (десятичн.) #0000...#FFFF (hex)



Данные	Диапазон
BCD	Одно слово для каждой из восьми уставок: #0000...#9999
Двоичное	Одно слово для каждой из восьми уставок: &0...&65535 (десятичн.) #0000...#FFFF (hex)

Примечание. Слова S, S+1,...S+7 должны принадлежать одной области.

Характеристики операндов

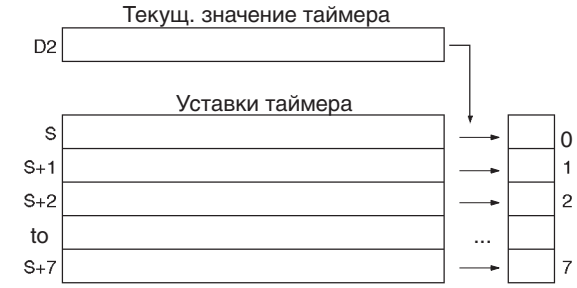
Область	D1	D2	S
Область CIO	CIO 0...CIO 6143		CIO 0...CIO 6136
Рабочая область	W0...W511		W0...W504
Область битов хранения	H0...H511		H0...H504
Область вспомогательных битов	A448...A959		A0...A952
Область таймеров	T0000...T4095		T0000...T4088
Область счетчиков	C0000...C4095		C0000...C4088
Область DM	D0...D32767		D0...D32760
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767		
Постоянные	---		
Регистры данных	---	DR0...DR15	---
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

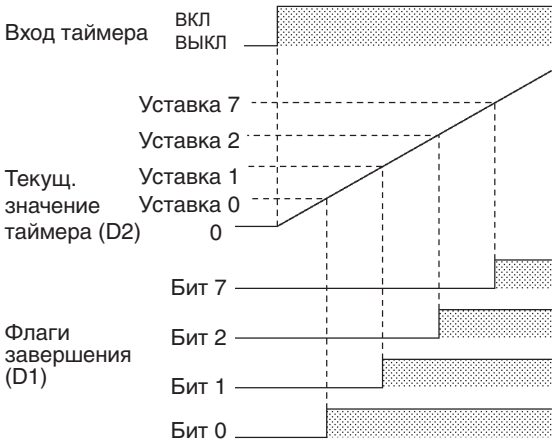
Описание

Если условие выполнения команды MTIM(543)/MTIMX(554) включено, а биты сброса и таймера оба выключены, команда MTIM(543)/MTIMX(554) производит отсчет времени, наращивая текущее значение в слове D2. При включении бита паузы таймер прекращает отсчет без сброса текущего значения. Команда MTIM(543)/MTIMX(554) возобновляет отсчет после выключения бита паузы.

Текущее значение (содержимое слова D2) сравнивается с восемью заданными значениями в словах S...S+7 при каждом выполнении команды MTIM(543)/MTIMX(554), и если какое-либо заданное значение меньше или равно текущему значению, включается соответствующий флаг завершения (слово D1, биты 00...07).

Достигнув 9999, текущее значение вновь становится равным 0000, все флаги завершения при этом сбрасываются. Если во время работы или паузы в работе таймера включается бит сброса, текущее значение сбрасывается в 0000, все флаги завершения сбрасываются.





В следующей таблице показана работа команды MTIM(543)/MTIMX(554) при четырех возможных комбинациях состояний битов сброса и паузы.

Бит сброса (бит 08)	Бит паузы (бит 09)	Работа
ВЫКЛ	ВЫКЛ	Текущее значение обновляется, соответствующий флаг завершения устанавливается, когда заданное значение ≤ текущее значение.
	ВКЛ	Текущее значение не обновляется, команда MTIM(543)/ MTIMX(554) воспринимается как NOP(000).
ВКЛ	ВЫКЛ	Текущее значение сбрасывается в 0000, флаги завершения сбрасываются. Текущее значение не обновляется.
	ВКЛ	

Биты сброса и паузы оказывают действие на работу команды MTIM(543)/ MTIMX(554) только в том случае, если установлено условие выполнения команды.

Флаги

Название	Обозна-чение	Операция
Флаг ошибки	ER	Включен, если текущее значение в слове D2 не представлено в формате BCD. Выключен во всех остальных случаях.

Меры предосторожности

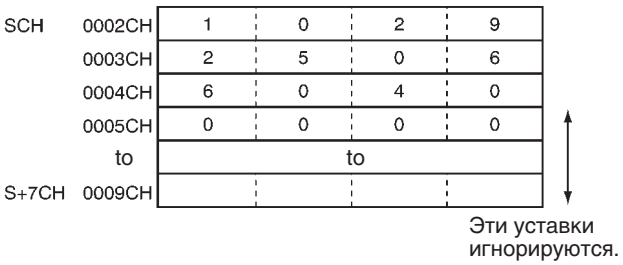
В отличие от большинства таймеров, команда MTIM(543)/MTIMX(554) не использует номер таймера. (Обновление текущих значений в области таймеров для MTIM(543)/MTIMX(554) не выполняется.)

Достигнув 9999, текущее значение вновь становится равным 0000, все флаги завершения при этом сбрасываются.

Если выбран формат BCD, а некоторое заданное значение в словах S...S+7 представлено не в формате BCD, это заданное значение игнорируется. Ошибки при этом не возникает, флаг ошибки не устанавливается.

Так как флаг завершения MTIM(543)/MTIMX(554) находится в области данных, его можно принудительно установить или принудительно сбросить, как любые другие биты, но текущее значение при этом не изменится.

Если уставок должно быть меньше восьми, введите в слово, следующее после последней уставки, значение 0000. Команда MTIM(543)/ MTIMX(554) проигнорирует уставку, равную 0000, как и все следующие за ней уставки.



Текущее значение таймера обновляется только при выполнении команды MTIM(543)/MTIMX(554), поэтому таймер будет работать неправильно, если время выполнения цикла превысит 100 мс (так как шаг отсчета времени данного таймера составляет 100 мс). Чтобы обеспечить точный отсчет времени и избежать проблем, вызванных большой длительностью цикла, вставьте одну и ту же команду MTIM(543)/MTIMX(554) в нескольких местах программы.

Флаг завершения таймера обновляется только при выполнении команды MTIM(543)/MTIMX(554), поэтому задержка установки флага завершения после завершения отсчета может составлять до одного цикла.

Если команда MTIM(543)/MTIMX(554) используется в сегменте между командами IL(002) и ILC(003) и данный сегмент программы блокируется, сохраняется предыдущее текущее значение (значение не сбрасывается). Обязательно принимайте во внимание этот факт при размещении команды MTIM(543)/MTIMX(554) между командами IL(002) и ILC(003).

Если работающий таймер, запущенный командой MTIM(543)/ MTIMX(554), расположен в сегменте программы между командами JMP(004) и JME(005) и этот сегмент программы пропускается, сохраняется предыдущее текущее значение. Обязательно принимайте во внимание этот факт при размещении команды MTIM(543)/MTIMX(554) между командами JMP(004) и JME(005).

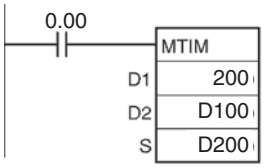
Позаботьтесь, чтобы слова, указанные для флага завершения и текущего значения (D1 и D2), не использовались другими командами. Если другие команды будут оказывать влияние на содержимое этих слов, таймер может работать неправильно.

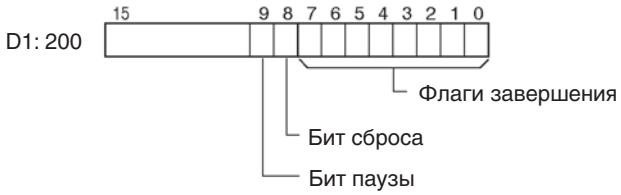
Если в качестве D1 указано слово из области CIO , для управления битами паузы и сброса можно использовать команды SET и RSET.

Пример

Если в следующем примере бит CIO 0.00 включен, а бит паузы (CIO 200.09) выключен, таймер начнет отсчет при выключении бита сброса (CIO 200.08). Отсчет времени ведется с 0000.

Восемь уставок, указанных в словах D200...D207, сравниваются с текущим значением. Если уставка ≤ текущее значение, устанавливается соответствующий флаг завершения (CIO 200.00...CIO 200.07).





Текущ. значение таймера
D2: D100

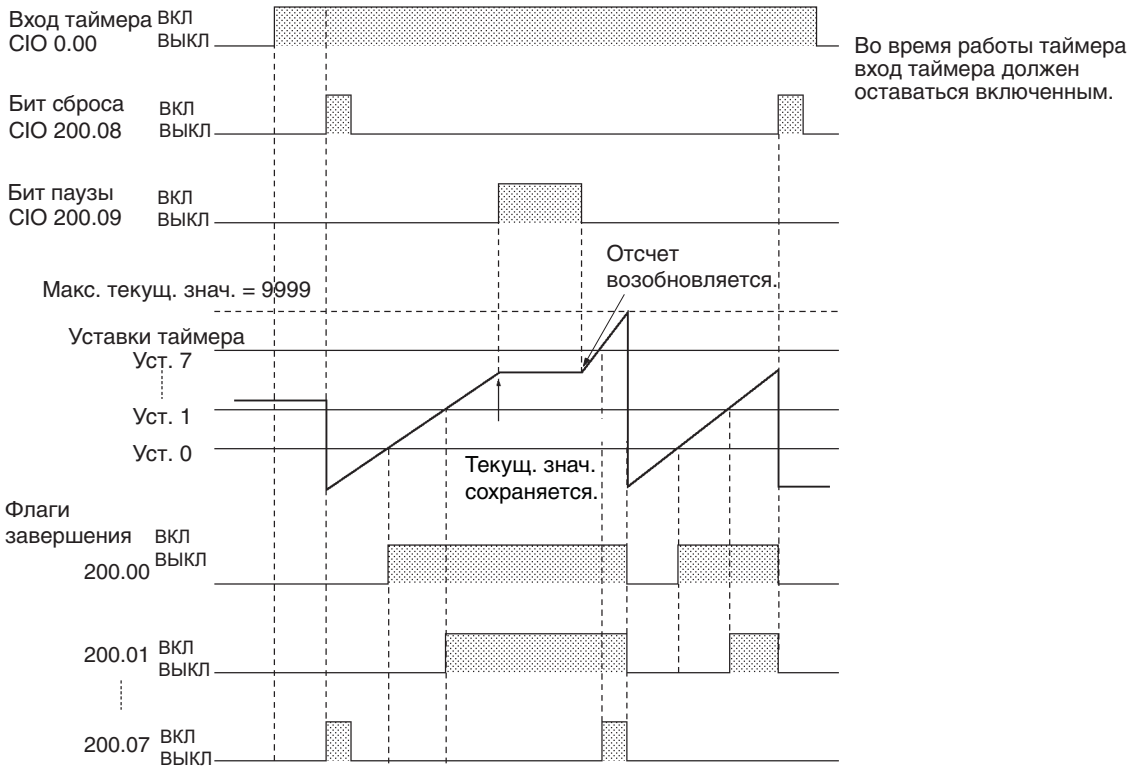
15			0
0	1	0	0

(Приращение)

Соответствующий флаг завершения = «1», когда уставка <= текущ. знач.

Уставки таймера

	15			0		
S: D200	0	0	8	0	1	0
S+1: D201	0	0	9	0	1	1
S+2: D202	0	1	0	0	1	2
S+3: D203	0	1	1	0	0	3
S+4: D204	0	1	2	0	0	4
S+5: D205	0	1	3	0	0	5
S+6: D206	0	1	5	0	0	6
S+7: D207	1	0	0	0	0	7



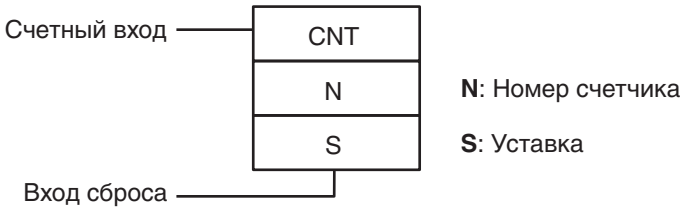
3-5-7 СЧЕТЧИК: CNT/CNTX(546)

Назначение

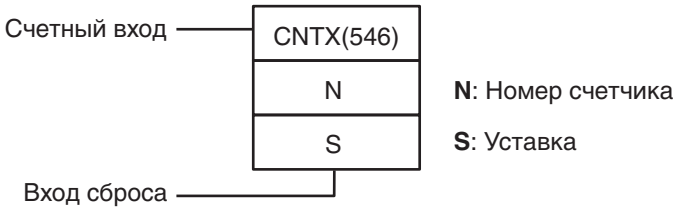
CNT/CNTX(546) запускает счетчик обратного счета. Задание счетчика должно находиться в диапазоне от 0 до 9999 (для CNT) и от 0 до 65535 (для CNTX(546)).

Символ РКС

BCD



Двоичное



Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	CNT/CNTX(546)
	Однократное выполнение по положительному фронту	Не предусмотрено
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	**Области пошаговых программ**	**Подпрограммы**	**Задачи обработки прерываний**
Не допускается	OK	OK	OK

Операнды

N: Номер счетчика

Номер счетчика должен быть указан в диапазоне от 0000 до 4095 (десятичн).

S: Уставка

Данные	**Диапазон**
BCD	#0000...#9999
Двоичное	&0...&65535 (десятичн.) #0000...#FFFF (hex)

Характеристики операндов

Область	**N**	**S**
Область CIO	---	CIO 0...CIO 6143
Рабочая область	---	W0...W511
Область битов хранения	---	H0...H511
Область вспомогательных битов	---	A0...A959

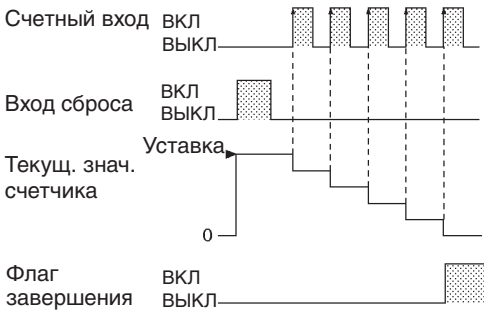
Область	N	S
Область таймеров	---	T0000...T4095
Область счетчиков	0000...4095 (десятичн.)	C0000...C4095
Область DM	---	D0...D32767
Косвенные адреса DM в двоичном формате	---	@ D0...@ D32767
Косвенные адреса DM в двоично-десятичном формате (BCD)	---	*D0...*D32767
Постоянные	---	BCD: #0000...9999 (BCD) «&» использовать нельзя. Двоичный: &0...&65535 (десятичн.) #0000...#FFFF (hex)
Регистры данных	---	DR0...DR15
Регистры указателей	---	---
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15	

Описание

Текущее значение счетчика уменьшается на 1 каждый раз, когда вход счетчика переходит из состояния «0» в состояние «1». Флаг завершения устанавливается, когда текущее значение достигает 0.

После установки флага завершения счетчик должен быть сброшен путем включения входа сброса или с помощью команды CNR(545)/CNRX(547). В противном случае счетчик нельзя будет перезапустить.

Пока включен вход сброса, счетчик находится в состоянии сброса, сигнал на входе счетчика при этом игнорируется. (При сбросе счетчика его текущее значение возвращается к значению уставки, флаг завершения сбрасывается.)



Флаги

Название	Обозначение	Операция
Флаг ошибки	ER	Включен, если адрес N, косвенно указанный через регистр указателя, не является адресом текущего значения счетчика. Включен, если выбран режим BCD, но значение S не представлено в формате BCD. Выключен во всех остальных случаях.
Флаг равенства	=	Выключен или не изменяется
Флаг отрицательного значения	N	Выключен или не изменяется

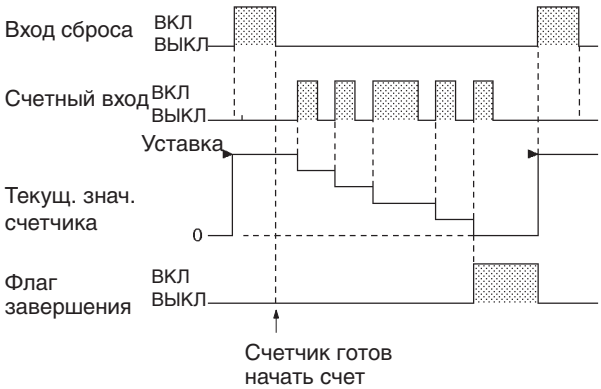
Меры предосторожности

Номера счетчиков являются общими для команд CNT, CNTX(546), CNTR(012), CNTRX(548), CNTW(814) и CNTWX(818). Если в программе используются две команды счетчиков с одинаковым номером счетчика, при проверке программы генерируется ошибка дублирования. При этом оба счетчика будут работать нормально, если их работа будет происходить в разное время. Счетчики с одинаковым номером счетчика будут работать неправильно, если они будут работать одновременно.

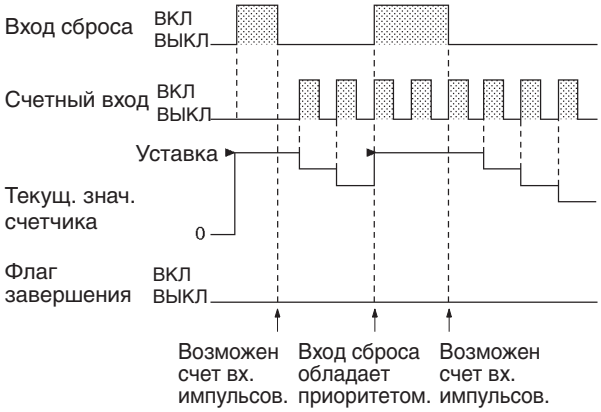
Текущее значение счетчика обновляется при включении входа счетчика, флаг завершения обновляется при каждом выполнении команды CNT/ CNTX(546). Флаг завершения установлен, если текущее значение равно 0. Флаг завершения сброшен, если текущее значение не равно 0.

В случае принудительной установки счетчика CNT/CNTX(546) его флаг завершения будет установлен, а текущее значение будет сброшено в 0000. В случае принудительного сброса счетчика CNT/CNTX(546) его флаг завершения будет сброшен, а текущее значение будет возвращено к значению уставки.

Обязательно сбросьте счетчик путем выключения → включения → → выключения входа сброса, прежде чем подавать сигнал на счетный вход счетчика (см. рисунок ниже). Если вход сброса включен, сигнал на счетном входе счетчика не воспринимается.



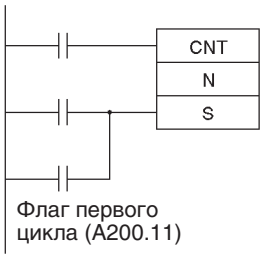
Вход сброса обладает приоритетом, поэтому если вход сброса и вход счетчика включаются одновременно, счетчик сбрасывается. (Текущее значение возвращается к значению уставки, флаг завершения сбрасывается.)



Работа флагов «=» и «N» зависит от модели модуля ЦПУ. Подробнее см. в таблице *Флаги* выше.

Примечание. Если счетчик вставляется в режиме онлайн-редактирования, он начнет работать правильно только после сброса. Если не сбросить счетчик, в качестве текущего значения счетчика будет использоваться предыдущее значение, поэтому вновь созданный счетчик может работать неправильно.

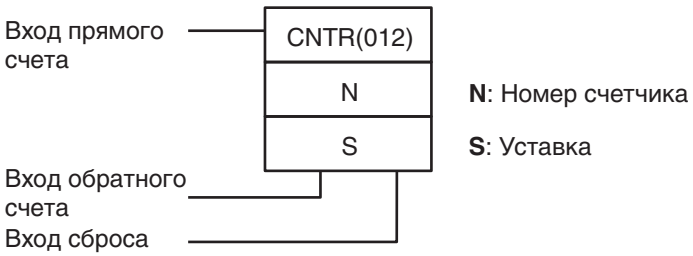
Текущие значения счетчика сохраняются даже в случае прерывания питания. Если нужно перезапустить счетчик и возобновить счет с уставки, а не с последнего текущего значения, на вход сброса счетчика следует подать флаг первого цикла (A200.11).



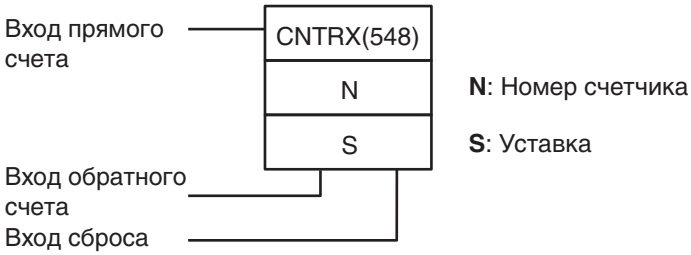
3-5-8 РЕВЕРСИВНЫЙ СЧЕТЧИК: CNTR(012)/CNTRX(548)

Назначение Команда CNTR(012)/CNTRX(548) запускает реверсивный счетчик.

Символ РКС **BCD**



Двоичное



Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	CNTR(012)/ CNTRX(548)
	Однократное выполнение по положительному фронту	Не предусмотрено
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
Не допускается	OK	OK	OK

Операнды

N: Номер счетчика

Номер счетчика должен быть указан в диапазоне от 0000 до 4095 (десятичн).

S: Уставка

Данные	Диапазон
BCD	#0000...#9999
Двоичное	&0...&65535 (десятичн.) #0000...#FFFF (hex)

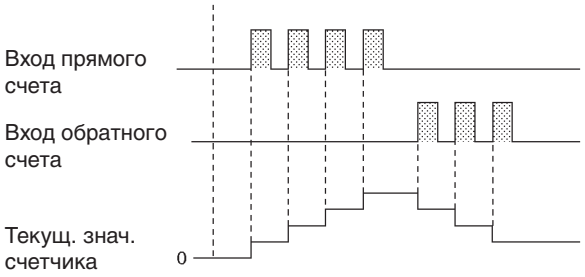
Характеристики операндов

Область	N	S
Область CIO	---	CIO 0...CIO 6143
Рабочая область	---	W0...W511
Область битов хранения	---	H0...H511
Область вспомогательных битов	---	A0...A959
Область таймеров	---	T0000...T4095
Область счетчиков	0000...4095 (десятичн.)	C0000...C4095
Область DM	---	D0...D32767
Косвенные адреса DM в двоичном формате	---	@ D0...@ D32767

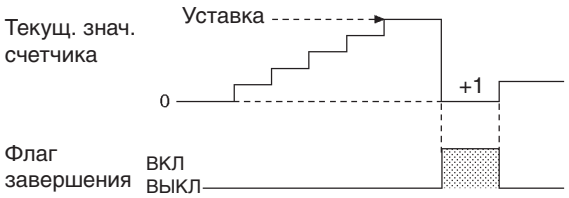
Область	N	S
Косвенные адреса DM в двоично-десятичном формате (BCD)	---	*D0...*D32767
Постоянные	---	BCD: #0000...9999 (BCD) «&» использовать нельзя. Двоичный: &0...&65535 (десятичн.) #0000...#FFFF (hex)
Регистры данных	---	DR0...DR15
Регистры указателей	---	---
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15	

Описание

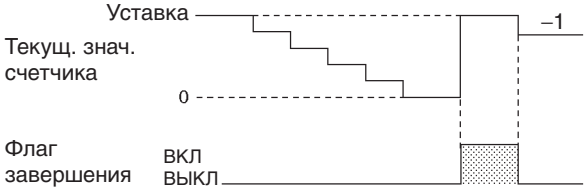
Текущее значение счетчика увеличивается на 1 при каждом переключении входа прямого счета из «0» в «1» и уменьшается на 1 при каждом переключении входа обратного счета из «0» в «1». Текущее значение может изменяться в пределах от 0 до заданной уставки.



При счете в прямом направлении флаг завершения устанавливается, когда текущее значение сбрасывается в 0 после достижения уставки. При увеличении текущего значения с 0 до 1 флаг завершения вновь сбрасывается.



При счете в обратном направлении флаг завершения устанавливается, когда текущее значение возвращается к уставке после достижения 0. После уменьшения с уставки до (уставка - 1) флаг завершения вновь сбрасывается.



Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

Флаги

Название	Обозначение	Операция
Флаг ошибки	ER	Включен, если адрес N, косвенно указанный через регистр указателя, не является адресом текущего значения счетчика. Включен, если выбран режим BCD, но значение S не представлено в формате BCD. Выключен во всех остальных случаях.

Меры предосторожности

Номера счетчиков являются общими для команд CNT, CNTX(546), CNTR(012), CNTRX(548), CNTW(814) и CNTWX(818). Если в программе используются две команды счетчиков с одинаковым номером счетчика, при проверке программы генерируется ошибка дублирования. При этом оба счетчика будут работать нормально, если их работа будет происходить в разное время. Счетчики с одинаковым номером счетчика будут работать неправильно, если они будут работать одновременно.

Текущее значение не изменяется при одновременном включении входов прямого и обратного счета. Если включен вход сброса, то текущее значение обнулено, а оба входа счетчика игнорируются.

Флаг завершения устанавливается только в момент реверса заданного значения: из уставки в 0 (при прямом счете) и из 0 в уставку (при обратном счете).

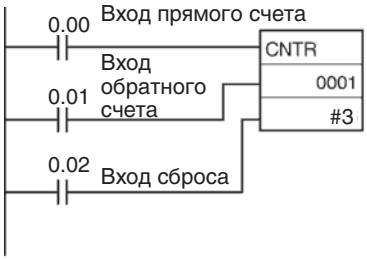
Ввод команды CNTR(012)/CNTRX(548) в мнемоническом представлении необходимо производить в следующей последовательности: сначала вход прямого счета (II), затем вход обратного счета (DI), затем вход сброса (R) и, наконец, команда CNTR(012)/CNTRX(548). Ввод в виде символа РКС необходимо производить в следующей последовательности: сначала вход прямого счета (II), затем команда CNTR(012)/CNTRX(548), затем вход обратного счета (DI) и, наконец, вход сброса (R).

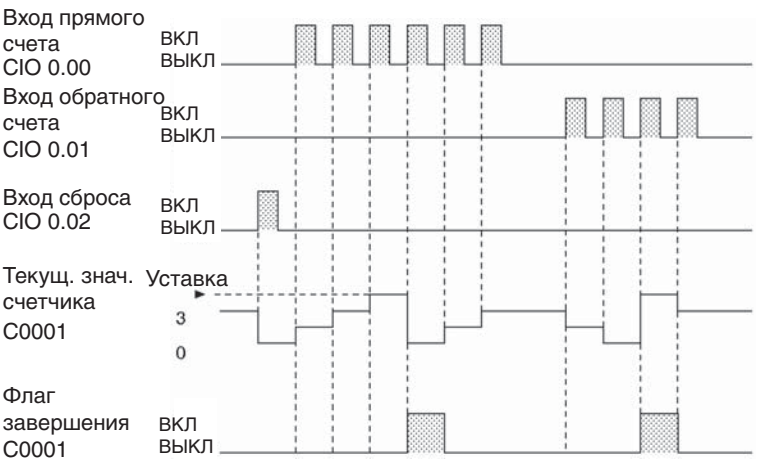
Примеры

Основное функционирование команды CNTR(012)/CNTRX(548)

Текущее значение счетчика сбрасывается в 0 путем последовательного включения и выключения входа сброса (CIO 0.02). Текущее значение увеличивается на 1 при каждом переключении входа прямого счета (CIO 0.00) из «0» в «1». После достижения уставки следующее увеличение приводит к сбросу текущего значения в 0, при этом устанавливается флаг завершения.

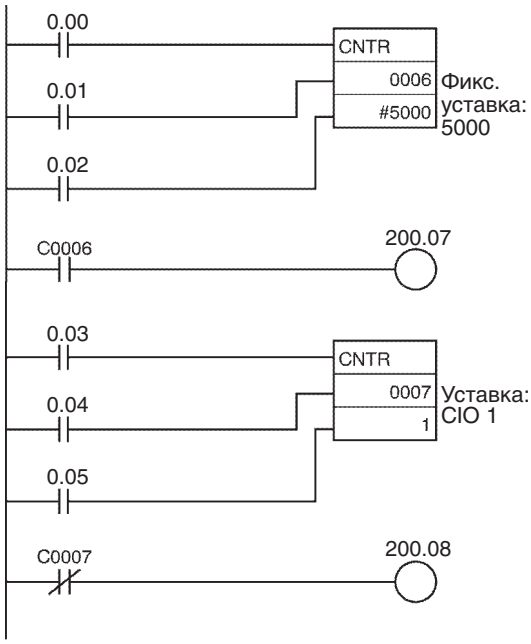
Аналогично, текущее значение уменьшается на 1 при каждом переключении входа обратного счета (CIO 0.01) из «0» в «1». После достижения нуля следующее уменьшение приводит к возврату текущего значения к значению уставки, при этом устанавливается флаг завершения.

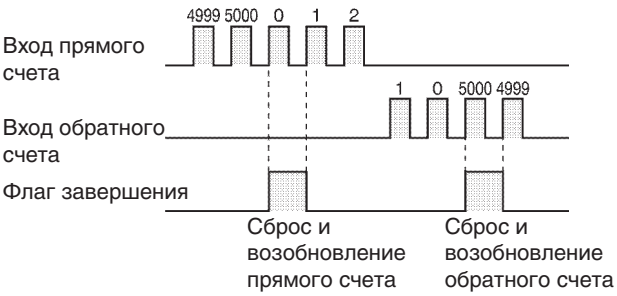




Ввод уставки с помощью слова

В следующем примере уставка для команды CNTR(012) (0007) определяется содержимым слова CIO 1. Для управления содержимым слова CIO 1 можно использовать внешний переключатель, что позволит изменять уставку вручную.





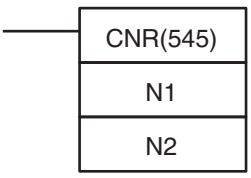
3-5-9 СБРОС ТАЙМЕРА/СЧЕТЧИКА: CNR(545)/CNRX(547)

Назначение

Команда сбрасывает таймеры или счетчики с номерами, принадлежащими указанному диапазону.

Символ РКС

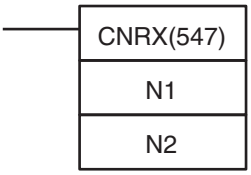
BCD



N_1: Первый номер диапазона

N_2: Последний номер диапазона

Двоичное



N_1: Первый номер диапазона

N_2: Последний номер диапазона

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	CNR(545)/ CNRX(547)
	Однократное выполнение по положительному фронту	@CNR(545)/ CNRX(547)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

N_1: Первый номер диапазона

N_1 должен содержать номер таймера в диапазоне от T0000 до T4095 либо номер счетчика в диапазоне от C0000 до C4095.

N_2: Последний номер диапазона

N_2 должен содержать номер таймера в диапазоне от T0000 до T4095 либо номер счетчика в диапазоне от C0000 до C4095.

Примечание. N_1 и N_2 должны принадлежать одной области, т. е. N_1 и N_2 должны быть номерами таймеров или номерами счетчиков.

Характеристики операндов

Область	N_1	N_2
Область CIO	---	---
Рабочая область	---	---
Область битов хранения	---	---
Область вспомогательных битов	---	---
Область таймеров	C0000...C4095	C0000...C4095
Область счетчиков	T0000...T4095	T0000...T4095
Область DM	---	---
Косвенные адреса DM в двоичном формате	---	---
Косвенные адреса DM в двоично-десятичном формате (BCD)	---	---
Постоянные	---	---
Регистры данных	---	---
Регистры указателей	---	---
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

Описание

Команда CNR(545)/CNRX(547) сбрасывает флаги завершения всех таймеров или счетчиков в диапазоне от N_1 до N_2. Одновременно во все текущие значения таймеров/счетчиков записывается максимальное значение (9999 для BCD и FFFF для двоичного формата). (При следующем выполнении команды управления таймером или счетчиком в текущее значение будет записана соответствующая уставка.)

Сброс таймеров командой CNR(545)/CNRX(547)

Следующие таймеры будут сброшены, если их номера попадают в указанный диапазон: TIM, TIMX(550), TIMH(015), TIMHX(551), TMHH(540), TMHHX(552), TTIM(087), TTIMX(555), TIMW(813), TIMWX(816), TMHW(815) и TMHWX(817). При сбросе таймера флаг завершения сбрасывается, а текущее значение становится равным максимальному значению: 9999.

Примечание. Команда CNR(545)/CNRX(547) не сбрасывает таймеры TIML(542), TIMLX(553), MTIM(543) и MTIMX(554), так как для них не используются номера таймеров.

Сброс счетчиков командой CNR(545)/CNRX(547)

Следующие счетчики будут сброшены, если их номера попадают в указанный диапазон: CNT, CNTX(546), CNTR(012), CNTRX(548), CNTW(814) и CNTWX(818). При сбросе счетчика флаг завершения сбрасывается, а текущее значение становится равным максимальному значению: 9999.

Флаги

Название	Обозначение	Операция
Флаг ошибки	ER	Включен, если адрес N_1, косвенно указанный через регистр указателя, не является адресом текущего значения таймера или счетчика. Включен, если адрес N_2, косвенно указанный через регистр указателя, не является адресом текущего значения таймера или счетчика. Включен, если N_1 и N_2 принадлежат разным областям. Выключен во всех остальных случаях.

Меры предосторожности

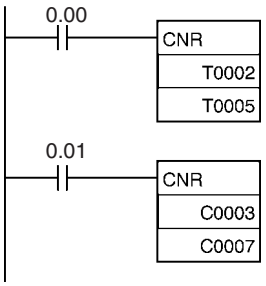
Команда CNR(545)/CNRX(547) не сбрасывает саму команду управления таймером/счетчиком, она сбрасывает текущие значения и флаги завершения, выделенные для этих команд. В большинстве случаев действие команды CNR(545)/CNRX(547) отличается от непосредственного сброса команд. Например, при непосредственном сбросе команды TIM/TIMX(550) ее текущее значение становится равным уставке, а при сбросе таймера командой CNR(545)/CNRX(547) его текущее значение становится равным 9999.

Если N1>N2, сбрасывается только флаг завершения таймера/счетчика с соответствующим номером.

Пример

Если в следующем примере включается бит CIO 0.00, то флаги завершения таймеров T0002...T0005 сбрасываются, а текущие значения этих таймеров становятся равными максимальному значению 9999.

Если включается бит CIO 0.01, то флаги завершения счетчиков C0003...C0007 сбрасываются, а текущие значения этих счетчиков становятся равными максимальному значению 9999.



3-5-10 Примеры применения таймеров и счетчиков

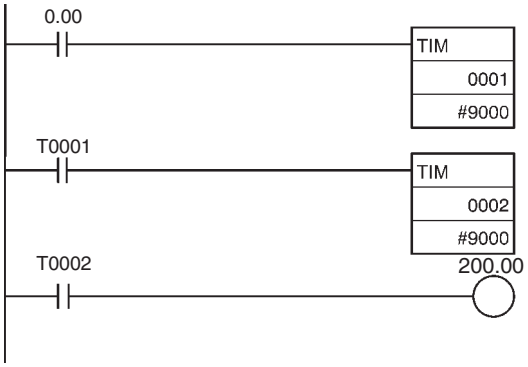
Ниже приведены различные примеры применения команд таймеров и счетчиков, в том числе применения долгосрочных таймеров, двухступенчатого счетчика, реализации задержки включения/выключения, одновибратора и мультивибратора.

Пример 1: Долгосрочные таймеры

В приведенных ниже примерах программ показаны три способа создания долгосрочных таймеров с помощью стандартных команд TIM и CNT.

Две команды TIM

В данном примере для создания 30-минутного таймера используются две команды TIM.

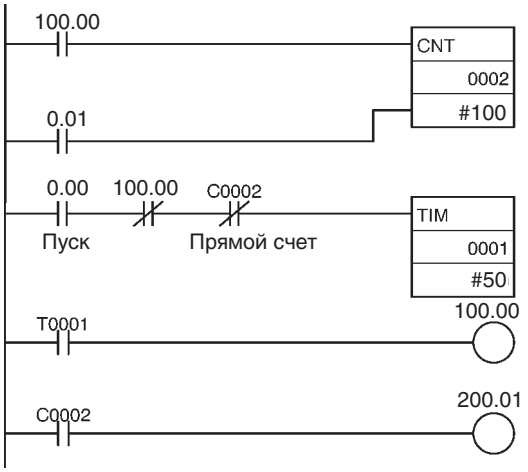


Адрес	Команда	Операнды
000000	LD	0.00
000001	TIM	0001
		#9000
000002	LD	T0001
000003	TIM	0002
		#9000
000004	LD	T0002
000005	OUT	200.00

Команды TIM и CNT

В данном примере для создания 500-секундного таймера используются команды TIM и CNT.

TIM 0001 генерирует импульсы с периодом 5 с, а команда CNT 0002 подсчитывает количество таких импульсов. В итоге отсчитывается интервал, равный: уставка таймера × уставка счетчика. В данном примере интервал отсчета равен 5 с × 100 = 500 с. При таком способе реализации текущее значение долгосрочного таймера равно текущему значению счетчика, которое сохраняется при прерывании питания.

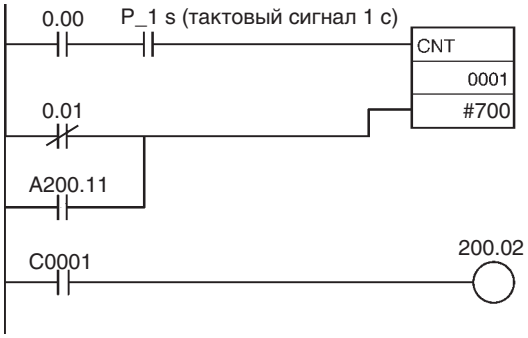


Адрес	Команда	Операнды
000000	LD	100.00
000001	LD	0.01
000002	CNT	0002
		#100
000003	LD	0.00
000004	AND NOT	100.00
000005	AND NOT	C0002
000006	TIM	0001
		#50
000007	LD	T0001
000008	OUT	100.00
000009	LD	C0002
000010	OUT	201.01

Тактовый генератор и команда CNT

В данном примере команда CNT подсчитывает импульсы 1-секундного генератора тактовых импульсов для создания 700-секундного таймера.

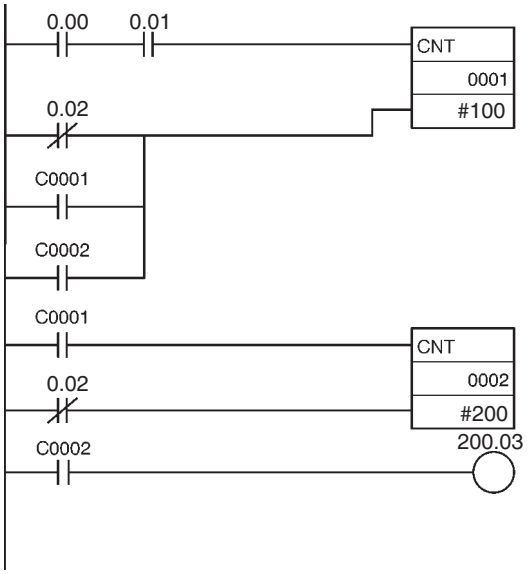
Если параллельно со входом сброса счетчика (CIO 0.01) включен флаг первого цикла (A200.11), в начале работы программы происходит сброс текущего значения к значению уставки (0700) вместо возобновления отсчета с предыдущего текущего значения.



Адрес	Команда	Операнды
000000	LD	0.00
000001	AND	1 s
000002	LD NOT	0.01
000003	OR	A200.11
000004	CNT	0001
		#700
000005	LD	C0001
000006	OUT	200.02

Пример 2: Двухступенчатый счетчик

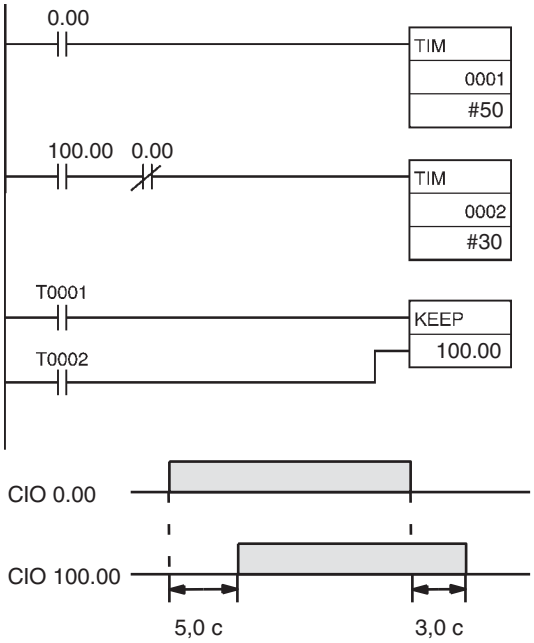
Если нужно задать уставку, превышающую 9999, можно использовать два счетчика, как показано в следующем примере. В данном случае две команды CNT используются для создания BCD-счетчика с уставкой 20 000.



Адрес	Команда	Операнды
000000	LD	0.00
000001	AND	0.01
000002	LD NOT	0.02
000003	OR	C0001
000004	OR	C0002
000005	CNT	0001
		#100
000006	LD	C0001
000007	LD NOT	0.02
000008	CNT	0002
		#200
000009	LD	C0002
000010	OUT	200.03

Пример 3: Задержка включения/ выключения

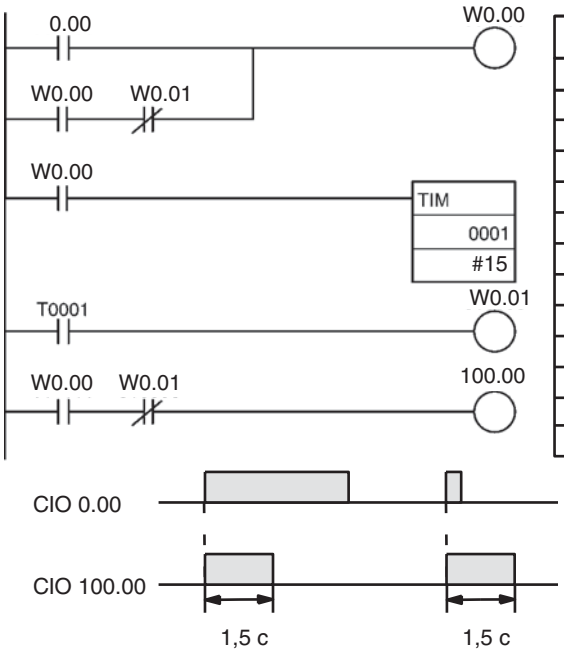
В данном примере два таймера TIM используются вместе с командой KEEP(011) для реализации задержки включения и задержки выключения. Бит CIO 100.00 включается через 5,0 секунд после включения бита CIO 0.00 и выключается через 3,0 секунды после выключения бита CIO 0.00.



Адрес	Команда	Операнды
000000	LD	0.00
000001	TIM	0001
		#50
000002	LD	100.00
000003	AND NOT	0.00
000004	TIM	0002
		#30
000005	LD	T0001
000006	LD	T0002
000007	KEEP(011)	100.00

Пример 4: Одновибратор

Таймер TIM можно использовать вместе с командами OUT или OUT NOT для управления длительностью пребывания определенного бита во включенном или выключенном состоянии. В данном примере бит CIO 100.00 будет включен в течение 1,5 секунд (уставка T0001) после включения CIO 0.00.



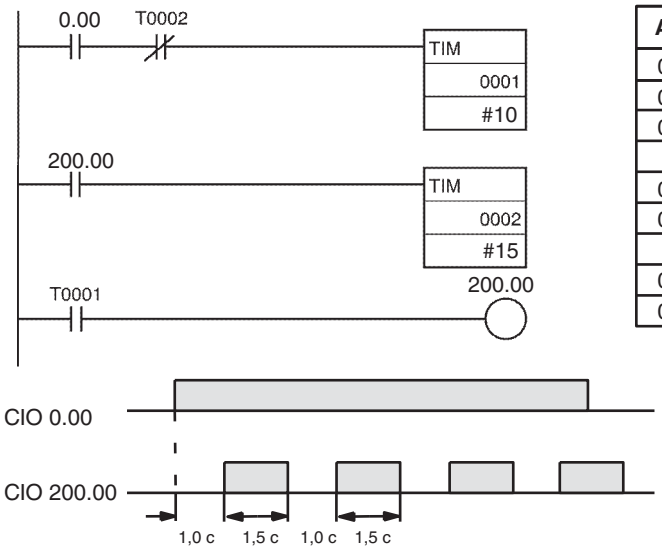
Адрес	Команда	Операнды
000000	LD	0.00
000001	LD	W0.00
000002	AND NOT	W0.01
000003	OR LD	–––
000004	OUT	W0.00
000005	LD	W0.00
000006	TIM	0001
		#15
000007	LD	T0001
000008	OUT	W0.01
000009	LD	W0.00
000010	AND NOT	W0.01
000011	OUT	100.00

Пример 5: Мультивибратор

В приведенных ниже примерах программ показаны два способа создания мультивибратора. Во втором примере просто воспроизводятся тактовые импульсы.

Две команды TIM

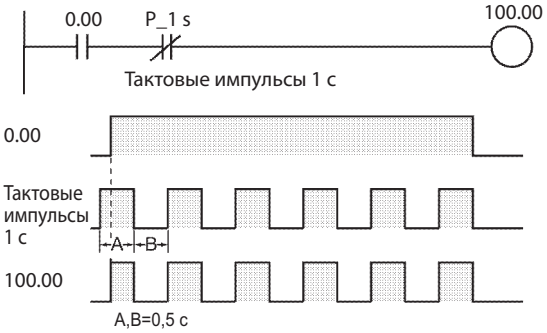
Для включения и выключения бита через равные интервалы времени при включенном условии выполнения можно использовать вместе два таймера TIM. В данном примере бит CIO 200.00 будет выключаться на 1,0 секунду, а затем включаться на 1,5 секунды, пока включен бит CIO 0.00.



Адрес	Команда	Операнды
000000	LD	0.00
000001	AND	T0002
000002	TIM	0001
		#10
000003	LD	200.00
000004	TIM	0002
		#15
000005	LD	T0001
000006	OUT	200.00

Тактовые импульсы

Для создания тактовых импульсов (с периодом 0,1 с / 0,2 с / 1,0 с) можно использовать генератор тактовых импульсов с требуемым условием выполнения.



Адрес	Команда	Операнды
000000	LD	0.00
000001	AND	1s
000002	OUT	100.00

3-5-11 Косвенное обращение к таймерам/счетчикам

Адреса текущих значений таймеров и счетчиков могут указываться косвенно с помощью регистров указателей. Для того чтобы использовать косвенную адресацию, с помощью команды MOVRW(561) (запись текущего значения таймера/счетчика в регистр) в требуемый регистр указателя следует записать адрес памяти ПЛК текущего значения требуемого таймера или счетчика.

Косвенная адресация через регистры указателей может применяться для следующих таймеров и счетчиков: TIM, TIMX(550), TIMH(015), TIMHX(551), TTIM(087), TTIMX(555), TMHH(540), TMHHX(552), TIMW(813), TIMWX(816), TMHW(815), TMHWX(817), CNT, CNTX(546), CNTR(012), CNTRX(548), CNTW(814) и CNTWX(818). (Это те таймеры и счетчики, для которых используются номера таймеров и счетчиков.)

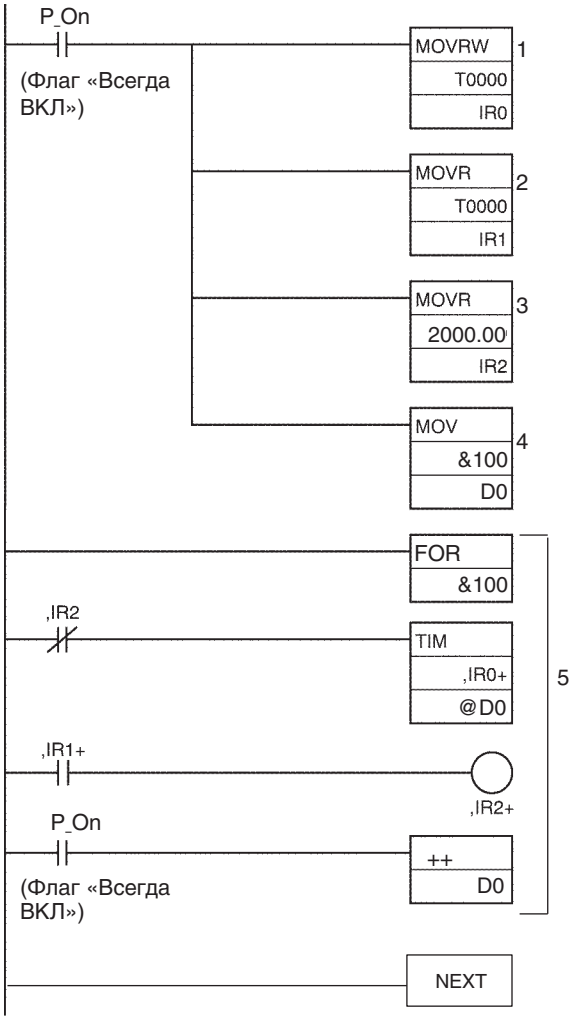
Команда управления таймером или счетчиком не выполняется, если адрес памяти ПЛК, хранящийся в указанном регистре указателя, не является адресом текущего значения таймера или счетчика.

Использование регистров указателей для косвенной адресации таймеров и счетчиков сокращает объем программы и повышает гибкость программирования. Например, можно создавать общие подпрограммы.

Пример

Ниже показан пример программы, использующей косвенную адресацию для создания и запуска 100 таймеров, уставки которых содержатся в словах D100...D199. Регистр IR0 содержит адрес памяти ПЛК текущего значения таймера, а регистр IR1 содержит адрес памяти ПЛК флага завершения таймера.

Адрес DM	Содержание	Функция
D100	0010	Уставка для T0000
D101	0100	Уставка для T0001
D102	0050	Уставка для T0002
. . .	. . .	. . .
D199	0999	Уставка для T0099



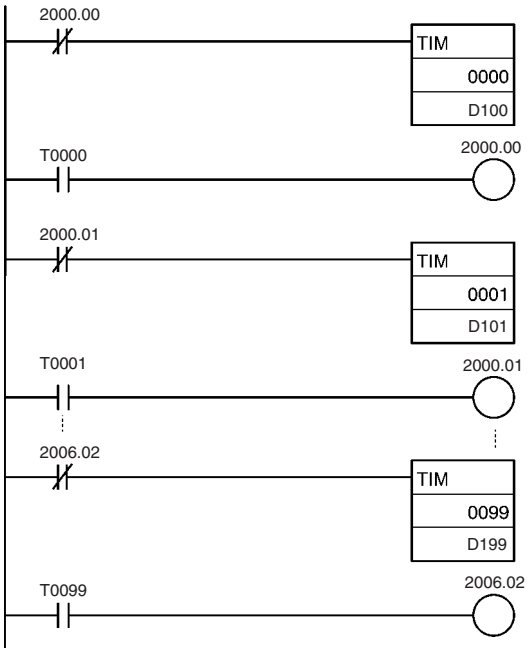
1,2,3...

1. Команда MOVRW(561) записывает адрес памяти ПЛК текущего значения таймера T0000 в регистр IR0. После этого IR0 можно использовать вместо номера таймера.
2. Команда MOVR(560) записывает адрес памяти ПЛК флага завершения работы таймера T0000 в регистр IR1.
3. Команда MOVR(560) записывает адрес памяти ПЛК бита CIO 2000.00 в регистр IR2.
4. Команда MOV(021) записывает &100 в D0 для косвенного указания адреса уставки таймера.
5. Содержимое IR0, IR1, IR2 и D0 увеличивается на 1 при каждом выполнении данного цикла, запуская таймеры T0...T99 (всего 100 раз).

В описанной выше программе 4 входных параметра цикла используются для запуска всех 100 таймеров посредством общей подпрограммы.

IR0 Адрес памяти ПЛК текущего значения таймера
IR1 Адрес памяти ПЛК флага завершения работы таймера.
IR2 Адрес памяти ПЛК условия выполнения таймера.
D0 Адрес области DM слова, содержащего уставку таймера.

Приведенная выше подпрограмма эквивалентна 400 командам, показанным ниже.


2000.00
TIM
0000
D100
T0000
2000.00
2000.01
TIM
0001
D101
T0001
2000.01
2006.02
TIM
0099
D199
T0099
2006.02

3-6 Команды сравнения

В данном разделе описаны команды, используемые для сравнения данных разной длины различными способами.

Команда	Мнемоническое обозначение	Код функции	Стр.
Входные команды сравнения	=, <>, <, <=, >, >= (S, L) (LD, AND, OR)	300...328	230
Команды сравнения значений времени	=DT, <>DT, <DT, <=DT, >DT, >=DT (LD, AND, OR)	341...346	236
СРАВНИТЬ	CMP	020	241
СРАВНИТЬ ДВОЙНЫЕ СЛОВА	CMPL	060	244
СРАВНИТЬ ДВОИЧНЫЕ ЗНАЧЕНИЯ СО ЗНАКОМ	CPS	114	247
СРАВНИТЬ ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ	CPSL	115	250
СРАВНИТЬ НЕСКОЛЬКО ЗНАЧЕНИЙ	MCMP	019	253
СРАВНИТЬ С ТАБЛИЦЕЙ	TCMP	085	256
СРАВНИТЬ С ДИАПАЗОНАМИ	BCMP	068	258
РАСШИРЕННОЕ СРАВНЕНИЕ С ДИАПАЗОНАМИ	BCMP2	502	261

3-6-1 Входные команды сравнения (300...328)

Назначение

Входные команды сравнения сравнивают два значения (константы и/или содержимое указанных слов) и включают условие выполнения, если выполнено условие сравнения. С помощью входных команд сравнения можно сравнивать данные со знаком или без знака длиной в одно или два слова.

Примечание. Описание входных команд сравнения для чисел с плавающей запятой одинарной точности см. в *3-14-21 Команды сравнения чисел с плавающей запятой одинарной точности*, для чисел с плавающей запятой двойной точности — в *3-15-21 Входные команды для чисел с плавающей запятой двойной точности*.

Символ РКС



S1: Сравниваемое значение 1

S2: Сравниваемое значение 2

Варианты выполнения

Варианты выполнения	**Включение условия выполнения в каждом цикле, в котором результат сравнения = ИСТИНА.**	Входная команда сравнения
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Описание операндов команд сравнения данных длиной в одно слово

Область	S_1	S_2
Область CIO	CIO 0...CIO 6143	
Рабочая область	W0...W511	
Область битов хранения	H0...H511	
Область вспомогательных битов	A0...A959	
Область таймеров	T0000...T4095	
Область счетчиков	C0000...C4095	
Область DM	D0...D32767	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Константы	#0000...#FFFF (двоичн.) &0...&65535 (десятичное без знака)	
Регистры данных	DR0...DR15	
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…-2048…+2047 ,IR15 DR0…DR15, IR0…IR15 ,IR0+(++)…,IR15+(++) ,–(– –)IR0…, –(– –)IR15	

Описание операндов команд сравнения данных длиной в два слова

Область	S_1	S_2
Область CIO	CIO 0...CIO 6142	
Рабочая область	W0...W510	
Область битов хранения	H0...H510	
Область вспомогательных битов	A0...A958	
Область таймеров	T0000...T4094	
Область счетчиков	C0000...C4094	
Область DM	D0...D32766	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Константы	#00000000...#FFFFFFFF (двоичное) &0...&4294967295 (десятичное без знака)	
Регистры данных	---	
Регистры указателей	IR0...IR15 (только для значений без знака)	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…-2048…+2047 ,IR15 DR0…DR15, IR0…IR15 ,IR0+(++)…,IR15+(++) ,–(– –)IR0…, –(– –)IR15	

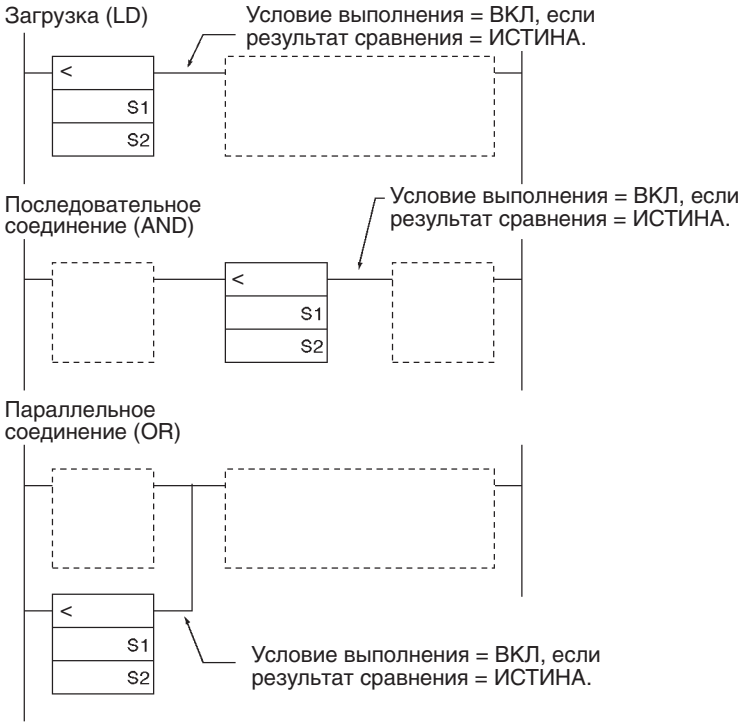
Описание

Входная команда сравнения сравнивает значения S_1 и S_2 как значения со знаком или без знака и включает условие выполнения, если выполнено условие сравнения. В отличие от таких команд, как CMP(020) и CMPL(060), результат входной команды сравнения сразу отражается в виде условия выполнения, что избавляет от необходимости считывать результат сравнения из флага арифметической операции, упрощая и ускоряя тем самым работу программы.

Применение команд в программе

Входные команды сравнения используются в программе так же, как команды LD, AND и OR. Они управляют выполнением следующих за ними команд.

Тип ввода	Описание работы
LD (Загрузка)	Команда может присоединяться непосредственно к левой шине.
AND (И)	Команда не может присоединяться непосредственно к левой шине.
OR (ИЛИ)	Команда может присоединяться непосредственно к левой шине.



Варианты применения

Входные команды сравнения могут сравнивать данные со знаком или без знака, длиной в одно или два слова. Если вариант применения специально не указан, сравнение проводится для данных без знака длиной в одно слово. Доступны шесть видов сравнения, три способа ввода команды, четыре типа данных: всего 72 комбинации.

Символ		Вариант (формат данных)	Вариант (длина данных)
= < > < <= > >=	(равно) (не равно) (меньше) (меньше или равно) (больше) (больше или равно)	Не указан: значение без знака S: значение со знаком	Не указан: одно слово L : два слова

Входные команды сравнения без знака (т. е. команды без опции S) работают с двоичными или двоично-десятичными данными без знака. Входные команды сравнения со знаком (т. е. команды с опцией S) работают с двоичными данными со знаком.

Краткий обзор входных команд сравнения

В приведенной ниже таблице перечислены коды функций, мнемонические коды, названия и назначение 72 входных команд сравнения. (Для сравнения данных длиной в одно слово C1=S_1, C2=S_2; для сравнения данных двойной длины C1=S_1+1, S_1, C2=S_2+1, S_2.)

Код	Мнемоническое обозначение	Название	Функция
300	LD=	ЗАГРУЗКА РАВНО	Истина, если C1 = C2
	AND=	И РАВНО	
	OR=	ИЛИ РАВНО	
301	LD=L	ЗАГРУЗКА ДВОЙНЫЕ СЛОВА РАВНЫ	
	AND=L	И ДВОЙНЫЕ СЛОВА РАВНЫ	
	OR=L	ИЛИ ДВОЙНЫЕ СЛОВА РАВНЫ	
302	LD=S	ЗАГРУЗКА СЛОВА СО ЗНАКОМ РАВНЫ	
	AND=S	И СЛОВА СО ЗНАКОМ РАВНЫ	
	OR=S	ИЛИ СЛОВА СО ЗНАКОМ РАВНЫ	
303	LD=SL	ЗАГРУЗКА ДВОЙНЫЕ СЛОВА СО ЗНАКОМ РАВНЫ	
	AND=SL	И ДВОЙНЫЕ СЛОВА СО ЗНАКОМ РАВНЫ	
	OR=SL	ИЛИ ДВОЙНЫЕ СЛОВА СО ЗНАКОМ РАВНЫ	
305	LD<>	ЗАГРУЗКА НЕ РАВНО	Истина, если C1 ≠ C2
	AND<>	И НЕ РАВНО	
	OR<>	ИЛИ НЕ РАВНО	
306	LD<>L	ЗАГРУЗКА ДВОЙНЫЕ СЛОВА НЕ РАВНЫ	
	AND<>L	И ДВОЙНЫЕ СЛОВА НЕ РАВНЫ	
	OR<>L	ИЛИ ДВОЙНЫЕ СЛОВА НЕ РАВНЫ	
307	LD<>S	ЗАГРУЗКА СЛОВА СО ЗНАКОМ НЕ РАВНЫ	
	AND<>S	И СЛОВА СО ЗНАКОМ НЕ РАВНЫ	
	OR<>S	ИЛИ СЛОВА СО ЗНАКОМ НЕ РАВНЫ	
308	LD<>SL	ЗАГРУЗКА ДВОЙНЫЕ СЛОВА СО ЗНАКОМ НЕ РАВНЫ	
	AND<>SL	И ДВОЙНЫЕ СЛОВА СО ЗНАКОМ НЕ РАВНЫ	
	OR<>SL	ИЛИ ДВОЙНЫЕ СЛОВА СО ЗНАКОМ НЕ РАВНЫ	
310	LD<	ЗАГРУЗКА МЕНЬШЕ	Истина, если C1 < C2
	AND<	И МЕНЬШЕ	
	OR<	ИЛИ МЕНЬШЕ	
311	LD<L	ЗАГРУЗКА ДВОЙНОЕ СЛОВО МЕНЬШЕ	
	AND<L	И ДВОЙНОЕ СЛОВО МЕНЬШЕ	
	OR<L	ИЛИ ДВОЙНОЕ СЛОВО МЕНЬШЕ	
312	LD<S	ЗАГРУЗКА СЛОВО СО ЗНАКОМ МЕНЬШЕ	
	AND<S	И СЛОВО СО ЗНАКОМ МЕНЬШЕ	
	OR<S	ИЛИ СЛОВО СО ЗНАКОМ МЕНЬШЕ	
313	LD<SL	ЗАГРУЗКА ДВОЙНОЕ СЛОВО СО ЗНАКОМ МЕНЬШЕ	
	AND<SL	И ДВОЙНОЕ СЛОВО СО ЗНАКОМ МЕНЬШЕ	
	OR<SL	ИЛИ ДВОЙНОЕ СЛОВО СО ЗНАКОМ МЕНЬШЕ	

Код	Мнемоническое обозначение	Название	Функция
315	LD<=	ЗАГРУЗКА МЕНЬШЕ ИЛИ РАВНО	Истина, если C1 ≤ C2
	AND<=	И МЕНЬШЕ ИЛИ РАВНО	
	OR<=	ИЛИ МЕНЬШЕ ИЛИ РАВНО	
316	LD<=L	ЗАГРУЗКА ДВОЙНОЕ СЛОВО МЕНЬШЕ ИЛИ РАВНО	
	AND<=L	И ДВОЙНОЕ СЛОВО МЕНЬШЕ ИЛИ РАВНО	
	OR<=L	ИЛИ ДВОЙНОЕ СЛОВО МЕНЬШЕ ИЛИ РАВНО	
317	LD<=S	ЗАГРУЗКА СЛОВО СО ЗНАКОМ МЕНЬШЕ ИЛИ РАВНО	
	AND<=S	И СЛОВО СО ЗНАКОМ МЕНЬШЕ ИЛИ РАВНО	
	OR<=S	ИЛИ СЛОВО СО ЗНАКОМ МЕНЬШЕ ИЛИ РАВНО	
318	LD<=SL	ЗАГРУЗКА ДВОЙНОЕ СЛОВО СО ЗНАКОМ МЕНЬШЕ ИЛИ РАВНО	Истина, если C1 ≤ C2
	AND<=SL	И ДВОЙНОЕ СЛОВО СО ЗНАКОМ МЕНЬШЕ ИЛИ РАВНО	
	OR<=SL	ИЛИ СЛОВО СО ЗНАКОМ МЕНЬШЕ ИЛИ РАВНО	
320	LD>	ЗАГРУЗКА БОЛЬШЕ	Истина, если C1 > C2
	AND>	И БОЛЬШЕ	
	OR>	ИЛИ БОЛЬШЕ	
321	LD>L	ЗАГРУЗКА ДВОЙНОЕ СЛОВО БОЛЬШЕ	
	AND>L	И ДВОЙНОЕ СЛОВО БОЛЬШЕ	
	OR>L	ИЛИ ДВОЙНОЕ СЛОВО БОЛЬШЕ	
322	LD>S	ЗАГРУЗКА СЛОВО СО ЗНАКОМ БОЛЬШЕ	
	AND>S	И СЛОВО СО ЗНАКОМ БОЛЬШЕ	
	OR>S	ИЛИ СЛОВО СО ЗНАКОМ БОЛЬШЕ	
323	LD>SL	ЗАГРУЗКА ДВОЙНОЕ СЛОВО СО ЗНАКОМ БОЛЬШЕ	
	AND>SL	И ДВОЙНОЕ СЛОВО СО ЗНАКОМ БОЛЬШЕ	
	OR>SL	ИЛИ ДВОЙНОЕ СЛОВО СО ЗНАКОМ БОЛЬШЕ	
325	LD>=	ЗАГРУЗКА БОЛЬШЕ ИЛИ РАВНО	Истина, если C1 ≥ C2
	AND>=	И БОЛЬШЕ ИЛИ РАВНО	
	OR>=	ИЛИ БОЛЬШЕ ИЛИ РАВНО	
326	LD>=L	ЗАГРУЗКА ДВОЙНОЕ СЛОВО БОЛЬШЕ ИЛИ РАВНО	
	AND>=L	И ДВОЙНОЕ СЛОВО БОЛЬШЕ ИЛИ РАВНО	
	OR>=L	ИЛИ ДВОЙНОЕ СЛОВО БОЛЬШЕ ИЛИ РАВНО	
327	LD>=S	ЗАГРУЗКА СЛОВО СО ЗНАКОМ БОЛЬШЕ ИЛИ РАВНО	
	AND>=S	И СЛОВО СО ЗНАКОМ БОЛЬШЕ ИЛИ РАВНО	
	OR>=S	ИЛИ СЛОВО СО ЗНАКОМ БОЛЬШЕ ИЛИ РАВНО	
328	LD>=SL	ЗАГРУЗКА ДВОЙНОЕ СЛОВО СО ЗНАКОМ БОЛЬШЕ ИЛИ РАВНО	
	AND>=SL	И ДВОЙНОЕ СЛОВО СО ЗНАКОМ БОЛЬШЕ ИЛИ РАВНО	
	OR>=SL	ИЛИ ДВОЙНОЕ СЛОВО СО ЗНАКОМ БОЛЬШЕ ИЛИ РАВНО	

Флаги

Название	Обо-значение	Описание работы
Флаг «Больше»	>	ВКЛ, если $S_1 > S_2$ для значений в одно слово. ВКЛ, если S_1+1, $S_1 > S_2+1$, S_2 для значений в два слова. Выключен во всех остальных случаях.
Флаг «Больше или равно»	> =	ВКЛ, если $S_1 \geq S_2$ для значений в одно слово. ВКЛ, если S_1+1, $S_1 \geq S_2+1$, S_2 для значений в два слова. Выключен во всех остальных случаях.
Флаг «Равно»	=	ВКЛ, если $S_1 = S_2$ для значений в одно слово. ВКЛ, если S_1+1, $S_1 = S_2+1$, S_2 для значений в два слова. Выключен во всех остальных случаях.
Флаг «Не равно»	=	ВКЛ, если $S_1 \neq S_2$ для значений в одно слово. ВКЛ, если S_1+1, $S_1 \neq S_2+1$, S_2 для значений в два слова. Выключен во всех остальных случаях.
Флаг «Меньше»	<	ВКЛ, если $S_1 < S_2$ для значений в одно слово. ВКЛ, если S_1+1, $S_1 < S_2+1$, S_2 для значений в два слова. Выключен во всех остальных случаях.
Флаг «Меньше или равно»	< =	ВКЛ, если $S_1 \leq S_2$ для значений в одно слово. ВКЛ, если S_1+1, $S_1 \leq S_2+1$, S_2 для значений в два слова. Выключен во всех остальных случаях.

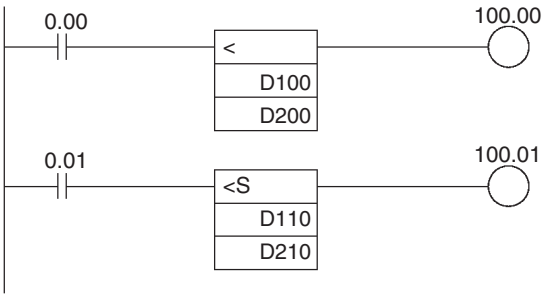
Меры предосторожности

Входная команда сравнения не может быть последней командой цепи. Между ней и правой шиной должна находиться как минимум еще одна команда.

Примеры

И МЕНЬШЕ: AND<(310)

Если в следующем примере бит CIO 0.00 включен, выполняется сравнение содержимого слов D100 и D200 как двоичных данных без знака. Если содержимое слова D100 меньше содержимого слова D200, включается бит CIO 100.00 и начинает выполняться следующая строка. Если содержимое D100 больше или равно содержимому D200, остальная часть строки программы пропускается и начинается выполнение следующей строки.



Сравнение МЕНЬШЕ без знака

S_1: D100 — 8714 — Десятичн.: 34580

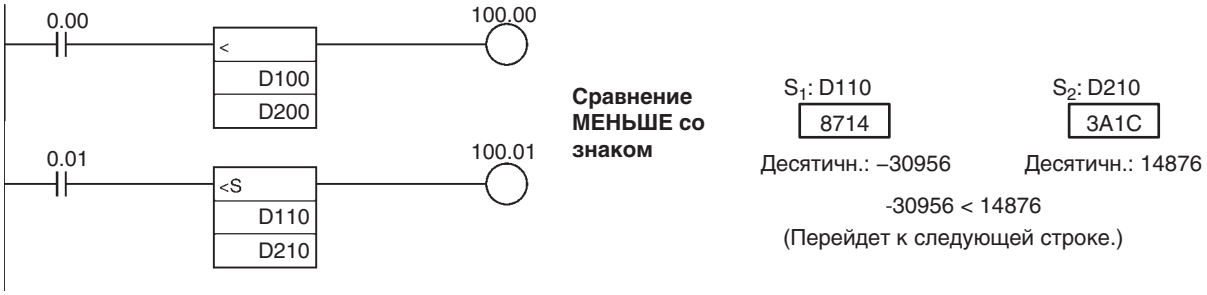
S_2: D200 — 3A1C — Десятичн.: 14876

34580 > 14876

(Не перейдет к следующей строке.)

И СЛОВО СО ЗНАКОМ МЕНЬШЕ: AND<S(312)

Если в следующем примере бит CIO 0.01 включен, выполняется сравнение содержимого слов D110 и D210 как двоичных данных со знаком. Если содержимое слова D110 меньше содержимого слова D210, включается бит CIO 100.01 и начинает выполняться следующая строка. Если содержимое D110 больше или равно содержимому D210, остальная часть строки программы пропускается и начинается выполнение следующей строки.



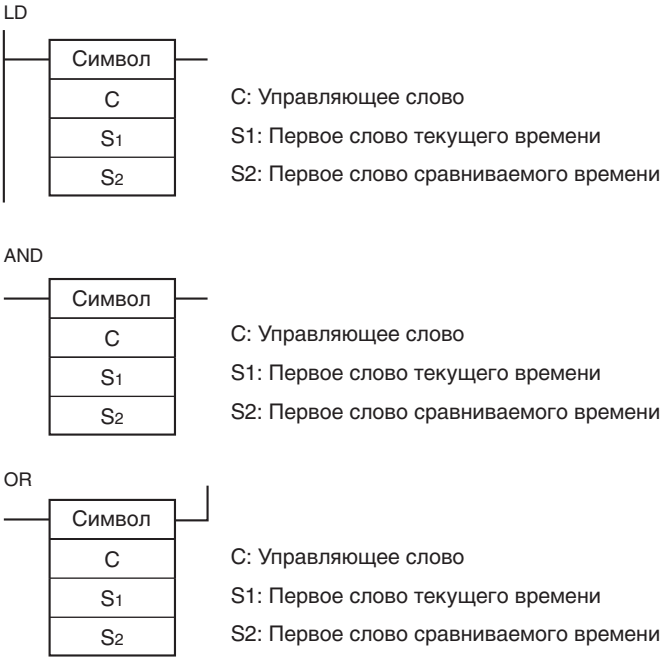
3-6-2 Команды сравнения значений времени (341...346)

Назначение

Команды сравнения значений времени сравнивают два значения времени в двоично-десятичном формате и включают условие выполнения, если условие сравнения выполнено.

Команды сравнения значений времени используются в программе так же, как команды LD, AND и OR. Они управляют выполнением следующих за ними команд.

Символ РКС



Варианты выполнения

Варианты выполнения	**Включение условия выполнения в каждом цикле, в котором результат сравнения = ИСТИНА.**	Команда сравнения значений времени
Модификатор мгновенного обновления		Не предусмотрено

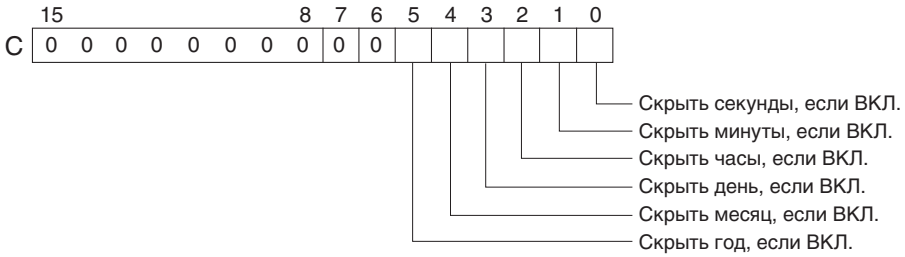
Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограм-мы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

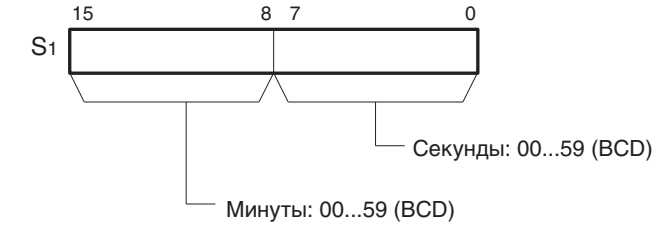
C: Управляющее слово

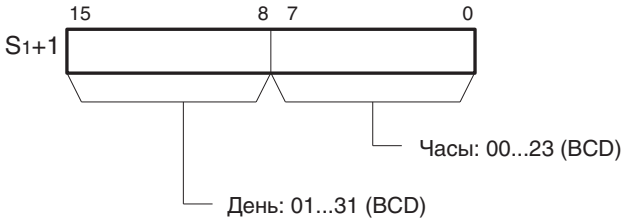
Биты 00...05 слова C указывают, должна ли использоваться маска для значений времени при выполнении сравнения. Биты 00...05 маскируют секунды, минуты, часы, день, месяц и год соответственно. Если все шесть значений замаскированы, команда не выполняется, условие выполнения будет выключено, включится флаг ошибки.

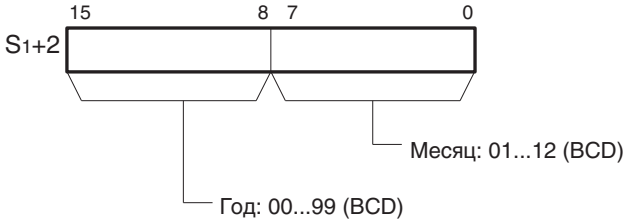


S_1 ... S_1+2: текущее время

Слова S_1...S_1+2 содержат значение текущего времени. Слова S_1...S_1+2 должны находиться в одной области данных.





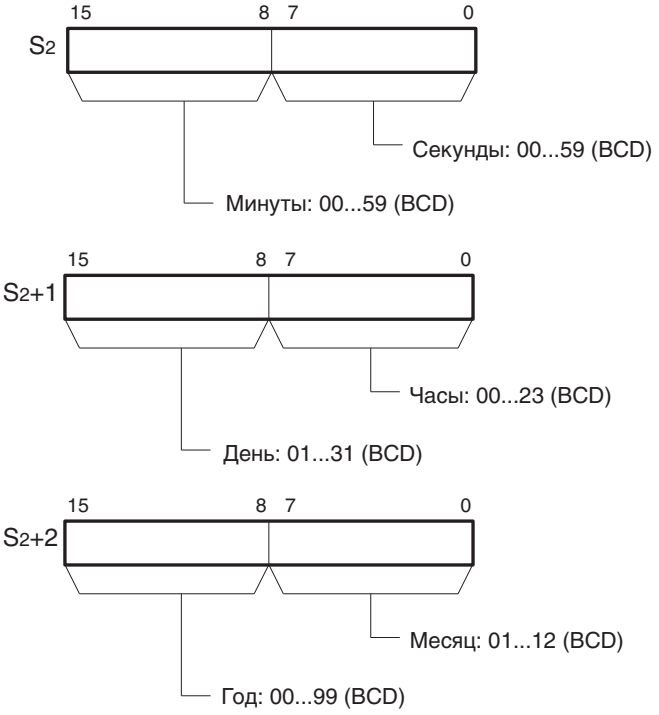


Примечание. Чтобы использовать показания внутренних часов модуля ЦПУ для сравнения, присвойте операнду S_1 значение A351. (Данные внутренних часов модуля ЦПУ содержатся в словах A351...A353.)

S_2 ... S_2+2: сравниваемое время

Слова S_2...S_2+2 содержат значение сравниваемого времени.

Слова S_2 ... S_2+2 должны находиться в одной области данных.



Характеристики операндов

Область	C	S_1	S_2
Область CIO	CIO 0...CIO 6143	CIO 0...CIO 6141	CIO 0...CIO 6142
Рабочая область	W0...W511	W0...W509	W0...W510
Область битов хранения	H0...H511	H0...H509	H0...H510
Область вспомогательных битов	A448...A959	A0...A957	A0...A958
Область таймеров	T0000...T4095	T0000...T4093	T0000...T4094
Область счетчиков	C0000...C4095	C0000...C4093	C0000...C4094
Область DM	D0...D32767	D0...D32765	D0...D32766
Косвенные адреса DM в двоичном формате	---	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	---	*D0...*D32767	
Константы	См. на предыдущей странице.	См. на предыдущей странице.	---
Регистры данных	---		
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…-2048…+2047 ,IR15 DR0…DR15, IR0…IR15 ,IR0+(++)…,IR15+(++) ,–(– –)IR0…, –(– –)IR15		

Описание

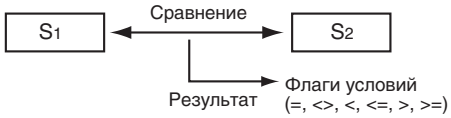
Команда сравнения значений времени сравнивает немаскированные значения (соответствующий бит C в состоянии «0») текущего времени, указанные операндами S_1...S_1+2, со значениями времени, указанными в S_2...S_2+2, и включает условие выполнения, если условие сравнения выполнено. Одновременно результат сравнения отражается во флагах арифметических операций (=, <>, <, <=, >, >=).

Доступны шесть видов сравнения, три способа ввода команды: всего 18 возможных комбинаций.

Значения времени, замаскированные в управляющем слове (C), при сравнении не учитываются.

В следующей таблице показаны состояния флагов для каждого результата сравнения.

Результат	Состояние флага					
	=	<>	<	<=	>	>=
$S_1 = S_2$	ВКЛ	ВЫКЛ	ВЫКЛ	ВКЛ	ВЫКЛ	ВКЛ
$S_1 > S_2$	ВЫКЛ	ВКЛ	ВЫКЛ	ВЫКЛ	ВКЛ	ВКЛ
$S_1 < S_2$	ВЫКЛ	ВКЛ	ВКЛ	ВКЛ	ВЫКЛ	ВЫКЛ

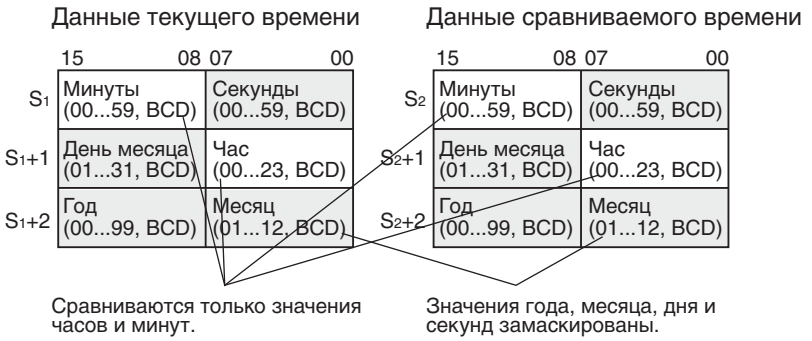


Маскировка значений времени

Отдельные значения времени можно маскировать и тем самым исключать из операции сравнения. Чтобы замаскировать значение времени, переключите в состояние «1» соответствующий бит управляющего слова (C). Биты 00...05 слова C маскируют, соответственно, секунды, минуты, часы, день, месяц и год.

Пример:

Если C = 39 hex, 6 младших битов имеют значения 111001 (год=1, месяц=1, день=1, часы=0, минуты=0, секунды=1), так что сравниваются только часы и минуты. Такую маску можно использовать для ежедневного выполнения определенной операции в определенное время (заданное часами и минутами).



Описанные ранее команды сравнения данных сравнивают данные блоками по 16 битов. В отличие от них команды сравнения значений времени сравнивают значения времени длиной в 8 битов.

В следующей таблице показана структура области внутреннего календаря/часов модуля ЦПУ.

Адреса	Содержание
A351.00...A351.07	Секунды (00...59, BCD)
A351.08...A351.15	Минуты (00...59, BCD)

Адреса	Содержание
A352.00...A352.07	Час (00...23, BCD)
A352.08...A352.15	День месяца (01...31, BCD)
A353.00...A353.07	Месяц (01...12, BCD)
A353.08...A353.15	Год (00...99, BCD)

Содержимое календаря/часов можно задать в CX-Programmer, а также с помощью команды DATE(735) или команды FINS «ЗАПИСАТЬ В ЧАСЫ» (0702 hex).

Краткий обзор команд сравнения значений времени

В приведенной ниже таблице перечислены коды функций, мнемонические коды, названия и назначение 18 команд сравнения значений времени.

Код	Мнемоническое обозначение	Название	Функция
341	LD= DT	ЗАГРУЗКА РАВНО	Истина, если S1 = S2
	AND=DT	И РАВНО	
	OR=DT	ИЛИ РАВНО	
342	LD<>DT	ЗАГРУЗКА НЕ РАВНО	Истина, если S1 ≠ S2
	AND<>DT	И НЕ РАВНО	
	OR<>DT	ИЛИ НЕ РАВНО	
343	LD<DT	ЗАГРУЗКА МЕНЬШЕ	Истина, если S1 < S2
	AND<DT	И МЕНЬШЕ	
	OR<DT	ИЛИ МЕНЬШЕ	
344	LD<=DT	ЗАГРУЗКА МЕНЬШЕ ИЛИ РАВНО	Истина, если S1 ≤ S2
	AND<=DT	И МЕНЬШЕ ИЛИ РАВНО	
	OR<=DT	ИЛИ МЕНЬШЕ ИЛИ РАВНО	
345	LD>DT	ЗАГРУЗКА БОЛЬШЕ	Истина, если S1 > S2
	AND>DT	И БОЛЬШЕ	
	OR>DT	ИЛИ БОЛЬШЕ	
346	LD>=DT	ЗАГРУЗКА БОЛЬШЕ ИЛИ РАВНО	Истина, если S1 ≥ S2
	AND>=DT	И БОЛЬШЕ ИЛИ РАВНО	
	OR>=DT	ИЛИ БОЛЬШЕ ИЛИ РАВНО	

Флаги

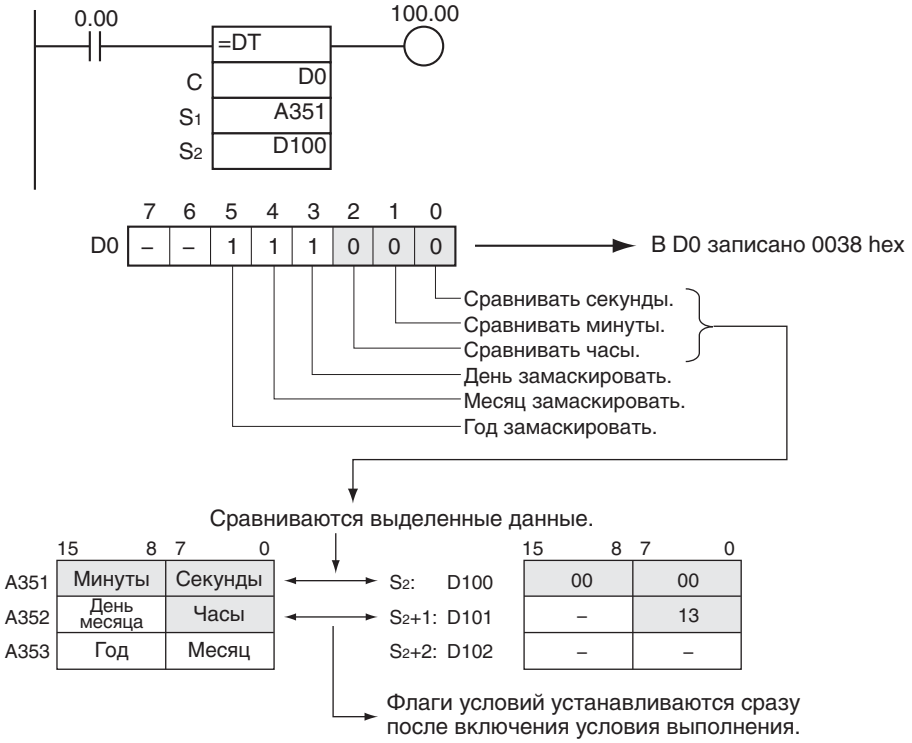
Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если включены все 6 битов маски (биты 00...05 слова C). Выключен во всех остальных случаях.
Флаг «Больше»	>	ВКЛ, если $S_1 > S_2$. Выключен во всех остальных случаях.
Флаг «Больше или равно»	> =	ВКЛ, если $S_1 \geq S_2$. Выключен во всех остальных случаях.
Флаг «Равно»	=	ВКЛ, если $S_1 = S_2$. Выключен во всех остальных случаях.
Флаг «Не равно»	=	ВКЛ, если $S_1 \neq S_2$. Выключен во всех остальных случаях.
Флаг «Меньше»	<	ВКЛ, если $S_1 < S_2$. Выключен во всех остальных случаях.
Флаг «Меньше или равно»	< =	ВКЛ, если $S_1 \leq S_2$. Выключен во всех остальных случаях.

Меры предосторожности

Команда сравнения времени не может быть последней командой цепи. Между ней и правой шиной должна находиться как минимум еще одна команда.

Пример

Если бит CIO 0.00 включен, то при значении времени 13:00:00 включится бит CIO 100.00. Содержимое слов A351...A353 (данные внутреннего календаря/часов модуля ЦПУ) используется в качестве значений текущего времени, а содержимое слов D100...D102 используется в качестве сравниваемого времени. Значения года, месяца и дня замаскированы, так что сравниваются только часы, минуты и секунды.



3-6-3 СРАВНИТЬ: CMP(020)

Назначение

Сравнение двух двоичных значений без знака (констант и/или содержимого указанных слов) и вывод результата во флаги арифметических операций во вспомогательной области.

Символ РКС



S1: Сравниваемое значение 1

S2: Сравниваемое значение 2

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	CMP(020)
	Однократное выполнение по положительному фронту	Не предусмотрено
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		!CMP(020)

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

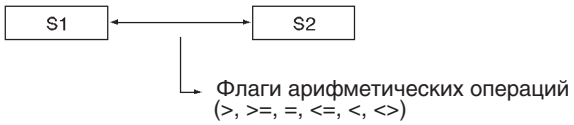
Характеристики операндов

Область	S_1	S_2
Область CIO	CIO 0...CIO 6143	
Рабочая область	W0...W511	
Область битов хранения	H0...H511	
Область вспомогательных битов	A0...A959	
Область таймеров	T0000...T4095	
Область счетчиков	C0000...C4095	
Область DM	D0...D32767	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Константы	#0000...#FFFF (двоичн.) &0...&65535 (десятичное без знака)	
Регистры данных	DR0...DR15	
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…-2048…+2047 ,IR15 DR0…DR15, IR0…IR15 ,IR0+(++)…,IR15+(++) ,–(– –)IR0…, –(– –)IR15	

Описание

Команда CMP(020) сравнивает два двоичных значения без знака, указанные в S_1 и S_2, и выводит результат во флаги арифметических операций («Больше», «Больше или равно», «Равно», «Меньше или равно», «Меньше» и «Не равно») во вспомогательной области.

Сравнение двоичных значений без знака



Состояния флагов условий

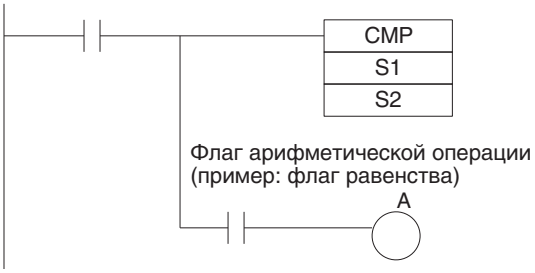
В следующей таблице показаны состояния флагов арифметических операций после выполнения команды CMP(020). («---» означает, что флаг может быть равновероятно включен или выключен).

Результат CMP(020)	Состояние флага					
	>	> =	=	< =	<	< >
$S_1 > S_2$	ВКЛ	ВКЛ	ВЫКЛ	ВЫКЛ	ВЫКЛ	ВКЛ
$S_1 = S_2$	ВЫКЛ	ВКЛ	ВКЛ	ВКЛ	ВЫКЛ	ВЫКЛ
$S_1 < S_2$	ВЫКЛ	ВЫКЛ	ВЫКЛ	ВКЛ	ВКЛ	ВКЛ

Использование результата CMP(020) в программе

Результат выполнения команды CMP(020) отражается во флагах арифметических операций. Управление требуемым выходом или последней командой цепи должно осуществляться ветвью, подключенной к тому же входному условию, которое управляет выполнением команды CMP(020), как показано на следующей схеме. В данном случае флаг равенства и выход A будут включены, когда $S_1 = S_2$.

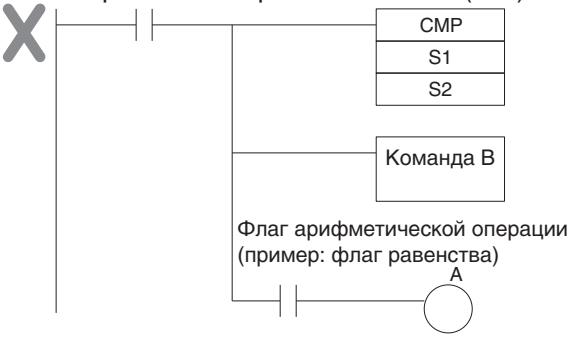
Правильное применение CMP(020)



Использование результата CMP(020) в программе

Не следует использовать другие команды между командой CMP(020) и командой, управляемой флагом арифметической операции, так как другая команда может изменить состояние этого флага. В этом случае результаты выполнения команды B могут изменить результаты команды CMP(020).

Неправильное применение CMP(020)



Если в S_1 и/или S_2 указаны слова, назначенные для внешних входов, команда может применяться с модификатором «мгновенное обновление» (!CMP(020)). При выполнении команды !CMP(020) сначала будут обновлены состояния входов, указанных в S_1 и/или S_2, и сравнение будет выполнено с использованием обновленных значений.

Флаги

Название	Обозначение в CX-Programmer	Описание работы
Флаг «Больше»	P_GT	ВКЛ, если $S_1 > S_2$. Выключен во всех остальных случаях.
Флаг «Больше или равно»	P_GE	ВКЛ, если $S_1 \geq S_2$. Выключен во всех остальных случаях.
Флаг «Равно»	P_EQ	ВКЛ, если $S_1 = S_2$. Выключен во всех остальных случаях.
Флаг «Не равно»	P_NE	ВКЛ, если $S_1 \neq S_2$. Выключен во всех остальных случаях.

Название	Обозначение в CX-Programmer	Описание работы
Флаг «Меньше»	P_LT	ВКЛ, если $S_1 < S_2$. Выключен во всех остальных случаях.
Флаг «Меньше или равно»	P_LE	ВКЛ, если $S_1 \leq S_2$. Выключен во всех остальных случаях.

Меры предосторожности

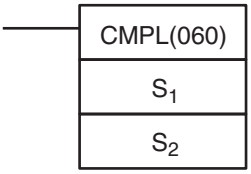
Не используйте другие команды между командой CMP(020) и входным условием, которое использует результат CMP(020), так как другая команда может изменить состояние флагов арифметических операций.

3-6-4 СРАВНИТЬ ДВОЙНЫЕ СЛОВА: CMPL(060)

Назначение

Сравнение двух двойных двоичных слов без знака (констант и/или содержимого указанных слов) и вывод результата во флаги арифметических операций во вспомогательной области.

Символ РКС



S1: Сравниваемое значение 1

S2: Сравниваемое значение 2

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	CMPL(060)
	Однократное выполнение по положительному фронту	Не предусмотрено
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

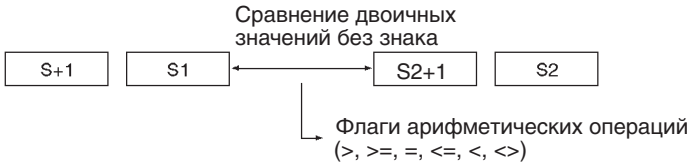
Характеристики операндов

Область	S_1	S_2
Область CIO	CIO 0...CIO 6142	
Рабочая область	W0...W510	
Область битов хранения	H0...H510	
Область вспомогательных битов	A0...A958	
Область таймеров	T0000...T4094	
Область счетчиков	C0000...C4094	
Область DM	D0...D32766	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Константы	#00000000...#FFFFFFFF (двоичное) &0...&4294967295 (десятичное без знака)	
Регистры данных	---	

Область	S_1	S_2
Регистры указателей	IR0...IR15	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…-2048…+2047 ,IR15 DR0…DR15, IR0…IR15 ,IR0+(++)…,IR15+(++) ,–(– –)IR0…, –(– –)IR15	

Описание

Команда CMPL(060) сравнивает двоичные значения без знака, указанные в S_1 +1, S_1 и S_2+1, S_2, и выводит результат во флаги арифметических операций («Больше», «Больше или равно», «Равно», «Меньше или равно», «Меньше» и «Не равно») во вспомогательной области.



Состояния флагов арифметических операций

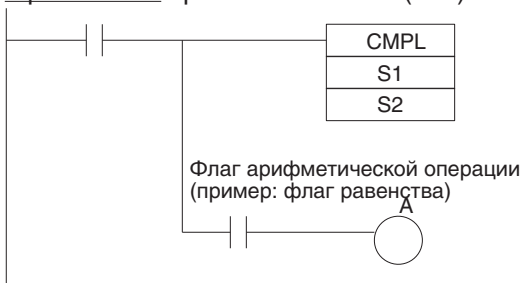
В следующей таблице показаны состояния флагов арифметических операций после выполнения команды CMPL(060). («---» означает, что флаг может быть равновероятно включен или выключен).

Результат CMPL(060)	Состояние флага					
	>	> =	=	< =	<	< >
S_1 +1, S_1 > S_2+1, S_2	ВКЛ	ВКЛ	ВЫКЛ	ВЫКЛ	ВЫКЛ	ВКЛ
S_1+1, S_1 = S_2+1, S_2	ВЫКЛ	ВКЛ	ВКЛ	ВКЛ	ВЫКЛ	ВЫКЛ
S_1 +1, S_1 < S_2+1, S_2	ВЫКЛ	ВЫКЛ	ВЫКЛ	ВКЛ	ВКЛ	ВКЛ

Использование результата CMPL(060) в программе

Результат выполнения команды CMPL(060) отражается во флагах арифметических операций. Управление требуемым выходом или последней командой цепи должно осуществляться ветвью, подключенной к тому же входному условию, которое управляет выполнением команды CMPL(060), как показано на следующей схеме. Здесь флаг равенства и выход A будут включены, когда S_1 +1, S_1 = S_2+1, S_2.

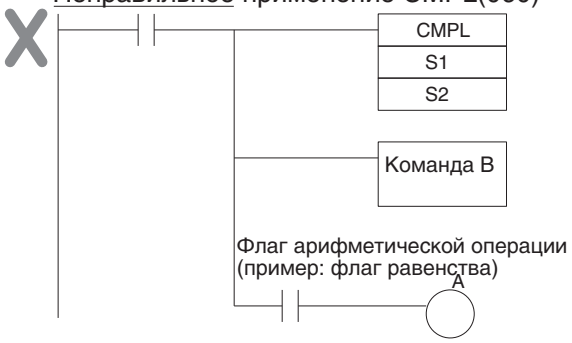
Правильное применение CMPL(060)



Использование результата CMPL(060) в программе

Не следует использовать другие команды между командой CMPL(060) и командой, управляемой флагом арифметической операции, так как другая команда может изменить состояние этого флага. В этом случае результаты выполнения команды B могут изменить результаты команды CMPL(060).

Неправильное применение CMPL(060)



Флаги

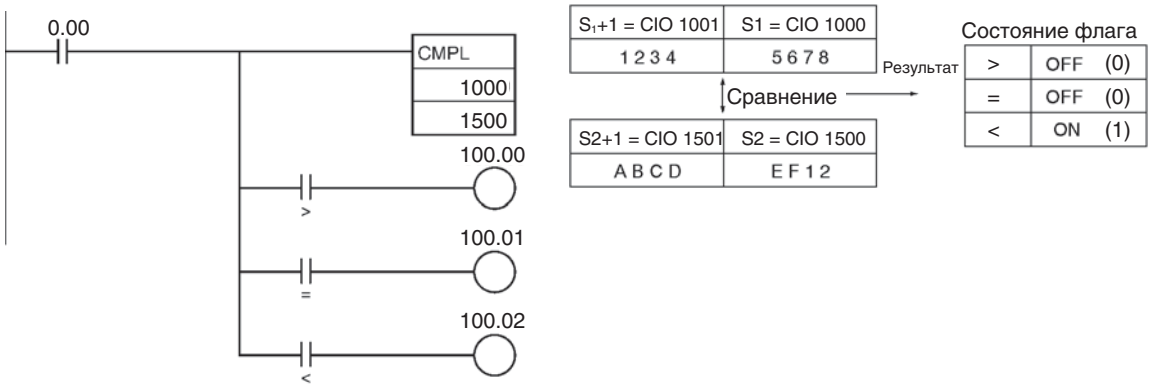
Название	Обозначение в CX-Programmer	Описание работы
Флаг «Больше»	P_GT	ВКЛ, если S_1 +1, $S_1 > S_2$+1, S_2. Выключен во всех остальных случаях.
Флаг «Больше или равно»	P_GE	ВКЛ, если S_1 +1, $S_1 \geq S_2$+1, S_2. Выключен во всех остальных случаях.
Флаг «Равно»	P_EQ	ВКЛ, если S_1 +1, $S_1 = S_2$+1, S_2. Выключен во всех остальных случаях.
Флаг «Не равно»	P_NE	ВКЛ, если S_1 +1, $S_1 \neq S_2$+1, S_2. Выключен во всех остальных случаях.
Флаг «Меньше»	P_LT	ВКЛ, если S_1 +1, $S_1 < S_2$+1, S_2. Выключен во всех остальных случаях.
Флаг «Меньше или равно»	P_LE	ВКЛ, если S_1 +1, $S_1 \leq S_2$+1, S_2. Выключен во всех остальных случаях.

Меры предосторожности

Не используйте другие команды между командой CMPL(060) и входным условием, которое использует результат CMPL(060), так как другая команда может изменить состояние флагов арифметических операций.

Пример

Если в следующем примере бит CIO 0.00 включен, 8-разрядное двоичное значение без знака, хранящееся в словах CIO 1001 и CIO 1000, сравнивается с 8-разрядным двоичным значением без знака, хранящимся в CIO 1501 и CIO 1500, результат выводится во флаги арифметических операций. Результаты, записанные во флаги «Больше», «Равно» и «Меньше», сохраняются по адресам CIO 100.00 (Больше), CIO 100.01 (Равно) и CIO 100.02 (Меньше).

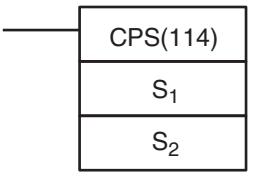


3-6-5 СРАВНИТЬ ДВОИЧНЫЕ ЗНАЧЕНИЯ СО ЗНАКОМ: CPS(114)

Назначение

Сравнение двух двоичных значений со знаком (констант и/или содержимого указанных слов) и вывод результата во флаги арифметических операций во вспомогательной области.

Символ РКС



S1: Сравниваемое значение 1

S2: Сравниваемое значение 2

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	CPS(114)
	Однократное выполнение по положительному фронту	Не предусмотрено
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		!CPS(114)

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

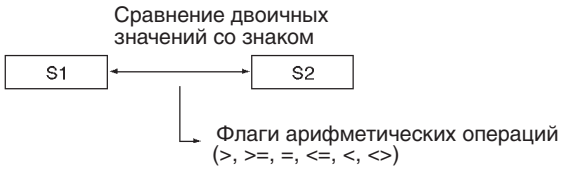
Характеристики операндов

Область	S_1	S_2
Область CIO	CIO 0...CIO 6143	
Рабочая область	W0...W511	
Область битов хранения	H0...H511	

Область	S_1	S_2
Область вспомогательных битов	A0...A959	
Область таймеров	T0000...T4095	
Область счетчиков	C0000...C4095	
Область DM	D0...D32767	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Константы	#0000...#FFFF (двоичн.) –32768...0...32767 (десятичное число со знаком)	
Регистры данных	DR0...DR15	
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…-2048…+2047 ,IR15 DR0…DR15, IR0…IR15 ,IR0+(++)…,IR15+(++) ,–(– –)IR0…, –(– –)IR15	

Описание

Команда CPS(114) сравнивает двоичные значения со знаком, указанные в S_1 и S_2, и выводит результат во флаги арифметических операций («Больше», «Больше или равно», «Равно», «Меньше или равно», «Меньше» и «Не равно») во вспомогательной области.



Примечание.

Команда CPS(114) оперирует данными в операндах S_1 и S_2 как двоичными значениями со знаком, лежащими в диапазоне 8000...7FFF (-32768...32767 в десятичном формате).

Состояния флагов арифметических операций

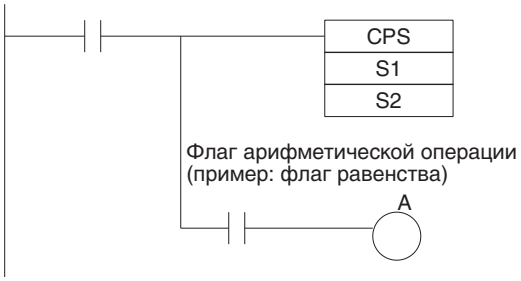
В следующей таблице показаны состояния флагов арифметических операций после выполнения команды CPS(114). («---» означает, что флаг может быть равновероятно включен или выключен).

Результат CPS(114)	Состояние флага					
	>	> =	=	< =	<	< >
$S_1 > S_2$	ВКЛ	ВКЛ	ВЫКЛ	ВЫКЛ	ВЫКЛ	ВКЛ
$S_1 = S_2$	ВЫКЛ	ВКЛ	ВКЛ	ВКЛ	ВЫКЛ	ВЫКЛ
$S_1 < S_2$	ВЫКЛ	ВЫКЛ	ВЫКЛ	ВКЛ	ВКЛ	ВКЛ

Использование результата CPS(114) в программе

Результат выполнения команды CPS(114) отражается во флагах арифметических операций. Управление требуемым выходом или последней командой цепи должно осуществляться ветвью, подключенной к тому же входному условию, которое управляет выполнением команды CPS(114), как показано на следующей схеме. В данном случае флаг равенства и выход A будут включены, когда $S_1 = S_2$.

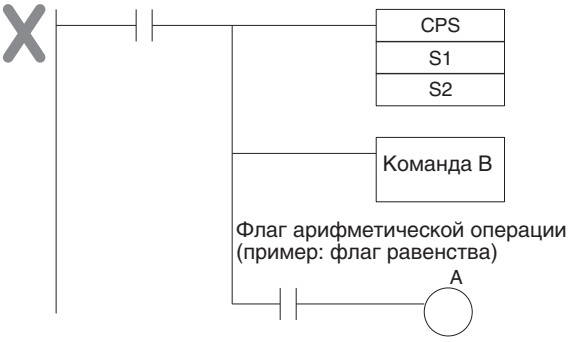
Правильное применение CPS(114)



Использование результата CPS(114) в программе

Не следует использовать другие команды между командой CPS(114) и командой, управляемой флагом арифметической операции, так как другая команда может изменить состояние этого флага. В этом случае результаты выполнения команды B могут изменить результаты команды CPS(114).

IНеправильное применение CPS(114)



Если в S_1 и/или S_2 указаны слова, назначенные для внешних входов, команда может применяться с модификатором «мгновенное обновление» (!CPS(114)). При выполнении команды !CPS(114) сначала будут обновлены состояния входов, указанных в S_1 и/или S_2, и сравнение будет выполнено с использованием обновленных значений.

Флаги

Название	Обо-значение	Описание работы
Флаг «Больше»	>	ВКЛ, если $S_1 > S_2$. ВЫКЛ в любых других случаях.
Флаг «Больше или равно»	> =	ВКЛ, если $S_1 \geq S_2$. ВЫКЛ в любых других случаях.
Флаг «Равно»	=	ВКЛ, если $S_1 = S_2$. ВЫКЛ в любых других случаях.
Флаг «Не равно»	<>	ВКЛ, если $S_1 \neq S_2$. ВЫКЛ в любых других случаях.

Название	Обозначение	Описание работы
Флаг «Меньше»	<	ВКЛ, если $S_1 < S_2$. ВЫКЛ в любых других случаях.
Флаг «Меньше или равно»	< =	ВКЛ, если $S_1 \leq S_2$. ВЫКЛ в любых других случаях.

Меры предосторожности

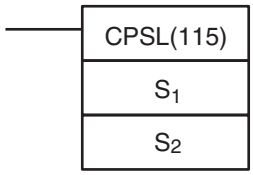
Не используйте другие команды между командой CPS(114) и входным условием, которое использует результат CPS(114), так как другая команда может изменить состояние флагов арифметических операций.

3-6-6 СРАВНИТЬ ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ: CPSL(115)

Назначение

Сравнение двух двойных двоичных слов со знаком (констант и/или содержимого указанных слов) и вывод результата во флаги арифметических операций во вспомогательной области.

Символ РКС



S1: Сравниваемое значение 1

S2: Сравниваемое значение 2

Варианты выполнения

Варианты выполнения		CPSL(115)
	Выполнение в каждом цикле при включенном условии выполнения	
	Однократное выполнение по положительному фронту	Не предусмотрено
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

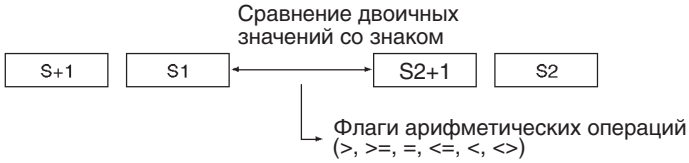
Характеристики операндов

Область	S_1	S_2
Область CIO	CIO 0...CIO 6142	
Рабочая область	W0...W510	
Область битов хранения	H0...H510	
Область вспомогательных битов	A0...A958	
Область таймеров	T0000...T4094	
Область счетчиков	C0000...C4094	
Область DM	D0...D32766	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Константы	#00000000...#FFFFFFFF (двоичное) -2147483648...0...2147483647 (десятичное число со знаком)	
Регистры данных	---	

Область	S_1	S_2
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…-2048…+2047 ,IR15 DR0…DR15, IR0…IR15 ,IR0+(++)…,IR15+(++) ,–(– –)IR0…, –(– –)IR15	

Описание

Команда CPSL(115) сравнивает двойные двоичные слова со знаком, указанные в S_1 +1, S_1 и S_2+1, S_2, и выводит результат во флаги арифметических операций («Больше», «Больше или равно», «Равно», «Меньше или равно», «Меньше» и «Не равно») во вспомогательной области.



Примечание.

Команда CPSL(115) оперирует данными в операндах S_1 и S_2 как двойными двоичными словами со знаком, значения которых лежат в диапазоне 8000 0000...7FFF FFFF (-2147483648...2147483647 в десятичном формате).

Состояния флагов арифметических операций

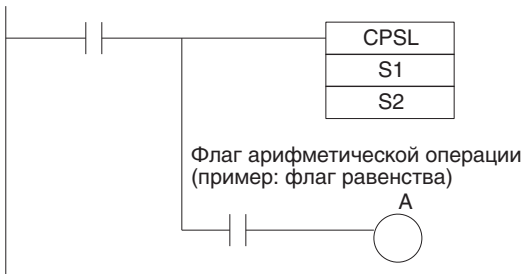
В следующей таблице показаны состояния флагов арифметических операций после выполнения команды CPSL(115). («---» означает, что флаг может быть равновероятно включен или выключен).

Результат CPSL(115)	Состояние флага					
	>	> =	=	< =	<	< >
S_1 +1, S_1 > S_2+1, S_2	ВКЛ	ВКЛ	ВЫКЛ	ВЫКЛ	ВЫКЛ	ВКЛ
S_1+1, S_1 = S_2+1, S_2	ВЫКЛ	ВКЛ	ВКЛ	ВКЛ	ВЫКЛ	ВЫКЛ
S_1 +1, S_1 < S_2+1, S_2	ВЫКЛ	ВЫКЛ	ВЫКЛ	ВКЛ	ВКЛ	ВКЛ

Использование результата CPSL(115) в программе

Результат выполнения команды CPSL(115) отражается во флагах арифметических операций. Управление требуемым выходом или последней командой цепи должно осуществляться ветвью, подключенной к тому же входному условию, которое управляет выполнением команды CPSL(115), как показано на следующей схеме. Здесь флаг равенства и выход A будут включены, когда S_1 +1, S_1 = S_2+1, S_2.

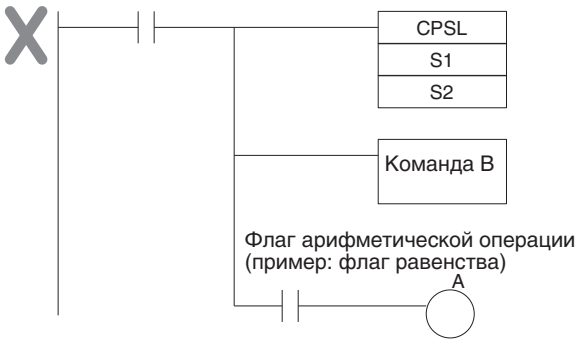
Правильное применение CPSL(115)



Использование результата CPSL(115) в программе

Не следует использовать другие команды между командой CPSL(115) и командой, управляемой флагом арифметической операции, так как другая команда может изменить состояние этого флага. В этом случае результаты выполнения команды B могут изменить результаты команды CPSL(115).

Неправильное применение CPSL(115)



Флаги

Название	Обо-значение	Описание работы
Флаг «Больше»	>	ВКЛ, если S_1 +1, $S_1 > S_2$+1, S_2. Выключен во всех остальных случаях.
Флаг «Больше или равно»	> =	ВКЛ, если S_1 +1, $S_1 \geq S_2$+1, S_2. Выключен во всех остальных случаях.
Флаг «Равно»	=	ВКЛ, если S_1 +1, $S_1 = S_2$+1, S_2. Выключен во всех остальных случаях.
Флаг «Не равно»	=	ВКЛ, если S_1 +1, $S_1 \neq S_2$+1, S_2. Выключен во всех остальных случаях.
Флаг «Меньше»	<	ВКЛ, если S_1 +1, $S_1 < S_2$+1, S_2. Выключен во всех остальных случаях.
Флаг «Меньше или равно»	< =	ВКЛ, если S_1 +1, $S_1 \leq S_2$+1, S_2. Выключен во всех остальных случаях.

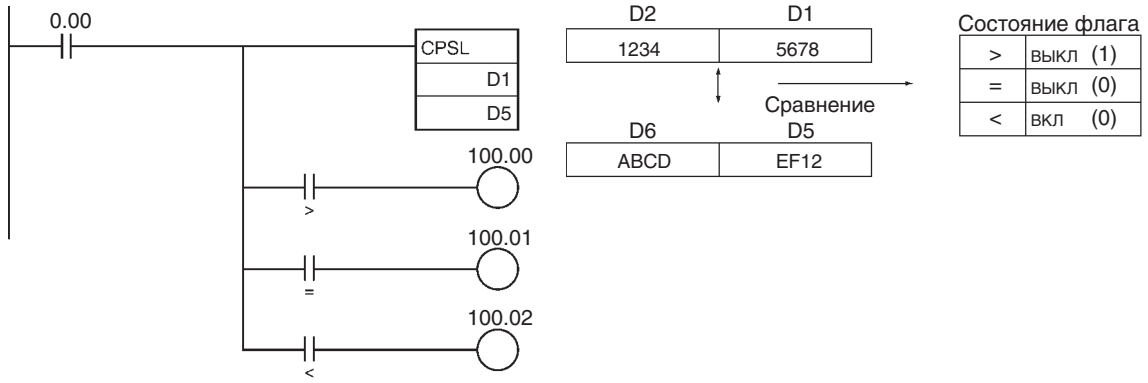
Меры предосторожности

Не используйте другие команды между командой CPSL(115) и входным условием, которое использует результат CPSL(115), так как другая команда может изменить состояние флагов арифметических операций.

Пример

Если в следующем примере бит CIO 0.00 включен, 8-разрядное двоичное значение со знаком, хранящееся в D1 и D2, сравнивается с 8-разрядным двоичным значением со знаком, хранящимся в D6 и D5, а результат выводится во флаги арифметических операций.

- Если содержимое слов D2 и D1 больше содержимого слов D6 и D5, включаются флаг «Больше» и бит CIO 100.00.
- Если содержимое слов D2 и D1 равно содержимому слов D6 и D5, включаются флаг «Равно» и бит CIO 100.01.
- Если содержимое слов D2 и D1 меньше содержимого слов D6 и D5, включаются флаг «Меньше» и бит CIO 100.02.

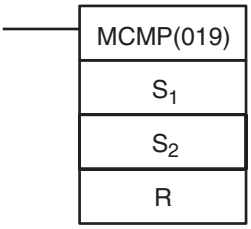


3-6-7 СРАВНИТЬ НЕСКОЛЬКО ЗНАЧЕНИЙ: MCMP(019)

Назначение

Сравнение 16 последовательных слов с другими 16 последовательными словами и включение бита слова результата, соответствующего словам, содержимое которых **не равно**.

Символ РКС



S1: Первое слово набора 1

S2: Первое слово набора 2

R: Слово результата

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	MCMP(019)
	Однократное выполнение по положительному фронту	@MCMP(019)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

S_1: Первое слово набора 1

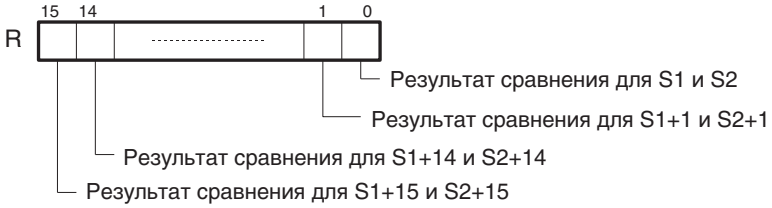
Указывает начало первого диапазона из 16 слов. S_1 и S_1+15 должны находиться в одной области данных.

S_2: Первое слово набора 2

Указывает начало второго диапазона из 16 слов. S_2 и S_2+15 должны находиться в одной области данных.

R: Слово результата

Каждый бит слова R содержит результат сравнения соответствующих слов двух групп. Бит n слова R (n = 00...15) содержит результат сравнения слов S_1+n и S_2+n.



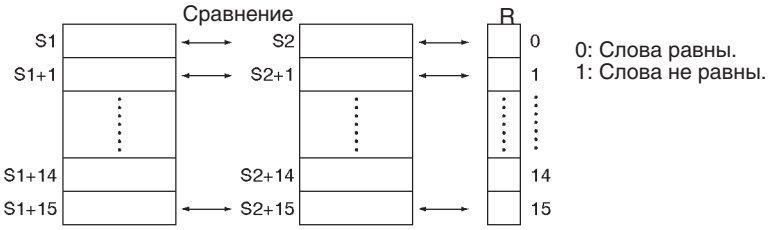
Характеристики операндов

Область	S_1	S_2	R
Область CIO	CIO 0...CIO 6128		CIO 0...CIO 6143
Рабочая область	W0...W496		W0...W511
Область битов хранения	H0...H496		H0...H511
Область вспомогательных битов	A0...A944		A448...A959
Область таймеров	T0000...T4080		T0000...T4095
Область счетчиков	C0000...C4080		C0000...C4095
Область DM	D0...D32752		D0...D32767
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Константы	---		
Регистры данных	---		DR0...DR15
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…-2048…+2047 ,IR15 DR0…DR15, IR0…IR15 ,IR0+(++)…,IR15+(++) ,–(– –)IR0…, –(– –)IR15		

Описание

Команда MCMP(019) сравнивает содержимое 16 слов диапазона S_1...S_1+15 с содержимым 16 слов диапазона S_2...S_2+15 и включает соответствующий бит слова R, если содержимое какой либо пары слов **не равно**.

Содержимое слова S_1 сравнивается с содержимым слова S_2, содержимое S_1+1 — с содержимым S_2+1, ..., содержимое S_1+15 — с содержимым S_2+15. Бит n в слове R выключается, если содержимое слова S_1+n равно содержимому слова S_2+n; бит n в слове R включается, если содержимое этих слов не равно. Если содержимое всех 16 пар слов совпадает, по завершении выполнения команды включается флаг «Равно».

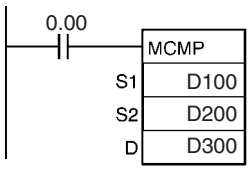


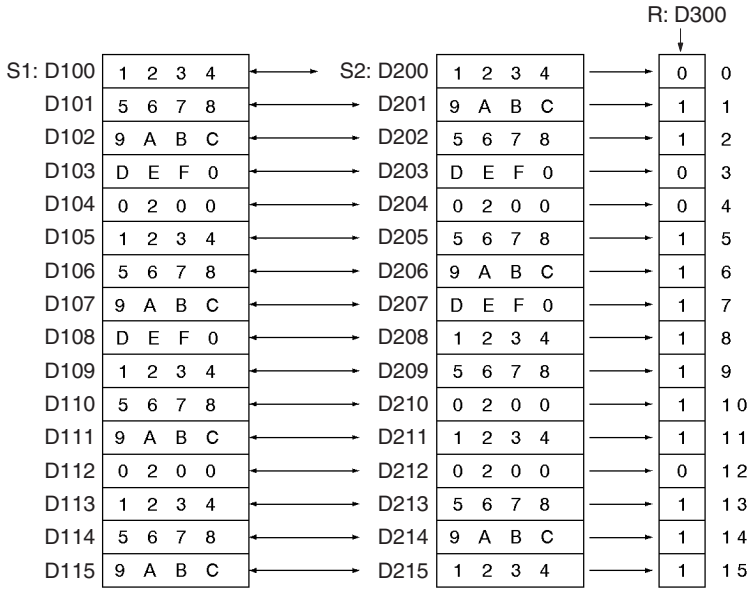
Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	ВЫКЛ
Флаг равенства	=	ВКЛ, если слово результата содержит значение 0000. (Два набора по 16 слов полностью совпадают.) Выключен во всех остальных случаях.

Пример

Если в следующем примере бит CIO 0.00 включен, команда MCMP(019) пословно сравнивает слова D100...D115 со словами D200...D215 и устанавливает соответствующие биты в слове D300, если сравниваемые слова **не равны**.



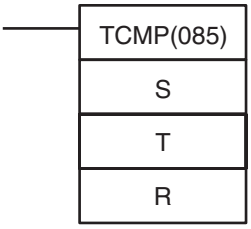


3-6-8 СРАВНИТЬ С ТАБЛИЦЕЙ: TCMP(085)

Назначение

Сравнение исходного значения с содержимым 16 последовательных слов и установка соответствующего бита в слове результата, если содержимое слов **равно**.

Символ РКС



S: Исходное значение

T: Первое слово таблицы

R: Слово результата

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	TCMP(085)
	Однократное выполнение по положительному фронту	@TCMP(085)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограм-мы	Задачи обработки прерываний
OK	OK	OK	OK

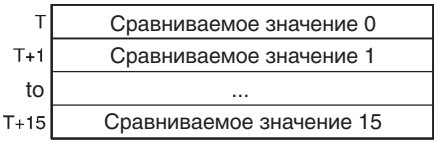
Операнды

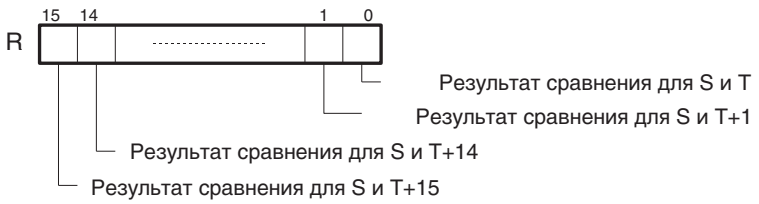
T: Первое слово таблицы

Указывает первое слово таблицы из 16 слов. Слова T...T+15 должны принадлежать одной области.

R: Слово результата

Каждый бит слова R содержит результат сравнения S с соответствующим словом таблицы. Бит n слова R (n = 00...15) содержит результат сравнения S с T+n.





Характеристики операндов

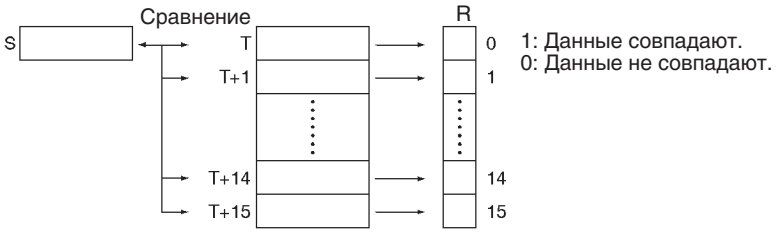
Область	S	T	R
Область CIO	CIO 0...CIO 6143	CIO 0...CIO 6128	CIO 0...CIO 6143
Рабочая область	W0...W511	W0...W496	W0...W511
Область битов хранения	H0...H511	H0...H496	H0...H511
Область вспомогательных битов	A0...A959	A0...A944	A448...A959

Область	S	T	R
Область таймеров	T0000...T4095	T0000...T4080	T0000...T4095
Область счетчиков	C0000...C4095	C0000...C4080	C0000...C4095
Область DM	D0...D32767	D0...D32752	D0...D32767
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Константы	#0000...#FFFF (двоичн.) &0...&65535 (десятичное без знака) -32768...0...32767 (десятичное число со знаком)	---	
Регистры данных	DR0...DR15	---	DR0...DR15
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…-2048…+2047 ,IR15 DR0…DR15, IR0…IR15 ,IR0+(++)…,IR15+(++) ,–(– –)IR0…, –(– –)IR15		

Описание

Команда TCMP(085) сравнивает входные данные (S) с каждым из 16 слов таблицы T...T+15 и включает соответствующий бит в слове R, если значения **равны**. Бит n слова R устанавливается, если содержимое T+n равно S, и не устанавливается, если слова не равны.

Сначала S сравнивается с содержимым T, бит 00 слова R устанавливается, если эти слова равны, и не устанавливается, если они не равны; затем S сравнивается с содержимым T+1, бит 01 слова R устанавливается, если эти слова равны, и не устанавливается, если они не равны; ...; наконец, S сравнивается с содержимым T+15, бит 15 слова R устанавливается, если эти слова равны, и не устанавливается, если они не равны.

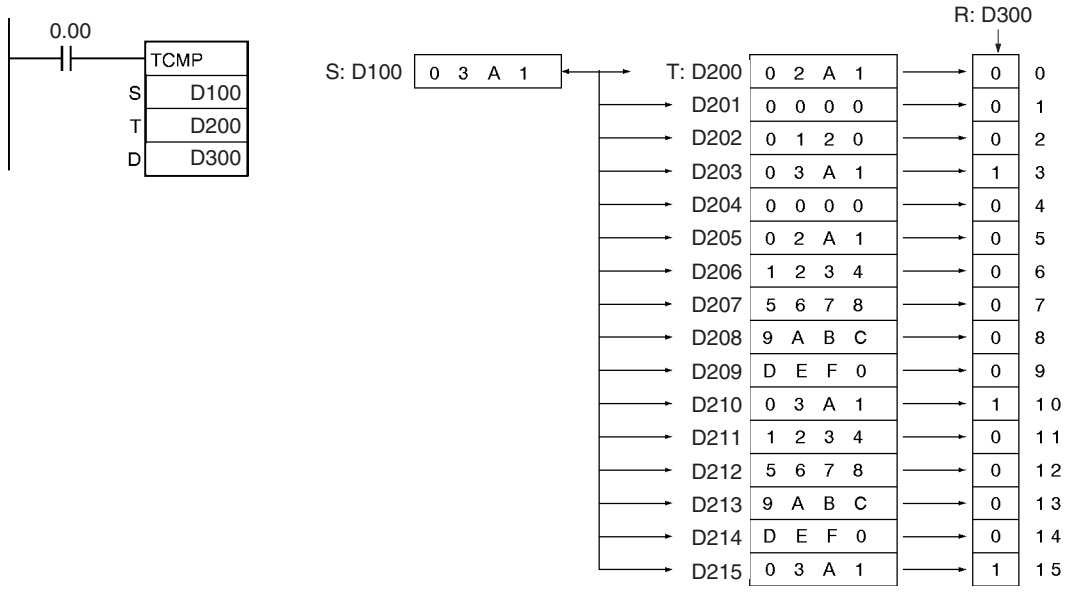


Флаги

Название	Обо-значение	Описание работы
Флаг ошибки	ER	ВЫКЛ
Флаг равенства	=	ВКЛ, если слово результата содержит значение 0000. (Ни одно из 16 слов таблицы не равно S.) Выключен во всех остальных случаях.

Пример

Если в следующем примере бит CIO 0.00 включен, команда TCMP(085) сравнивает содержимое слова D100 с содержимым слов D200...D215 и устанавливает соответствующие биты в слове D300, если содержимое пары слов совпадает, или не устанавливает, если содержимое пары слов не совпадает.

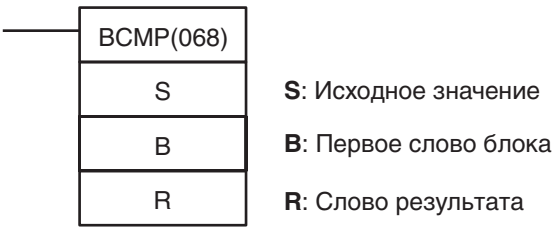


3-6-9 СРАВНИТЬ С ДИАПАЗОНАМИ: BCMP(068)

Назначение

Проверка принадлежности входного значения 16-ти диапазонам (заданным 16-тью нижними и 16-тью верхними границами) и установка соответствующего бита в слове результата, если входное значение принадлежат диапазону.

Символ РКС



S: Исходное значение

B: Первое слово блока

R: Слово результата

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	BCMP(068)
	Однократное выполнение по положительному фронту	@BCMP(068)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

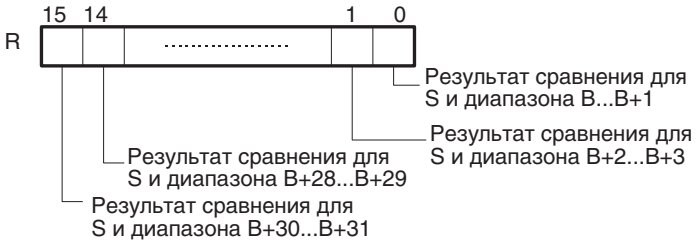
Операнды

B: Первое слово блока

Указывает первое слово блока из 32 слов (16 пар нижних/верхних границ). Слова B...B+31 должны принадлежать одной области.

R: Слово результата

Каждый бит слова R содержит результат сравнения S с одним из 16 диапазонов, заданных блоком из 32 слов. Бит n слова R (n = 00...15) содержит результат сравнения S с n-й парой слов.



Характеристики операндов

Область	S	B	R
Область CIO	CIO 0...CIO 6143	CIO 0...CIO 6112	CIO 0...CIO 6143
Рабочая область	W0...W511	W0...W480	W0...W511
Область битов хранения	H0...H511	H0...H480	H0...H511
Область вспомогательных битов	A0...A959	A0...A928	A448...A959
Область таймеров	T0000...T4095	T0000...T4064	T0000...T4095
Область счетчиков	C0000...C4095	C0000...C4064	C0000...C4095
Область DM	D0...D32767	D0...D32736	D0...D32767
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Константы	#0000...#FFFF (двоичн.) &0...&65535 (десятичное без знака) -32768...0...32767 (десятичное число со знаком)	---	
Регистры данных	DR0...DR15	---	DR0...DR15
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…-2048…+2047 ,IR15 DR0…DR15, IR0…IR15 ,IR0+(++)…,IR15+(++) ,–(– –)IR0…, –(– –)IR15		

Описание

Команда BCMP(068) сравнивает входное значение (S) с 16-тью диапазонами, заданными парами верхних и нижних границ, указанными в словах B...B+31. Первое слово каждой пары (B+2n) указывает нижнюю границу, а второе слово (B+2n+1) указывает верхнюю границу диапазона n (n = 0...15). Если S принадлежит какому-либо из этих диапазонов (включая равенство верхней или нижней границе), включается соответствующий бит слова R. Остальные биты в R будут выключены.

B	≤ S ≤	B+1	Бит 00 слова R
B+2	≤ S ≤	B+3	Бит 01 слова R
B+4	≤ S ≤	B+5	Бит 02 слова R
B+6	≤ S ≤	B+7	Бит 03 слова R
B+8	≤ S ≤	B+9	Бит 04 слова R
B+10	≤ S ≤	B+11	Бит 05 слова R
B+12	≤ S ≤	B+13	Бит 06 слова R
B+14	≤ S ≤	B+15	Бит 07 слова R
B+16	≤ S ≤	B+17	Бит 08 слова R
B+18	≤ S ≤	B+19	Бит 09 слова R
B+20	≤ S ≤	B+21	Бит 10 слова R
B+22	≤ S ≤	B+23	Бит 11 слова R
B+24	≤ S ≤	B+25	Бит 12 слова R
B+26	≤ S ≤	B+27	Бит 13 слова R
B+28	≤ S ≤	B+29	Бит 14 слова R
B+30	≤ S ≤	B+31	Бит 15 слова R

Например, бит 00 слова R устанавливается, если S принадлежит первому диапазону (B ≤ S ≤ B+1), бит 01 слова R устанавливается, если S принадлежит в второму диапазону (B+2 ≤ S ≤ B+3), ..., бит 15 слова R устанавливается, если S принадлежит пятнадцатому диапазону (B+30 ≤ S ≤ B+31). Все остальные биты слова R остаются сброшенными.

Флаги

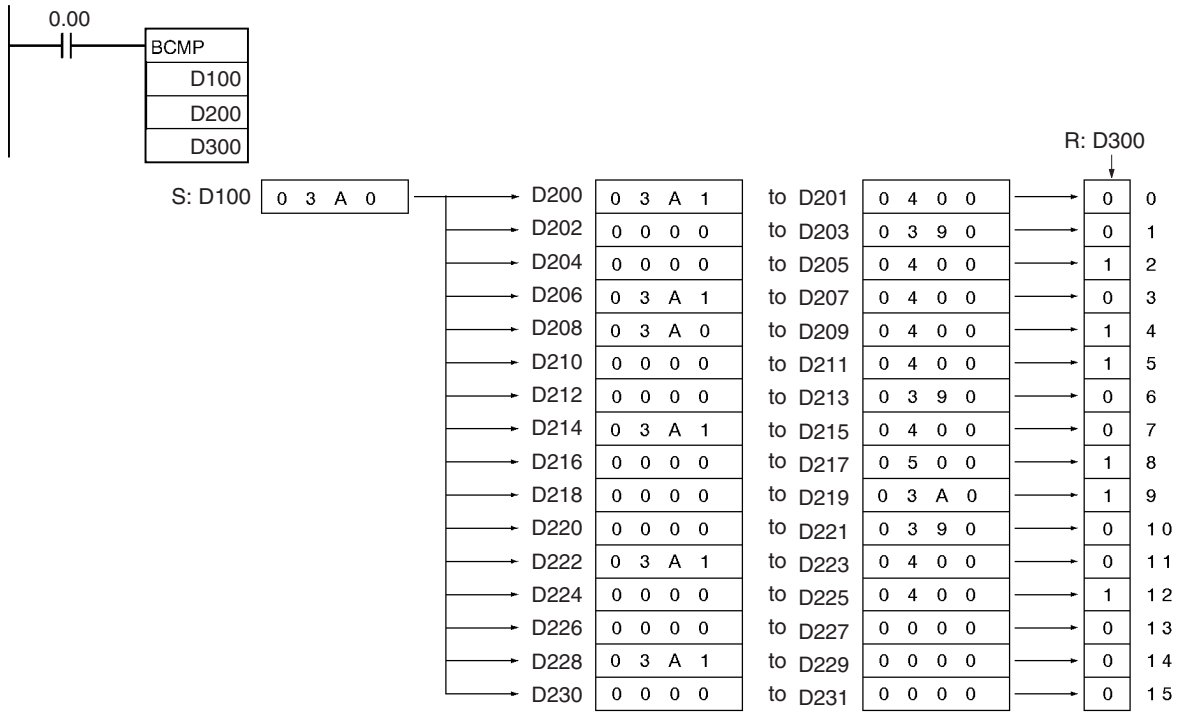
Название	Обо-значение	Описание работы
Флаг ошибки	ER	ВЫКЛ
Флаг равенства	=	ВКЛ, если слово результата содержит значение 0000. (S не принадлежит ни одному из 16 диапазонов.) Выключен во всех остальных случаях.

Меры предосторожности

Если нижняя граница больше верхней границы, ошибка не генерируется, но в соответствующий бит слова R записывается значение 0 (не принадлежит диапазону).

Пример

Если в следующем примере бит CIO 0.00 включен, команда BCMP(068) сравнивает содержимое слова D100 с 16-тью диапазонами, заданными словами D200...D231 (т. е. D200 и D201, D202 и D203 и т. п.), и устанавливает соответствующий бит слова D300, если S принадлежит диапазону, либо не устанавливает, если S не принадлежит диапазону.

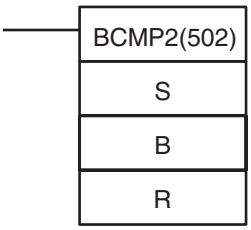


3-6-10 РАСШИРЕННОЕ СРАВНЕНИЕ С ДИАПАЗОНАМИ: BCMP2(502)

Назначение

Проверка принадлежности входного значения максимум 256-ти диапазонам (заданным 256-тью нижними и 256-тью верхними границами) и установка соответствующего бита в слове результата, если входное слово принадлежит соответствующему диапазону.

Символ РКС



S: Исходное значение

B: Первое слово блока

R: Первое слово результата

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	BCMP2(502)
	Однократное выполнение по положительному фронту	@BCMP2(502)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

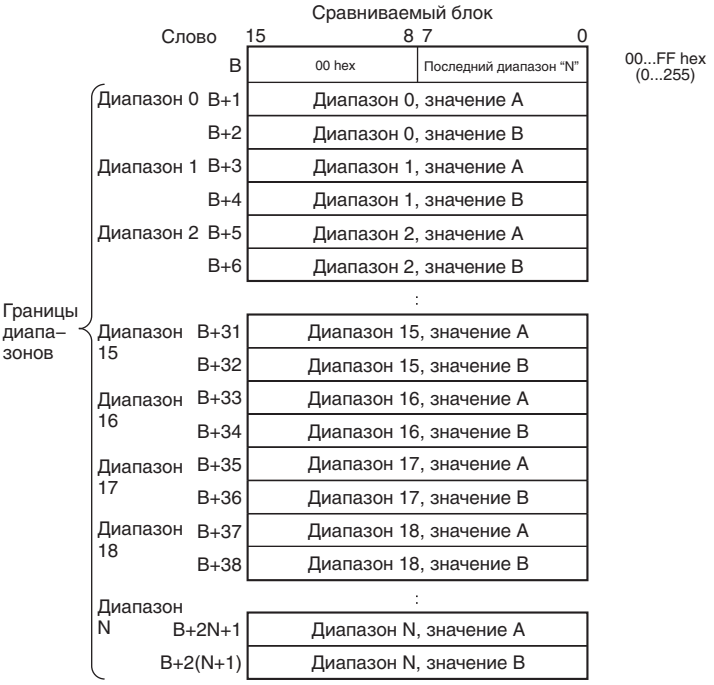
Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

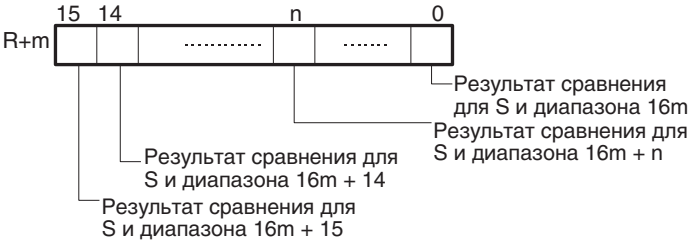
B: Первое слово блока

Указывает начало сравниваемого блока, содержащего до 513 слов (то есть до 256 пар нижних и верхних границ). Все слова должны принадлежать одной области.



R: Первое слово результата

Каждый бит каждого слова результата содержит результат сравнения S с одним из диапазонов, заданных сравниваемым блоком. В результате может быть не более 16 слов, т. е. m = 0...15.



Характеристики операндов

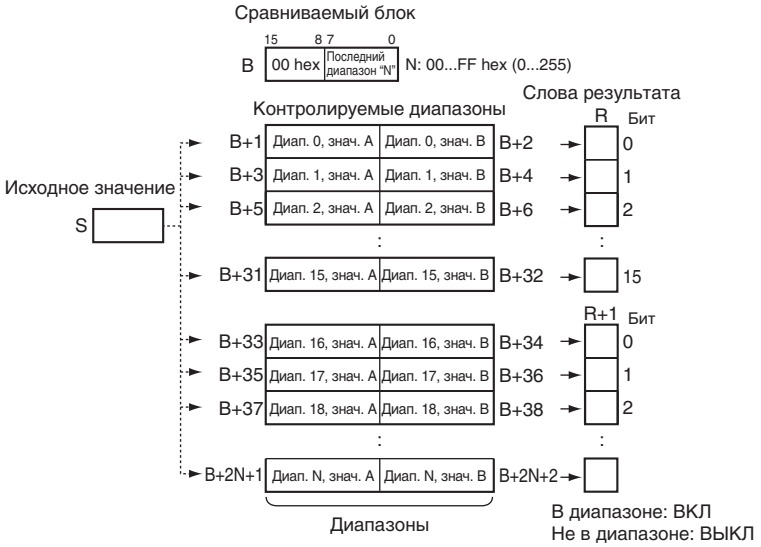
Область	S	B	R
Область CIO	CIO 0...CIO 6143		
Рабочая область	W0...W511		
Область битов хранения	H0...H511		
Область вспомогательных битов	A0...A959		A448...A959
Область таймеров	T0000...T4095		

Область	S	B	R
Область счетчиков	C0000...C4095		
Область DM	D0...D32767		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Константы	#0000...#FFFF (двоичн.) &0...&65535 (десятичное без знака) -32768...0...32767 (десятичное число со знаком)	---	
Регистры данных	DR0...DR15	---	
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…-2048…+2047 ,IR15 DR0…DR15, IR0…IR15 ,IR0+(++)…,IR15+(++) ,–(– –)IR0…, –(– –)IR15		

Описание

Команда BCMP2(502) сравнивает входное значение (S) с диапазонами, которые заданы парами нижних и верхних границ, указанных в сравниваемом блоке. Если S принадлежит какому-либо из этих диапазонов (включая равенство верхней или нижней границе), включаются соответствующие биты слов результата (R...R+15). Остальные биты в R будут выключены.

Количество диапазонов определяется значением N, указанным в младшем байте B. Допустимый диапазон значений N: 0...255. Старший байт B должен содержать значение 00 hex.

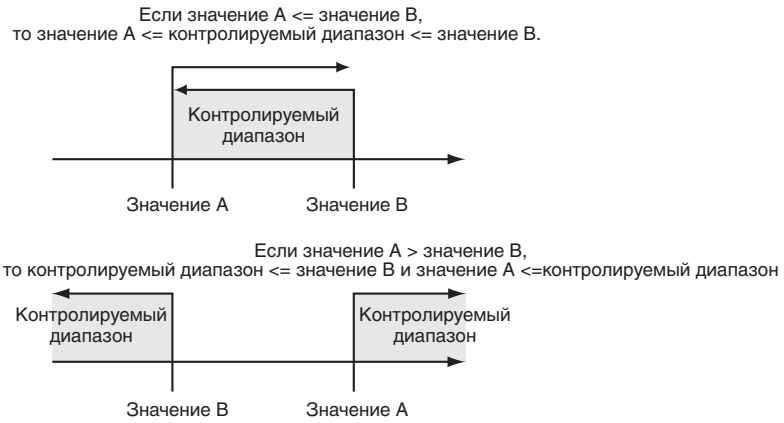


Количество диапазонов

Количество диапазонов в сравниваемом блоке указывается в первом слове блока. Можно задать до 256 диапазонов.

Указание диапазонов

Выполнение сравнения зависит от того, какое из значений больше: A или B, что показано на рисунке ниже.



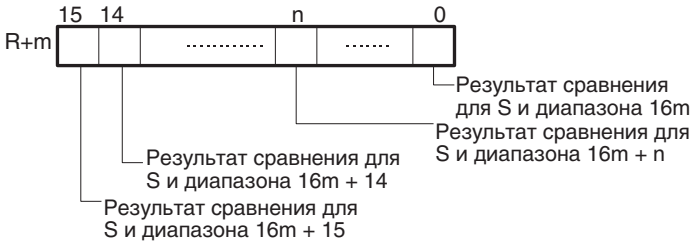
Пример

Когда B+1 ≤ B+2
Если B+1 ≤ S ≤ B+2, бит 0 слова R будет установлен,
если B+3 ≤ S ≤ B+4, бит 1 слова R будет установлен,
если S < B+5 и B+6 < S, бит 2 слова R не будет установлен, и
если S < B+7 и B+8 < S, бит 3 слова R не будет установлен.

Когда B+1 > B+2
Если S ≤ B+2 и B+1 ≤ S, бит 0 слова R будет установлен,
если S ≤ B+4 и B+3 ≤ S, бит 1 слова R будет установлен,
если B+6 < S < B+5, бит 2 слова R не будет установлен, и
если B+8 < S < B+7, бит 3 слова R не будет установлен.

Место хранения результатов

Результаты выводятся в соответствующие биты слова R. Если сравниваемых диапазонов больше чем 16, будут использованы слова, следующие за R. Максимальное количество слов результата: 16 (т. е. m = 0...15).



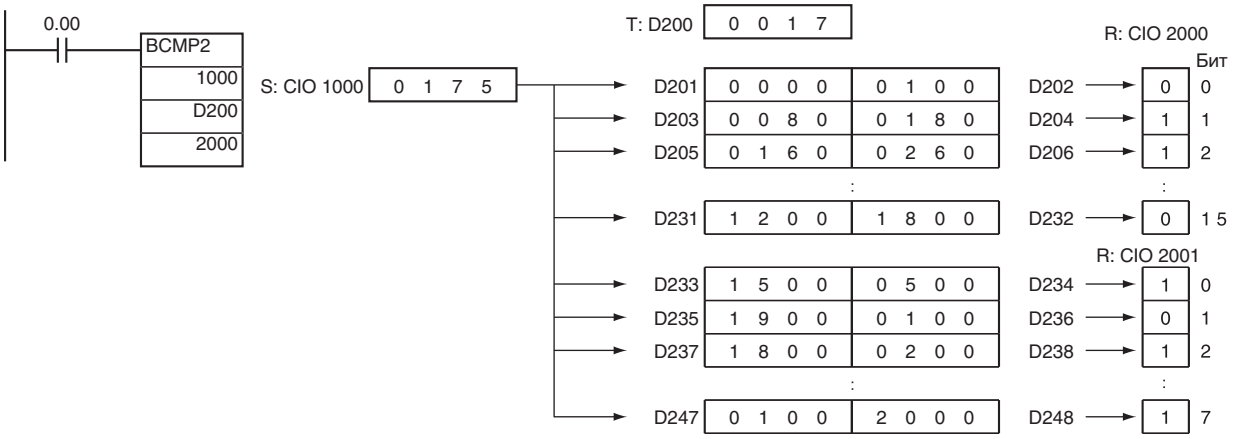
Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	ВЫКЛ

Пример

Если в следующем примере бит CIO 0.00 включен, команда BCMP2(502) сравнивает содержимое слова CIO 1000 с 24-мя диапазонами, указанными в словах D200...D247 (N = 17 hex = 23 десятичн., т. е. 24 диапазона), и устанавливает соответствующие биты слов CIO 2000 и CIO 2001, если S принадлежит диапазону, либо не устанавливает, если S

не принадлежит диапазону. Например, если значение слова CIO 1000 принадлежит диапазону, заданному словами D201 и D202, устанавливается бит 00 слова CIO 2000; если оно не принадлежит этому диапазону, бит 00 слова CIO 2000 не устанавливается. Аналогичным образом входное значение, хранящееся в слове CIO 1000, сравнивается с диапазонами, заданными словами D203 и D204, D247 и D248,... и другими словами сравниваемого блока, при этом бит 1 в CIO 2000, бит 7 в CIO 2001 и прочие биты в словах результата устанавливаются или не устанавливаются в зависимости от результатов сравнения.

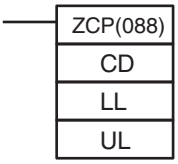


3-6-11 СРАВНИТЬ СЛОВО С ДИАПАЗОНОМ: ZCP(088)

Назначение

Проверка принадлежности 16-битового двоичного значения без знака (CD) диапазону, заданному нижней границей (LL) и верхней границей (UL). Результаты выводятся во флаги арифметических операций.

Символ РКС



CD: Сравниваемое значение
LL: Нижняя граница диапазона
UL: Верхняя граница диапазона

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	ZCP(088)
	Однократное выполнение по положительному фронту	Не предусмотрено
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	CD	LL	UL
Область CIO	CIO 0...CIO 6143		
Рабочая область	W0...W511		
Область битов хранения	H0...H511		

Область	CD	LL	UL
Область вспомогательных битов	A0...A959		
Область таймеров	T0000...T4095		
Область счетчиков	C0000...C4095		
Область DM	D0...D32767		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Константы	#0000...#FFFF (двоичн.) &0...&65535 (десятичное без знака)		
Регистры данных	DR0...DR15		
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…-2048…+2047 ,IR15 DR0…DR15, IR0…IR15 ,IR0+(++)…,IR15+(++) ,–(– –)IR0…, –(– –)IR15		

Описание

Команда ZCP(088) сравнивает 16-битовое двоичное значение без знака, заданное операндом CD, с диапазоном, указанным операндами LL и UL, и выводит результат во флаги «Больше», «Равно» и «Меньше» во вспомогательной области (состояния флагов «Меньше или равно», «Больше или равно» и «Не равно» не изменяются).

Состояния флагов арифметических операций

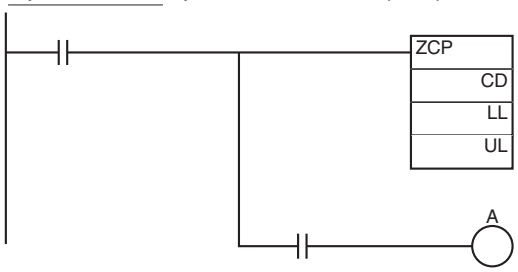
В следующей таблице показаны состояния флагов арифметических операций после выполнения команды ZCP(088).

Результат ZCP(088)	Состояние флага		
	>	=	<
CD > UL	ВКЛ	ВЫКЛ	ВЫКЛ
CD = UL	ВЫКЛ	ВКЛ	
LL < CD < UL			
CD = LL			
CD < LL		ВЫКЛ	ВКЛ

Использование результатов ZCP(088) в программе

Результат выполнения команды ZCP(088) отражается во флагах арифметических операций. Управление требуемым выходом или последней командой цепи должно осуществляться ветвью, подключенной к тому же входному условию, которое управляет выполнением команды ZCP(088), как показано на следующей схеме. В данном случае флаг равенства и выход A будут включены, когда LL ≤ CD ≤ UL.

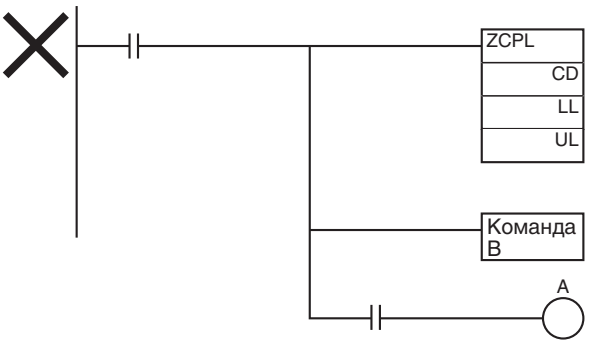
Правильное применение ZCP(088)



Флаг арифметической операции
(пример: флаг равенства)

Не следует использовать другие команды между командой ZCP(088) и командой, управляемой флагом арифметической операции, так как другая команда может изменить состояние этого флага. В этом случае результаты выполнения команды B могут изменить результаты команды ZCP(088).

Неправильное применение ZCP(088)



Флаг арифметической операции
(пример: флаг равенства)

Флаги

Название	Обо-значение	Описание работы
Флаг ошибки	ER	ВКЛ, если LL > UL.
Флаг «Больше»	>	ВКЛ, если CD > UL. Выключен во всех остальных случаях.
Флаг «Больше или равно»	> =	Не изменяется.
Флаг «Равно»	=	ВКЛ, если LL ≤ CD ≤ UL. Выключен во всех остальных случаях.
Флаг «Не равно»	<>	Не изменяется.
Флаг «Меньше»	<	ВКЛ, если CD < LL. Выключен во всех остальных случаях.
Флаг «Меньше или равно»	< =	Не изменяется.
Флаг «Меньше нуля»	N	Не изменяется.

Меры предосторожности

Не используйте другие команды между командой ZCP(088) и входным условием, которое использует результат ZCP(088), так как другая команда может изменить состояние флагов арифметических операций.

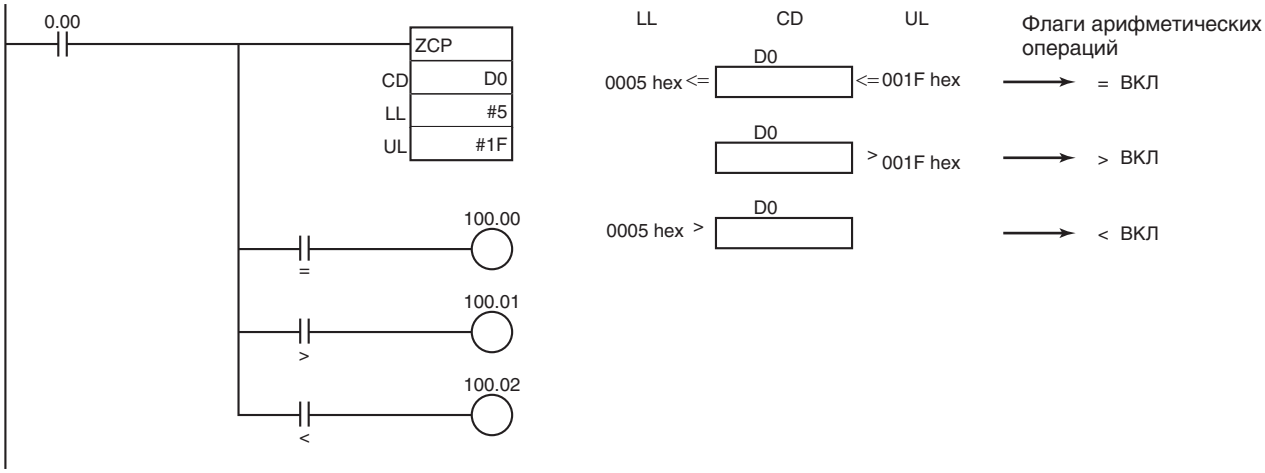
Пример

Если в следующем примере бит CIO 0.00 включен, 16-битовое двоичное значение без знака, указанное в слове D0, сравнивается с диапазоном 0005... 001F hex (5...31 в десятичном формате), а результат выводится во флаги арифметических операций.

Бит CIO 100.00 включается, если 0005 hex ≤ содержимое слова D0 ≤ 001F hex.

CIO 100.01 включается, если содержимое слова D0 > 001F hex.

CIO 100.02 включается, если содержимое слова D0 < 0005 hex.

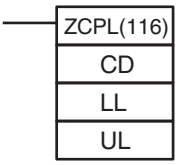


3-6-12 СРАВНИТЬ ДВОЙНОЕ СЛОВО С ДИАПАЗОНОМ: ZCPL(116)

Назначение

Проверка принадлежности 32-битового двоичного значения без знака (CD+1, CD) диапазону, заданному нижней границей (LL+1, LL) и верхней границей (UL+1, UL). Результаты выводятся во флаги арифметических операций.

Символ РКС



CD: Первое слово сравниваемого значения

LL: Первое слово нижней границы

UL: Первое слово верхней границы

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	ZCPL(116)
	Однократное выполнение по положительному фронту	Не предусмотрено
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	CD	LL	UL
Область CIO	CIO 0...CIO 6142		
Рабочая область	W0...W510		
Область битов хранения	H0...H510		
Область вспомогательных битов	A0...A958		
Область таймеров	T0000...T4094		
Область счетчиков	C0000...C4094		
Область DM	D0...D32766		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Константы	#0000 0000... #FFFF FFFF (двоичн.) &0...&4294967295 (десятичное без знака)		
Регистры данных	---		
Регистры указателей	IR0...IR15		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…-2048…+2047 ,IR15 DR0…DR15, IR0…IR15 ,IR0+(++)…,IR15+(++) ,–(– –)IR0…, –(– –)IR15		

Описание

Команда ZCPL(116) сравнивает 32-битовое двоичное значение со знаком, заданное в [CD+1, CD], с диапазоном, указанным [LL+1, LL] и [UL+1, UL], и выводит результат во флаги «Больше», «Равно» и «Меньше» во вспомогательной области (состояния флагов «Меньше или равно», «Больше или равно» и «Не равно» не изменяются).

Состояния флагов арифметических операций

В следующей таблице показаны состояния флагов арифметических операций после выполнения команды ZCPL(116).

Результат ZCPL(116)	Состояние флага		
	>	=	<
CD+1, CD > UL+1, UL	ВКЛ	ВЫКЛ	ВЫКЛ
CD+1, CD = UL+1, UL	ВЫКЛ	ВКЛ	
LL+1, LL < CD+1, CD < UL+1, UL			
CD+1, CD = LL+1, LL			
CD+1, CD < LL+1, LL		ВЫКЛ	ВКЛ

Использование результата ZCPL(116) в программе

Результат выполнения команды ZCPL(116) отражается во флагах арифметических операций. Управление требуемым выходом или последней командой цепи должно осуществляться ветвью, подключенной к тому же входному условию, которое управляет выполнением команды ZCPL(116).

Не следует использовать другие команды между командой ZCPL(116) и командой, управляемой флагом арифметической операции, так как другая команда может изменить состояние этого флага.

Команда ZCPL(116) отличается от команды ZCP(088) только тем, что ZCPL(116) сравнивает 32-битовые, а не 16-битовые значения. Схемы, показывающие использование результатов в программе, и пример сегмента программы см. в *3-6-11 СРАВНИТЬ СЛОВО С ДИАПАЗОНОМ: ZCP(088)*.

Флаги

Название	Обо-значение	Описание работы
Флаг ошибки	ER	ВКЛ, если LL+1, LL > UL+1, UL.
Флаг «Больше»	>	ВКЛ, если CD > UL+1, UL. Выключен во всех остальных случаях.
Флаг «Больше или равно»	> =	Не изменяется.
Флаг «Равно»	=	ВКЛ, если LL+1, LL ≤ CD+1, CD ≤ UL+1, UL. Выключен во всех остальных случаях.
Флаг «Не равно»	<>	Не изменяется.
Флаг «Меньше»	<	ВКЛ, если CD+1, CD < LL+1, LL. Выключен во всех остальных случаях.
Флаг «Меньше или равно»	< =	Не изменяется.
Флаг «Меньше нуля»	N	Не изменяется.

Меры предосторожности

Не используйте другие команды между командой ZCPL(116) и входным условием, которое использует результат ZCPL(116), так как другая команда может изменить состояние флагов арифметических операций.

3-7 Команды передачи данных

В данном разделе описаны команды, используемые для передачи данных различными способами.

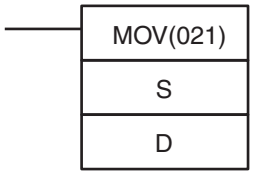
Команда	Мнемоническое обозначение	Код функции	Стр.
ПЕРЕДАТЬ	MOV	021	271
ПЕРЕДАТЬ ДОПОЛНЕНИЕ	MVN	022	273
ПЕРЕДАТЬ ДВОЙНОЕ СЛОВО	MOVL	498	274
ПЕРЕДАТЬ ДОПОЛНЕНИЕ К ДВОЙНОМУ СЛОВУ	MVNL	499	276
ПЕРЕДАТЬ БИТ	MOVB	082	277
ПЕРЕДАТЬ РАЗРЯД	MOVD	083	279
ПЕРЕДАТЬ НЕСКОЛЬКО БИТОВ	XFRB	062	282
ПЕРЕДАЧА БЛОКА	XFER	070	285
ЗАПОЛНЕНИЕ БЛОКА	BSET	071	287
ОБМЕН ДАННЫМИ	XCHG	073	289
ОБМЕН ДВОЙНЫМИ СЛОВАМИ	XCGL	562	290
РАЗМЕЩЕНИЕ ОДИНОЧНОГО СЛОВА	DIST	080	292
СБОР ДАННЫХ	COLL	081	294
ПЕРЕДАТЬ В РЕГИСТР	MOVR	560	296
ПЕРЕДАТЬ ТЕКУЩЕЕ ЗНАЧЕНИЕ ТАЙМЕРА/ СЧЕТЧИКА В РЕГИСТР	MOVRW	561	298

3-7-1 ПЕРЕДАТЬ: MOV(021)

Назначение

Передача слова данных в указанное слово.

Символ РКС



S: Исходное слово

D: Слово назначения

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	MOV(021)
	Однократное выполнение по положительному фронту	@MOV(021)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		!MOV(021)
Комбинированная модификация	**Однократное выполнение и мгновенное обновление слова назначения по положительному фронту.**	!@MOV(021)

Доступные области программы

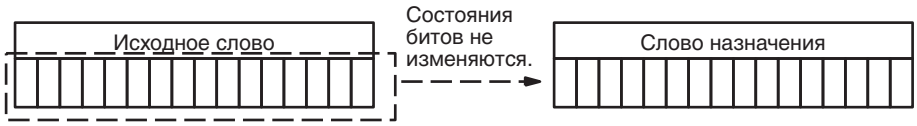
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	S	D
Область CIO	CIO 0...CIO 6143	
Рабочая область	W0...W511	
Область битов хранения	H0...H511	
Область вспомогательных битов	A0...A959	A448...A959
Область таймеров	T0000...T4095	
Область счетчиков	C0000...C4095	
Область DM	D0...D32767	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Константы	#0000...#FFFF (двоичн.)	---
Регистры данных	DR0...DR15	
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…-2048…+2047 ,IR15 DR0…DR15, IR0…IR15 ,IR0+(++)…,IR15+(++) ,–(– –)IR0…, –(– –)IR15	

Описание

Передача значения S в D. Указывая в качестве S константу, команду можно использовать для установки требуемого значения по требуемому адресу.



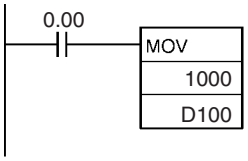
Для команды MOV(021) предусмотрен вариант выполнения с мгновенным обновлением (!MOV(021)). В качестве S могут быть указаны биты внешних входов, в качестве D могут быть указаны биты внешних выходов. Биты входов, указанные в S, обновляются непосредственно перед выполнением команды; биты выходов, указанные в D, обновляются сразу после выполнения команды.

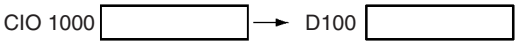
Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	ВЫКЛ
Флаг «Равно»	=	Включен, если передаваемое значение = 0000. Выключен во всех остальных случаях.
Флаг «Меньше нуля»	N	Включен, если старший бит передаваемого значения = «1». Выключен во всех остальных случаях.

Пример

Если в следующем примере бит CIO 0.00 включен, содержимое слова CIO 1000 копируется в слово D100.



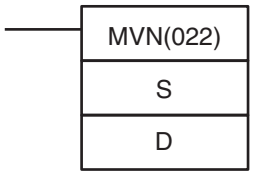


3-7-2 ПЕРЕДАТЬ ДОПОЛНЕНИЕ: MVN(022)

Назначение

Передача дополнения к слову данных в указанное слово.

Символ РКС



S: Исходное слово

D: Слово назначения

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	MVN(022)
	Однократное выполнение по положительному фронту	@MVN(022)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

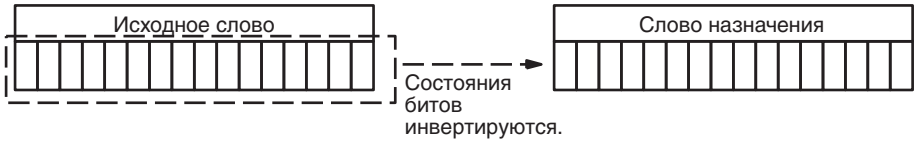
Характеристики операндов

Область	S	D
Область CIO	CIO 0...CIO 6143	
Рабочая область	W0...W511	
Область битов хранения	H0...H511	
Область вспомогательных битов	A0...A959	A448...A959
Область таймеров	T0000...T4095	
Область счетчиков	C0000...C4095	
Область DM	D0...D32767	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Константы	#0000...#FFFF (двоичн.)	---
Регистры данных	DR0...DR15	

Область	S	D
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…-2048…+2047 ,IR15 DR0…DR15, IR0…IR15 ,IR0+(++)…,IR15+(++) ,–(– –)IR0…, –(– –)IR15	

Описание

Команда MVN(022) выполняет инверсию битов в слове S и передает результат в слово D. Содержимое S остается неизменным.

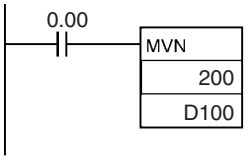


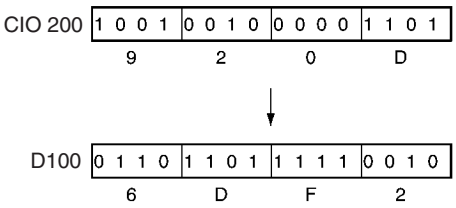
Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	ВЫКЛ
Флаг «Равно»	=	Включен, если содержимое D имеет значение «0000» после выполнения команды. Выключен во всех остальных случаях.
Флаг «Меньше нуля»	N	Включен, если старший бит D находится в состоянии «1» после выполнения команды. Выключен во всех остальных случаях.

Пример

Если в следующем примере бит CIO 0.00 включен, состояние битов в слове CIO 200 меняется на противоположное, а результат копируется в слово D100.



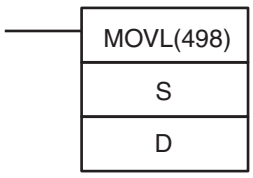


3-7-3 ПЕРЕДАТЬ ДВОЙНОЕ СЛОВО: MOVL(498)

Назначение

Передача двух слов данных в указанные слова.

Символ РКС



S: Первое исходное слово

D: Первое слово назначения

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	MOVL(498)
	Однократное выполнение по положительному фронту	@MOVL(498)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

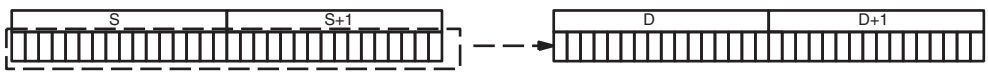
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	S	D
Область CIO	CIO 0...CIO 6142	
Рабочая область	W0...W510	
Область битов хранения	H0...H510	
Область вспомогательных битов	A0...A958	A448...A958
Область таймеров	T0000...T4094	
Область счетчиков	C0000...C4094	
Область DM	D0...D32766	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Константы	#00000000...#FFFFFFFF (двоичн.)	---
Регистры данных	---	
Регистры указателей	IR0...IR15	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…-2048…+2047 ,IR15 DR0…DR15, IR0…IR15 ,IR0+(++)…,IR15+(++) ,–(– –) IR0 …, 1–(– –) IR5	

Описание

Команда MOVL(498) передает слова S+1 и S в слова D+1 и D. Указывая в качестве S+1 и S константу, команду можно использовать для установки требуемого значения по требуемому адресу.



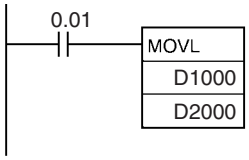
Состояния битов не изменяются.

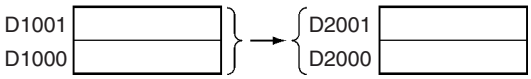
Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	ВЫКЛ
Флаг «Равно»	=	ВКЛ, если после выполнения в словах D+1 и D содержится «0000 0000». Выключен во всех остальных случаях.
Флаг «Меньше нуля»	N	ВКЛ, если после выполнения старший бит слова D+1 = «1». Выключен во всех остальных случаях.

Пример

Если в следующем примере бит CIO 0.01 включен, содержимое слов D1001 и D1000 копируется в слова D2001 и D2000.



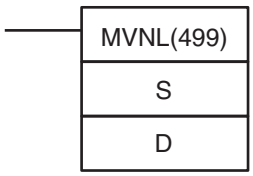


3-7-4 ПЕРЕДАТЬ ДОПОЛНЕНИЕ К ДВОЙНОМУ СЛОВУ: MVNL(499)

Назначение

Передача дополнения к двум словам данных в указанные слова.

Символ РКС



S: Первое исходное слово

D: Первое слово назначения

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	MVNL(499)
	Однократное выполнение по положительному фронту	@MVNL(499)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	S	D
Область CIO	CIO 0...CIO 6142	
Рабочая область	W0...W510	
Область битов хранения	H0...H510	
Область вспомогательных битов	A0...A958	A448...A958
Область таймеров	T0000...T4094	
Область счетчиков	C0000...C4094	
Область DM	D0...D32766	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Константы	#00000000...#FFFFFFFF (двоичн.)	---
Регистры данных	---	

Область	S	D
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…-2048…+2047 ,IR15 DR0…DR15, IR0…IR15 ,IR0+(++)…,IR15+(++) ,–(– –)IR0…, –(– –)IR15	

Описание

Команда MVNL(499) выполняет инверсию битов в словах S+1 и S и передает результат в слова D+1 и D. Содержимое слов S+1 и S остается неизменным.

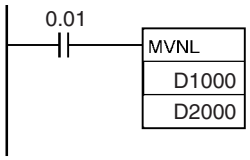


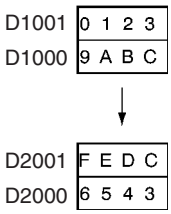
Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	ВЫКЛ
Флаг «Равно»	=	ВКЛ, если после выполнения в словах D+1 и D содержится «0000 0000». Выключен во всех остальных случаях.
Флаг «Меньше нуля»	N	ВКЛ, если после выполнения старший бит слова D+1 = «1». Выключен во всех остальных случаях.

Примеры

Если в следующем примере бит CIO 0.01 включен, состояние битов в словах D1001 и D1000 меняется на противоположное, результат записывается в слова D2001 и D2000. (Исходное содержимое слов D1001 и D1000 остается неизменным.)



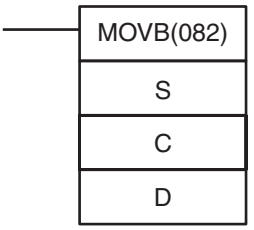


3-7-5 ПЕРЕДАТЬ БИТ: MOVB(082)

Назначение

Передает состояние указанного бита.

Символ РКС



S: Исходное слово или данные

C: Управляющее слово

D: Слово назначения

Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	MOVB(082)
	Однократное выполнение по положительному фронту	@MOVB(082)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

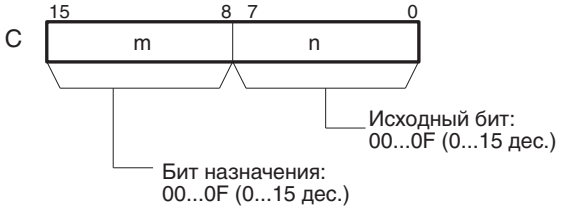
Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

C: Управляющее слово

Два младших разряда слова C указывают требуемый бит в исходном объекте S, а два старших разряда слова C указывают требуемый бит в конечном объекте D.

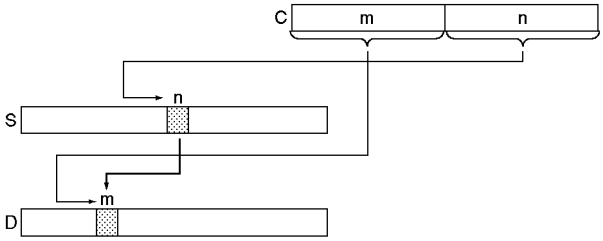


Характеристики операндов

Область	S	C	D
Область CIO	CIO 0...CIO 6143		
Рабочая область	W0...W511		
Область битов хранения	H0...H511		
Область вспомогательных битов	A0...A959		A448...A959
Область таймеров	T0000...T4095		
Область счетчиков	C0000...C4095		
Область DM	D0...D32767		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Константы	#0000...#FFFF (двоичн.)	Только указанные значения	---
Регистры данных	DR0...DR15		
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…-2048…+2047 ,IR15 DR0…DR15, IR0…IR15 ,IR0+(++)…,IR15+(++) ,–(– –)IR0…, –(– –)IR15		

Описание

Команда MOVB(082) копирует состояние указанного бита (n) слова S в указанный бит (m) слова D. Прочие биты слова адресата остаются неизменными.



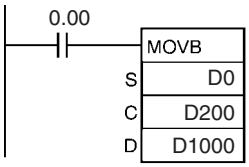
Примечание. В операндах S и D может быть указано одно и то же слово. В этом случае бит будет скопирован в другой бит в пределах данного слова.

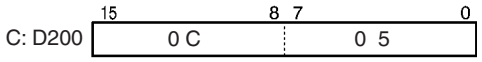
Флаги

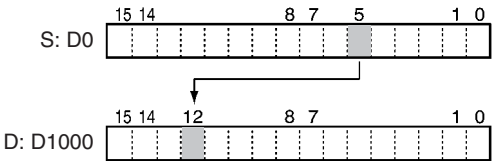
Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если значение пары младших или пары старших разрядов C выходит за диапазон 00...0F (hex). Выключен во всех остальных случаях.

Примеры

Если бит CIO 0.00 включен в следующем примере, 5-й бит исходного слова (D0) копируется в 12-й бит слова адресата (B1000) в соответствии со значением управляющего слова 0C05.





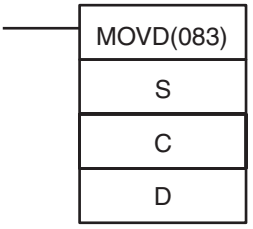


3-7-6 ПЕРЕДАТЬ РАЗРЯД: MOVD(083)

Назначение

Передача указанного разряда или разрядов. (Подразумевается шестнадцатеричный разряд, состоящий из 4 двоичных битов.)

Символ РКС



S: Исходное слово или данные

C: Управляющее слово

D: Слово назначения

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	MOVD(083)
	Однократное выполнение по положительному фронту	@MOVD(083)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

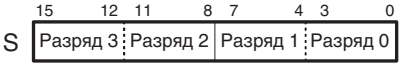
Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

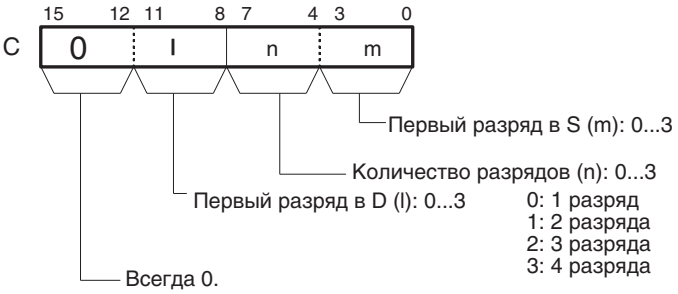
S: Исходное слово

Исходные разряды считываются справа налево с возвратом к младшему разряду (разряд 0), если это необходимо.



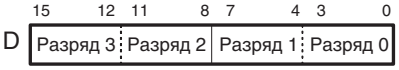
C: Управляющее слово

Три первых разряда слова C указывают: первый разряд (m) исходного слова, количество передаваемых разрядов (n) и первый разряд (ℓ) слова назначения (см. рисунок ниже).



D: Адресуемое слово

Адресуемые разряды записываются справа налево с возвратом к младшему разряду (разряд 0), если это необходимо.



Характеристики операндов

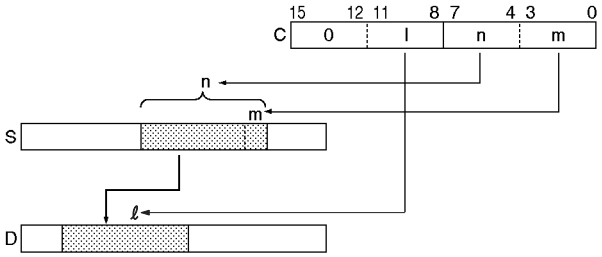
Область	S	C	D
Область CIO	CIO 0...CIO 6143		
Рабочая область	W0...W511		
Область битов хранения	H0...H511		
Область вспомогательных битов	A0...A959		A448...A959
Область таймеров	T0000...T4095		
Область счетчиков	C0000...C4095		
Область DM	D0...D32767		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		

Область	S	C	D
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Константы	#0000...#FFFF (двоичн.)	Только указанные значения	---
Регистры данных	DR0...DR15		
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…-2048…+2047 ,IR15 DR0…DR15, IR0…IR15 ,IR0+(++)…,IR15+(++) ,–(– –)IR0…, –(– –)IR15		

Описание

Команда MOVD(083) копирует содержимое n разрядов слова S (начиная с разряда m) в слово D (начиная с разряда ℓ). Изменяются только указанные разряды; остальные разряды остаются неизменными.

Если указанное количество считываемых или записываемых разрядов приводит к выходу за пределы старшего разряда S или D, команда MOVD(083) переходит к младшему разряду того же слова.



Примечание. В операндах S и D может быть указано одно и то же слово. В этом случае разряд будет скопирован в другой разряд в пределах данного слова.

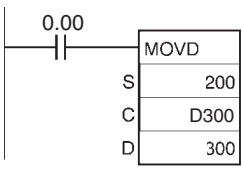
Флаги

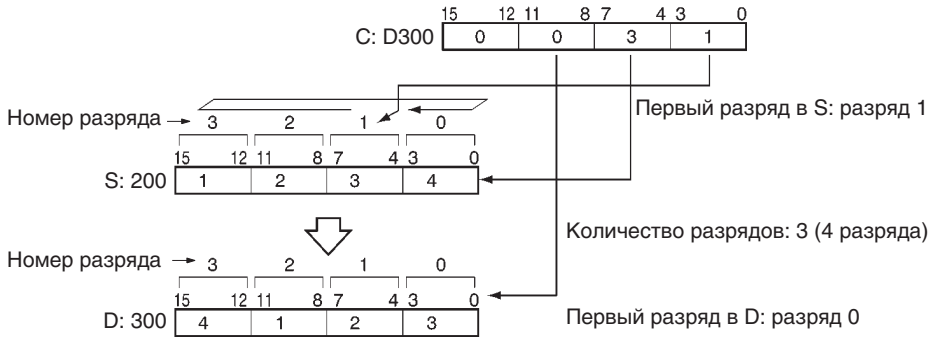
Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если один из трех первых разрядов слова C выходит за диапазон 0...3. Выключен во всех остальных случаях.

Примеры

Передача четырех разрядов

Если в следующем примере бит CIO 0.00 включен, четыре разряда данных копируются из слова CIO 200 в слово CIO 300. В соответствии со значением управляющего слова (0031 hex), выполнение начинается с того, что 1-й разряд слова CIO 200 копируется в 0-й разряд слова CIO 300.

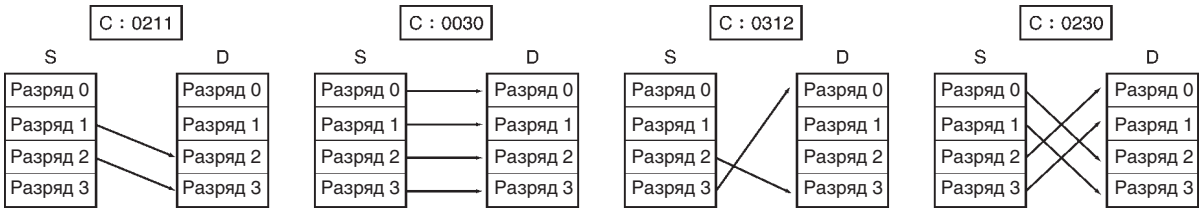




Примечание. После считывания старшего разряда слова S (разряд 3) команда MOVD(083) возвращается к младшему разряду (разряд 0).

Примеры настройки слова C

На следующем рисунке показаны примеры передачи данных для разных значений C.

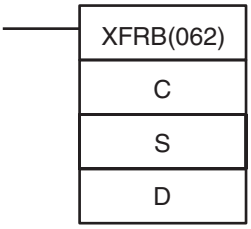


3-7-7 ПЕРЕДАТЬ НЕСКОЛЬКО БИТОВ: XFRB(062)

Назначение

Передача указанного количества последовательно расположенных битов.

Символ РКС



C: Управляющее слово

S: Первое исходное слово

D: Первое слово назначения

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	XFRB(062)
	Однократное выполнение по положительному фронту	@XFRB(062)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

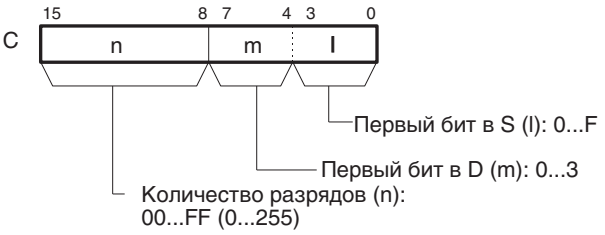
Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

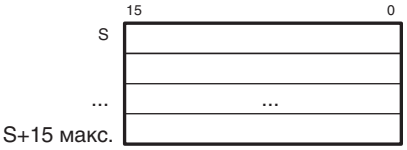
C: Управляющее слово

Три первых разряда слова C указывают: первый разряд (m) исходного слова, количество передаваемых разрядов (n) и первый разряд (ℓ) слова назначения (см. рисунок ниже).



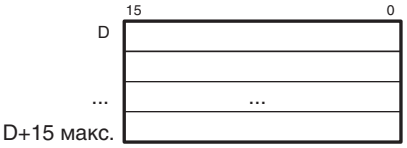
S: Первое исходное слово

Указывает первое исходное слово. Биты считываются справа налево, при необходимости продолжается считывание битов следующих слов (до S+16).



D: Первое слово назначения

Указывает первое слово назначения. Биты записываются справа налево, при необходимости продолжается запись в биты следующих слов (до D+16).



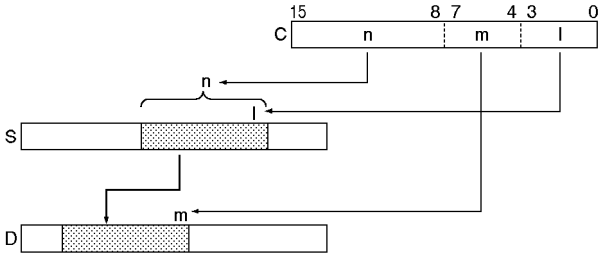
Характеристики операндов

Область	C	S	D
Область CIO	CIO 0...CIO 6143		
Рабочая область	W0...W511		
Область битов хранения	H0...H511		
Область вспомогательных битов	A0...A959		A448...A959
Область таймеров	T0000...T4095		
Область счетчиков	C0000...C4095		
Область DM	D0...D32767		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Константы	Только указанные значения	---	---
Регистры данных	DR0...DR15	---	
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…-2048…+2047 ,IR15 DR0…DR15, IR0…IR15 ,IR0+(++)...5+(++) ,–(– –)IR0…, –(– –)IR15		

Описание

Команда XFRB(062) передает до 255 последовательных битов исходных слов (начиная с бита ℓ слова S) в слова назначения (начиная с бита m слова D). Биты слов назначения, в которые не были записаны исходные биты, остаются неизменными.

Начальные биты и количество битов указываются в C, как показано на следующем рисунке.



Исходные слова и слова назначения могут перекрываться. Путем передачи данных с перекрытием нескольких слов можно добиться более эффективного расположения данных в области данных. (В частности, это может быть полезным при обработке данных положения при позиционировании.)

Так как исходные слова и слова назначения могут накладываться, команду XFRB(062) можно использовать вместе с командой ANDW(034) для сдвига m битов на n позиций.

Флаги

Название	Обо-значение	Описание работы
Флаг ошибки	ER	ВЫКЛ

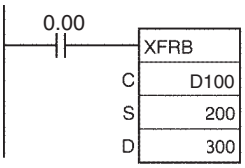
Меры предосторожности

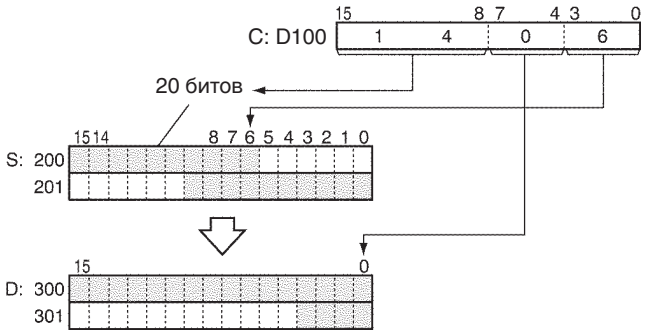
При каждом выполнении команды XFRB(062) можно передать до 255 битов данных.

Убедитесь, что исходные слова и слова назначения не выходят за пределы области данных.

Примеры

Если в следующем примере бит CIO 0.00 включен, 20 битов, начиная с бита CIO 200.06, копируются в 20 битов, начиная с бита CIO 300.00.



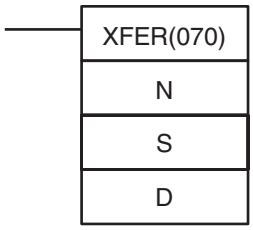


3-7-8 ПЕРЕДАЧА БЛОКА: XFER(070)

Назначение

Передает указанное количество последовательно расположенных слов.

Символ РКС



N: Количество слов

S: Первое исходное слово

D: Первое слово назначения

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	XFER(070)
	Однократное выполнение по положительному фронту	@XFER(070)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

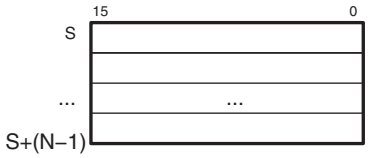
Операнды

N: Количество слов

Указывает количество слов, которое должно быть передано. Допустимый диапазон значений N: 0000...FFFF (0...65535 в десятичном формате).

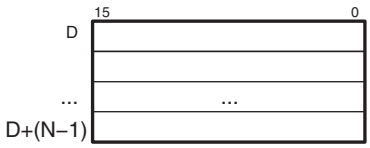
S: Первое исходное слово

Указывает первое исходное слово.



D: Первое слово назначения

Указывает первое слово назначения.



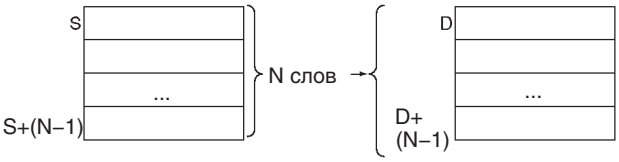
Характеристики операндов

Область	N	S	D
Область CIO	CIO 0...CIO 6143		
Рабочая область	W0...W511		
Область битов хранения	H0...H511		
Область вспомогательных битов	A0...A959		A448...A959
Область таймеров	T0000...T4095		
Область счетчиков	C0000...C4095		

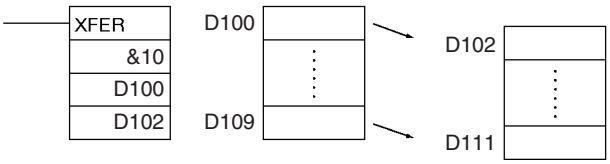
Область	N	S	D
Область DM	D0...D32767		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Константы	#0000...#FFFF (двоичный) или &0...&65535	---	---
Регистры данных	DR0...DR15	---	
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…-2048…+2047 ,IR15 DR0…DR15, IR0…IR15 ,IR0+(++)…,IR15+(++) ,–(– –)IR0…, –(– –)IR15		

Описание

Команда XFER(070) копирует N слов, начиная со слова S (S...S+(N-1)), в N слов, начиная со слова D (D...D+(N-1)).



Исходные слова и слова назначения могут перекрываться, так что с помощью команды XFER(070) можно выполнять сдвиг слов.



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	ВЫКЛ

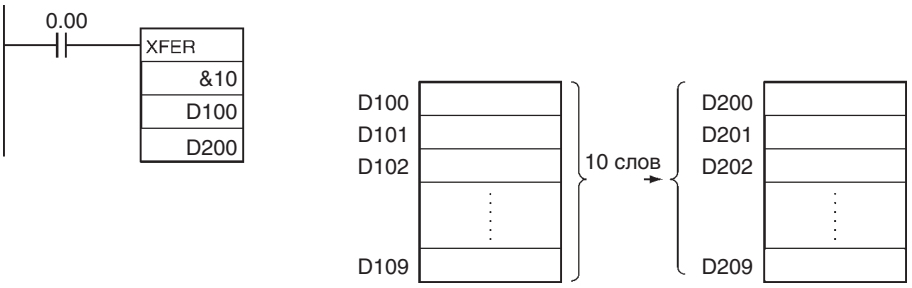
Меры предосторожности

Проследите, чтобы исходные слова (S...S+N-1) и слова назначения (D...D+N-1) не вышли за границу области данных.

Копирование большого количества слов командой XFER(070) может занимать продолжительное время. Если во время выполнения команды будет прервано питание, операция копирования может не завершиться.

Пример

Если в следующем примере бит CIO 0.00 включен, значения десяти слов, расположенных по адресам от D100 до D109, копируются в соответствующие слова, расположенные по адресам от D200 до D209.



3-7-9 ЗАПОЛНЕНИЕ БЛОКА: BSET(071)

Назначение

Запись одного и того же значения в группу последовательно расположенных слов.

Символ РКС



Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	BSET(071)
	Однократное выполнение по положительному фронту	@BSET(071)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

S: Исходное слово

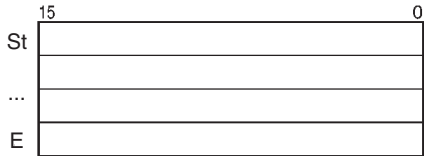
Указывает входное значение или слово, содержащее входное значение.

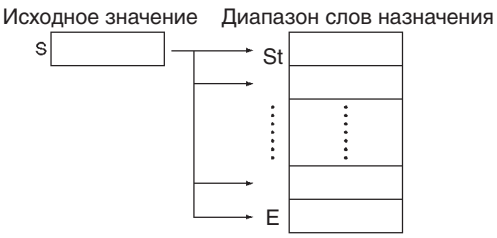
St: Начальное слово

Указывает первое слово выходного диапазона слов.

E: Конечное слово

Указывает последнее слово выходного диапазона слов.





Примечание. St и E должны находиться в одной области данных.

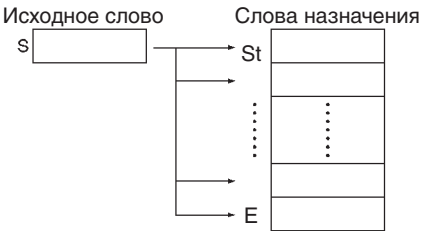
Характеристики операндов

Область	S	St	E
Область CIO	CIO 0...CIO 6143		
Рабочая область	W0...W511		

Область	S	St	E
Область битов хранения	H0...H511		
Область вспомогательных битов	A0...A959	A448...A959	
Область таймеров	T0000...T4095		
Область счетчиков	C0000...C4095		
Область DM	D0...D32767		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Константы	#0000...#FFFF (двоичн.)	---	
Регистры данных	DR0...DR15	---	
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…-2048…+2047 ,IR15 DR0…DR15, IR0…IR15 ,IR0+(++)…,IR15+(++) ,–(– –) IR0..., 15–(– –) IR		

Описание

Команда BSET(071) копирует одно и то же входное слово (S) во все слова назначения в диапазоне St...E.



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если St больше E. Выключен во всех остальных случаях.

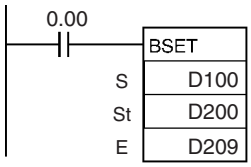
Меры предосторожности

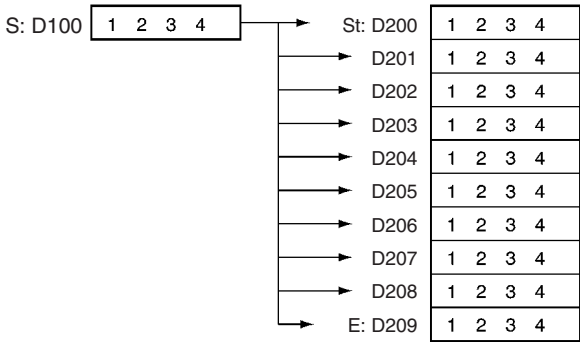
Убедитесь, что первое слово (St) и последнее слово (E) принадлежат одной области данных и что St ≤ E.

Копирование большого количества слов командой BSET(071) может занимать продолжительное время. Если во время выполнения команды BSET(071) будет прервано питание, передача данных может быть не завершена.

Пример

Если в следующем примере бит CIO 0.00 включен, входное значение, указанное в D100, копируется во все слова диапазона D200...D209.



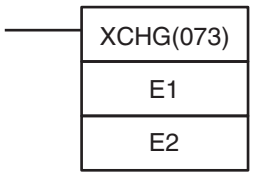


3-7-10 ОБМЕН ДАННЫМИ: XCHG(073)

Назначение

Обмен содержимым между двумя указанными словами.

Символ РКС



E1: Первое слово

E2: Второе слово

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	XCHG(073)
	Однократное выполнение по положительному фронту	@XCHG(073)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

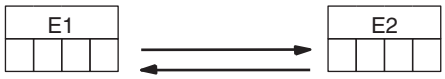
Характеристики операндов

Область	E1	E2
Область CIO	CIO 0...CIO 6143	
Рабочая область	W0...W511	
Область битов хранения	H0...H511	
Область вспомогательных битов	A448...A959	

Область	E1	E2
Область таймеров	T0000...T4095	
Область счетчиков	C0000...C4095	
Область DM	D0...D32767	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Константы	---	
Регистры данных	DR0...DR15	
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…-2048…+2047 ,IR15 DR0…DR15, IR0…IR15 ,IR0+(++)…,IR15+(++) ,–(– –)IR0…, –(– –)IR15	

Описание

XCHG(073) производит обмен содержимым между словами E1 и E2.

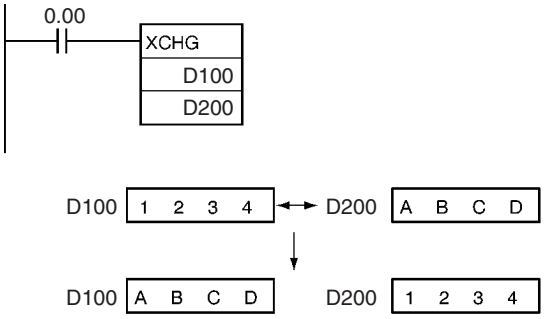


Флаги

Данная команда не изменяет состояние какого-либо флага.

Пример

Если в следующем примере бит CIO 0.00 включен, происходит обмен значениями слов D100 и D200.

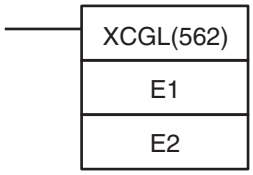


3-7-11 ОБМЕН ДВОЙНЫМИ СЛОВАМИ: XCGL(562)

Назначение

Обмен содержимым между одной парой последовательных слов и другой парой последовательных слов.

Символ РКС



E1: Первое слово

E2: Второе слово

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	XCGL(562)
	Однократное выполнение по положительному фронту	@XCGL(562)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

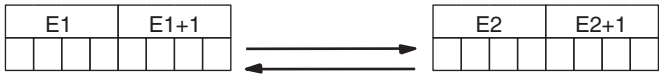
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

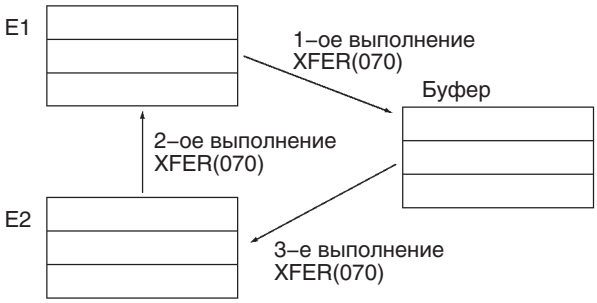
Область	E1	E2
Область CIO	CIO 0...CIO 6142	
Рабочая область	W0...W510	
Область битов хранения	H0...H510	
Область вспомогательных битов	A448...A958	
Область таймеров	T0000...T4094	
Область счетчиков	C0000...C4094	
Область DM	D0...D32766	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Константы	---	---
Регистры данных	---	
Регистры указателей	IR0...IR15	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…-2048…+2047 ,IR15 DR0…DR15, IR0…IR15 ,IR0+(++)…,IR15+(++) ,–(– –)IR0…, –(– –)IR15	

Описание

Команда XCGL(562) производит обмен содержимым между словами E1+1 и E1 и словами E2+1 и E2.



Чтобы выполнить обмен содержимым между тремя словами и более, используйте команду XFER(070) для передачи содержимого слов в дополнительную группу слов (буфер), как показано на следующем рисунке.

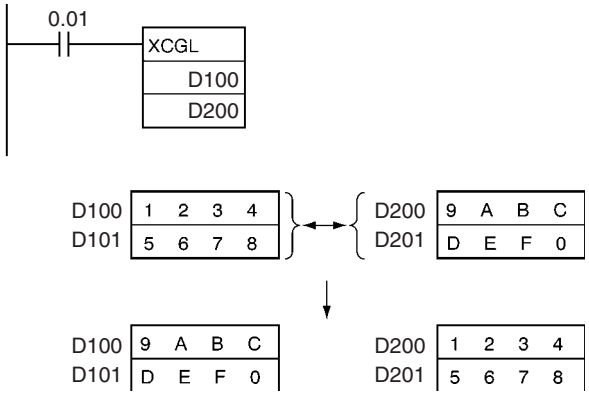


Флаги

Данная команда не изменяет состояние какого-либо флага.

Пример

Если в следующем примере бит CIO 0.01 включен, происходит обмен значениями между словами D100, D101 и словами D200, D201.

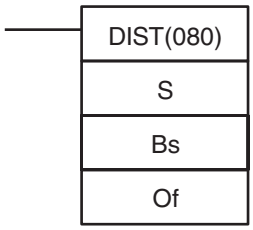


3-7-12 РАЗМЕЩЕНИЕ ОДИНОЧНОГО СЛОВА: DIST(080)

Назначение

Передает содержимое исходного слова в слово назначения, адрес которого определяется путем смещения базового адреса на указанную величину.

Символ РКС



S: Исходное слово

Bs: Базовый адрес слова назначения

Of: Смещение

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	DIST(080)
	Однократное выполнение по положительному фронту	@DIST(080)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

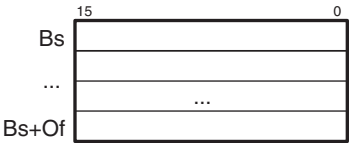
Операнды

Bs: Базовый адрес слова назначения

Указывает базовый адрес слова назначения. Фактический адрес слова назначения определяется путем добавления величины смещения к данному адресу.

Of: Смещение

Это значение добавляется к базовому адресу для определения фактического адреса слова назначения. Может быть указано любое значение смещения от 0000 до FFFF (от 0 до 65535 в десятичном формате), однако Bs и Bs+Of должны находиться в одной области данных.

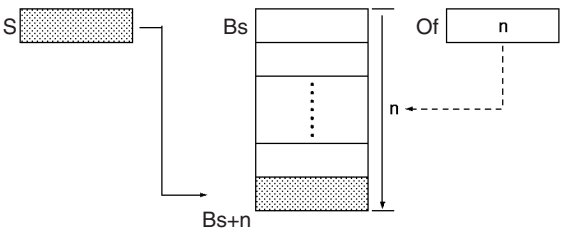


Характеристики операндов

Область	S	Bs	Of
Область CIO	CIO 0...CIO 6143		
Рабочая область	W0...W511		
Область битов хранения	H0...H511		
Область вспомогательных битов	A0...A959	A448...A959	A0...A959
Область таймеров	T0000...T4095		
Область счетчиков	C0000...C4095		
Область DM	D0...D32767		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Константы	#0000...#FFFF (двоичн.)	---	#0000...#FFFF (двоичный) или &0...&65535
Регистры данных	DR0...DR15	---	DR0...DR15
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…-2048…+2047 ,IR15 DR0…DR15, IR0…IR15 ,IR0+(++)…,IR15+(++) ,–(– –)IR0…, –(– –)IR15		

Описание

Команда DIST(080) копирует содержимое входного слова (S) в слово назначения, адрес которого определяется путем добавления величины смещения (Of) к базовому адресу (Bs). Используя одну и ту же команду DIST(080) и изменяя смещение адреса (Of), содержимое входного слова можно разместить в несколько различных слов некоторой области данных.



Флаги

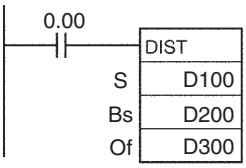
Название	Обо-значение	Описание работы
Флаг ошибки	ER	ВЫКЛ
Флаг «Равно»	=	ВКЛ, если исходное слово = 0000. Выключен во всех остальных случаях.
Флаг «Меньше нуля»	N	ВКЛ, если старший бит исходного слова = «1». Выключен во всех остальных случаях.

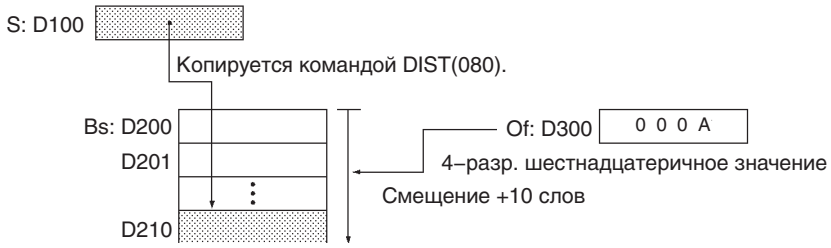
Меры предосторожности

Проследите, чтобы величина смещения не приводила к выходу за границу области данных (то есть, чтобы Bs и Bs+Of были в одной области данных).

Пример

Если в следующем примере включен бит CIO 0.00, содержимое слова D100 копируется в слово D210 (D200 + 10), если D300 = 10 (0A hex). Изменяя величину смещения в D300, содержимое D100 можно скопировать в другие слова.



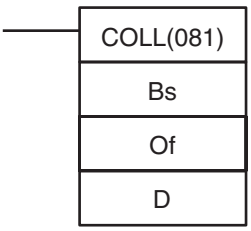


3-7-13 СБОР ДАННЫХ: COLL(081)

Назначение

Передает в слово назначения содержимое исходного слова, адрес которого определяется путем смещения базового адреса на указанную величину.

Символ РКС



Bs: Базовый адрес исходного слова

Of: Смещение

D: Слово назначения

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	COLL(081)
	Однократное выполнение по положительному фронту	@COLL(081)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограм-мы	Задачи обработки прерываний
OK	OK	OK	OK

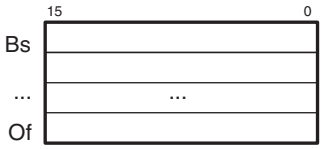
Операнды

Bs: Базовый адрес исходного слова

Указывает базовый адрес исходного слова. Фактический адрес исходного слова определяется путем добавления величины смещения к данному адресу.

Of: Смещение

Это значение добавляется к базовому адресу для определения адреса исходного слова. Может быть указано любое значение смещения от 0000 до FFFF (от 0 до 65535 в десятичном формате), однако Bs и Bs+Of должны находиться в одной области данных.

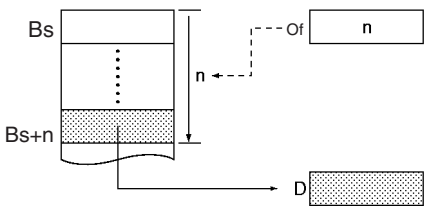


Характеристики операндов

Область	Bs	Of	D
Область CIO	CIO 0...CIO 6143		
Рабочая область	W0...W511		
Область битов хранения	H0...H511		
Область вспомогательных битов	A0...A959		A448...A959
Область таймеров	T0000...T4095		
Область счетчиков	C0000...C4095		
Область DM	D0...D32767		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Константы	---	#0000...#FFFF (двоичный) или &0...&65535	---
Регистры данных	---	DR0...DR15	
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…-2048…+2047 ,IR15 DR0…DR15, IR0…IR15 ,IR0+(++)…,IR15+(++) ,–(– –)IR0…, –(– –)IR15		

Описание

Команда COLL(081) копирует исходное слово (адрес которого определяется путем добавления Of к Bs) в слово назначения. Используя одну и ту же команду COLL(081) и изменяя смещение адреса (Of), можно собирать данные из различных исходных слов области данных.



Флаги

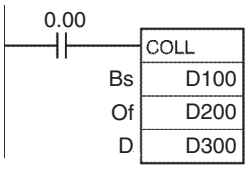
Название	Обозначение	Описание работы
Флаг ошибки	ER	ВЫКЛ
Флаг «Равно»	=	ВКЛ, если исходное слово = 0000. Выключен во всех остальных случаях.
Флаг «Меньше нуля»	N	ВКЛ, если старший бит исходного слова = «1». Выключен во всех остальных случаях.

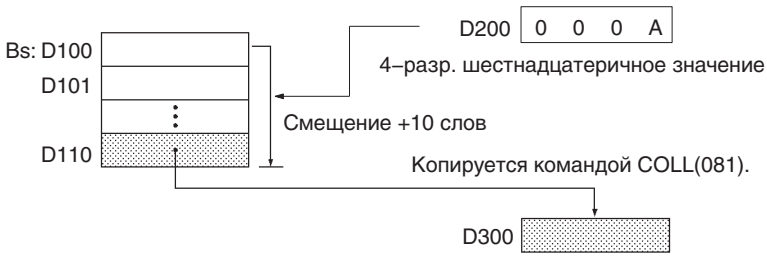
Меры предосторожности

Проследите, чтобы величина смещения не приводила к выходу за границу области данных (то есть, чтобы Bs и Bs+Of были в одной области данных).

Пример

Если в следующем примере включен бит CIO 0.00, содержимое слова D110 (D100 + 10) копируется в слово D300, если D200 = 10 (0A hex). Изменяя величину смещения в D200, в D300 можно скопировать содержимое других слов.



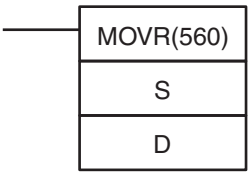


3-7-14 ПЕРЕДАТЬ В РЕГИСТР: MOVR(560)

Назначение

Запись адреса памяти ПЛК указанного слова, бита или флага завершения таймера/счетчика в указанный регистр указателей. (Чтобы записать в регистр указателя адрес текущего значения таймера/счетчика в памяти ПЛК, используйте команду MOVRW(561).)

Символ РКС



S: Источник (требуемое слово или бит)

D: Адресат (регистр указателя)

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	MOVR(560)
	Однократное выполнение по положительному фронту	@MOVR(560)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

D: Адресат

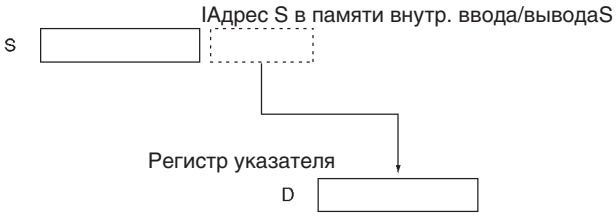
Адресатом должен быть регистр указателя (IR0...IR15).

Характеристики операндов

Область	S	D
Область CIO	CIO 0...CIO 6143 CIO 0.00...CIO 6143.15	---
Рабочая область	W0...W511 W0.00...W511.15	---
Область битов хранения	H0...H511 H0.00...H511.15	---
Область вспомогательных битов	A0...A447 A448...A959 A0.00...A447.15 A448.00...A959.15	---
Область таймеров	T0000...T4095 (флаг завершения)	---
Область счетчиков	C0000...C4095 (флаг завершения)	---
Флаг задачи	TK00...TK31	---
Область DM	D0...D32767	---
Косвенные адреса DM в двоичном формате	---	
Косвенные адреса DM в BCD-формате	---	
Константы	---	
Регистры данных	---	
Регистры указателей	---	IR0...IR15
Косвенная адресация с помощью регистров указателей	---	

Описание

Команда MOVR(560) выполняет поиск адреса объекта S в памяти ПЛК (абсолютного адреса) и записывает этот адрес в D (регистр указателей).



Если в S указан таймер или счетчик, команда MOVR(560) записывает в D адрес флага завершения таймера/счетчика в памяти ПЛК. Чтобы записать в D адрес текущего значения таймера/счетчика в памяти ПЛК, используйте команду MOVRW(561).

Флаги

Название	Обо-значение	Описание работы
Флаг ошибки	ER	ВЫКЛ или не изменяется
Флаг «Равно»	=	ВЫКЛ или не изменяется
Флаг «Меньше нуля»	N	ВЫКЛ или не изменяется

Меры предосторожности

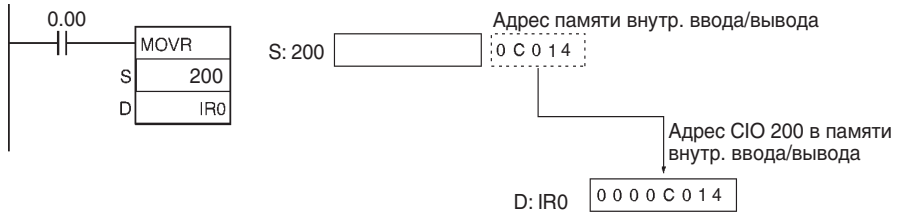
С помощью команды MOVR(560) невозможно записать адреса текущих значений таймера/счетчика в памяти ПЛК. Чтобы записать адреса текущих значений таймеров/счетчиков в памяти ПЛК, используйте команду MOVRW(561).

Невозможно предсказать, что будет содержаться в регистре указателя при вызове задачи обработки прерывания, если значение специально не задано. Прежде чем использовать регистр, задайте его значение с помощью команды MOVR(560) в самой задаче обработки прерывания.

Изменение содержимого регистра указателей или регистра данных в задаче обработки прерывания не влияет на содержимое регистра в циклической задаче.

Пример

Если в следующем примере бит CIO 0.00 включен, команда MOVR(560) записывает адрес слова CIO 200 в памяти ПЛК в регистр IR0.

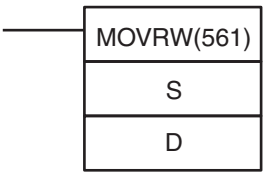


3-7-15 ПЕРЕДАТЬ ТЕКУЩЕЕ ЗНАЧЕНИЕ ТАЙМЕРА/СЧЕТЧИКА В РЕГИСТР: MOVRW(561)

Назначение

Запись адреса текущего значения указанного таймера/счетчика в памяти ПЛК в указанный регистр указателей. (Чтобы записать в регистр указателя адрес памяти ПЛК для слова, бита или флага завершения таймера/счетчика, используйте команду MOVR(560).)

Символ РКС



S: Источник (требуемый номер ТС)

D: Адресат (регистр указателя)

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	MOVR(561)
	Однократное выполнение по положительному фронту	@MOVR(561)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

D: Адресат

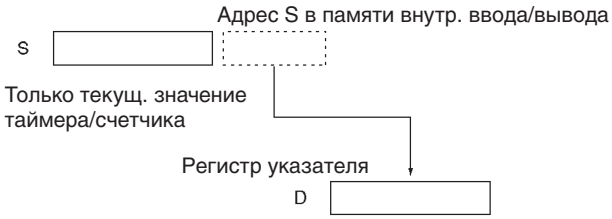
Адресатом должен быть регистр указателя (IR0...IR15).

Характеристики операндов

Область	S	D
Область CIO	---	
Рабочая область	---	
Область битов хранения	---	
Область вспомогательных битов	---	
Область таймеров	T0000...T4095 (текущее значение)	---
Область счетчиков	C0000...C4095 (текущее значение)	---
Область DM	---	
Косвенные адреса DM в двоичном формате	---	
Косвенные адреса DM в BCD-формате	---	
Константы	---	
Регистры данных	---	
Регистры указателей	---	IR0...IR15
Косвенная адресация с помощью регистров указателей	---	

Описание

Команда MOVRW(561) выполняет поиск адреса памяти ПЛК для текущего значения таймера/счетчика, указанного операндом S, и записывает этот адрес в D (регистр указателя).



Команда MOVRW(561) записывает в D адрес текущего значения таймера/счетчика в памяти ПЛК. Чтобы записать в D адрес флага завершения таймера/счетчика в памяти ПЛК, используйте команду MOVR(560).

Флаги

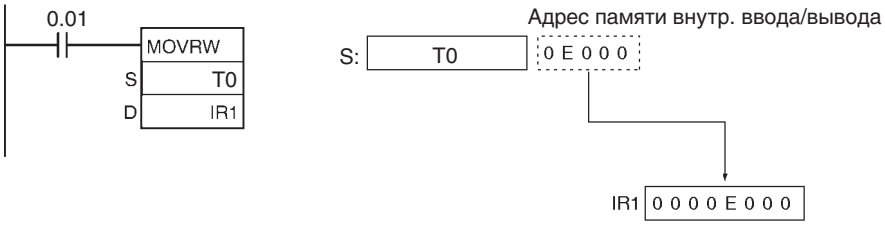
Название	Обо-значение	Описание работы
Флаг ошибки	ER	ВЫКЛ или не изменяется
Флаг «Равно»	=	ВЫКЛ или не изменяется
Флаг «Меньше нуля»	N	ВЫКЛ или не изменяется

Меры предосторожности

С помощью команды MOVRW(561) невозможно записать адреса памяти ПЛК для слов области данных, битов или флагов завершения таймеров/счетчиков. Для этих объектов используйте MOVR(560).

Пример

Если в следующем примере бит CIO 0.01 включен, команда MOVRW(561) записывает адрес текущего значения таймера T0 в памяти ПЛК в регистр IR1.



3-8 Команды сдвига данных

В данном разделе описаны команды, используемые для сдвига данных в пределах слова или из одних слов в другие в различных объемах и направлениях.

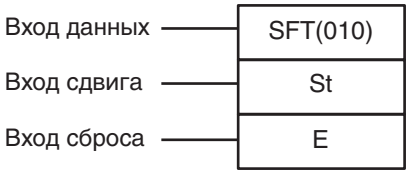
Команда	Мнемоническое обозначение	Код функции	Стр.
РЕГИСТР СДВИГА	SFT	010	302
РЕВЕРСИВНЫЙ РЕГИСТР СДВИГА	SFTR	084	304
АСИНХРОННЫЙ РЕГИСТР СДВИГА	ASFT	017	306
СДВИГ ПО СЛОВАМ	WSFT	016	309
АРИФМЕТИЧЕСКИЙ СДВИГ ВЛЕВО	ASL	025	311
СДВИГ ДВОЙНОГО СЛОВА ВЛЕВО	ASLL	570	312
АРИФМЕТИЧЕСКИЙ СДВИГ ВПРАВО	ASR	026	314
СДВИГ ДВОЙНОГО СЛОВА ВПРАВО	ASRL	571	316
ЦИКЛИЧЕСКИЙ СДВИГ ВЛЕВО	ROL	027	317
ЦИКЛИЧЕСКИЙ СДВИГ ДВОЙНОГО СЛОВА ВЛЕВО	ROLL	572	319
ЦИКЛИЧЕСКИЙ СДВИГ ВЛЕВО БЕЗ ПЕРЕНОСА	RLNC	574	324
ЦИКЛИЧЕСКИЙ СДВИГ ДВОЙНОГО СЛОВА ВЛЕВО БЕЗ ПЕРЕНОСА	RLNL	576	326
ЦИКЛИЧЕСКИЙ СДВИГ ВПРАВО	ROR	028	321
ЦИКЛИЧЕСКИЙ СДВИГ ДВОЙНОГО СЛОВА ВПРАВО	RORL	573	322
ЦИКЛИЧЕСКИЙ СДВИГ ВПРАВО БЕЗ ПЕРЕНОСА	RRNC	575	327
ЦИКЛИЧЕСКИЙ СДВИГ ДВОЙНОГО СЛОВА ВПРАВО БЕЗ ПЕРЕНОСА	RRNL	577	329
СДВИГ НА ОДИН РАЗРЯД ВЛЕВО	SLD	074	331
СДВИГ НА ОДИН РАЗРЯД ВПРАВО	SRD	075	332
СДВИГ N БИТОВ ДАННЫХ ВЛЕВО	NSFL	578	334
СДВИГ N БИТОВ ДАННЫХ ВПРАВО	NSFR	579	336
СДВИГ НА N БИТОВ ВЛЕВО	NASL	580	338
СДВИГ ДВОЙНОГО СЛОВА НА N БИТОВ ВЛЕВО	NSLL	582	341
СДВИГ НА N БИТОВ ВПРАВО	NASR	581	344
СДВИГ ДВОЙНОГО СЛОВА НА N БИТОВ ВПРАВО	NSRL	583	346

3-8-1 РЕГИСТР СДВИГА: SFT(010)

Назначение

Реализация регистра сдвига.

Символ РКС



St: Начальное слово

E: Конечное слово

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	SFT(010)
	Однократное выполнение по положительному фронту	Не предусмотрено
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограм-мы	Задачи обработки прерываний
Не допускается	OK	OK	OK

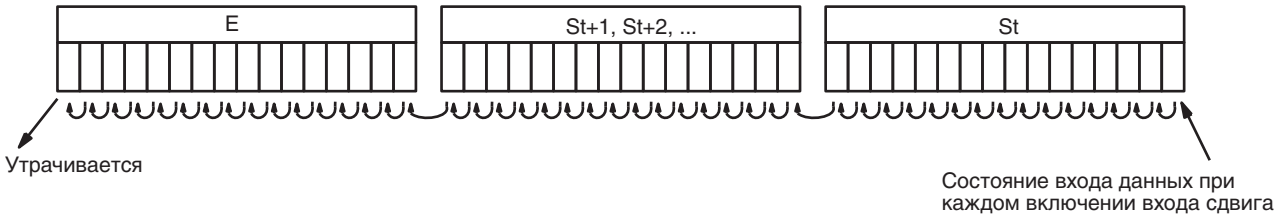
Примечание. St и E должны находиться в одной области данных.

Характеристики операндов

Область	St	E
Область CIO	CIO 0...CIO 6143	
Рабочая область	W0...W511	
Область битов хранения	H0...H511	
Область вспомогательных битов	A448...A959	
Область таймеров	---	
Область счетчиков	---	
Область DM	---	
Косвенные адреса DM в двоичном формате	---	
Косвенные адреса DM в BCD-формате	---	
Постоянные	---	
Регистры данных	---	
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…-2048…+2047 ,IR15 DR0…DR15, IR0…IR15	

Описание

Когда условие выполнения на входе сдвига переключается из «0» в «1», все данные в диапазоне St...E сдвигаются влево на один бит (от младшего бита к старшему биту), а в самый младший бит записывается состояние входа данных («0» или «1»).



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если косвенный адрес St и E, указанный регистром указателя, находится за пределами областей данных CIO, AR, HR или WR. Выключен во всех остальных случаях.

Меры предосторожности

Состояния битов, сдвигаемые за пределы регистра сдвига, утрачиваются.

При включении входа сброса все биты регистра сдвига, начиная с младшего бита указанного слова (St) и заканчивая старшим битом указанного слова (E), будут сброшены (т. е. будут находиться в состоянии «0»). Вход сброса имеет приоритет по отношению ко всем прочим входам.

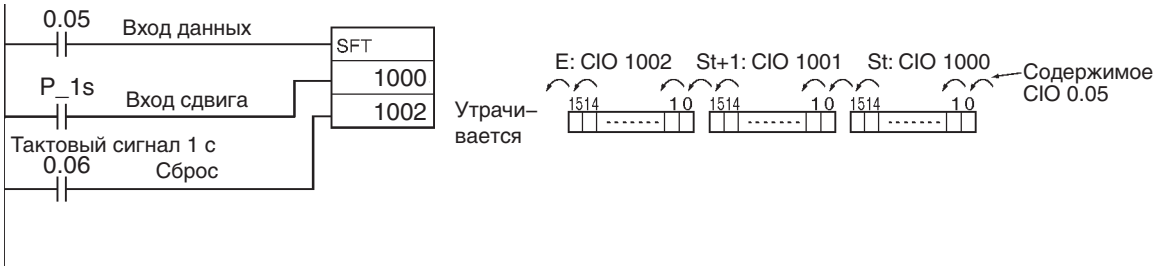
Адрес St должен быть меньше или равен адресу E, однако если указанный адрес St больше адреса E, ошибка не генерируется, сдвигается только одно слово данных по адресу St.

Если St и E указываются косвенно с помощью регистров указателей и фактические адреса в памяти ввода/вывода находятся за пределами областей памяти, предназначенных для хранения данных, будет сгенерирована ошибка и установится флаг ошибки.

Примеры

Регистр сдвига размером больше 16 битов

В следующем примере показан 48-битовый регистр сдвига, занимающий слова CIO 1000...CIO 1002. Используются тактовые импульсы с периодом 1 с, так что состояние бита CIO 0.05 каждую секунду сдвигается в регистр и перемещается по регистру из трех слов в направлении от бита CIO 1000.00 к биту CIO 1002.15.



Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

3-8-2 РЕВЕРСИВНЫЙ РЕГИСТР СДВИГА: SFTR(084)

Назначение

Реализация регистра сдвига, в котором данные сдвигаются либо вправо, либо влево.

Символ РКС



Варианты выполнения

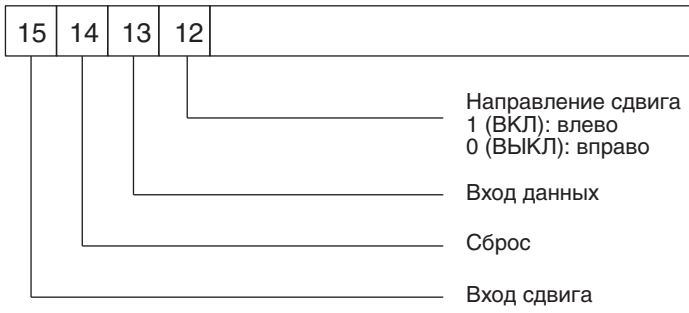
Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	SFTR(084)
	Однократное выполнение по положительному фронту	@SFTR(084)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

C: Управляющее слово



Примечание. St и E должны находиться в одной области данных.

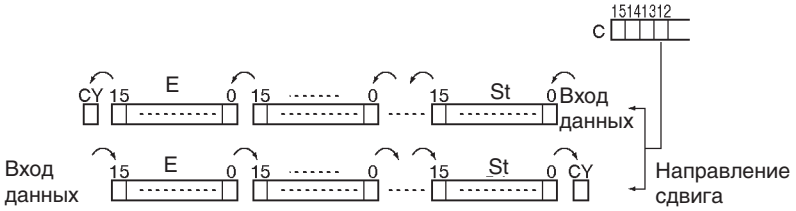
Характеристики операндов

Область	C	St	E
Область CIO	CIO 0...CIO 6143		
Рабочая область	W0...W511		
Область битов хранения	H0...H511		
Область вспомогательных битов	A0...A959	A448...A959	
Область таймеров	T0000...T4095		
Область счетчиков	C0000...C4095		
Область DM	D0...D32767		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	---		

Область	C	St	E
Регистры данных	DR0...DR15	---	
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…-2048…+2047 ,IR15 DR0…DR15, IR0…IR15 ,IR0+(++)…,IR15+(++) ,–(– –)IR0…, –(– –)IR15		

Описание

Когда условие выполнения на входе сдвига (бит 14 слова C) переключается из «0» в «1», все данные в диапазоне St...E сдвигаются в указанном направлении (оно указывается битом 12 слова C), при этом в младший или старший бит записывается состояние входа данных («0» или «1»). Состояние бита, сдвинутое за пределы регистра сдвига, записывается во флаг переноса (CY).



Флаги

Название	Обо-значение	Описание работы
Флаг ошибки	ER	Включен, если St больше E. Выключен во всех остальных случаях.
Флаг переноса	CY	Включен, если в него сдвинуто «1». Выключен, если в него сдвинуто «0». Выключен, если установлен бит сброса.

Меры предосторожности

Описанные выше операции сдвига выполняются, если не установлен бит сброса (бит 15 слова C).

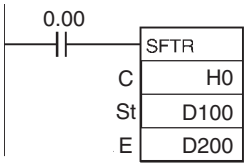
Установка бита сброса (бит 15 слова C) приводит к сбросу в «0» всех битов регистра сдвига в диапазоне St...E.

Если адрес St будет больше адреса E, будет сгенерирована ошибка и установится флаг ошибки.

Примеры

<u>Сдвиг данных</u>

Если вход сдвига H0.14 включается, когда бит CIO 0.00 = 1 и бит сброса H0.15 = 0, слова D100...D102 сдвигаются на один бит в направлении, указанном битом H0.12 (например, «1»: влево), а состояние входного бита H0.13 сдвигается в младший бит D100. Состояние бита 15 слова D102 при этом сдвигается во флаг переноса (CY).





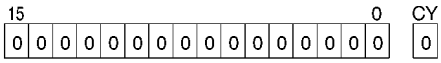
Очистка данных

Если бит H0.14 включается, когда бит CIO 0.00 = 1 и бит сброса H0.15 = 1, содержимое слов D100...D102 и флага переноса обнуляются.

Управление данными

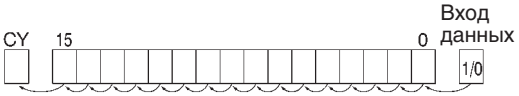
Очистка данных

Все биты слов St...E и флаг переноса принимают состояние «0», и пока бит входа сброса (бит 15 слова C) включен, никакие другие состояния не могут быть приняты.



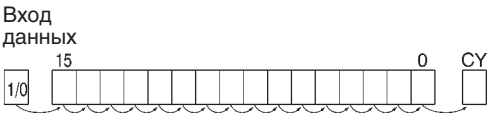
Сдвиг данных влево (от младшего бита к старшему)

Если бит входа сдвига (бит 14 слова C) включен, состояние входного бита (бит 13 слова C) сдвигается в бит 00 первого слова, а каждый из последующих битов сдвигается на один бит влево. Состояние бита 15 последнего слова сдвигается во флаг переноса.



Сдвиг данных вправо (от старшего бита к младшему)

Если бит входа сдвига (бит 14 слова C) включен, состояние входного бита (бит 13 слова C) сдвигается в бит 15 последнего слова, а каждый из последующих битов сдвигается на один бит вправо. Состояние бита 00 первого слова сдвигается во флаг переноса.



3-8-3 АСИНХРОННЫЙ РЕГИСТР СДВИГА: ASFT(017)

Назначение

Сдвиг всех ненулевых слов данных в пределах указанного диапазона слов в направлении слова St или слова E с замещением нулевых (0000 hex) слов данных.

Символ РКС



Варианты выполнения

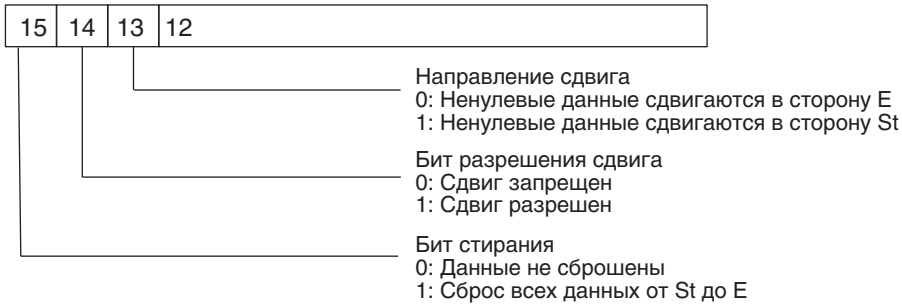
Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	ASFT(017)
	Однократное выполнение по положительному фронту	@ASFT(017)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

C: Управляющее слово



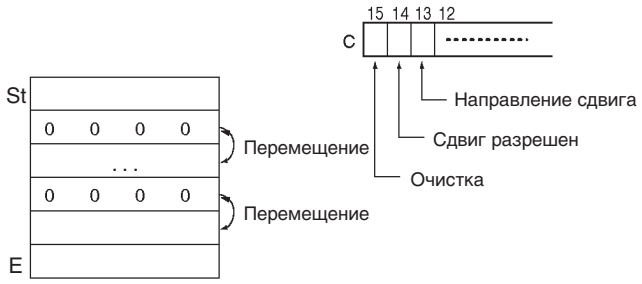
Примечание. St и E должны находиться в одной области данных.

Характеристики операндов

Область	C	St	E
Область CIO	CIO 0...CIO 6143		
Рабочая область	W0...W511		
Область битов хранения	H0...H511		
Область вспомогательных битов	A0...A959	A448...A959	
Область таймеров	T0000...T4095		
Область счетчиков	C0000...C4095		
Область DM	D0...D32767		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	---		
Регистры данных	DR0...DR15	---	
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…-2048…+2047 ,IR15 DR0…DR15, IR0…IR15 ,IR0+(++)…,IR15+(++) ,–(– –)IR0…, –(– –)IR15		

Описание

Если бит разрешения сдвига (бит 14 слова C) включен, все слова с ненулевым содержимым в диапазоне St...E сдвигаются на одно слово в направлении, указанном битом направления сдвига (бит 13 слова C), если слово в направлении сдвига содержит только нули. Если повторить выполнение команды ASFT(017) достаточное количество раз, все ненулевые слова будут собраны в одну группу, нулевые слова между ними будут исключены.





Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если St больше E. Включен, если указано выполнение в фоновом режиме и при этом выключен флаг доступности порта связи, соответствующий номеру порта связи, указанному в *Com Port number (Номер COM-порта)* для *Background Execution (Выполнение в фоновом режиме)* (только для CP1H). Выключен во всех остальных случаях.

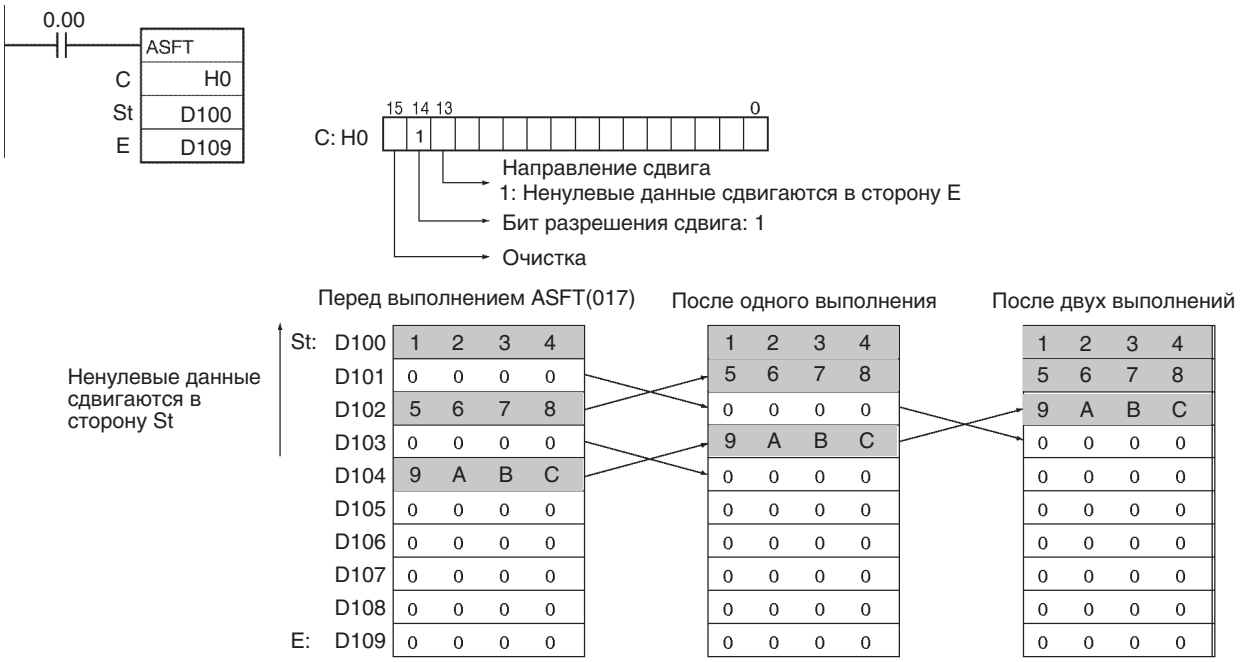
Меры предосторожности

Установка флага очистки (бит 15 слова C) приводит к сбросу в «0» всех битов регистра сдвига в диапазоне St...E. Флаг очистки обладает приоритетом по отношению к биту разрешения сдвига (бит 14 слова C).
Если адрес St будет больше адреса E, будет сгенерирована ошибка и установится флаг ошибки.

Примеры

Сдвиг данных:

Если бит разрешения сдвига H0.14 включается при включенном бите CIO 0.00, все слова с ненулевым содержимым в диапазоне D100...D109 сдвигаются в направлении, указанном битом направления сдвига H0.13 (например, состояние «1» означает сдвиг в направлении St), если слева от ненулевых данных имеется хотя бы одно слово, содержащее только нули.



3-8-4 СДВИГ ПО СЛОВАМ: WSFT(016)

Назначение

Пословный сдвиг данных в диапазоне слов от St по E.

Символ РКС



Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	WSFT(016)
	Однократное выполнение по положительному фронту	@WSFT(016)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

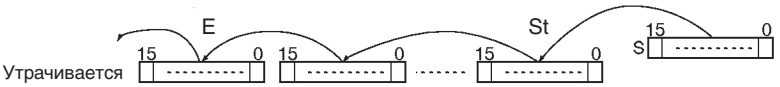
Примечание. St и E должны находиться в одной области данных.

Характеристики операндов

Область	S	St	E
Область CIO	CIO 0...CIO 6143		
Рабочая область	W0...W511		
Область битов хранения	H0...H511		
Область вспомогательных битов	A0...A959	A448...A959	
Область таймеров	T0000...T4095		
Область счетчиков	C0000...C4095		
Область DM	D0...D32767		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	#0000...#FFFF (двоичн.)	---	
Регистры данных	DR0...DR15	---	
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…-2048…+2047 ,IR15 DR0…DR15, IR0…IR15 ,IR0+(++)…,IR15+(++) ,–(– –)IR0…, –(– –)IR15		

Описание

Команда WSFT(016) сдвигает данные в диапазоне St...E целыми словами и помещает содержимое входного слова S в слово по адресу St. Содержимое слова по адресу E теряется.



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если St больше E. Выключен во всех остальных случаях.

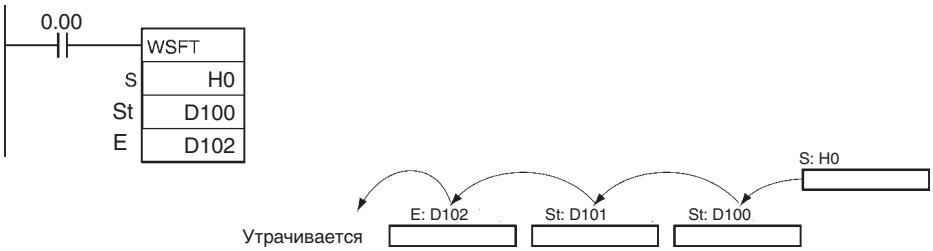
Меры предосторожности

Если адрес St будет больше адреса E, будет сгенерирована ошибка и установится флаг ошибки.

Примечание. При сдвиге большого объема данных команда выполняется довольно долго. Следите за тем, чтобы во время выполнения команды WSFT(016) не прерывалось питание, в противном случае операция сдвига может быть не завершена.

Примеры

Если бит CIO 0.00 включен, данные в диапазоне D100...D102 сдвигаются на одно слово в направлении слова E. Содержимое H0 записывается в D100, а содержимое D102 утрачивается.



3-8-5 АРИФМЕТИЧЕСКИЙ СДВИГ ВЛЕВО: ASL(025)

Назначение

Сдвиг содержимого Wd на один бит влево.

Символ РКС



Wd: Слово

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	ASL(025)
	Однократное выполнение по положительному фронту	@ASL(025)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

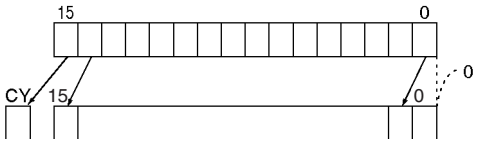
Характеристики операндов

Область	Wd
Область CIO	CIO 0...CIO 6143
Рабочая область	W0...W511
Область битов хранения	H0...H511
Область вспомогательных битов	A448...A959
Область таймеров	T0000...T4095
Область счетчиков	C0000...C4095
Область DM	D0...D32767
Косвенные адреса DM в двоичном формате	@ D0...@ D32767
Косвенные адреса DM в BCD-формате	*D0...*D32767
Постоянные	---
Регистры данных	DR0...DR15

Область	Wd
Регистры указателей	---
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…-2048…+2047 ,IR15 DR0…DR15, IR0…IR15 ,IR0+(++)…,IR15+(++) ,–(– –)IR0…, –(– –)IR15

Описание

Команда ASL(025) сдвигает содержимое слова Wd на один бит влево (от младшего бита к старшему биту). В младший бит записывается состояние «0», состояние старшего бита сдвигается во флаг переноса (CY).



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	ВЫКЛ
Флаг равенства	=	Включен, если результат операции сдвига = 0. Выключен во всех остальных случаях.
Флаг переноса	CY	Включен, если во флаг переноса (CY) сдвинуто «1». Выключен во всех остальных случаях.
Флаг «Меньше нуля»	N	Включен, если в результате сдвига старший бит = «1». Выключен во всех остальных случаях.

Меры предосторожности

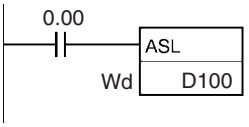
При выполнении ASL(025) флаг ошибки выключается.

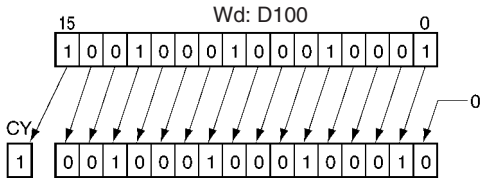
Если в результате операции сдвига слово Wd окажется равным «0», будет установлен флаг равенства.

Если в результате операции сдвига старший бит слова Wd окажется равным «1», будет установлен флаг отрицательного значения.

Примеры

Если бит CIO 0.00 включен, слово D100 будет сдвинуто на один бит влево. При этом в бит 00 слова D100 будет записано состояние «0», а содержимое бита 15 слова D100 сдвинется во флаг переноса (CY).



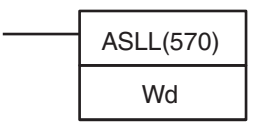


3-8-6 СДВИГ ДВОЙНОГО СЛОВА ВЛЕВО: ASLL(570)

Назначение

Сдвиг содержимого слов Wd и Wd + 1 на один бит влево.

Символ РКС



Wd: Слово

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	ASLL(570)
	Однократное выполнение по положительному фронту	@ASLL(570)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

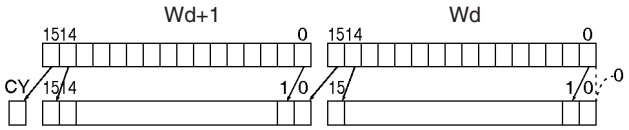
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	Wd
Область CIO	CIO 0...CIO 6142
Рабочая область	W0...W510
Область битов хранения	H0...H510
Область вспомогательных битов	A448...A958
Область таймеров	T0000...T4094
Область счетчиков	C0000...C4094
Область DM	D0...D32766
Косвенные адреса DM в двоичном формате	@ D0...@ D32767
Косвенные адреса DM в BCD-формате	*D0...*D32767
Постоянные	---
Регистры данных	---
Регистры указателей	---
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…-2048…+2047 ,IR15 DR0…DR15, IR0…IR15 ,IR0+(++)…,IR15+(++) ,–(– –)IR0…, –(– –)IR15

Описание

Команда ASLL(570) сдвигает содержимое слов Wd и Wd +1 на один бит влево (от младшего бита к старшему биту). В младший бит слова Wd записывается состояние «0», содержимое старшего бита двойного слова Wd и Wd +1 сдвигается во флаг переноса (CY).



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	ВЫКЛ
Флаг равенства	=	Включен, если результат операции сдвига = 0. Выключен во всех остальных случаях.

Название	Обозначение	Описание работы
Флаг переноса	CY	Включен, если во флаг переноса (CY) сдвинуто «1». Выключен во всех остальных случаях.
Флаг «Меньше нуля»	N	Включен, если в результате сдвига старший бит = «1». Выключен во всех остальных случаях.

Меры предосторожности

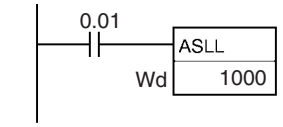
При выполнении ASLL(570) флаг ошибки выключается.

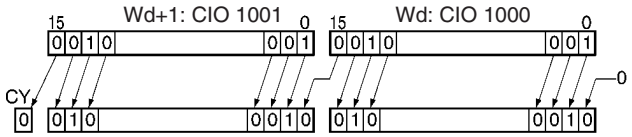
Если в результате операции сдвига содержимое слов Wd и Wd +1 окажется равным «0», будет установлен флаг равенства.

Если в результате операции сдвига старший бит слова Wd +1 окажется равным «1», будет установлен флаг отрицательного значения.

Примеры

Если бит CIO 0.01 включен, слова CIO 1000 и CIO 1001 сдвигаются на один бит влево. При этом в бит CIO 1000.00 записывается состояние «0», а содержимое CIO 1001.15 сдвигается во флаг переноса (CY).



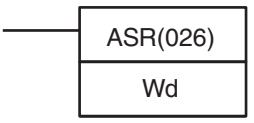


3-8-7 АРИФМЕТИЧЕСКИЙ СДВИГ ВПРАВО: ASR(026)

Назначение

Сдвиг содержимого Wd на один бит вправо.

Символ РКС



Wd: Слово

Варианты выполнения

Варианты выполнения		
	Выполнение в каждом цикле при включенном условии выполнения	ASR(026)
	Однократное выполнение по положительному фронту	@ASR(026)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

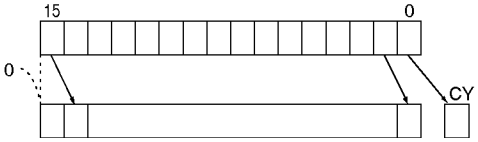
Характеристики операндов

Область	Wd
Область CIO	CIO 0...CIO 6143
Рабочая область	W0...W511
Область битов хранения	H0...H511

Область	Wd
Область вспомогательных битов	A448...A959
Область таймеров	T0000...T4095
Область счетчиков	C0000...C4095
Область DM	D0...D32767
Косвенные адреса DM в двоичном формате	@ D0...@ D32767
Косвенные адреса DM в BCD-формате	*D0...*D32767
Постоянные	---
Регистры данных	DR0...DR15
Регистры указателей	---
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…-2048…+2047 ,IR15 DR0…DR15, IR0…IR15 ,IR0+(++)…,IR15+(++) ,–(– –)IR0…, –(– –)IR15

Описание

Команда ASR(026) сдвигает содержимое слова Wd на один бит вправо (от старшего бита к младшему биту). В старший бит записывается состояние «0», содержимое младшего бита сдвигается во флаг переноса (CY).



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	ВЫКЛ
Флаг равенства	=	Включен, если результат операции сдвига = 0. Выключен во всех остальных случаях.
Флаг переноса	CY	Включен, если во флаг переноса (CY) сдвинуто «1». Выключен во всех остальных случаях.
Флаг «Меньше нуля»	N	ВЫКЛ

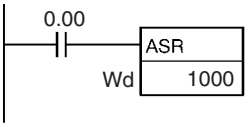
Меры предосторожности

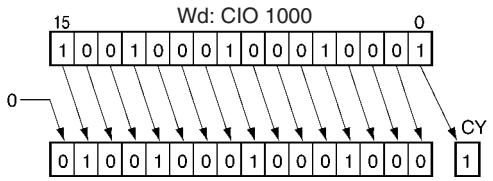
При выполнении ASR(026) выключаются флаг ошибки и флаг отрицательного значения.

Если в результате операции сдвига слово Wd окажется равным «0», будет установлен флаг равенства.

Примеры

Если бит CIO 0.00 включен, слово CIO 1000 сдвигается на один бит вправо. При этом в бит CIO 1000.15 записывается состояние «0», а содержимое бита CIO 1000.00 сдвигается во флаг переноса (CY).



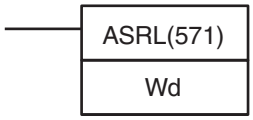


3-8-8 СДВИГ ДВОЙНОГО СЛОВА ВПРАВО: ASRL(571)

Назначение

Сдвиг содержимого слов Wd и Wd + 1 на один бит вправо.

Символ РКС



Wd: Слово

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	ASRL(571)
	Однократное выполнение по положительному фронту	@ASRL(571)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

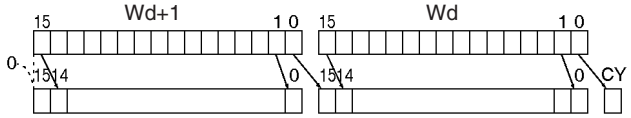
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	Wd
Область CIO	CIO 0...CIO 6142
Рабочая область	W0...W510
Область битов хранения	H0...H510
Область вспомогательных битов	A448...A958
Область таймеров	T0000...T4094
Область счетчиков	C0000...C4094
Область DM	D0...D32766
Косвенные адреса DM в двоичном формате	@ D0...@ D32767
Косвенные адреса DM в BCD-формате	*D0...*D32767
Постоянные	---
Регистры данных	---
Регистры указателей	---
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…-2048…+2047 ,IR15 DR0…DR15, IR0…IR15 ,IR0+(++)…,IR15+(++) ,–(– –)IR0…, –(– –)IR15

Описание

Команда ASRL(571) сдвигает содержимое слов Wd и Wd +1 на один бит вправо (от старшего бита к младшему биту). В старший бит Wd +1 записывается состояние «0», содержимое младшего бита слова Wd сдвигается во флаг переноса (CY).



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	ВЫКЛ
Флаг равенства	=	Включен, если результат операции сдвига = 0. Выключен во всех остальных случаях.
Флаг переноса	CY	Включен, если во флаг переноса (CY) сдвинуто «1». Выключен во всех остальных случаях.
Флаг «Меньше нуля»	N	ВЫКЛ

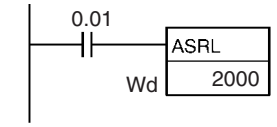
Меры предосторожности

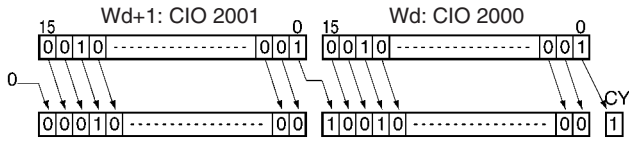
При выполнении ASRL (571) выключаются флаг ошибки и флаг отрицательного значения.

Если в результате операции сдвига содержимое слов Wd и Wd +1 окажется равным «0», будет установлен флаг равенства.

Примеры

Если бит CIO 0.01 включен, слова CIO 2000 и CIO 2001 сдвигаются на один бит вправо. При этом в бит CIO 2001.15 записывается состояние «0», а содержимое CIO 2000.00 сдвигается во флаг переноса (CY).



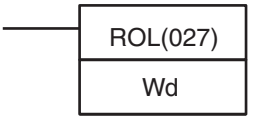


3-8-9 ЦИКЛИЧЕСКИЙ СДВИГ ВЛЕВО: ROL(027)

Назначение

Сдвиг на один бит влево всех битов слова Wd, включая флаг переноса (CY).

Символ РКС



Wd: Слово

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	ROL(027)
	Однократное выполнение по положительному фронту	@ROL(027)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

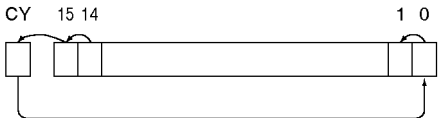
Область	Wd
Область CIO	CIO 0...CIO 6143
Рабочая область	W0...W511

Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

Область	Wd
Область битов хранения	H0...H511
Область вспомогательных битов	A448...A959
Область таймеров	T0000...T4095
Область счетчиков	C0000...C4095
Область DM	D0...D32767
Косвенные адреса DM в двоичном формате	@ D0...@ D32767
Косвенные адреса DM в BCD-формате	*D0...*D32767
Постоянные	---
Регистры данных	DR0...DR15
Регистры указателей	---
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…-2048…+2047 ,IR15 DR0…DR15, IR0…IR15 ,IR0+(++)…,IR15+(++) ,–(– –)IR0…, –(– –)IR15

Описание

Команда ROL(027) сдвигает влево (от младшего бита к старшему биту) все биты слова Wd, включая флаг переноса (CY).



Флаги

Название	Обо-значение	Описание работы
Флаг ошибки	ER	ВЫКЛ
Флаг равенства	=	Включен, если результат операции сдвига = 0. Выключен во всех остальных случаях.
Флаг переноса	CY	Включен, если во флаг переноса (CY) сдвинуто «1». Выключен во всех остальных случаях.
Флаг «Меньше нуля»	N	Включен, если в результате сдвига старший бит = «1». Выключен во всех остальных случаях.

Меры предосторожности

При выполнении ROL(027) флаг ошибки выключается.

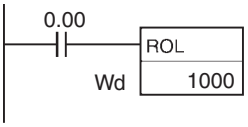
Если в результате операции сдвига слово Wd окажется равным «0», будет установлен флаг равенства.

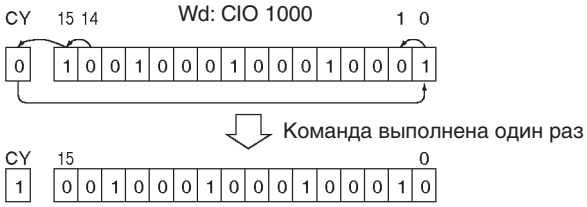
Если в результате операции сдвига старший бит слова Wd окажется равным «1», будет установлен флаг отрицательного значения.

Примечание. Флаг переноса можно установить или сбросить непосредственно перед выполнением данной команды с помощью команд STC(040) (Установка флага переноса) или CLC(041) (Сброс флага переноса).

Примеры

Если бит CIO 0.00 включен, слово CIO 1000 и флаг переноса (CY) сдвинутся на один бит влево. Содержимое бита CIO 1000.15 будет сдвинуто во флаг переноса (CY), а содержимое флага переноса сдвинется в бит CIO 1000.00.



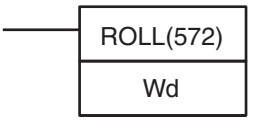


3-8-10 ЦИКЛИЧЕСКИЙ СДВИГ ДВОЙНОГО СЛОВА ВЛЕВО: ROLL(572)

Назначение

Сдвиг на один бит влево всех битов слов Wd и Wd +1, включая флаг переноса (CY).

Символ РКС



Wd: Слово

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	ROLL(572)
	Однократное выполнение по положительному фронту	@ROLL(572)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

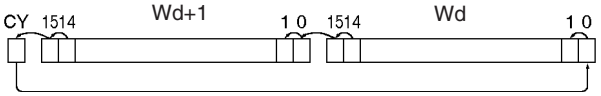
Характеристики операндов

Область	Wd
Область CIO	CIO 0...CIO 6142
Рабочая область	W0...W510
Область битов хранения	H0...H510
Область вспомогательных битов	A448...A958
Область таймеров	T0000...T4094
Область счетчиков	C0000...C4094
Область DM	D0...D32766
Косвенные адреса DM в двоичном формате	@ D0...@ D32767
Косвенные адреса DM в BCD-формате	*D0...*D32767

Область	Wd
Постоянные	---
Регистры данных	---
Регистры указателей	---
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…-2048…+2047 ,IR15 DR0…DR15, IR0…IR15 ,IR0+(++)…,IR15+(++) ,–(– –)IR0…, –(– –)IR15

Описание

Команда ROLL(572) сдвигает влево (от младшего бита к старшему биту) все биты слов Wd и Wd +1, включая флаг переноса (CY).



Флаги

Название	Обо-значение	Описание работы
Флаг ошибки	ER	ВЫКЛ
Флаг равенства	=	Включен, если результат операции сдвига = 0. Выключен во всех остальных случаях.
Флаг переноса	CY	Включен, если во флаг переноса (CY) сдвинуто «1». Выключен во всех остальных случаях.
Флаг «Меньше нуля»	N	Включен, если в результате сдвига старший бит = «1». Выключен во всех остальных случаях.

Меры предосторожности

При выполнении ROLL(572) флаг ошибки выключается.

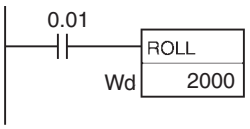
Если в результате операции сдвига содержимое слов Wd и Wd +1 окажется равным «0», будет установлен флаг равенства.

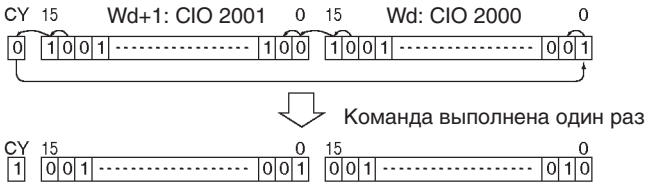
Если в результате операции сдвига старший бит слова Wd + 1 окажется равным «1», будет установлен флаг отрицательного значения.

Примечание. Флаг переноса можно установить или сбросить непосредственно перед выполнением данной команды с помощью команд STC(040) (Установка флага переноса) или CLC(041) (Сброс флага переноса).

Примеры

Если бит CIO 0.01 включен, слова CIO 2000 и CIO 2001, а также флаг переноса (CY) сдвинутся на один бит влево. Содержимое бита CIO 2001.15 будет сдвинуто во флаг переноса (CY), а содержимое флага переноса сдвинется в бит CIO 2000.00.



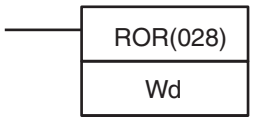


3-8-11 ЦИКЛИЧЕСКИЙ СДВИГ ВПРАВО: ROR(028)

Назначение

Сдвиг на один бит вправо всех битов слова Wd, включая флаг переноса (CY).

Символ РКС



Wd: Слово

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	ROR(028)
	Однократное выполнение по положительному фронту	@ROR(028)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

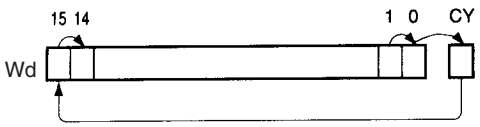
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	Wd
Область CIO	CIO 0...CIO 6143
Рабочая область	W0...W511
Область битов хранения	H0...H511
Область вспомогательных битов	A448...A959
Область таймеров	T0000...T4095
Область счетчиков	C0000...C4095
Область DM	D0...D32767
Косвенные адреса DM в двоичном формате	@ D0...@ D32767
Косвенные адреса DM в BCD-формате	*D0...*D32767
Постоянные	---
Регистры данных	DR0...DR15
Регистры указателей	---
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…-2048…+2047 ,IR15 DR0…DR15, IR0…IR15 ,IR0+(++)…,IR15+(++) ,–(– –)IR0…, –(– –)IR15

Описание

Команда ROR(028) сдвигает вправо (от старшего бита к младшему биту) все биты слова Wd, включая флаг переноса (CY).



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	ВЫКЛ
Флаг равенства	=	Включен, если результат операции сдвига = 0. Выключен во всех остальных случаях.
Флаг переноса	CY	Включен, если во флаг переноса (CY) сдвинуто «1». Выключен во всех остальных случаях.
Флаг «Меньше нуля»	N	Включен, если в результате сдвига старший бит = «1». Выключен во всех остальных случаях.

Меры предосторожности

При выполнении ROR(028) флаг ошибки выключается.

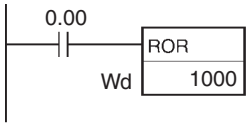
Если в результате операции сдвига слово Wd окажется равным «0», будет установлен флаг равенства.

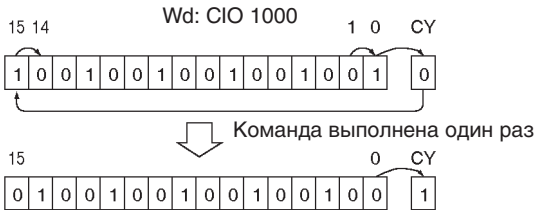
Если в результате операции сдвига старший бит слова Wd окажется равным «1», будет установлен флаг отрицательного значения.

Примечание. Флаг переноса можно установить или сбросить непосредственно перед выполнением данной команды с помощью команд STC(040) (Установка флага переноса) или CLC(041) (Сброс флага переноса).

Примеры

Если бит CIO 0.00 включен, слово CIO 1000 и флаг переноса (CY) сдвинутся на один бит вправо. Содержимое бита CIO 1000.00 будет сдвинуто во флаг переноса (CY), а содержимое флага переноса сдвинется в бит CIO 1000.15.



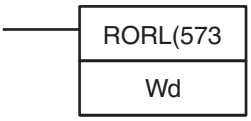


3-8-12 ЦИКЛИЧЕСКИЙ СДВИГ ДВОЙНОГО СЛОВА ВПРАВО: RORL(573)

Назначение

Сдвиг на один бит вправо всех битов слов Wd и Wd +1, включая флаг переноса (CY).

Символ РКС



Wd: Слово

Варианты выполнения

Варианты выполнения		
	Выполнение в каждом цикле при включенном условии выполнения	RORL(573)
	Однократное выполнение по положительному фронту	@RORL(573)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

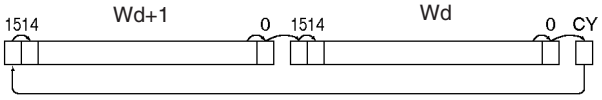
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	Wd
Область CIO	CIO 0...CIO 6142
Рабочая область	W0...W510
Область битов хранения	H0...H510
Область вспомогательных битов	A448...A958
Область таймеров	T0000...T4094
Область счетчиков	C0000...C4094
Область DM	D0...D32766
Косвенные адреса DM в двоичном формате	@ D0...@ D32767
Косвенные адреса DM в BCD-формате	*D0...*D32767
Постоянные	---
Регистры данных	---
Регистры указателей	---
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…-2048…+2047 ,IR15 DR0…DR15, IR0…IR15 ,IR0+(++)…,IR15+(++) ,–(– –)IR0…, –(– –)IR15

Описание

Команда RORL(573) сдвигает вправо (от старшего бита к младшему биту) все биты слов Wd и Wd +1, включая флаг переноса (CY).



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	ВЫКЛ
Флаг равенства	=	Включен, если результат операции сдвига = 0. Выключен во всех остальных случаях.
Флаг переноса	CY	Включен, если во флаг переноса (CY) сдвинуто «1». Выключен во всех остальных случаях.
Флаг «Меньше нуля»	N	Включен, если в результате сдвига старший бит = «1». Выключен во всех остальных случаях.

Меры предосторожности

При выполнении RORL(573) флаг ошибки выключается.

Если в результате операции сдвига содержимое слов Wd и Wd +1 окажется равным «0», будет установлен флаг равенства.

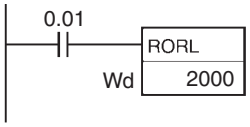
Если в результате операции сдвига старший бит слова Wd + 1 окажется равным «1», будет установлен флаг отрицательного значения.

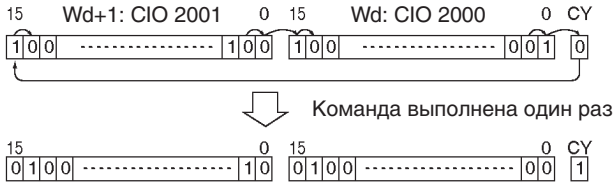
Примечание. Флаг переноса можно установить или сбросить непосредственно перед выполнением данной команды с помощью команд STC(040) (Установка флага переноса) или CLC(041) (Сброс флага переноса).

Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

Примеры

Если бит CIO 0.01 включен, слова CIO 2000 и CIO 2001, а также флаг переноса (CY) сдвинутся на один бит вправо. Содержимое бита CIO 2000.00 будет сдвинуто во флаг переноса (CY), а содержимое флага переноса сдвинется в бит CIO 2001.15.



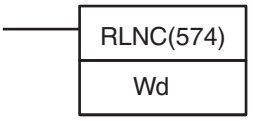


3-8-13 ЦИКЛИЧЕСКИЙ СДВИГ ВЛЕВО БЕЗ ПЕРЕНОСА: RLNC(574)

Назначение

Сдвиг на один бит влево всех битов слова Wd, за исключением флага переноса (CY).

Символ РКС



Wd: Слово

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	RLNC(574)
	Однократное выполнение по положительному фронту	@RLNC(574)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

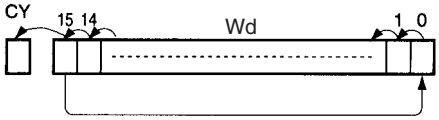
Характеристики операндов

Область	Wd
Область CIO	CIO 0...CIO 6143
Рабочая область	W0...W511
Область битов хранения	H0...H511
Область вспомогательных битов	A448...A959
Область таймеров	T0000...T4095
Область счетчиков	C0000...C4095
Область DM	D0...D32767
Косвенные адреса DM в двоичном формате	@ D0...@ D32767
Косвенные адреса DM в BCD-формате	*D0...*D32767
Постоянные	---

Область	Wd
Регистры данных	DR0...DR15
Регистры указателей	---
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…-2048…+2047 ,IR15 DR0…DR15, IR0…IR15 ,IR0+(++)…,IR15+(++) ,–(– –)IR0…, –(– –)IR15

Описание

Команда RLNC(574) сдвигает влево (от младшего бита к старшему биту) все биты слова Wd. Содержимое старшего бита слова Wd сдвигается в младший бит и во флаг переноса (CY).



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	ВЫКЛ
Флаг равенства	=	Включен, если результат операции сдвига = 0. Выключен во всех остальных случаях.
Флаг переноса	CY	Включен, если во флаг переноса (CY) сдвинуто «1». Выключен во всех остальных случаях.
Флаг «Меньше нуля»	N	Включен, если в результате сдвига старший бит = «1». Выключен во всех остальных случаях.

Меры предосторожности

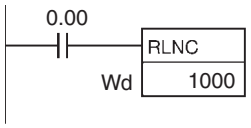
При выполнении RLNC(574) флаг ошибки выключается.

Если в результате операции сдвига слово Wd окажется равным «0», будет установлен флаг равенства.

Если в результате операции сдвига старший бит слова Wd окажется равным «1», будет установлен флаг отрицательного значения.

Примеры

Если бит CIO 0.00 включен, слово CIO 1000 сдвигается на один бит влево (за исключением флага переноса (CY)). Содержимое бита CIO 1000.15 будет сдвинуто в CIO 1000.00.



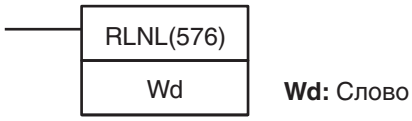


3-8-14 ЦИКЛИЧЕСКИЙ СДВИГ ДВОЙНОГО СЛОВА ВЛЕВО БЕЗ ПЕРЕНОСА: RLNL(576)

Назначение

Сдвиг на один бит влево всех битов слов Wd и Wd +1, за исключением флага переноса (CY).

Символ РКС



Wd: Слово

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	RLNL(576)
	Однократное выполнение по положительному фронту	@RLNL(576)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

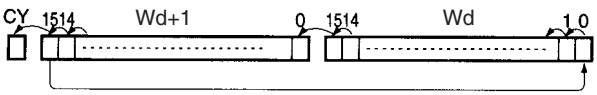
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	Wd
Область CIO	CIO 0...CIO 6142
Рабочая область	W0...W510
Область битов хранения	H0...H510
Область вспомогательных битов	A448...A958
Область таймеров	T0000...T4094
Область счетчиков	C0000...C4094
Область DM	D0...D32766
Косвенные адреса DM в двоичном формате	@ D0...@ D32767
Косвенные адреса DM в BCD-формате	*D0...*D32767
Постоянные	---
Регистры данных	---
Регистры указателей	---
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…-2048…+2047 ,IR15 DR0…DR15, IR0…IR15 ,IR0+(++)…,IR15+(++) ,–(– –)IR0…, –(– –)IR15

Описание

Команда RLNL(576) сдвигает влево (от младшего бита к старшему биту) все биты слов Wd и Wd +1. Содержимое старшего бита слова Wd +1 сдвигается в младший бит слова Wd и во флаг переноса (CY).



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	ВЫКЛ
Флаг равенства	=	Включен, если результат операции сдвига = 0. Выключен во всех остальных случаях.
Флаг переноса	CY	Включен, если во флаг переноса (CY) сдвинуто «1». Выключен во всех остальных случаях.
Флаг «Меньше нуля»	N	Включен, если в результате сдвига старший бит = «1». Выключен во всех остальных случаях.

Меры предосторожности

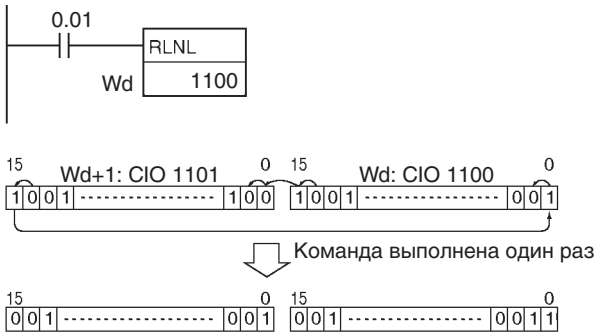
При выполнении RLNL(576) флаг ошибки выключается.

Если в результате операции сдвига содержимое слов Wd и Wd +1 окажется равным «0», будет установлен флаг равенства.

Если в результате операции сдвига старший бит слова Wd + 1 окажется равным «1», будет установлен флаг отрицательного значения.

Примеры

Если бит CIO 0.01 включен, слова CIO 1100 и CIO 1101 сдвигаются на один бит влево (за исключением флага переноса (CY)). Содержимое бита CIO 1101.15 будет сдвинуто в CIO 1100.00.

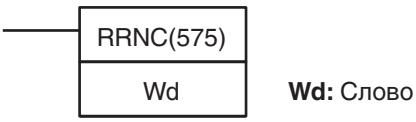


3-8-15 ЦИКЛИЧЕСКИЙ СДВИГ ВПРАВО БЕЗ ПЕРЕНОСА: RRNC(575)

Назначение

Сдвиг на один бит вправо всех битов слова Wd, за исключением флага переноса (CY). Содержимое младшего бита слова Wd сдвигается в старший бит и во флаг переноса (CY).

Символ РКС



Wd: Слово

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	RRNC(575)
	Однократное выполнение по положительному фронту	@RRNC(575)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

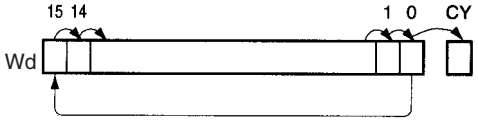
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	Wd
Область CIO	CIO 0...CIO 6143
Рабочая область	W0...W511
Область битов хранения	H0...H511
Область вспомогательных битов	A448...A959
Область таймеров	T0000...T4095
Область счетчиков	C0000...C4095
Область DM	D0...D32767
Косвенные адреса DM в двоичном формате	@ D0...@ D32767
Косвенные адреса DM в BCD-формате	*D0...*D32767
Постоянные	---
Регистры данных	DR0...DR15
Регистры указателей	---
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…-2048…+2047 ,IR15 DR0…DR15, IR0…IR15 ,IR0+(++)…,IR15+(++) ,–(– –)IR0…, –(– –)IR15

Описание

Команда RRNC(575) сдвигает вправо (от старшего бита к младшему биту) все биты слова Wd, за исключением флага переноса (CY).



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	ВЫКЛ
Флаг равенства	=	Включен, если результат операции сдвига = 0. Выключен во всех остальных случаях.
Флаг переноса	CY	Включен, если во флаг переноса (CY) сдвинуто «1». Выключен во всех остальных случаях.
Флаг «Меньше нуля»	N	Включен, если в результате сдвига старший бит = «1». Выключен во всех остальных случаях.

Меры предосторожности

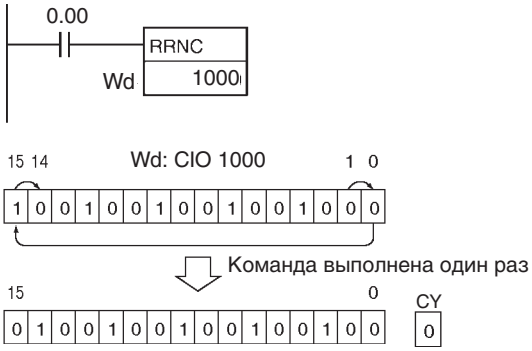
При выполнении команды RRNC(575) флаг ошибки выключается.

Если в результате операции сдвига слово Wd окажется равным «0», будет установлен флаг равенства.

Если в результате операции сдвига старший бит слова Wd окажется равным «1», будет установлен флаг отрицательного значения.

Примеры

Если бит CIO 0.00 включен, слово CIO 1000 сдвигается на один бит вправо (за исключением флага переноса (CY)). Содержимое бита CIO 1000.00 будет сдвинуто в CIO 1000.15.

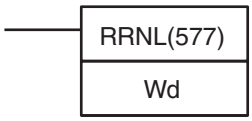


3-8-16 ЦИКЛИЧЕСКИЙ СДВИГ ДВОЙНОГО СЛОВА ВПРАВО БЕЗ ПЕРЕНОСА: RRNL(577)

Назначение

Сдвиг на один бит вправо всех битов слов Wd и Wd +1, за исключением флага переноса (CY). Содержимое младшего бита слова Wd + 1 сдвигается в старший бит слова Wd и во флаг переноса (CY).

Символ РКС



Wd: Слово

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	RRNL(577)
	Однократное выполнение по положительному фронту	@RRNL(577)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

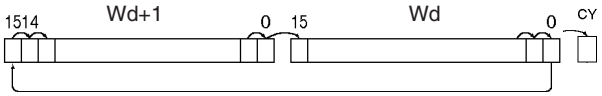
Область	Wd
Область CIO	CIO 0...CIO 6142
Рабочая область	W0...W510
Область битов хранения	H0...H510
Область вспомогательных битов	A448...A958
Область таймеров	T0000...T4094

Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

Область	Wd
Область счетчиков	C0000...C4094
Область DM	D0...D32766
Косвенные адреса DM в двоичном формате	@ D0...@ D32767
Косвенные адреса DM в BCD-формате	*D0...*D32767
Постоянные	---
Регистры данных	---
Регистры указателей	---
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…-2048…+2047 ,IR15 DR0…DR15, IR0…IR15 ,IR0+(++)…,IR15+(++) ,–(– –)IR0…, –(– –)IR15

Описание

Команда RRNL(577) сдвигает вправо (от старшего бита к младшему биту) все биты слов Wd и Wd +1, за исключением флага переноса (CY).



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	ВЫКЛ
Флаг равенства	=	Включен, если результат операции сдвига = 0. Выключен во всех остальных случаях.
Флаг переноса	CY	Включен, если во флаг переноса (CY) сдвинуто «1». Выключен во всех остальных случаях.
Флаг «Меньше нуля»	N	Включен, если в результате сдвига старший бит = «1». Выключен во всех остальных случаях.

Меры предосторожности

При выполнении RRNL(577) флаг ошибки выключается.

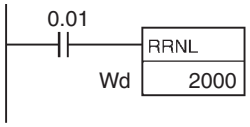
Если в результате операции сдвига содержимое слов Wd и Wd +1 окажется равным «0», будет установлен флаг равенства.

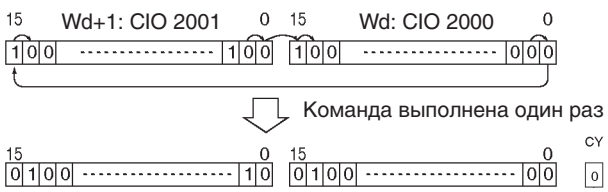
Если в результате операции сдвига старший бит слова Wd + 1 окажется равным «1», будет установлен флаг отрицательного значения.

Примечание. Флаг переноса можно установить или сбросить непосредственно перед выполнением данной команды с помощью команд STC(040) (Установка флага переноса) или CLC(041) (Сброс флага переноса).

Примеры

Если бит CIO 0.01 включен, слова CIO 2000 и CIO 2001 сдвигаются на один бит вправо (за исключением флага переноса (CY)). Содержимое бита CIO 2001.00 будет сдвинуто в CIO 2000.15.



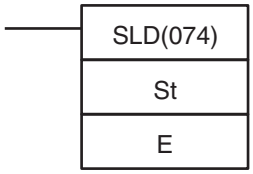


3-8-17 СДВИГ НА ОДИН РАЗРЯД ВЛЕВО: SLD(074)

Назначение

Сдвиг данных на один разряд (4 бита) влево.

Символ РКС



St: Начальное слово

E: Конечное слово

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	SLD(074)
	Однократное выполнение по положительному фронту	@SLD(074)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

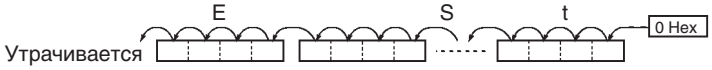
Примечание. St и E должны находиться в одной области данных.

Характеристики операндов

Область	St	E
Область CIO	CIO 0...CIO 6143	
Рабочая область	W0...W511	
Область битов хранения	H0...H511	
Область вспомогательных битов	A448...A959	
Область таймеров	T0000...T4095	
Область счетчиков	C0000...C4095	
Область DM	D0...D32767	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Постоянные	---	
Регистры данных	---	
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…-2048…+2047 ,IR15 DR0…DR15, IR0…IR15 ,IR0+(++)…,IR15+(++) ,–(– –)IR0…, –(– –)IR15	

Описание

Команда SLD(074) сдвигает данные в диапазоне St...E на один разряд (4 бита) влево. В младший разряд (биты 3...0 слова St) записывается значение «0», содержимое старшего разряда (биты 15...12 слова E) утрачивается.



Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

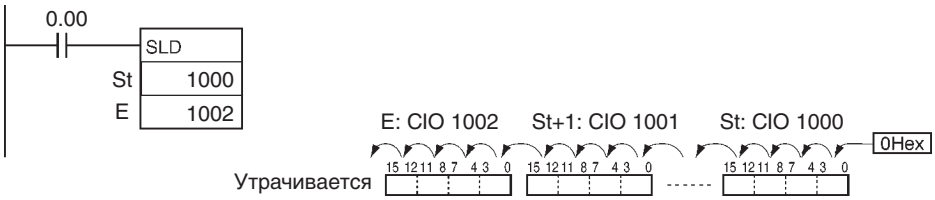
Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если St больше E. Выключен во всех остальных случаях.

Меры предосторожности Если адрес St будет больше адреса E, будет сгенерирована ошибка и установится флаг ошибки.

Примечание. При сдвиге большого объема данных команда выполняется довольно долго. Следите за тем, чтобы во время выполнения команды SLD(074) не прерывалось питание, в противном случае операция сдвига может быть не завершена.

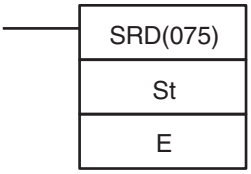
Примеры Если бит CIO 0.00 включен, слова CIO 1000...CIO 1002 сдвигаются на один разряд (4 бита) влево. При этом в биты 0...3 слова CIO 1000 записывается значение «0», а содержимое битов 12...15 слова CIO 1002 утрачивается.



3-8-18 СДВИГ НА ОДИН РАЗРЯД ВПРАВО: SRD(075)

Назначение Сдвиг данных на один разряд (4 бита) вправо.

Символ РКС



St: Начальное слово

E: Конечное слово

Варианты выполнения

Варианты выполнения		
	Выполнение в каждом цикле при включенном условии выполнения	SRD(075)
	Однократное выполнение по положительному фронту	@SRD(075)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Примечание. St и E должны находиться в одной области данных.

Характеристики операндов

Область	St	E
Область CIO	CIO 0...CIO 6143	
Рабочая область	W0...W511	
Область битов хранения	H0...H511	

Область	St	E
Область вспомогательных битов	A448...A959	
Область таймеров	T0000...T4095	
Область счетчиков	C0000...C4095	
Область DM	D0...D32767	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Постоянные	---	
Регистры данных	---	
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…-2048…+2047 ,IR15 DR0…DR15, IR0…IR15 ,IR0+(++)…,IR15+(++) ,–(– –)IR0…, –(– –)IR15	

Описание

Команда SRD(075) сдвигает данные в диапазоне St...E на один разряд (4 бита) вправо. В старший разряд (биты 15...12 слова E) записывается значение «0», содержимое младшего разряда (биты 3...0 слова St) утрачивается.



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если St больше E. Выключен во всех остальных случаях.

Меры предосторожности

Если адрес St будет больше адреса E, будет сгенерирована ошибка и установится флаг ошибки.

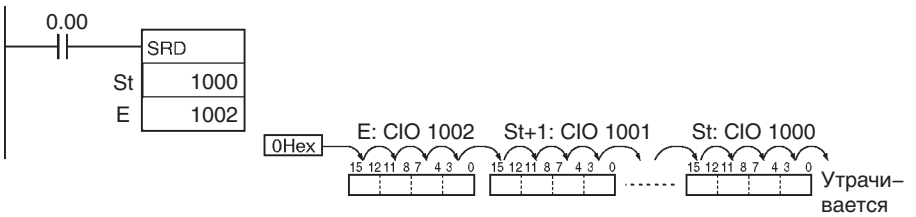
При выполнении команды SRD(075) выключаются флаг ошибки и флаг отрицательного значения.

Примечание.

При сдвиге большого объема данных команда выполняется довольно долго. Всегда следите за тем, чтобы во время выполнения SRD(075) не прерывалось питание, в противном случае операция сдвига может быть не завершена.

Примеры

Если бит CIO 0.00 включен, слова CIO 1000 и CIO 1002 сдвигаются на один разряд (4 бита) вправо. При этом в биты 12...15 слова CIO 1002 записывается значение «0», а содержимое битов 0...3 слова CIO 1000 утрачивается.

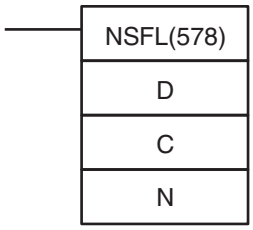


3-8-19 СДВИГ N БИТОВ ДАННЫХ ВЛЕВО: NSFL(578)

Назначение

Сдвиг указанного количества битов влево.

Символ РКС



D: Начальное слово для сдвига

C: Первый бит

N: Длина сдвигаемых данных

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	NSFL(578)
	Однократное выполнение по положительному фронту	@NSFL(578)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

C: 0000...000F hex (0...15)
N: 0000...FFFF hex (0...65535)

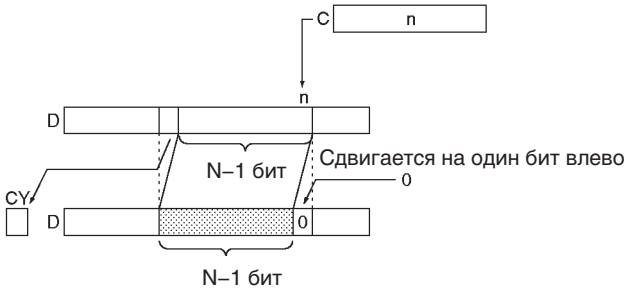
Примечание. Все слова в регистре сдвига должны принадлежать одной области.

Характеристики операндов

Область	D	C	N
Область CIO	CIO 0...CIO 6143		
Рабочая область	W0...W511		
Область битов хранения	H0...H511		
Область вспомогательных битов	A448...A959	A0...A959	
Область таймеров	T0000...T4095		
Область счетчиков	C0000...C4095		
Область DM	D0...D32767		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	---	#0000...#000F (двоичный) или &0...&15	#0000...#FFFF (двоичный) или &0...&65535
Регистры данных	---	DR0...DR15	
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…-2048…+2047 ,IR15 DR0…DR15, IR0…IR15 ,IR0+(++)…,IR15+(++) ,–(– –)IR0…, –(– –)IR15		

Описание

Команда NSFL(578) сдвигает на один бит влево (в направлении старшего слова и старшего бита) указанное количество битов (N), начиная с бита в позиции (C) младшего слова (D). В начальный бит записывается состояние «0», содержимое старшего бита области сдвига сдвигается во флаг переноса (CY).



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если значение C выходит за диапазон 0000...000F hex. Выключен во всех остальных случаях.
Флаг переноса	CY	Включен, если во флаг переноса (CY) сдвинуто «1». Выключен во всех остальных случаях.

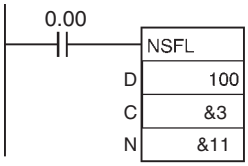
Меры предосторожности

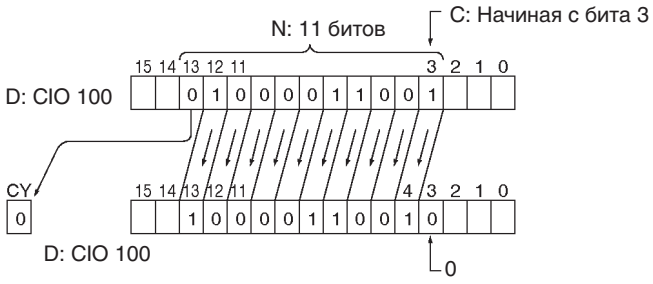
При нулевом количестве сдвигаемых битов (N = 0) состояние начального бита будет скопировано во флаг переноса (CY), но само при этом не изменится.

Изменяются только состояния битов в пределах области сдвига.

Примеры

Если бит CIO 0.00 включен, все биты, начиная с бита 3 и до конца диапазона, заданного количеством сдвигаемых битов (B hex), сдвигаются на один бит влево (от младшего бита к старшему биту). При этом в бит 3 слова CIO 100 будет записано состояние «0». Содержимое старшего бита области сдвига (бит 13 слова CIO 100) будет скопировано во флаг переноса (CY).





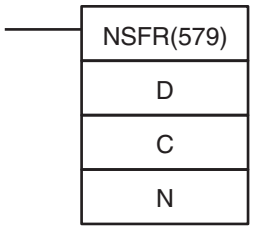
Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

3-8-20 СДВИГ N БИТОВ ДАННЫХ ВПРАВО: NSFR(579)

Назначение

Сдвиг указанного количества битов вправо.

Символ РКС



D: Начальное слово для сдвига

C: Первый бит

N: Длина сдвигаемых данных

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	NSFR(579)
	Однократное выполнение по положительному фронту	@NSFR(579)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

C: 0000...000F hex (0...15)
N: 0000...FFFF hex (0...65535)

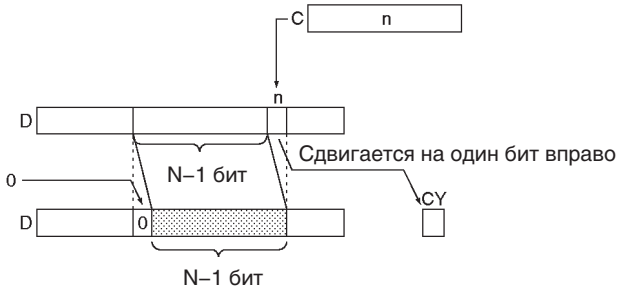
Примечание. Все слова в регистре сдвига должны принадлежать одной области.

Характеристики операндов

Область	D	C	N
Область CIO	CIO 0...CIO 6143		
Рабочая область	W0...W511		
Область битов хранения	H0...H511		
Область вспомогательных битов	A448...A959	A0...A959	
Область таймеров	T0000...T4095		
Область счетчиков	C0000...C4095		
Область DM	D0...D32767		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	---	#0000...#000F (двоичный) или &0...&15	#0000...#FFFF (двоичный) или &0...&65535
Регистры данных	---	DR0...DR15	
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…-2048…+2047 ,IR15 DR0…DR15, IR0…IR15 ,IR0+(++)…,IR15+(++) ,–(– –)IR0…, –(– –)IR15		

Описание

Команда NSFR(579) сдвигает на один бит вправо (в направлении младшего слова и младшего бита) указанное количество битов (N), начиная с бита в позиции (C) младшего слова (D). В начальный бит записывается состояние «0», содержимое младшего бита области сдвига сдвигается во флаг переноса (CY).



Флаги

Название	Обо-значение	Описание работы
Флаг ошибки	ER	Включен, если значение C выходит за диапазон 0000...000F hex. Выключен во всех остальных случаях.
Флаг переноса	CY	Включен, если во флаг переноса (CY) сдвинуто «1». Выключен во всех остальных случаях.

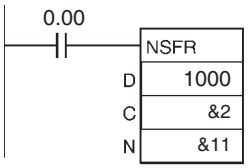
Меры предосторожности

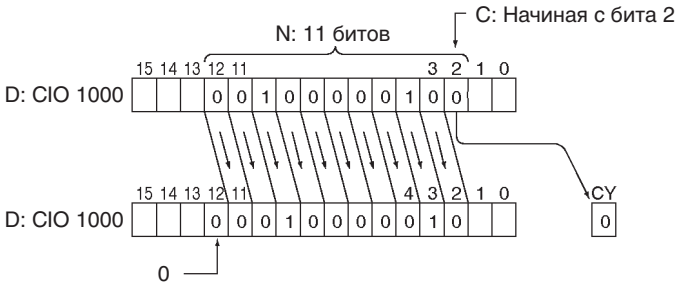
При нулевом количестве сдвигаемых битов (N = 0) состояние начального бита будет скопировано во флаг переноса (CY), но само при этом не изменится.

Изменяются только состояния битов в пределах области сдвига.

Примеры

Если бит CIO 0.00 включен, все биты, начиная с бита 2 и до конца диапазона (всего 11 битов (B hex)), сдвигаются на один бит вправо (от старшего бита к младшему биту). При этом в бит 12 слова CIO 1000 записывается состояние «0». Содержимое младшего бита области сдвига (бит 2 слова CIO 1000) копируется во флаг переноса (CY).





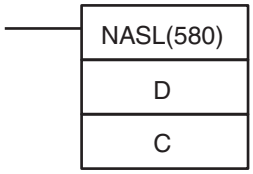
Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

3-8-21 СДВИГ НА N БИТОВ ВЛЕВО: NASL(580)

Назначение

Сдвигает содержимое указанного слова (16 битов) влево на указанное количество битов.

Символ РКС



D: Сдвигаемое слово

C: Управляющее слово

Варианты выполнения

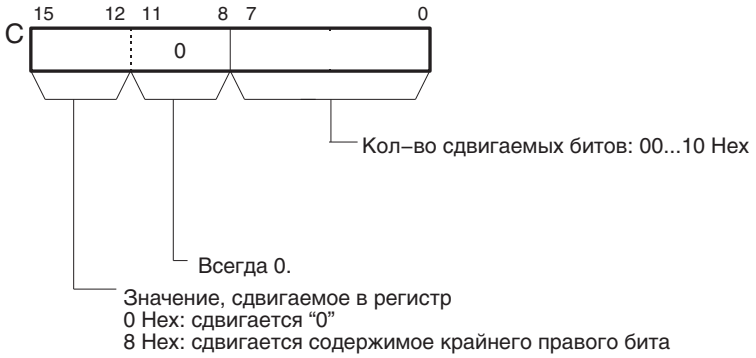
Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	NASL(580)
	Однократное выполнение по положительному фронту	@NASL(580)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

C: Управляющее слово



Характеристики операндов

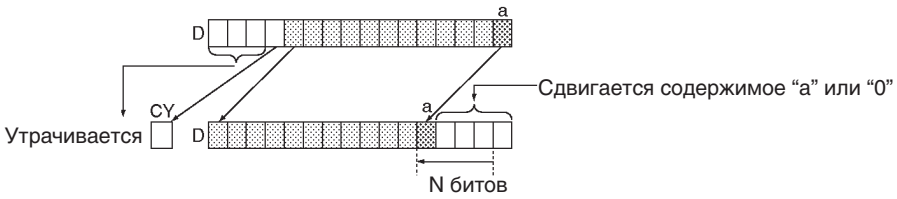
Область	D	C
Область CIO	CIO 0...CIO 6143	
Рабочая область	W0...W511	
Область битов хранения	H0...H511	
Область вспомогательных битов	A448...A959	A0...A959
Область таймеров	T0000...T4095	
Область счетчиков	C0000...C4095	
Область DM	D0...D32767	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Постоянные	---	Только указанные значения

Область	D	C
Регистры данных	DR0...DR15	
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…-2048…+2047 ,IR15 DR0…DR15, IR0…IR15 ,IR0+(++)…,IR15+(++) ,–(– –)IR0…, –(– –)IR15	

Описание

Команда NASL(580) сдвигает слово D на указанное количество (указывается в C) двоичных битов влево (от младшего бита к старшему биту). В указанное количество битов сдвигаемого слова, начиная с младшего бита, записываются нули или состояние младшего бита.





Флаги

Название	Обо-значение	Описание работы
Флаг ошибки	ER	Включен, если управляющее слово C (количество сдвигаемых битов) выходит за допустимый диапазон. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если результат операции сдвига = 0. Выключен во всех остальных случаях.
Флаг переноса	CY	Включен, если во флаг переноса (CY) сдвинуто «1». Выключен во всех остальных случаях.
Флаг «Меньше нуля»	N	Включен, если в результате сдвига старший бит = «1». Выключен во всех остальных случаях.

Меры предосторожности

Содержимое всех битов, сдвигаемых за пределы указанного слова, утрачивается, за исключением самого последнего бита, который сдвигается во флаг переноса (CY).

При нулевом количестве сдвигаемых битов (указанном в C) данные сдвинуты не будут. Однако соответствующие флаги все равно будут установлены либо сброшены в соответствии со значением указанного слова.

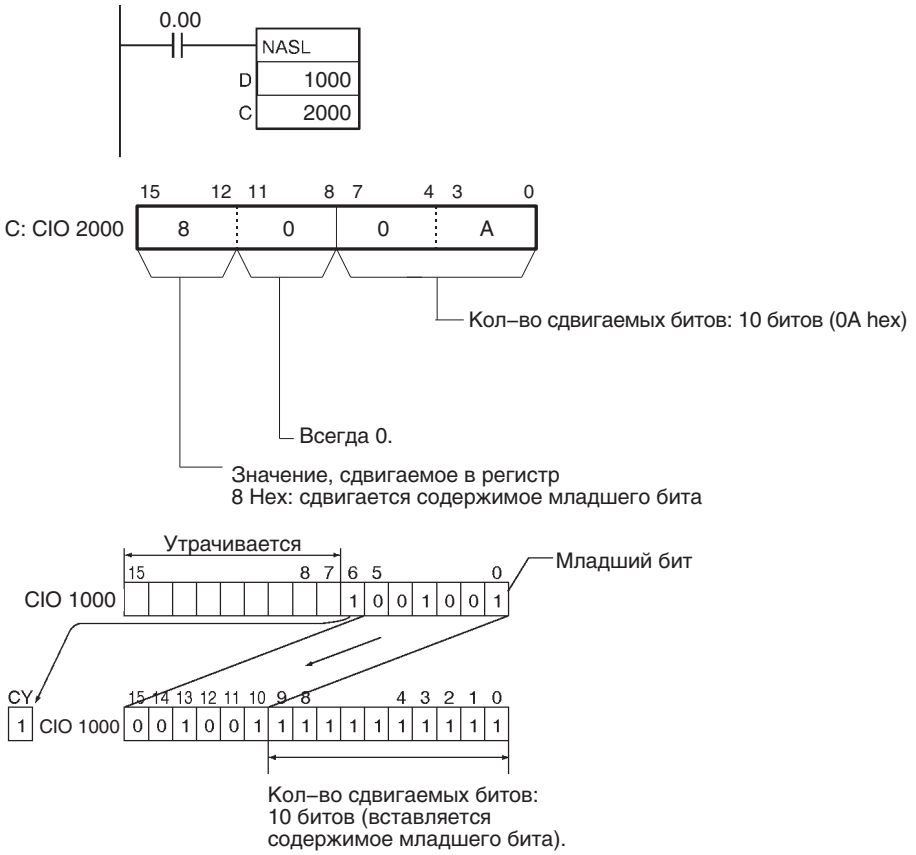
Если содержимое управляющего слова C выходит за допустимый диапазон, будет сгенерирована ошибка и установится флаг ошибки.

Если в результате операции сдвига слово D окажется равным 0000 hex, будет установлен флаг равенства.

Если в результате операции сдвига старший бит слова D окажется равным «1», будет установлен флаг отрицательного значения.

Примеры

Если бит CIO 0.00 включен, содержимое слова CIO 1000 сдвигается на 10 битов влево (от младшего бита к старшему биту). Количество сдвигаемых битов указывается в битах 0...7 управляющего слова CIO 2000. В «освободившиеся» после сдвига биты копируется состояние бита 0 слова CIO 1000, а флаг переноса (CY) принимает состояние последнего (крайнего правого) бита, сдвинутого за пределы слова. Остальные данные утрачиваются.

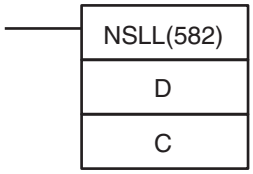


3-8-22 СДВИГ ДВОЙНОГО СЛОВА НА N БИТОВ ВЛЕВО: NSLL(582)

Назначение

Сдвигает содержимое указанного двойного слова (32 бита) влево на указанное количество битов.

Символ РКС



D: Сдвигаемое слово

C: Управляющее слово

Варианты выполнения

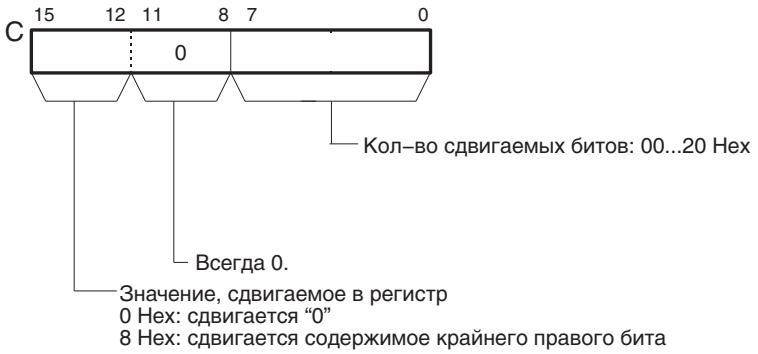
Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	NSLL(582)
	Однократное выполнение по положительному фронту	@NSLL(582)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

C: Управляющее слово



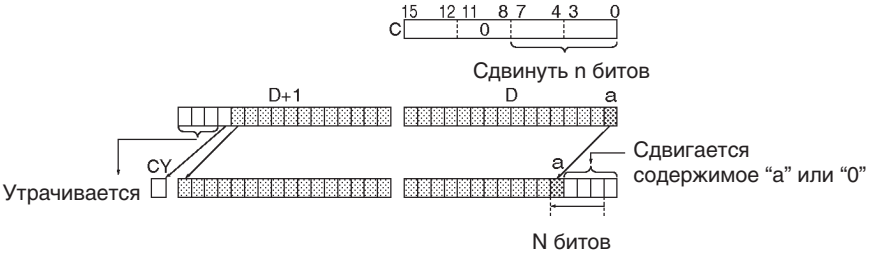
Характеристики операндов

Область	D	C
Область CIO	CIO 0...CIO 6142	CIO 0...CIO 6143
Рабочая область	W0...W510	W0...W511
Область битов хранения	H0...H510	H0...H511
Область вспомогательных битов	A448...A958	A0...A959
Область таймеров	T0000...T4094	T0000...T4095
Область счетчиков	C0000...C4094	C0000...C4095
Область DM	D0...D32766	D0...D32767
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Постоянные	---	Только указанные значения
Регистры данных	---	DR0...DR15

Область	D	C
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…-2048…+2047 ,IR15 DR0…DR15, IR0…IR15 ,IR0+(++)…,IR15+(++) ,–(– –)IR0…, –(– –)IR15	

Описание

Команда NSLL(582) сдвигает слова D и D+1 на указанное количество двоичных битов (указывается в C) влево (от младшего бита к старшему биту). В указанное количество битов сдвигаемого слова, начиная с младшего бита, записываются нули или состояние младшего бита.



Флаги

Название	Обо-значение	Описание работы
Флаг ошибки	ER	Включен, если управляющее слово C (количество сдвигаемых битов) выходит за допустимый диапазон. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если результат операции сдвига = 0. Выключен во всех остальных случаях.
Флаг переноса	CY	Включен, если во флаг переноса (CY) сдвинуто «1». Выключен во всех остальных случаях.
Флаг «Меньше нуля»	N	Включен, если в результате сдвига старший бит = «1». Выключен во всех остальных случаях.

Меры предосторожности

Содержимое всех битов, сдвигаемых за пределы указанного слова, утрачивается, за исключением самого последнего бита, который сдвигается во флаг переноса (CY).

При нулевом количестве сдвигаемых битов (указанном в C) данные сдвинуты не будут. Однако соответствующие флаги все равно будут установлены либо сброшены в соответствии со значением указанного слова.

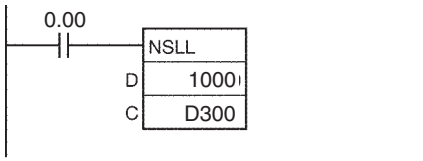
Если содержимое управляющего слова C выходит за допустимый диапазон, будет сгенерирована ошибка и установится флаг ошибки.

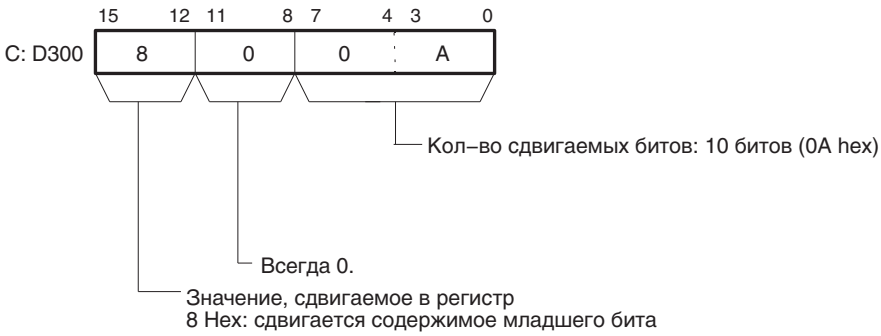
Если в результате операции сдвига слово D окажется равным «0000», будет установлен флаг равенства.

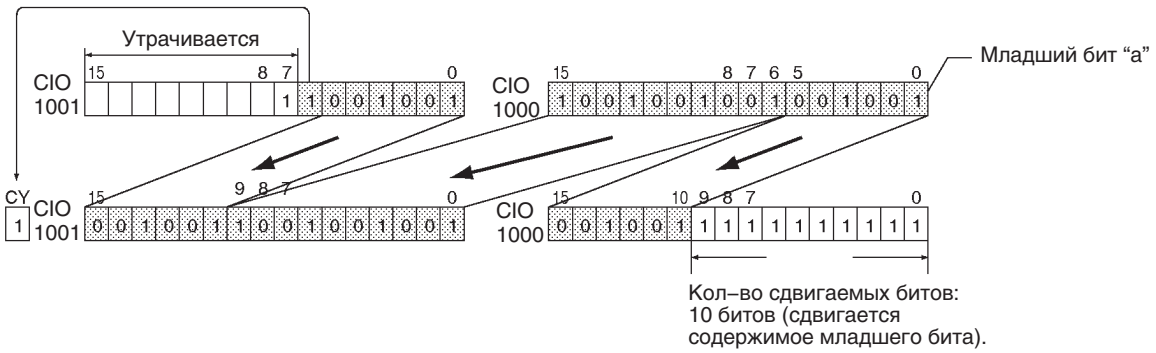
Если в результате операции сдвига старший бит двойного слова D, D+1 окажется равным «1», будет установлен флаг отрицательного значения.

Примеры

Если бит CIO 0.00 включен, слова CIO 1000 и CIO 1001 сдвигаются влево (от младшего бита к старшему) на 10 битов. Количество сдвигаемых битов указывается в битах 0...7 управляющего слова D300. В «освободившиеся» после сдвига биты копируется состояние бита 0 слова CIO 1000, а флаг переноса (CY) принимает состояние последнего (крайнего правого) бита, сдвинутого за пределы слова. Остальные данные утрачиваются.





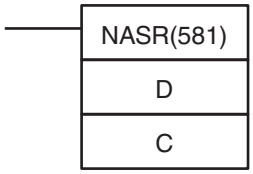


3-8-23 СДВИГ НА N БИТОВ ВПРАВО: NASR(581)

Назначение

Сдвигает содержимое указанного слова (16 битов) вправо на указанное количество битов.

Символ РКС



D: Сдвигаемое слово

C: Управляющее слово

Варианты выполнения

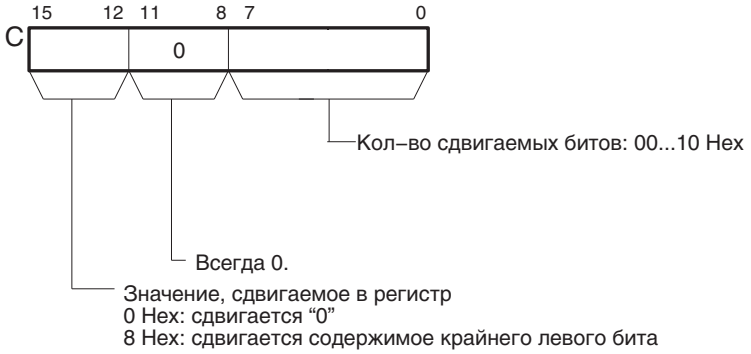
Варианты выполнения		
	Выполнение в каждом цикле при включенном условии выполнения	NASR(581)
	Однократное выполнение по положительному фронту	@NASR(581)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

C: Управляющее слово



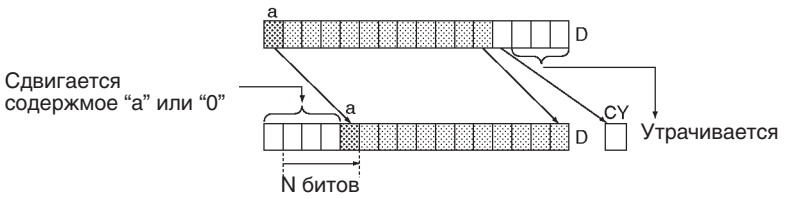
Характеристики операндов

Область	D	C
Область CIO	CIO 0...CIO 6143	
Рабочая область	W0...W511	
Область битов хранения	H0...H511	
Область вспомогательных битов	A448...A959	A0...A447 A448...A959
Область таймеров	T0000...T4095	
Область счетчиков	C0000...C4095	
Область DM	D0...D32767	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Постоянные	---	Только указанные значения

Область	D	C
Регистры данных	DR0...DR15	
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…-2048…+2047 ,IR15 DR0…DR15, IR0…IR15 ,IR0+(++)…,IR15+(++) ,–(– –)IR0…, –(– –)IR15	

Описание

Команда NASR(581) сдвигает слово D на указанное количество двоичных битов (указывается в C) вправо (от старшего бита к младшему биту). В указанное количество битов сдвигаемого слова, начиная со старшего бита, записываются нули или состояние старшего бита.



Флаги

Название	Обо-значение	Описание работы
Флаг ошибки	ER	Включен, если управляющее слово C (количество сдвигаемых битов) выходит за допустимый диапазон. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если результат операции сдвига = 0. Выключен во всех остальных случаях.
Флаг переноса	CY	Включен, если во флаг переноса (CY) сдвинуто «1». Выключен во всех остальных случаях.
Флаг «Меньше нуля»	N	Включен, если в результате сдвига старший бит = «1». Выключен во всех остальных случаях.

Меры предосторожности

Содержимое всех битов, сдвигаемых за пределы указанного слова, утрачивается, за исключением самого последнего бита, который сдвигается во флаг переноса (CY).

При нулевом количестве сдвигаемых битов (указанном в C) данные сдвинуты не будут. Однако соответствующие флаги все равно будут установлены либо сброшены в соответствии со значением указанного слова.

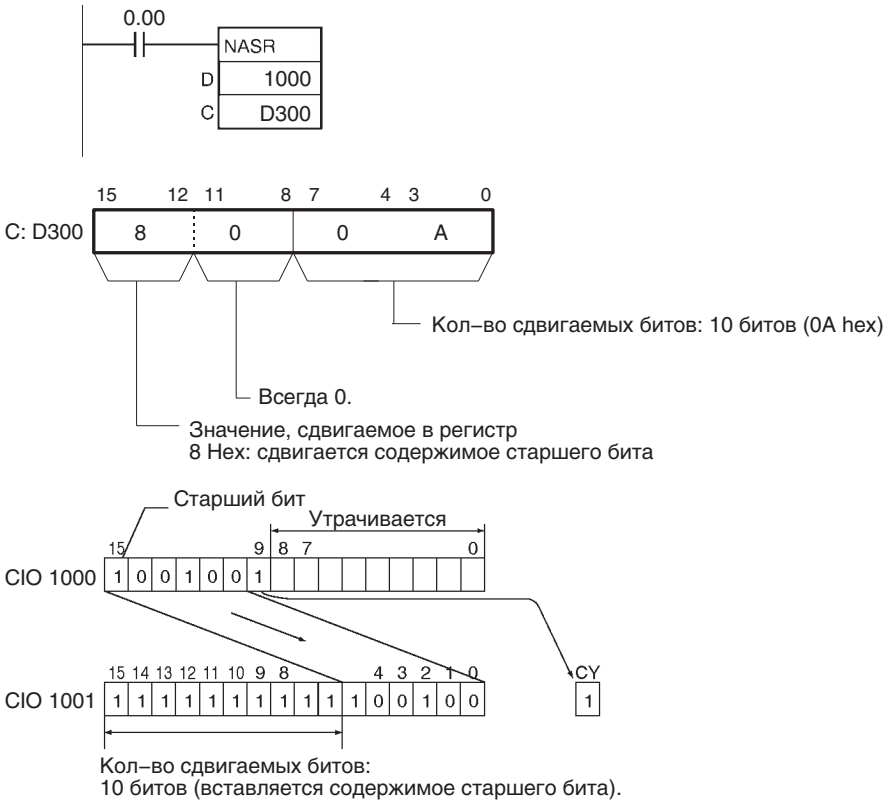
Если содержимое управляющего слова C выходит за допустимый диапазон, будет сгенерирована ошибка и установится флаг ошибки.

Если в результате операции сдвига слово D окажется равным 0000 hex, будет установлен флаг равенства.

Если в результате операции сдвига старший бит слова D окажется равным «1», будет установлен флаг отрицательного значения.

Примеры

Если бит CIO 0.00 включен, содержимое слова CIO 1000 сдвигается на 10 битов вправо (от старшего бита к младшему биту). Количество сдвигаемых битов указывается в битах 0...7 управляющего слова D300. В «освободившиеся» после сдвига биты копируется состояние бита 15 слова CIO 1000, а флаг переноса (CY) принимает состояние крайнего левого бита, сдвинутого за пределы слова. Остальные данные утрачиваются.

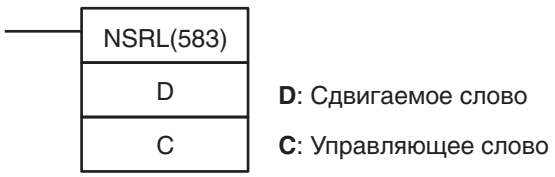


3-8-24 СДВИГ ДВОЙНОГО СЛОВА НА N БИТОВ ВПРАВО: NSRL(583)

Назначение

Сдвигает содержимое указанного двойного слова (32 бита) вправо на указанное количество битов.

Символ РКС



D: Сдвигаемое слово

C: Управляющее слово

Варианты выполнения

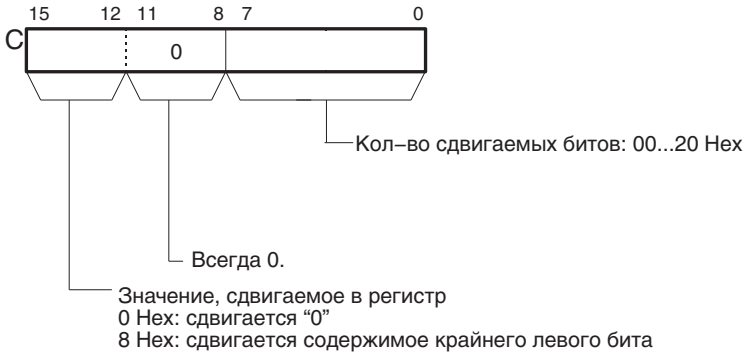
Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	NSRL(583)
	Однократное выполнение по положительному фронту	@NSRL(583)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограм-мы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

C: Управляющее слово

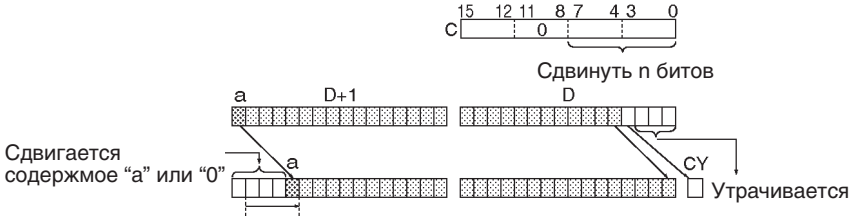


Характеристики операндов

Область	D	C
Область CIO	CIO 0...CIO 6142	CIO 0...CIO 6143
Рабочая область	W0...W510	W0...W511
Область битов хранения	H0...H510	H0...H511
Область вспомогательных битов	A448...A958	A0...A959
Область таймеров	T0000...T4094	T0000...T4095
Область счетчиков	C0000...C4094	C0000...C4095
Область DM	D0...D32766	D0...D32767
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Постоянные	---	Только указанные значения
Регистры данных	---	DR0...DR15
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…-2048…+2047 ,IR15 DR0…DR15, IR0…IR15 ,IR0+(++)…,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

Описание

Команда NSRL(583) сдвигает слова D и D+1 на указанное количество двоичных битов (указывается в C) вправо (от старшего бита к младшему биту). В указанное количество битов сдвигаемого слова, начиная со старшего бита, записываются нули или состояние старшего бита.



Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если управляющее слово C (количество сдвигаемых битов) выходит за допустимый диапазон. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если результат операции сдвига = 0. Выключен во всех остальных случаях.
Флаг переноса	CY	Включен, если во флаг переноса (CY) сдвинуто «1». Выключен во всех остальных случаях.
Флаг «Меньше нуля»	N	Включен, если в результате сдвига старший бит = «1». Выключен во всех остальных случаях.

Меры предосторожности

Содержимое всех битов, сдвигаемых за пределы указанного слова, утрачивается, за исключением самого последнего бита, который сдвигается во флаг переноса (CY).

При нулевом количестве сдвигаемых битов (указанном в C) данные сдвинуты не будут. Однако соответствующие флаги все равно будут установлены или сброшены в соответствии со значением указанного слова.

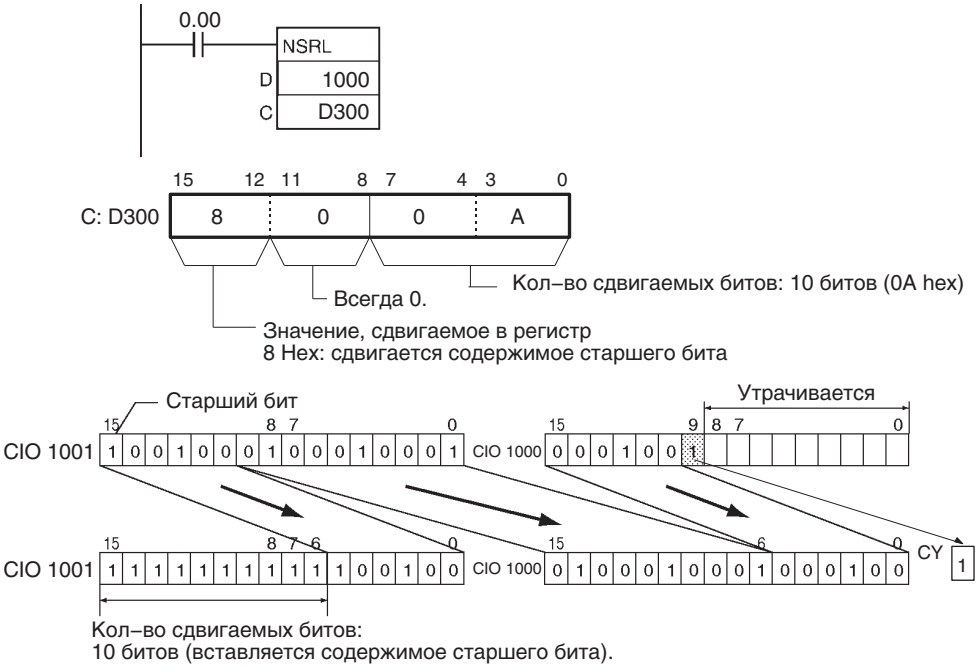
Если содержимое управляющего слова C выходит за допустимый диапазон, будет сгенерирована ошибка и установится флаг ошибки.

Если в результате операции сдвига двойное слово D, D + 1 окажется равным 0000 0000 hex, будет установлен флаг равенства.

Если в результате операции сдвига старший бит слова D +1 окажется равным «1», будет установлен флаг отрицательного значения.

Примеры

Если бит CIO 0.00 включен, содержимое слов CIO 1000 и CIO 1001 сдвигается на 10 битов вправо (от старшего бита к младшему биту). Количество сдвигаемых битов указывается в битах 0...7 управляющего слова D300. В «освободившиеся» после сдвига биты копируется состояние бита 15 слова CIO 1001, а флаг переноса (CY) принимает состояние крайнего левого бита, сдвинутого за пределы слова. Остальные данные утрачиваются.



3-9 Команды увеличения/уменьшения

В данном разделе описаны команды, используемые для увеличения/уменьшения значений.

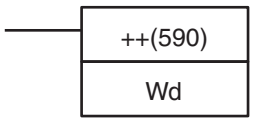
Команда	Мнемоническое обозначение	Код функции	Стр.
УВЕЛИЧИТЬ ДВОИЧНОЕ СЛОВО	++	590	349
УВЕЛИЧИТЬ ДВОЙНОЕ ДВОИЧНОЕ СЛОВО	++L	591	351
УМЕНЬШИТЬ ДВОИЧНОЕ СЛОВО	– –	592	353
УМЕНЬШИТЬ ДВОЙНОЕ ДВОИЧНОЕ СЛОВО	– –L	593	355
УВЕЛИЧИТЬ ДВОИЧНО-ДЕСЯТИЧНОЕ СЛОВО	++B	594	358
УВЕЛИЧИТЬ ДВОЙНОЕ ДВОИЧНО-ДЕСЯТИЧНОЕ СЛОВО	++BL	595	360
УМЕНЬШИТЬ ДВОИЧНО-ДЕСЯТИЧНОЕ СЛОВО	– –B	596	362
УМЕНЬШИТЬ ДВОЙНОЕ ДВОИЧНО-ДЕСЯТИЧНОЕ СЛОВО	– –BL	597	364

3-9-1 УВЕЛИЧИТЬ ДВОИЧНОЕ СЛОВО: ++(590)

Назначение

Увеличение 4-разрядного шестнадцатеричного значения указанного слова на 1.

Символ РКС



Wd: Слово

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	++(590)
	Однократное выполнение по положительному фронту	@++(590)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

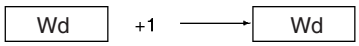
Характеристики операндов

Область	Wd
Область CIO	CIO 0...CIO 6143
Рабочая область	W0...W511
Область битов хранения	H0...H511

Область	Wd
Область вспомогательных битов	A448...A959
Область таймеров	T0000...T4095
Область счетчиков	C0000...C4095
Область DM	D0...D32767
Косвенные адреса DM в двоичном формате	@ D0...@ D32767
Косвенные адреса DM в BCD-формате	*D0...*D32767
Постоянные	---
Регистры данных	DR0...DR15
Регистры указателей	---
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…-2048…+2047 ,IR15 DR0…DR15, IR0…IR15 ,IR0+(++)…,IR15+(++) ,–(– –)IR0…, –(– –)IR15

Описание

Команда ++(590) добавляет 1 к двоичному значению слова Wd. Указанное слово увеличивается на 1 в каждом цикле, пока условие выполнения ++(590) находится во включенном состоянии. Если же выбрано выполнение по положительному фронту (@++(590)), указанное слово увеличивается только при переключении условия выполнения из «0» в «1».



Если результат равен «0000», включается флаг равенства. Если значение разряда (тетрады) меняется с «F» на «0», включается флаг переноса. Если в результате выполнения устанавливается бит 15 слова Wd, включается флаг отрицательного значения.

Если значение слова Wd изменяется с «FFFF» на «0000», одновременно включаются флаг равенства и флаг переноса.

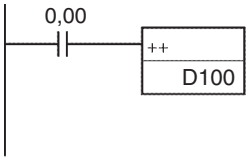
Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	ВЫКЛ
Флаг равенства	=	Включен, если после выполнения Wd содержит 0000. Выключен во всех остальных случаях.
Флаг переноса	CY	Включен, если во время выполнения значение разряда слова Wd поменялось с «F» на «0». Выключен во всех остальных случаях.
Флаг «Меньше нуля»	N	Включен, если после выполнения бит 15 слова Wd = «1». Выключен во всех остальных случаях.

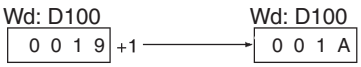
Примеры

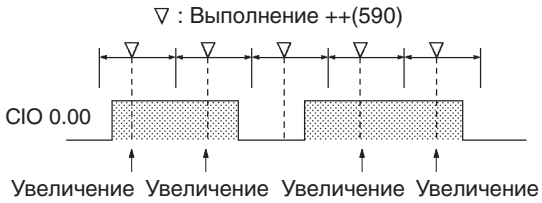
Выполнение команды ++(590)

В следующем примере значение слова D100 увеличивается на 1 в каждом цикле, пока включен бит CIO 0.00.



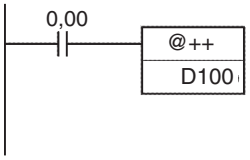
Увеличивается на 1 в каждом цикле, пока CIO 0.00 = «1».





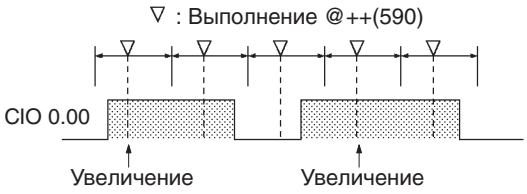
Выполнение команды @++(590)

В следующем примере используется модификатор положительного фронта, поэтому содержимое D100 будет возрастать на 1 только при переключении CIO 0.00 из «0» в «1».



Увеличивается на 1 только по положительному фронту



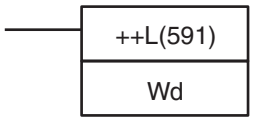


3-9-2 УВЕЛИЧИТЬ ДВОЙНОЕ ДВОИЧНОЕ СЛОВО: ++L(591)

Назначение

Увеличение 8-разрядного шестнадцатеричного значения указанных слов на 1.

Символ РКС



Wd: Первое слово

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	++L(591)
	Однократное выполнение по положительному фронту	@++L(591)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

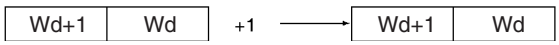
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	Wd
Область CIO	CIO 0...CIO 6142
Рабочая область	W0...W510
Область битов хранения	H0...H510
Область вспомогательных битов	A448...A958
Область таймеров	T0000...T4094
Область счетчиков	C0000...C4094
Область DM	D0...D32766
Косвенные адреса DM в двоичном формате	@ D0...@ D32767
Косвенные адреса DM в BCD-формате	*D0...*D32767
Постоянные	---
Регистры данных	---
Регистры указателей	IR0...IR15
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…-2048…+2047 ,IR15 DR0…DR15, IR0…IR15 ,IR0+(++)…,IR15+(++) ,–(– –)IR0…, –(– –)IR15

Описание

Команда ++L(591) добавляет 1 к 8-разрядному шестнадцатеричному значению слов Wd+1 и Wd. Значение указанного двойного слова увеличивается на 1 в каждом цикле, пока условие выполнения ++L(591) находится во включенном состоянии. Если же выбрано выполнение по положительному фронту (@++L(591)), указанное двойное слово увеличивается только при переключении условия выполнения из «0» в «1».



Если результат равен «0000 0000», включается флаг равенства. Если значение разряда (тетрады) меняется с «F» на «0», включается флаг переноса. Если в результате выполнения устанавливается бит 15 слова Wd + 1, включается флаг отрицательного значения.

Если значение двойного слова изменяется с «FFFF FFFF» на «0000 0000», одновременно включаются флаг равенства и флаг переноса.

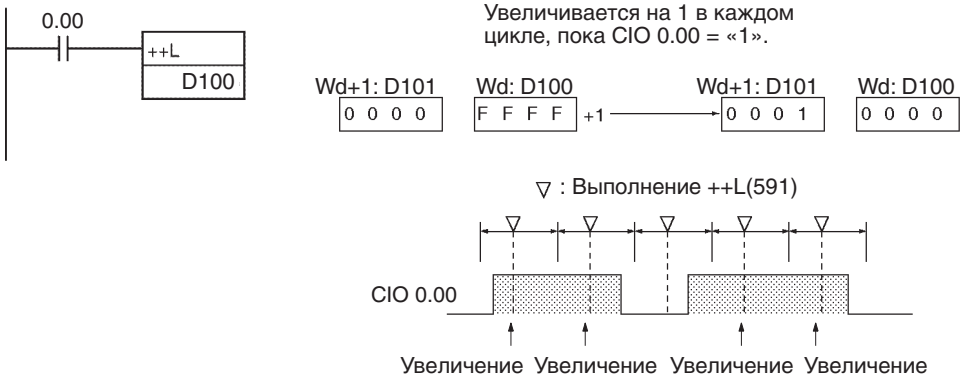
Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	ВЫКЛ
Флаг равенства	=	Включен, если после выполнения результат = 0000 0000. Выключен во всех остальных случаях.
Флаг переноса	CY	Включен, если во время выполнения значение разряда слова Wd+1 или Wd поменялось с «F» на «0». Выключен во всех остальных случаях.
Флаг «Меньше нуля»	N	Включен, если после выполнения бит 15 слова Wd+1 = «1». Выключен во всех остальных случаях.

Примеры

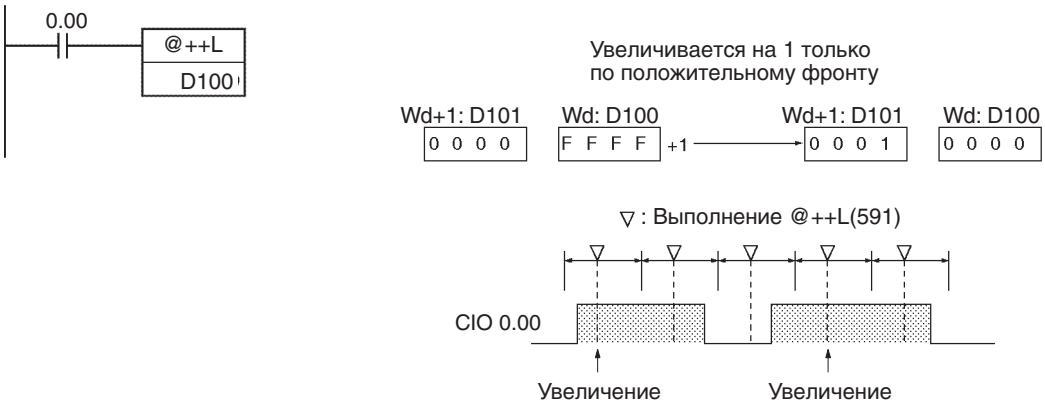
Выполнение команды ++L(591)

В следующем примере 8-разрядное шестнадцатеричное значение слов D101 и D100 увеличивается на 1 в каждом цикле, пока включен бит CIO 0.00.



Выполнение команды @++L(591)

В следующем примере используется модификатор положительного фронта, поэтому содержимое слов D101 и D100 будет возрастать на 1 только при переключении CIO 0.00 из «0» в «1».

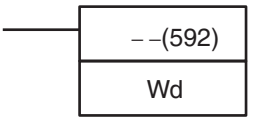


3-9-3 УМЕНЬШИТЬ ДВОИЧНОЕ СЛОВО: – –(592)

Назначение

Уменьшение 4-разрядного шестнадцатеричного значения указанного слова на 1.

Символ РКС



Wd: Слово

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	– – (592)
	Однократное выполнение по положительному фронту	@– – (592)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

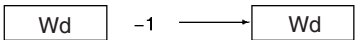
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	Wd
Область CIO	CIO 0...CIO 6143
Рабочая область	W0...W511
Область битов хранения	H0...H511
Область вспомогательных битов	A448...A959
Область таймеров	T0000...T4095
Область счетчиков	C0000...C4095
Область DM	D0...D32767
Косвенные адреса DM в двоичном формате	@ D0...@ D32767
Косвенные адреса DM в BCD-формате	*D0...*D32767
Постоянные	---
Регистры данных	DR0...DR15
Регистры указателей	---
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…-2048…+2047 ,IR15 DR0…DR15, IR0…IR15 ,IR0+(++)…,IR15+(++) ,–(– –)IR0…, –(– –)IR15

Описание

Команда – –(592) вычитает 1 из двоичного значения слова Wd. Указанное слово уменьшается на 1 в каждом цикле, пока условие выполнения – –(592) находится во включенном состоянии. Если же выбрано выполнение по положительному фронту (@– –(592)), указанное слово уменьшается только при переключении условия выполнения из «0» в «1».



Если результат равен «0000», включается флаг равенства. Если значение разряда (тетрады) меняется с «0» на «F», включается флаг переноса. Если в результате выполнения устанавливается бит 15 слова Wd, включается флаг отрицательного значения.

Если значение слова Wd изменяется с «0000» на «FFFF», одновременно включаются флаг переноса и флаг отрицательного значения.

Флаги

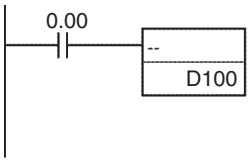
Название	Обозначение	Описание работы
Флаг ошибки	ER	ВЫКЛ
Флаг равенства	=	Включен, если после выполнения Wd содержит 0000. Выключен во всех остальных случаях.

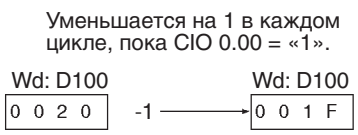
Название	Обозначение	Описание работы
Флаг переноса	CY	Включен, если во время выполнения значение разряда слова Wd поменялось с «0» на «F». Выключен во всех остальных случаях.
Флаг «Меньше нуля»	N	Включен, если после выполнения бит 15 слова Wd = «1». Выключен во всех остальных случаях.

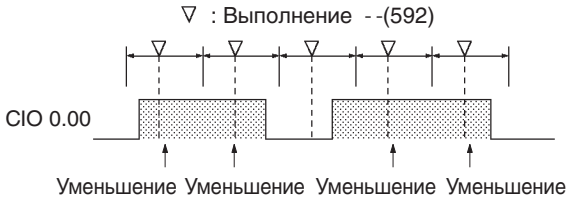
Примеры

Выполнение команды – –(592)

В следующем примере значение слова D100 уменьшается на 1 в каждом цикле, пока включен бит CIO 0.00.

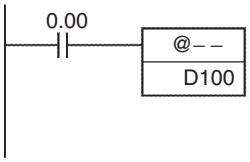


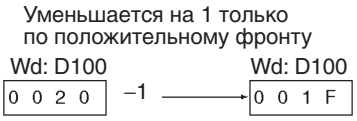


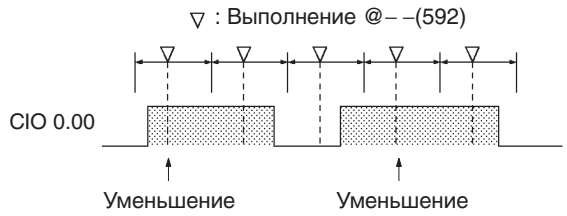


Выполнение команды @– –(592)

В следующем примере используется модификатор положительного фронта, поэтому содержимое D100 будет уменьшаться на 1 только при переключении CIO 0.00 из «0» в «1».





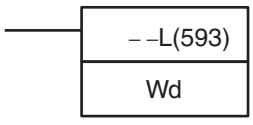


3-9-4 УМЕНЬШИТЬ ДВОЙНОЕ ДВОИЧНОЕ СЛОВО: – –L(593)

Назначение

Уменьшение 8-разрядного шестнадцатеричного значения указанных слов на 1.

Символ РКС



Wd: Первое слово

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	– –L(593)
	Однократное выполнение по положительному фронту	@– –L(593)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

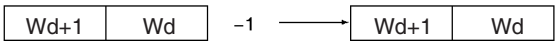
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	Wd
Область CIO	CIO 0...CIO 6142
Рабочая область	W0...W510
Область битов хранения	H0...H510
Область вспомогательных битов	A448...A958
Область таймеров	T0000...T4094
Область счетчиков	C0000...C4094
Область DM	D0...D32766
Косвенные адреса DM в двоичном формате	@ D0...@ D32767
Косвенные адреса DM в BCD-формате	*D0...*D32767
Постоянные	---
Регистры данных	---
Регистры указателей	---
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…-2048…+2047 ,IR15 DR0…DR15, IR0…IR15 ,IR0+(++)…,IR15+(++) ,–(– –)IR0…, –(– –)IR15

Описание

Команда – –L(593) вычитает 1 из 8-разрядного шестнадцатеричного значения слов Wd+1 и Wd. Значение указанного двойного слова уменьшается на 1 в каждом цикле, пока условие выполнения – –L(593) находится во включенном состоянии. Если же выбрано выполнение по положительному фронту (@– –L(593)), значение указанного двойного слова уменьшается только при переключении условия выполнения из «0» в «1».



Если результат равен «0000 0000», включается флаг равенства. Если значение разряда (тетрады) меняется с «0» на «F», включается флаг переноса. Если в результате выполнения устанавливается бит 15 слова Wd + 1, включается флаг отрицательного значения.

Если значение слов изменяется с «0000 0000» на «FFFF FFFF», одновременно включаются флаг переноса и флаг отрицательного значения.

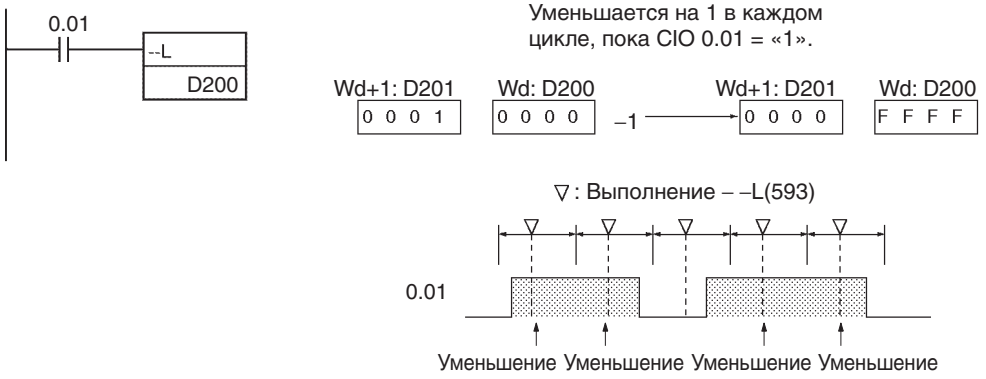
Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	ВЫКЛ
Флаг равенства	=	Включен, если после выполнения результат = 0000 0000. Выключен во всех остальных случаях.
Флаг переноса	CY	Включен, если во время выполнения значение разряда слова Wd+1 или Wd поменялось с «0» на «F». Выключен во всех остальных случаях.
Флаг «Меньше нуля»	N	Включен, если после выполнения бит 15 слова Wd+1 = «1». Выключен во всех остальных случаях.

Примеры

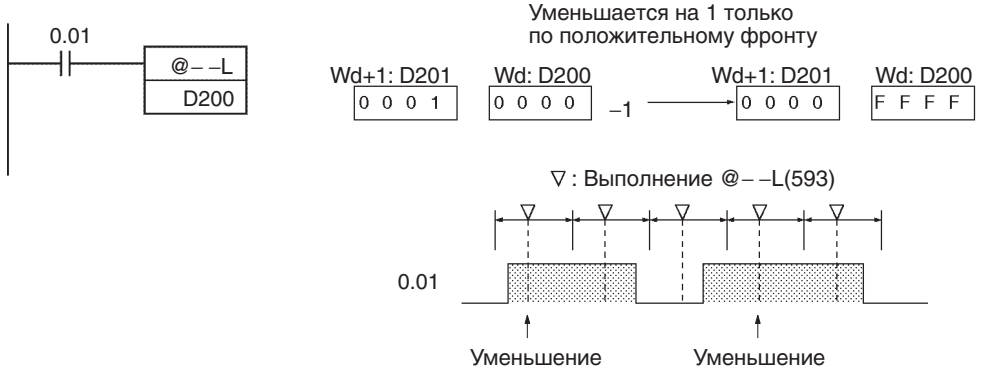
Выполнение команды – –L(593)

В следующем примере 8-разрядное шестнадцатеричное значение слов D201 и D200 уменьшается на 1 в каждом цикле, пока включен бит CIO 0.01.



Выполнение команды @– –L(593)

В следующем примере используется модификатор положительного фронта, поэтому содержимое слов D201 и D200 будет уменьшаться на 1 только при переключении CIO 0.01 из «0» в «1».



3-9-5 УВЕЛИЧИТЬ ДВОИЧНО-ДЕСЯТИЧНОЕ СЛОВО: ++B(594)

Назначение

Увеличение 4-разрядного двоично-десятичного значения указанного слова на 1.

Символ РКС



Wd: Слово

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	++B(594)
	Однократное выполнение по положительному фронту	@++B(594)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

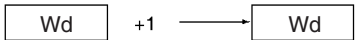
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	Wd
Область CIO	CIO 0...CIO 6143
Рабочая область	W0...W511
Область битов хранения	H0...H511
Область вспомогательных битов	A448...A959
Область таймеров	T0000...T4095
Область счетчиков	C0000...C4095
Область DM	D0...D32767
Косвенные адреса DM в BCD-формате	@ D0...@ D32767
Косвенные адреса DM в BCD-формате	*D0...*D32767
Постоянные	---
Регистры данных	DR0...DR15
Регистры указателей	---
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…-2048…+2047 ,IR15 DR0…DR15, IR0…IR15 ,IR0+(++)…,IR15+(++) ,–(– –)IR0…, –(– –)IR15

Описание

Команда ++B(594) добавляет 1 к двоично-десятичному значению слова Wd. Указанное слово увеличивается на 1 в каждом цикле, пока условие выполнения ++B(594) находится во включенном состоянии. Если же выбрано выполнение по положительному фронту (@++B(594)), указанное слово увеличивается только при переключении условия выполнения из «0» в «1».



Если результат = 0000, будет установлен флаг равенства. Если значение разряда поменяется с «9» на «0», будет установлен флаг переноса.

Если значение слова Wd изменяется с «9999» на «0000», одновременно включаются флаг равенства и флаг переноса.

Флаги

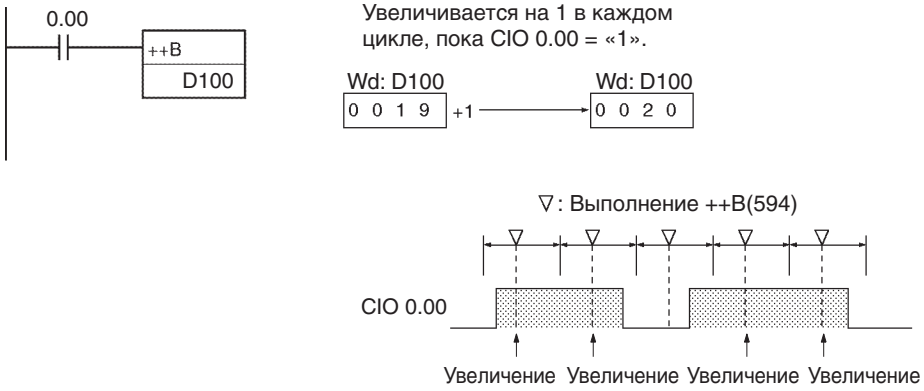
Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если содержимое Wd не в формате BCD. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если после выполнения Wd содержит 0000. Выключен во всех остальных случаях.
Флаг переноса	CY	Включен, если во время выполнения значение разряда Wd поменялось с «9» на «0». Выключен во всех остальных случаях.

Меры предосторожности Wd должно содержать двоично-десятичное значение (BCD). Иначе возникнет ошибка и будет установлен флаг ошибки

Примеры

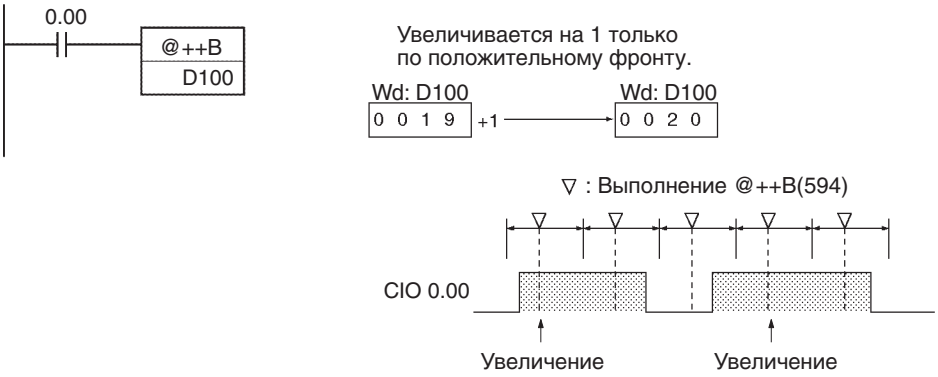
Выполнение команды ++B(594)

В следующем примере двоично-десятичное значение слова D100 увеличивается на 1 в каждом цикле, пока включен бит CIO 0.00.



Выполнение команды @++B(594)

В следующем примере используется модификатор положительного фронта, поэтому содержимое D100 будет возрастать на 1 только при переключении CIO 0.00 из «0» в «1».



3-9-6 УВЕЛИЧИТЬ ДВОЙНОЕ ДВОИЧНО-ДЕСЯТИЧНОЕ СЛОВО: ++BL(595)

Назначение

Увеличение 8-разрядного двоично-десятичного значения указанных слов на 1.

Символ РКС



Wd: Первое слово

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	++BL(595)
	Однократное выполнение по положительному фронту	@++BL(595)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

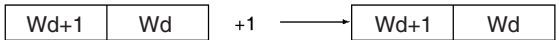
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	Wd
Область CIO	CIO 0...CIO 6142
Рабочая область	W0...W510
Область битов хранения	H0...H510
Область вспомогательных битов	A448...A958
Область таймеров	T0000...T4094
Область счетчиков	C0000...C4094
Область DM	D0...D32766
Косвенные адреса DM в BCD-формате	@ D0...@ D32767
Косвенные адреса DM в BCD-формате	*D0...*D32767
Постоянные	---
Регистры данных	---
Регистры указателей	---
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…-2048…+2047 ,IR15 DR0…DR15, IR0…IR15 ,IR0+(++)…,IR15+(++) ,–(– –)IR0…, –(– –)IR15

Описание

Команда ++BL(595) добавляет 1 к 8-разрядному двоично-десятичному значению слов Wd+1 и Wd. Значение указанного двойного слова увеличивается на 1 в каждом цикле, пока условие выполнения ++BL(595) находится во включенном состоянии. Если же выбрано выполнение по положительному фронту (@++BL(595)), значение указанного двойного слова увеличивается только при переключении условия выполнения из «0» в «1».



Если результат = 0000 0000, будет установлен флаг равенства. Если значение разряда поменяется с «9» на «0», будет установлен флаг переноса.

Если значение слов изменяется с «9999 9999 » на «0000 0000», одновременно включаются флаг равенства и флаг переноса.

Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если содержимое Wd+1 и Wd не в формате BCD. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если после выполнения результат = 0000 0000. Выключен во всех остальных случаях.
Флаг переноса	CY	Включен, если во время выполнения значение разряда слова Wd+1 или Wd поменялось с «9» на «0». Выключен во всех остальных случаях.

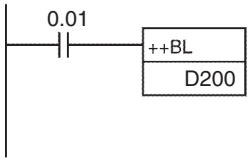
Меры предосторожности

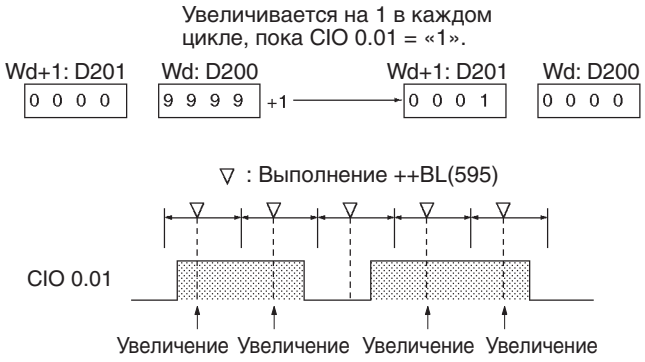
Wd+1 и Wd должны содержать двоично-десятичное значение (BCD). Иначе возникнет ошибка и будет установлен флаг ошибки

Примеры

Выполнение команды ++BL(595)

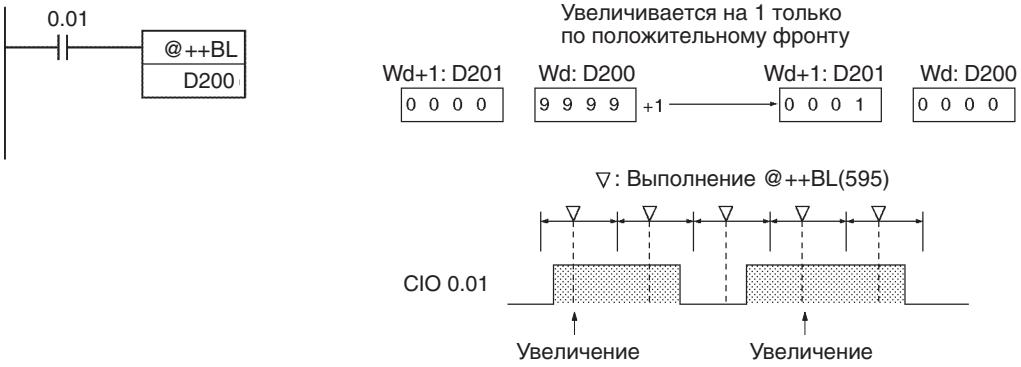
В следующем примере 8-разрядное двоично-десятичного значение слов D201 и D200 увеличивается на 1 в каждом цикле, пока включен бит CIO 0.01.





Выполнение команды @++BL(595)

В следующем примере используется модификатор положительного фронта, поэтому двоично-десятичное содержимое слов D201 и D200 будет увеличиваться на 1 только при переключении CIO 0.01 из «0» в «1».

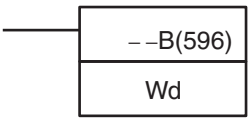


3-9-7 УМЕНЬШИТЬ ДВОИЧНО-ДЕСЯТИЧНОЕ СЛОВО: – –B(596)

Назначение

Уменьшение 4-разрядного двоично-десятичного значения указанного слова на 1.

Символ РКС



Wd: Слово

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	– –B(596)
	Однократное выполнение по положительному фронту	@– –B(596)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

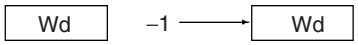
Характеристики операндов

Область	Wd
Область CIO	CIO 0...CIO 6143
Рабочая область	W0...W511
Область битов хранения	H0...H511
Область вспомогательных битов	A448...A959
Область таймеров	T0000...T4095
Область счетчиков	C0000...C4095
Область DM	D0...D32767

Область	Wd
Косвенные адреса DM в BCD-формате	@ D0...@ D32767
Косвенные адреса DM в BCD-формате	*D0...*D32767
Постоянные	---
Регистры данных	DR0...DR15
Регистры указателей	---
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…-2048…+2047 ,IR15 DR0…DR15, IR0…IR15 ,IR0+(++)…,IR15+(++) ,–(– –)IR0…, –(– –)IR15

Описание

Команда – –B(596) вычитает 1 из двоично-десятичного значения слова Wd. Указанное слово уменьшается на 1 в каждом цикле, пока условие выполнения – –B(596) находится во включенном состоянии. Если же выбрано выполнение по положительному фронту (@– –B(596)), указанное слово уменьшается только при переключении условия выполнения из «0» в «1».



Если результат = 0000, будет установлен флаг равенства. Если значение разряда поменяется с «0» на «9», будет установлен флаг переноса.

Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если содержимое Wd не в формате BCD. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если после выполнения Wd содержит 0000. Выключен во всех остальных случаях.
Флаг переноса	CY	Включен, если во время выполнения значение разряда слова Wd поменялось с «0» на «9». Выключен во всех остальных случаях.

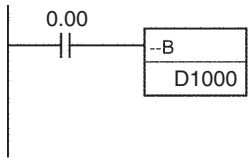
Меры предосторожности

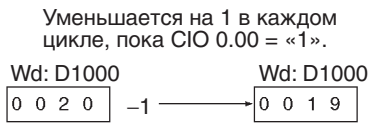
Wd должно содержать двоично-десятичное значение (BCD). Иначе возникнет ошибка и будет установлен флаг ошибки

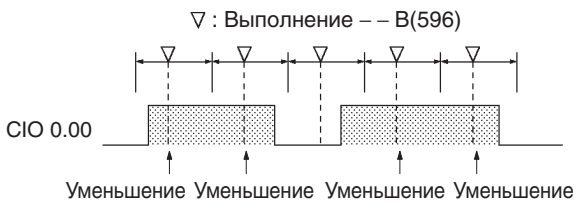
Примеры

Выполнение команды – –B(596)

В следующем примере двоично-десятичное значение слова D1000 уменьшается на 1 в каждом цикле, пока включен бит CIO 0.00.

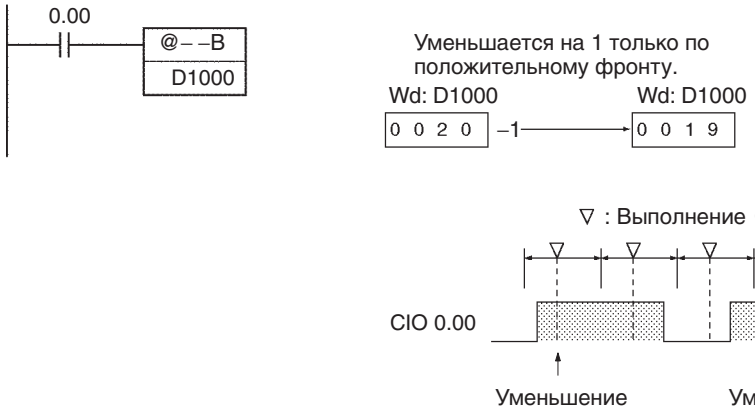






Выполнение команды @– –B(596)

В следующем примере используется модификатор положительного фронта, поэтому двоично-десятичное содержимое слова D1000 будет уменьшаться на 1 только при переключении CIO 0.00 из «0» в «1».

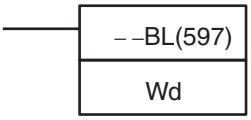


3-9-8 УМЕНЬШИТЬ ДВОЙНОЕ ДВОИЧНО-ДЕСЯТИЧНОЕ СЛОВО: – –BL(597)

Назначение

Уменьшение 8-разрядного двоично-десятичного значения указанных слов на 1.

Символ РКС



Wd: Первое слово

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	– –BL(597)
	Однократное выполнение по положительному фронту	@– –BL(597)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

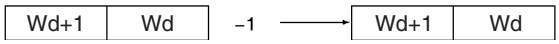
Характеристики операндов

Область	Wd
Область CIO	CIO 0...CIO 6142
Рабочая область	W0...W510
Область битов хранения	H0...H510
Область вспомогательных битов	A448...A958
Область таймеров	T0000...T4094
Область счетчиков	C0000...C4094

Область	Wd
Область DM	D0...D32766
Косвенные адреса DM в BCD-формате	@ D0...@ D32767
Косвенные адреса DM в BCD-формате	*D0...*D32767
Постоянные	---
Регистры данных	---
Регистры указателей	---
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…-2048…+2047 ,IR15 DR0…DR15, IR0…IR15 ,IR0+(++)…,IR15+(++) ,–(– –)IR0…, –(– –)IR15

Описание

Команда – –BL(597) вычитает 1 из 8-разрядного двоично-десятичного значения слов Wd+1 и Wd. Значение указанного двойного слова уменьшается на 1 в каждом цикле, пока условие выполнения – –BL(597) находится во включенном состоянии. Если же выбрано выполнение по положительному фронту (@– –BL(597)), значение указанного двойного слова уменьшается только при переключении условия выполнения из «0» в «1».



Если результат = 0000 0000, будет установлен флаг равенства. Если значение разряда поменяется с «0» на «9», будет установлен флаг переноса.

Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если содержимое Wd+1 и Wd не в формате BCD. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если после выполнения результат = 0000 0000. Выключен во всех остальных случаях.
Флаг переноса	CY	Включен, если во время выполнения значение разряда слова Wd+1 или Wd поменялось с «0» на «9». Выключен во всех остальных случаях.

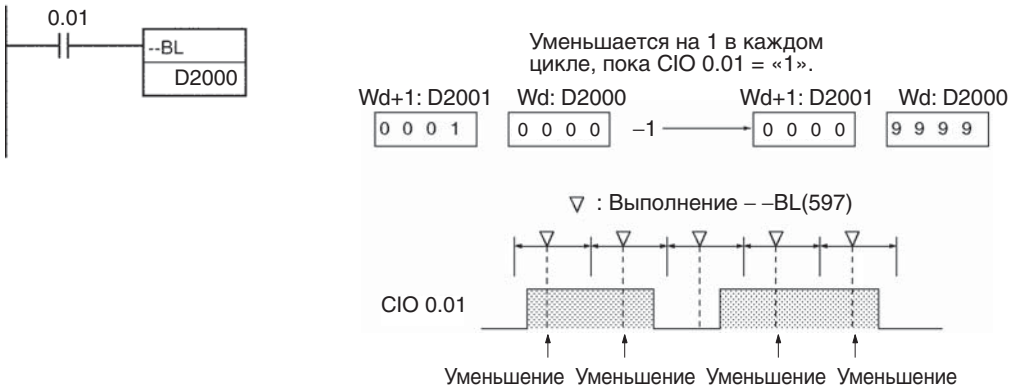
Меры предосторожности

Wd+1 и Wd должны содержать двоично-десятичное значение (BCD). Иначе возникнет ошибка и будет установлен флаг ошибки

Примеры

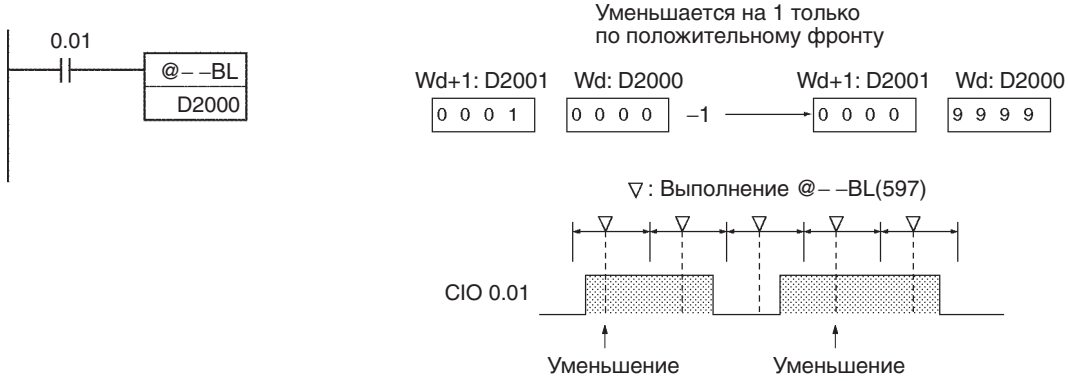
Выполнение команды – –BL(597)

В следующем примере 8-разрядное двоично-десятичное значение слов D2001 и D2000 уменьшается на 1 в каждом цикле, пока включен бит CIO 0.01.



Выполнение команды @– –BL(597)

В следующем примере используется модификатор положительного фронта, поэтому двоично-десятичное содержимое слов D2001 и D2000 будет уменьшаться на 1 только при переключении CIO 0.01 из «0» в «1».



3-10 Символьные математические команды

В настоящем разделе описаны символьные математические команды, выполняющие арифметические операции над данными в двоичном и двоично-десятичном (BCD) формате.

Команда	Мнемоническое обозначение	Код функции	Стр.
СЛОЖИТЬ БЕЗ ПЕРЕНОСА ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ	+	400	367
СЛОЖИТЬ БЕЗ ПЕРЕНОСА ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ	+L	401	370
СЛОЖИТЬ С ПЕРЕНОСОМ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ	+C	402	371
СЛОЖИТЬ С ПЕРЕНОСОМ ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ	+CL	403	374
СЛОЖИТЬ БЕЗ ПЕРЕНОСА ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА	+B	404	376
СЛОЖИТЬ БЕЗ ПЕРЕНОСА ДВОЙНЫЕ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА	+BL	405	377
СЛОЖИТЬ С ПЕРЕНОСОМ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА	+BC	406	379
СЛОЖИТЬ С ПЕРЕНОСОМ ДВОЙНЫЕ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА	+BCL	407	381
ВЫЧЕСТЬ БЕЗ ПЕРЕНОСА ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ	–	410	382

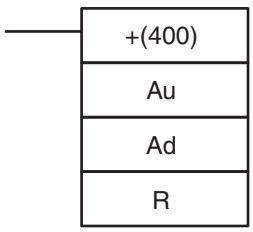
Команда	Мнемоническое обозначение	Код функции	Стр.
ВЫЧЕСТЬ БЕЗ ПЕРЕНОСА ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ	–L	411	385
ВЫЧЕСТЬ С ПЕРЕНОСОМ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ	–C	412	389
ВЫЧЕСТЬ С ПЕРЕНОСОМ ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ	–CL	413	391
ВЫЧЕСТЬ БЕЗ ПЕРЕНОСА ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА	–B	414	394
ВЫЧЕСТЬ БЕЗ ПЕРЕНОСА ДВОЙНЫЕ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА	–BL	415	396
ВЫЧЕСТЬ С ПЕРЕНОСОМ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА	–BC	416	399
ВЫЧЕСТЬ С ПЕРЕНОСОМ ДВОЙНЫЕ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА	–BCL	417	401
УМНОЖИТЬ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ	*	420	403
УМНОЖИТЬ ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ	*L	421	405
УМНОЖИТЬ ДВОИЧНЫЕ СЛОВА БЕЗ ЗНАКА	*U	422	407
УМНОЖИТЬ ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА БЕЗ ЗНАКА	*UL	423	408
УМНОЖИТЬ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА	*B	424	410
УМНОЖИТЬ ДВОЙНЫЕ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА	*BL	425	412
ДЕЛИТЬ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ	/	430	413
ДЕЛИТЬ ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ	/L	431	415
ДЕЛИТЬ ДВОИЧНЫЕ СЛОВА БЕЗ ЗНАКА	/U	432	417
ДЕЛИТЬ ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА БЕЗ ЗНАКА	/UL	433	419
ДЕЛИТЬ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА	/B	434	421
ДЕЛИТЬ ДВОЙНЫЕ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА	/BL	435	422

3-10-1 СЛОЖИТЬ БЕЗ ПЕРЕНОСА ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ: +(400)

Назначение

Операция сложения 4-разрядных (одно слово) шестнадцатеричных чисел и/или констант.

Символ РКС



Au: Слово 1–го слагаемого

Ad: Слово 2–го слагаемого

R: Слово результата

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	+(400)
	Однократное выполнение по положительному фронту	@+(400)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

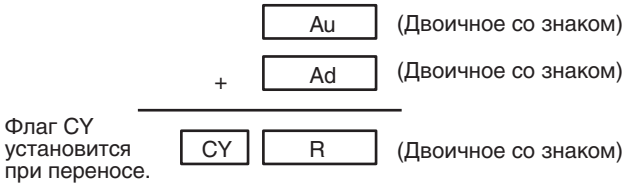
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	Au	Ad	R
Область CIO	CIO 0...CIO 6143		
Рабочая область	W0...W511		
Область битов хранения	H0...H511		
Область вспомогательных битов	A0...A959		A448...A959
Область таймеров	T0000...T4095		
Область счетчиков	C0000...C4095		
Область DM	D0...D32767		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	#0000...#FFFF (двоичное) &0...&65535 (десятичное без знака) -32768...0...32767 (десятичное со знаком)		---
Регистры данных	DR0...DR15		
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда +(400) складывает двоичные значения Au и Ad и выводит результат в R.



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Выключен
Флаг равенства	=	Включен, если результат = 0. Выключен во всех остальных случаях.
Флаг переноса	CY	Включен, если сложение приводит к переносу. Выключен во всех остальных случаях.
Флаг переполнения	OF	Включен, если результат сложения двух положительных чисел принадлежит диапазону 8000...FFFF hex. Выключен во всех остальных случаях.
Флаг потери значимости	UF	Включен, если результат сложения двух отрицательных чисел принадлежит диапазону 0000...7FFF hex. Выключен во всех остальных случаях.
Флаг отрицательного значения	N	Включен, если старший бит результата = «1». Выключен во всех остальных случаях.

Меры предосторожности

При выполнении команды +(400) флаг ошибки выключается.

Если в результате операции сложения содержимое R окажется равным 0000 hex, будет установлен флаг равенства.

Если сложение приведет к переносу, установится флаг переноса.

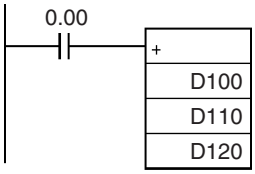
Если результат сложения двух положительных чисел является отрицательным числом (в диапазоне 8000...FFFF hex), включается флаг переполнения.

Если результат сложения двух отрицательных чисел является положительным числом (в диапазоне 0000...7FFF hex), включается флаг потери значимости.

Если в результате операции сложения старший бит R окажется равным «1», будет установлен флаг отрицательного значения.

Примеры

Если в следующем примере бит CIO 0.00 включен, выполняется сложение содержимого слов D100 и D110 как 4-разрядных двоичных чисел со знаком, результат выводится в слово D120.

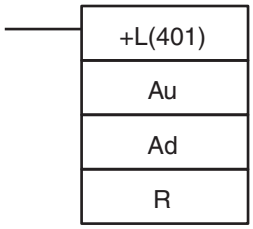


3-10-2 СЛОЖИТЬ БЕЗ ПЕРЕНОСА ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ: +L(401)

Назначение

Операция сложения 8-разрядных (двойное слово) шестнадцатеричных чисел и/или констант.

Символ РКС



Au: Первое слово 1–го слагаемого

Ad: Первое слово 2–го слагаемого

R: Первое слово результата

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	+L(401)
	Однократное выполнение по положительному фронту	@+L(401)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

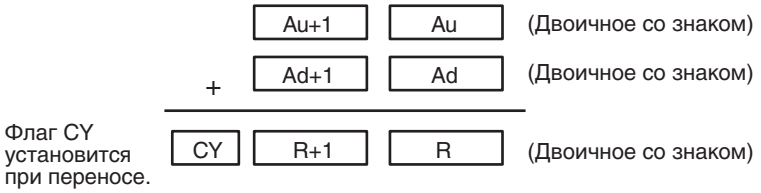
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	Au	Ad	R
Область CIO	CIO 0...CIO 6142		
Рабочая область	W0...W510		
Область битов хранения	H0...H510		
Область вспомогательных битов	A0...A958		A448...A958
Область таймеров	T0000...T4094		
Область счетчиков	C0000...C4094		
Область DM	D0...D32766		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	#00000000...#FFFFFFFF (двоичное) &0...&4294967295 (десятичное без знака) -2147483648...2147483647 (десятичное со знаком)		---
Регистры данных	---		
Регистры указателей	IR0...IR15		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда +L(401) складывает двоичное значение [Au, Au+1] со значением [Ad, Ad+1] и выводит результат в [R, R+1].



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Выключен
Флаг равенства	=	Включен, если результат = 0. Выключен во всех остальных случаях.
Флаг переноса	CY	Включен, если сложение приводит к переносу. Выключен во всех остальных случаях.
Флаг переполнения	OF	Включен, если результат сложения двух положительных чисел принадлежит диапазону 00000000...7FFFFFFF hex. Выключен во всех остальных случаях.
Флаг потери значимости	UF	Включен, если результат сложения двух отрицательных чисел принадлежит диапазону 00000000...7FFFFFFF hex. Выключен во всех остальных случаях.
Флаг отрицательного значения	N	Включен, если старший бит результата = «1». Выключен во всех остальных случаях.

Меры предосторожности

При выполнении команды +L(401) флаг ошибки выключается.

Если в результате операции сложения содержимое [R, R+1] окажется равным 00000000 hex, будет установлен флаг равенства.

Если сложение приведет к переносу, установится флаг переноса.

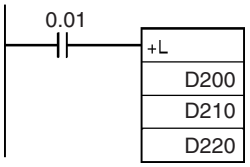
Если результат сложения двух положительных чисел является отрицательным числом (в диапазоне 80000000...FFFFFFFF hex), включается флаг переполнения.

Если результат сложения двух отрицательных чисел является положительным числом (в диапазоне 00000000...7FFFFFFF hex), включается флаг потери значимости.

Если в результате операции сложения старший бит R+1 окажется равным «1», будет установлен флаг отрицательного значения.

Примеры

Если в следующем примере бит CIO 0.01 включен, выполняется сложение содержимого слов [D200, D201] и слов [D211, D210] как 8-разрядных двоичных чисел со знаком, результат выводится в слова [D221, D220].



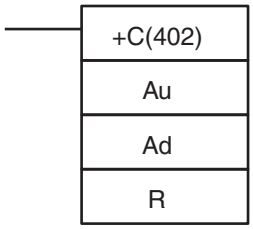
3-10-3 СЛОЖИТЬ С ПЕРЕНОСОМ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ: +C(402)

Назначение

Операция сложения 4-разрядных (одно слово) шестнадцатеричных чисел и/или констант с участием флага переноса (CY).

Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

Символ РКС



Au: Слово 1–го слагаемого

Ad: Слово 2–го слагаемого

R: Слово результата

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	+C(402)
	Однократное выполнение по положительному фронту	@+C(402)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

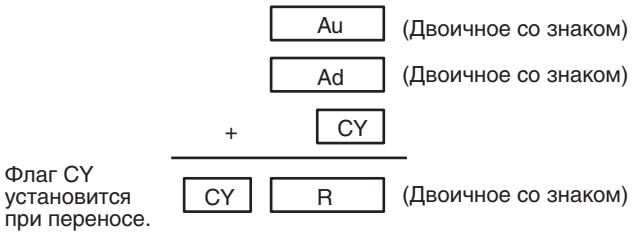
Области программных блоков	**Области пошаговых программ**	**Подпрограм-мы**	**Задачи обработки прерываний**
OK	OK	OK	OK

Характеристики операндов

Область	**Au**	**Ad**	**R**
Область CIO	CIO 0...CIO 6143		
Рабочая область	W0...W511		
Область битов хранения	H0...H511		
Область вспомогательных битов	A0...A959		A448...A959
Область таймеров	T0000...T4095		
Область счетчиков	C0000...C4095		
Область DM	D0...D32767		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	#0000...#FFFF (двоичное) &0...&65535 (десятичное без знака) -32768...0...32767 (десятичное со знаком)		---
Регистры данных	DR0...DR15		
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда +C(402) складывает двоичные значения Au, Ad и флага CY и выводит результат в R.



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Выключен
Флаг равенства	=	Включен, если результат сложения = 0. Выключен во всех остальных случаях.
Флаг переноса	CY	Включен, если сложение приводит к переносу. Выключен во всех остальных случаях.
Флаг переполнения	OF	Включен, если результат сложения двух положительных чисел и CY принадлежит диапазону 8000...FFFF hex. Выключен во всех остальных случаях.
Флаг потери значимости	UF	Включен, если результат сложения двух отрицательных чисел и CY принадлежит диапазону 0000...7FFF hex. Выключен во всех остальных случаях.
Флаг отрицательного значения	N	Включен, если старший бит результата = «1». Выключен во всех остальных случаях.

Меры предосторожности

При выполнении команды +C(402) флаг ошибки выключается.

Если в результате операции сложения содержимое R окажется равным 0000 hex, будет установлен флаг равенства.

Если сложение приведет к переносу, установится флаг переноса.

Если результат сложения двух положительных чисел и CY является отрицательным числом (в диапазоне 8000...FFFF hex), включается флаг переполнения.

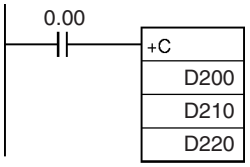
Если результат сложения двух отрицательных чисел и CY является положительным числом (в диапазоне 0000...7FFF hex), включается флаг потери значимости.

Если в результате операции сложения старший бит R окажется равным «1», будет установлен флаг отрицательного значения.

Примечание. Для того чтобы очистить флаг переноса (CY), выполните команду CLC(041) (Сброс флага переноса).

Примеры

Если в следующем примере бит CIO 0.00 включен, с участием флага CY выполняется сложение содержимого слов D200 и D210 как 4-разрядных двоичных чисел со знаком, результат выводится в слово D220.

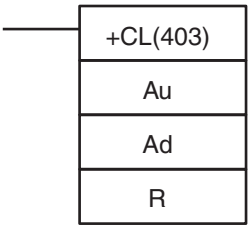


3-10-4 СЛОЖИТЬ С ПЕРЕНОСОМ ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ: +CL(403)

Назначение

Операция сложения 8-разрядных (двойное слово) шестнадцатеричных чисел и/или констант с участием флага переноса (CY).

Символ РКС



Au: Первое слово 1–го слагаемого

Ad: Первое слово 2–го слагаемого

R: Первое слово результата

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	+CL(403)
	Однократное выполнение по положительному фронту	@+CL(403)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

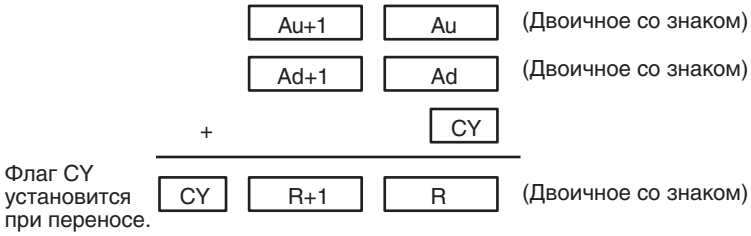
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	Au	Ad	R
Область CIO	CIO 0...CIO 6142		
Рабочая область	W0...W510		
Область битов хранения	H0...H510		
Область вспомогательных битов	A0...A958		A448...A958
Область таймеров	T0000...T4094		
Область счетчиков	C0000...C4094		
Область DM	D0...D32766		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	#00000000...#FFFFFFFF (двоичное) &0...&4294967295 (десятичное без знака) -2147483648...2147483647 (десятичное со знаком)		---
Регистры данных	---		
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда +CL(403) складывает двоичное значение [Au, Au+1] с двоичным значением [Ad, Ad+1 и CY] и выводит результат в [R, R+1].



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Выключен
Флаг равенства	=	Включен, если результат = 0. Выключен во всех остальных случаях.
Флаг переноса	CY	Включен, если сложение приводит к переносу. Выключен во всех остальных случаях.
Флаг переполнения	OF	Включен, если результат сложения двух положительных чисел и CY принадлежит диапазону 80000000...FFFFFFFF hex. Выключен во всех остальных случаях.
Флаг потери значимости	UF	Включен, если результат сложения двух отрицательных чисел и CY принадлежит диапазону 00000000...7FFFFFFF hex. Выключен во всех остальных случаях.
Флаг отрицательного значения	N	Включен, если старший бит результата = «1». Выключен во всех остальных случаях.

Меры предосторожности

При выполнении команды +CL(403) флаг ошибки выключается.

Если в результате операции сложения содержимое [R, R+1] окажется равным 00000000 hex, будет установлен флаг равенства.

Если сложение приведет к переносу, установится флаг переноса.

Если результат сложения двух положительных чисел и CY является отрицательным числом (в диапазоне 80000000...FFFFFFFF hex), включается флаг переполнения.

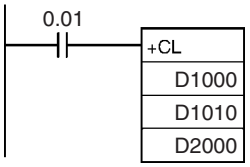
Если результат сложения двух отрицательных чисел и CY является положительным числом (в диапазоне 00000000...7FFFFFFF hex), включается флаг потери значимости.

Если в результате операции сложения старший бит R+1 окажется равным «1», будет установлен флаг отрицательного значения.

Примечание. Для того чтобы очистить флаг переноса (CY), выполните команду CLC(041) (Сброс флага переноса).

Примеры

Если в следующем примере бит CIO 0.01 включен, с участием флага CY выполняется сложение содержимого слов [D1001, D1000] и [D1011, D1010] как 8-разрядных двоичных чисел со знаком, результат выводится в слова [D2001, D2000].

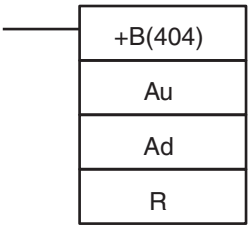


3-10-5 СЛОЖИТЬ БЕЗ ПЕРЕНОСА ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА: +B(404)

Назначение

Операция сложения 4-разрядных (одно слово) чисел и/или констант в двоично-десятичном формате.

Символ РКС



Au: Слово 1–го слагаемого

Ad: Слово 2–го слагаемого

R: Слово результата

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	+B(404)
	Однократное выполнение по положительному фронту	@+B(404)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

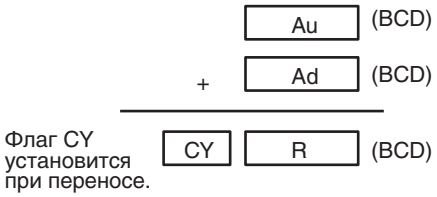
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	Au	Ad	R
Область CIO	CIO 0...CIO 6143		
Рабочая область	W0...W511		
Область битов хранения	H0...H511		
Область вспомогательных битов	A0...A959		A448...A959
Область таймеров	T0000...T4095		
Область счетчиков	C0000...C4095		
Область DM	D0...D32767		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	0000...9999 (BCD)		---
Регистры данных	DR0...DR15		
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда +B(404) складывает значения Au и Ad в двоично-десятичном формате и выводит результат в R.



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если Au не в формате BCD. Включен, если Ad не в формате BCD. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если результат = 0. Выключен во всех остальных случаях.
Флаг переноса	CY	Включен, если сложение приводит к переносу. Выключен во всех остальных случаях.

Меры предосторожности

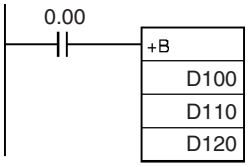
Если Au или Ad не является двоично-десятичным значением, будет сгенерирована ошибка и установится флаг ошибки.

Если в результате операции сложения содержимое R окажется равным 0000 hex, будет установлен флаг равенства.

Если сложение приведет к переносу, установится флаг переноса.

Примеры

Если в следующем примере бит CIO 0.00 включен, выполняется сложение содержимого слов D100 и D110 как 4-разрядных двоично-десятичных чисел, результат выводится в слово D120.

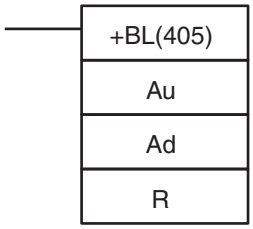


3-10-6 СЛОЖИТЬ БЕЗ ПЕРЕНОСА ДВОЙНЫЕ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА: +BL(405)

Назначение

Операция сложения 8-разрядных (двойное слово) чисел и/или констант в двоично-десятичном формате.

Символ РКС



Au: Первое слово 1–го слагаемого

Ad: Первое слово 2–го слагаемого

R: Первое слово результата

Варианты выполнения

Варианты выполнения		
	Выполнение в каждом цикле при включенном условии выполнения	+BL(405)
	Однократное выполнение по положительному фронту	@+BL(405)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

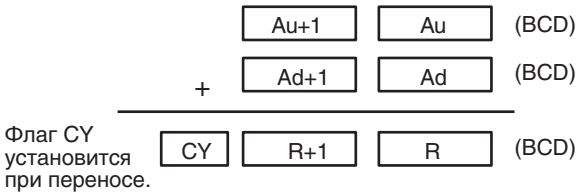
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	Au	Ad	R
Область CIO	CIO 0...CIO 6142		
Рабочая область	W0...W510		
Область битов хранения	H0...H510		
Область вспомогательных битов	A0...A958		A448...A958
Область таймеров	T0000...T4094		
Область счетчиков	C0000...C4094		
Область DM	D0...D32766		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	#00000000...#99999999 (BCD)		---
Регистры данных	---		
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда +BL(405) складывает двоично-десятичное значение [Au, Au+1] с двоично-десятичным значением [Ad, Ad+1] и выводит результат в [R, R+1].



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если [Au, Au +1] не в формате BCD. Включен, если [Ad, Ad +1] не в формате BCD. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если результат = 0. Выключен во всех остальных случаях.
Флаг переноса	CY	Включен, если сложение приводит к переносу. Выключен во всех остальных случаях.

Меры предосторожности

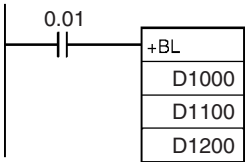
Если [Au, Au +1] или [Ad, Ad +1] не является двоично-десятичным значением, будет сгенерирована ошибка и установится флаг ошибки.

Если в результате операции сложения содержимое [R, R+1] окажется равным 00000000 hex, будет установлен флаг равенства.

Если сложение приведет к переносу, установится флаг переноса.

Примеры

Если в следующем примере бит CIO 0.01 включен, выполняется сложение содержимого слов [D1001, D1000] с содержимым слов [D1101, D1100] как 8-разрядных двоично-десятичных чисел, результат выводится в слова [D1201, D1200].

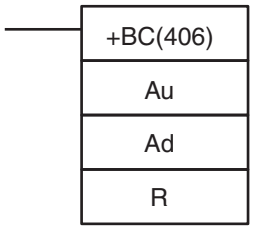


3-10-7 СЛОЖИТЬ С ПЕРЕНОСОМ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА: +BC(406)

Назначение

Операция сложения 4-разрядных (одно слово) чисел и/или констант в двоично-десятичном формате с участием флага переноса (CY).

Символ РКС



Au: Слово 1–го слагаемого

Ad: Слово 2–го слагаемого

R: Слово результата

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	+BC(406)
	Однократное выполнение по положительному фронту	@+BC(406)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

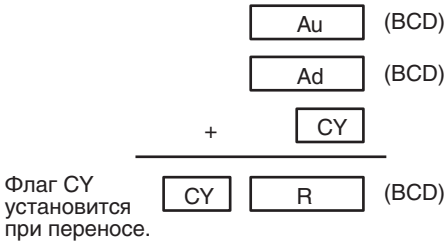
Характеристики операндов

Область	Au	Ad	R
Область CIO	CIO 0...CIO 6143		
Рабочая область	W0...W511		
Область битов хранения	H0...H511		
Область вспомогательных битов	A0...A959		A448...A959
Область таймеров	T0000...T4095		
Область счетчиков	C0000...C4095		
Область DM	D0...D32767		

Область	Au	Ad	R
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	#0000 ... 9999 (BCD)		---
Регистры данных	DR0...DR15		
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда +BC(406) складывает двоично-десятичные значения Au и Ad с участием флага CY и выводит результат в R.



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если Au не в формате BCD. Включен, если Ad не в формате BCD. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если результат = 0. Выключен во всех остальных случаях.
Флаг переноса	CY	Включен, если сложение приводит к переносу. Выключен во всех остальных случаях.

Меры предосторожности

Если Au или Ad не является двоично-десятичным значением, будет сгенерирована ошибка и установится флаг ошибки.

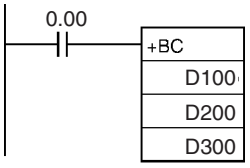
Если в результате операции сложения содержимое R окажется равным 0000 hex, будет установлен флаг равенства.

Если сложение приведет к переносу, установится флаг переноса.

Примечание. Для того чтобы очистить флаг переноса (CY), выполните команду CLC(041) (Сброс флага переноса).

Примеры

Если в следующем примере бит CIO 0.00 включен, с участием флага CY выполняется сложение содержимого слов D100 и D200 как 4-разрядных двоично-десятичных чисел, результат выводится в слово D300.

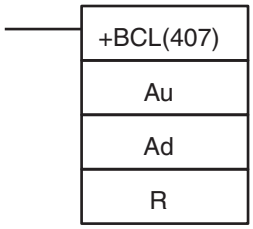


3-10-8 СЛОЖИТЬ С ПЕРЕНОСОМ ДВОЙНЫЕ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА: +BCL(407)

Назначение

Операция сложения 8-разрядных (двойное слово) чисел и/или констант в двоично-десятичном формате с участием флага переноса (CY).

Символ РКС



Au: Первое слово 1–го слагаемого

Ad: Первое слово 2–го слагаемого

R: Первое слово результата

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	+BCL(407)
	Однократное выполнение по положительному фронту	@+BCL(407)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

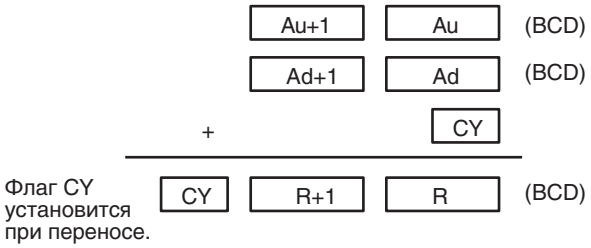
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	Au	Ad	R
Область CIO	CIO 0...CIO 6142		
Рабочая область	W0...W510		
Область битов хранения	H0...H510		
Область вспомогательных битов	A0...A958		A448...A958
Область таймеров	T0000...T4094		
Область счетчиков	C0000...C4094		
Область DM	D0...D32766		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	#00000000...#99999999 (BCD)		---
Регистры данных	---		
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда +BCL(407) складывает двоично-десятичное значение [Au, Au+1] с двоично-десятичным значением [Ad, Ad+1 и CY] и выводит результат в [R, R+1].



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если [Au, Au +1] не в формате BCD. Включен, если [Ad, Ad +1] не в формате BCD. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если результат = 0. Выключен во всех остальных случаях.
Флаг переноса	CY	Включен, если сложение приводит к переносу. Выключен во всех остальных случаях.

Меры предосторожности

Если [Au, Au +1] или [Ad, Ad +1] не является двоично-десятичным значением, будет сгенерирована ошибка и установится флаг ошибки.

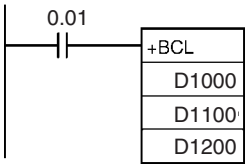
Если в результате операции сложения содержимое [R, R+1] окажется равным 00000000 hex, будет установлен флаг равенства.

Если сложение приведет к переносу, установится флаг переноса.

Примечание. Для того чтобы очистить флаг переноса (CY), выполните команду CLC(041) (Сброс флага переноса).

Примеры

Если в следующем примере бит CIO 0.01 включен, с участием флага CY выполняется сложение содержимого слов [D1001, D1000] с содержимым слов [D1101, D1100] как 8-разрядных двоично-десятичных чисел, результат выводится в слова [D1201, D1200].

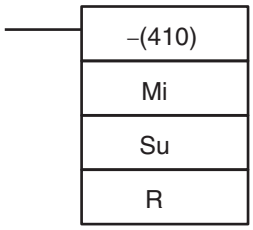


3-10-9 ВЫЧЕСТЬ БЕЗ ПЕРЕНОСА ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ: –(410)

Назначение

Операция вычитания 4-разрядных (одно слово) шестнадцатеричных чисел и/или констант.

Символ РКС



Mi: Слово уменьшаемого

Su: Слово вычитаемого

R: Слово результата

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	–(410)
	Однократное выполнение по положительному фронту	@–(410)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

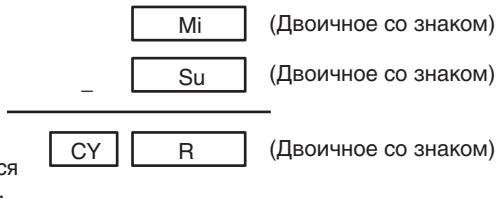
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	Mi	Su	R
Область CIO	CIO 0...CIO 6143		
Рабочая область	W0...W511		
Область битов хранения	H0...H511		
Область вспомогательных битов	A0...A959		A448...A959
Область таймеров	T0000...T4095		
Область счетчиков	C0000...C4095		
Область DM	D0...D4095		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	#0000...#FFFF (двоичное) &0...&65535 (десятичное без знака) -32768.. 32767 (десятичное со знаком)		---
Регистры данных	DR0...DR15		
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда –(400) вычитает двоичное значение Su из двоичного значения Mi и выводит результат в R. Если в результате получается отрицательное число, в R выводится дополнение до двух. (Пример работы с дополнениями до двух см. в *3-10-10 ВЫЧЕСТЬ БЕЗ ПЕРЕНОСА ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ: –L(411)*.)



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Выключен
Флаг равенства	=	Включен, если результат = 0. Выключен во всех остальных случаях.
Флаг переноса	CY	Включен, если вычитание приводит к заему. Выключен во всех остальных случаях.
Флаг переполнения	OF	Включен, если результат вычитания отрицательного числа из положительного числа принадлежит диапазону 8000...FFFF hex. Выключен во всех остальных случаях.
Флаг потери значимости	UF	Включен, если результат вычитания положительного числа из отрицательного числа принадлежит диапазону 0000...7FFF hex. Выключен во всех остальных случаях.
Флаг отрицательного значения	N	Включен, если старший бит результата = «1». Выключен во всех остальных случаях.

Меры предосторожности

При выполнении команды –(410) флаг ошибки выключается.

Если в результате операции вычитания содержимое R окажется равным 0000 hex, будет установлен флаг равенства.

Если вычитание приведет к заему, установится флаг переноса.

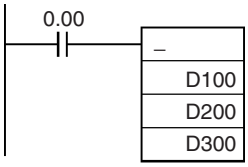
Если результат вычитания отрицательного числа из положительного числа является отрицательным числом (в диапазоне 8000...FFFF hex), включается флаг переполнения.

Если результат вычитания положительного числа из отрицательного числа является положительным числом (в диапазоне 0000...7FFF hex), включается флаг потери значимости.

Если в результате операции вычитания старший бит R окажется равным «1», будет установлен флаг отрицательного значения.

Примеры

Если в следующем примере бит CIO 0.00 включен, выполняется вычитание значения слова D200 из значения слова D100 как 4-разрядных двоичных чисел со знаком, результат выводится в слово D300.

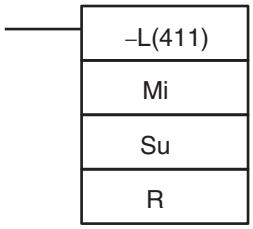


3-10-10 ВЫЧЕСТЬ БЕЗ ПЕРЕНОСА ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ: –L(411)

Назначение

Операция вычитания 8-разрядных (двойное слово) шестнадцатеричных чисел и/или констант.

Символ РКС



Mi: Слово уменьшаемого

Su: Слово вычитаемого

R: Слово результата

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	–L(411)
	Однократное выполнение по положительному фронту	@–L(411)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

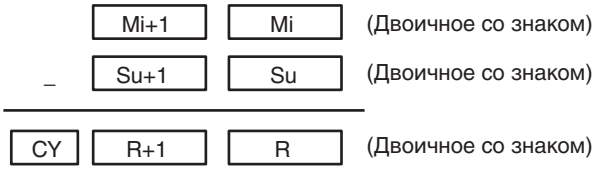
Характеристики операндов

Область	Mi	Su	R
Область CIO	CIO 0...CIO 6142		
Рабочая область	W0...W510		
Область битов хранения	H0...H510		

Область	Mi	Su	R
Область вспомогательных битов	A0...A958		A448...A958
Область таймеров	T0000...T4094		
Область счетчиков	C0000...C4094		
Область DM	D0...D32766		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	#00000000...#FFFFFFFF (двоичное) &0...&4294967295 (десятичное без знака) -2147483648...2147483647 (десятичное со знаком)		---
Регистры данных	---		
Регистры указателей	IR0...IR15		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда –L(411) вычитает двоичное значение [Su, Su+1] из двоичного значения [Mi, Mi+1] и выводит результат в [R, R+1]. Если в результате получается отрицательное число, оно выводится в [R, R+1] как дополнение до двух.



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Выключен
Флаг равенства	=	Включен, если результат = 0. Выключен во всех остальных случаях.
Флаг переноса	CY	Включен, если вычитание приводит к заему. Выключен во всех остальных случаях.
Флаг переполнения	OF	Включен, если результат вычитания отрицательного числа из положительного числа принадлежит диапазону 80000000...FFFFFFFF hex. Выключен во всех остальных случаях.
Флаг потери значимости	UF	Включен, если результат вычитания положительного числа из отрицательного числа принадлежит диапазону 00000000...7FFFFFFF hex. Выключен во всех остальных случаях.
Флаг отрицательного значения	N	Включен, если старший бит результата = «1». Выключен во всех остальных случаях.

Меры предосторожности

При выполнении команды –L(411) флаг ошибки выключается.

Если в результате операции вычитания содержимое [R, R+1] окажется равным 00000000 hex, будет установлен флаг равенства.

Если вычитание приведет к заему, установится флаг переноса.

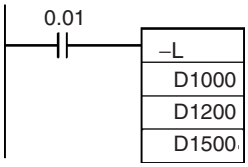
Если результат вычитания отрицательного числа из положительного числа является отрицательным числом (в диапазоне 80000000...FFFFFFFF hex), включается флаг переполнения.

Если результат вычитания положительного числа из отрицательного числа является положительным числом (в диапазоне 00000000...7FFFFFFF hex), включается флаг потери значимости.

Если в результате операции вычитания старший бит R+1 окажется равным «1», будет установлен флаг отрицательного значения.

Примеры

Если в следующем примере бит CIO 0.01 включен, выполняется вычитание значения слов [D1201, D1200] из значения слов [D1001, D1000] как 8-разрядных двоичных чисел со знаком, результат выводится в слова [D1501, D1500].



Примеры

Если результат вычитания является отрицательным числом (Mi<Su или [Mi+1, Mi] < [Su+1, Su]), в слово результата выводится дополнение этого числа до двух, при этом включается флаг переноса (CY), указывая на то, что результат вычитания — отрицательное число. Для получения фактического значения из дополнения до двух требуется программа, вычитающая результат из 0 и использующая флаг переноса (CY) в качестве условия выполнения.

Примечание. **Дополнение до двух и дополнительный код**

Дополнение двоичного числа до двух: получают путем вычитания каждого разряда этого числа из 1 (фактически, инверсией каждого разряда) и добавления 1 к результату. Дополнительный код отрицательного двоичного числа: к полученному дополнению до двух слева дописывают «1» в качестве разряда знака. Например, дополнение до двух для числа 1101 вычисляется следующим образом: 1111 (F hex) - 1101 (D hex) + 1 (1 hex) = 0011 (3 hex). Дополнение числа 3039 (hex) до двух рассчитывается следующим образом: FFFF (hex) - 3039 (hex) + 0001 (hex) = CFC7 (hex). Следовательно, дополнение до двух 4-разрядного шестнадцатеричного значения можно рассчитать следующим образом: FFFF (hex) - a (hex) + 0001 (hex) = b (hex). Фактическое значение из дополнения до двух b (hex): a (hex) = 10000 (hex) - b (hex). Например, для получения фактического значения из дополнения до двух CFC7 (hex): 10000 (hex) - CFC7 (hex) = 3039 (hex).

Пример 1

	Значение со знаком	Значение без знака
FFFF Hex →	–1	65535
–) 0001 Hex →	–) +1	–) 1
FFFE Hex →	–2 Примеч. 1	65534 Примеч. 2

Флаг «Меньше нуля» = ВКЛ
Флаг переноса = ВЫКЛ

Примеч. 1. Так как флаг «Меньше нуля» включен, результат (FFFE hex, дополнение до двух) является отрицательным и, следовательно, равен –2.

2. Так как флаг переноса выключен, результат (FFFE hex) является положительным значением без знака и равен 65534.

Пример 2

	Значение со знаком	Значение без знака
FFFD Hex →	–3	65533
–)FFFF Hex →	–) –1	–) 65535
FFFE Hex →	–2 Примеч. 3	65534 Примеч. 4

Флаг «Меньше нуля» = ВКЛ
Флаг переноса = ВЫКЛ

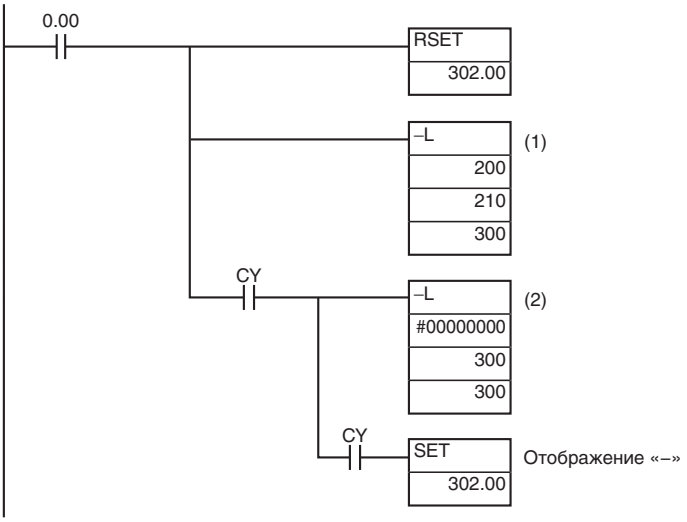
3. Так как флаг «Меньше нуля» включен, результат (FFFE hex, дополнение до двух) является отрицательным и, следовательно, равен –2.

4. Так как флаг переноса включен, результат (FFFE hex, дополнение до двух) является отрицательным и принимает значение –2 после преобразования к фактическому значению.

Пример программы

20F55A10 - B8A360E3 = -97AE06D3.

В данном примере 8-разрядное двоичное значение слов [CIO 211, CIO 210] вычитается из значения слов [CIO 201, CIO 200], а результат выводится в слова [CIO 301, CIO 300] как 8-разрядное двоичное число. Если результат — отрицательное число, выполняется команда (2) и результат выводится в слова [CIO 301, CIO 300].



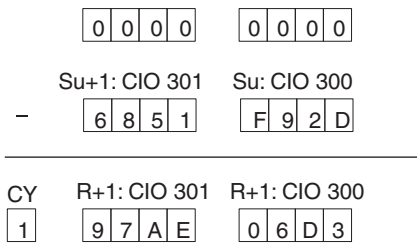
Вычитание в п. 1

	Mi+1: CIO 201	Mi: CIO 200
	2 0 F 5	5 A 1 0
	Su+1: CIO 211	Su: CIO 210
–	B 8 A 3	6 0 E 3

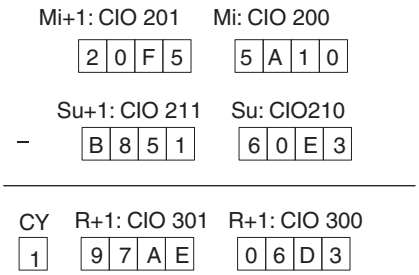
CY	R+1: D301	R+1: D300
1	6 8 5 1	F 9 2 D

Флаг переноса (CY) включен, поэтому для получения фактического значения результат вычитается из 0000 0000.

Вычитание в п. 2



Конечный результат вычитани



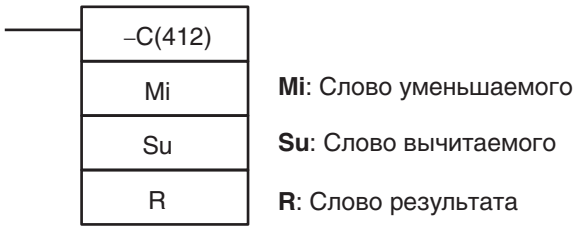
Флаг переноса (CY) включен, так что фактическое значение равно - 97AE06D3. Так как слова [CIO 301, CIO 300] содержат отрицательное значение, флаг CY используется для установки бита CIO 302.00, чтобы указать на это.

3-10-11 ВЫЧЕСТЬ С ПЕРЕНОСОМ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ: –C(412)

Назначение

Операция вычитания 4-разрядных (одно слово) шестнадцатеричных чисел и/или констант с участием флага переноса (CY).

Символ РКС



Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	–C(412)
	Однократное выполнение по положительному фронту	@–C(412)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

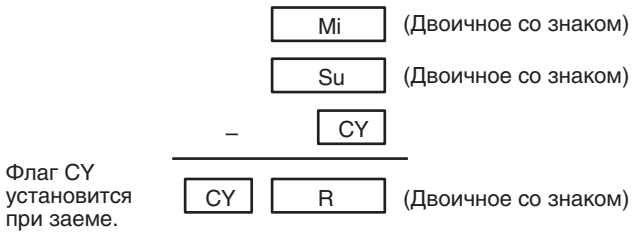
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	Mi	Su	R
Область CIO	CIO 0...CIO 6143		
Рабочая область	W0...W511		
Область битов хранения	H0...H511		
Область вспомогательных битов	A0...A959		A448...A959
Область таймеров	T0000...T4095		
Область счетчиков	C0000...C4095		
Область DM	D0...D32767		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	#0000...#FFFF (двоичное) &0...&65535 (десятичное без знака) -32768...0...32767 (десятичное со знаком)		---
Регистры данных	DR0...DR15		
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда –(412) вычитает двоичное значение Su и флага CY из двоичного значения Mi и выводит результат в R. Если в результате получается отрицательное число, в R выводится дополнение до двух.



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Выключен
Флаг равенства	=	Включен, если результат вычитания = 0. Выключен во всех остальных случаях.
Флаг переноса	CY	Включен, если вычитание приводит к заему. Выключен во всех остальных случаях.
Флаг переполнения	OF	Включен, если результат вычитания отрицательного числа и CY из положительного числа принадлежит диапазону 8000...FFFF hex. Выключен во всех остальных случаях.

Название	Обозначение	Описание работы
Флаг потери значимости	UF	Включен, если результат вычитания положительного числа и CY из отрицательного числа принадлежит диапазону 0000...7FFF hex. Выключен во всех остальных случаях.
Флаг отрицательного значения	N	Включен, если старший бит результата = «1». Выключен во всех остальных случаях.

Меры предосторожности

При выполнении команды –C(412) флаг ошибки выключается.

Если в результате операции вычитания содержимое R окажется равным 0000 hex, будет установлен флаг равенства.

Если вычитание приведет к заему, установится флаг переноса.

Если результат вычитания отрицательного числа и CY из положительного числа является отрицательным числом (в диапазоне 8000...FFFF hex), включается флаг переполнения.

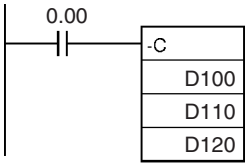
Если результат вычитания положительного числа и CY из отрицательного числа является положительным числом (в диапазоне 0000...7FFF hex), включается флаг потери значимости.

Если в результате операции вычитания старший бит R окажется равным «1», будет установлен флаг отрицательного значения.

Примечание. Для того чтобы очистить флаг переноса (CY), выполните команду CLC(041) (Сброс флага переноса).

Примеры

Если в следующем примере бит CIO 0.00 включен, с участием флага CY выполняется вычитание значения слова D110 из значения слова D100 как 4-разрядных двоичных чисел со знаком, результат выводится в слово D120.

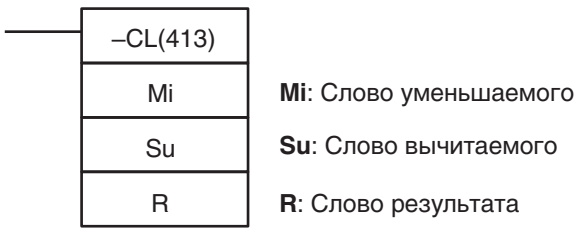


3-10-12 ВЫЧЕСТЬ С ПЕРЕНОСОМ ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ: –CL(413)

Назначение

Операция вычитания 8-разрядных (двойное слово) шестнадцатеричных чисел и/или констант с участием флага переноса (CY).

Символ РКС



Mi: Слово уменьшаемого

Su: Слово вычитаемого

R: Слово результата

Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	–CL(413)
	Однократное выполнение по положительному фронту	@–CL(413)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

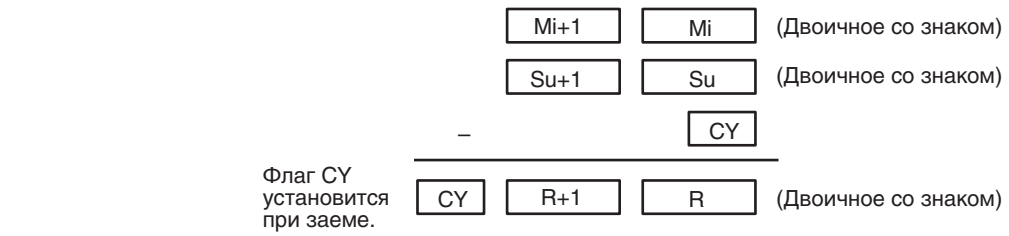
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	Mi	Su	R
Область CIO	CIO 0...CIO 6142		
Рабочая область	W0...W510		
Область битов хранения	H0...H510		
Область вспомогательных битов	A0...A958		A448...A958
Область таймеров	T0000...T4094		
Область счетчиков	C0000...C4094		
Область DM	D0...D32766		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	#00000000...#FFFFFFFF (двоичное) &0...&4294967295 (десятичное без знака) -2147483648...2147483647 (десятичное со знаком)		---
Регистры данных	---		
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда –CL(413) вычитает двоичное значение [Su, Su+1 и CY] из двоичного значения [Mi, Mi+1] и выводит результат в [R, R+1]. Если в результате получается отрицательное число, в [R, R+1] выводится дополнение до двух.



Предварительная версия
Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Выключен
Флаг равенства	=	Включен, если результат = 0. Выключен во всех остальных случаях.
Флаг переноса	CY	Включен, если вычитание приводит к заему. Выключен во всех остальных случаях.
Флаг переполнения	OF	Включен, если результат вычитания отрицательного числа и CY из положительного числа принадлежит диапазону 80000000...FFFFFFFF hex. Выключен во всех остальных случаях.
Флаг потери значимости	UF	Включен, если результат вычитания положительного числа и CY из отрицательного числа принадлежит диапазону 00000000...7FFFFFFF hex. Выключен во всех остальных случаях.
Флаг отрицательного значения	N	Включен, если старший бит результата = «1». Выключен во всех остальных случаях.

Меры предосторожности

При выполнении команды –CL(413) флаг ошибки выключается.

Если в результате операции вычитания содержимое [R, R+1] окажется равным 00000000 hex, будет установлен флаг равенства.

Если вычитание приведет к заему, установится флаг переноса.

Если результат вычитания отрицательного числа и флага CY из положительного числа является отрицательным числом (в диапазоне 80000000...FFFFFFFF hex), включается флаг переполнения.

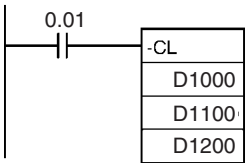
Если результат вычитания положительного числа и флага CY из отрицательного числа является положительным числом (в диапазоне 00000000...7FFFFFFF hex), включается флаг потери значимости.

Если в результате операции вычитания старший бит R+1 окажется равным «1», будет установлен флаг отрицательного значения.

Примечание. Для того чтобы очистить флаг переноса (CY), выполните команду CLC(041) (Сброс флага переноса).

Примеры

Если в следующем примере бит CIO 0.01 включен, с участием флага CY выполняется вычитание содержимого слов [D1101, D1100] из содержимого слов [D1001, D1000] как 8-разрядных двоичных чисел, результат выводится в слова [D1201, D1200].



Если в результате вычитания получается отрицательное число (Mi<Su или [Mi+1, Mi] < [Su+1, Su]), в качестве результата выдается дополнение до двух. При этом устанавливается флаг переноса (CY). Для получения фактического значения из дополнения до двух требуется программа, вычитающая результат из 0 и использующая флаг переноса (CY) в качестве условия выполнения. Фактически, установка флага переноса означает, что результат вычитания является отрицательным числом.

Примечание. **Дополнение до двух и дополнительный код**

Дополнение двоичного числа до двух: получают путем вычитания каждого разряда этого числа из 1 (фактически, инверсией каждого

разряда) и добавления 1 к результату. Дополнительный код отрицательного двоичного числа: к полученному дополнению до двух слева дописывают «1» в качестве разряда знака.

Пример: Дополнение двоичного числа 1101 до двух рассчитывается следующим образом:

1111 (F hex) - 1101 (D hex) + 1 (1 hex) = 0011 (3 hex).

Пример: Дополнение 4-разрядного шестнадцатеричного числа 3039 до двух рассчитывается следующим образом:

FFFF hex - 3039 hex + 0001 hex = CFC7 hex.

Таким образом, дополнение 4-разрядного шестнадцатеричного числа «a» до двух рассчитывается следующим образом:

FFFF hex - a hex + 0001 hex = b hex.

Фактическое значение a (hex) из дополнения до двух b (hex):

a hex = 10000 hex - b hex.

Пример: Фактическое значение из дополнения до двух CFC7 hex:

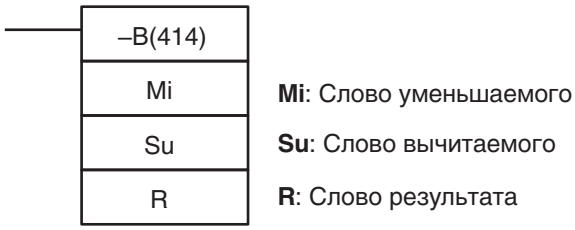
10000 hex - CFC7 hex = 3039 hex.

3-10-13 ВЫЧЕСТЬ БЕЗ ПЕРЕНОСА ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА: –B(414)

Назначение

Операция вычитания 4-разрядных (одно слово) чисел и/или констант в двоично-десятичном формате.

Символ РКС



Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	–B(414)
	Однократное выполнение по положительному фронту	@–B(414)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

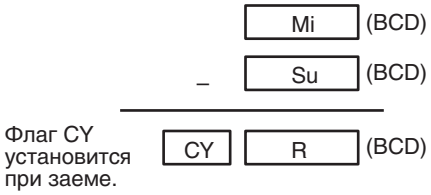
Характеристики операндов

Область	Mi	Su	R
Область CIO	CIO 0...CIO 6143		
Рабочая область	W0...W511		
Область битов хранения	H0...H511		
Область вспомогательных битов	A0...A959		A448...A959
Область таймеров	T0000...T4095		

Область	Mi	Su	R
Область счетчиков	C0000...C4095		
Область DM	D0...D32767		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	0000...9999 (BCD)		---
Регистры данных	DR0...DR15		
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда –B(414) вычитает значение Su из значения Mi в двоично-десятичном формате и выводит результат в R. Если в результате вычитания получается отрицательное число, в качестве результата выдается дополнение до 10.



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если Mi не в формате BCD. Включен, если Su не в формате BCD. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если результат = 0. Выключен во всех остальных случаях.
Флаг переноса	CY	Включен, если вычитание приводит к заему. Выключен во всех остальных случаях.

Меры предосторожности

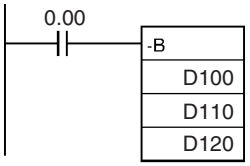
Если Mi и/или Su не является двоично-десятичным значением, будет сгенерирована ошибка и установится флаг ошибки.

Если в результате операции вычитания содержимое R окажется равным 0000 hex, будет установлен флаг равенства.

Если вычитание приведет к заему, установится флаг переноса.

Примеры

Если в следующем примере бит CIO 0.00 включен, выполняется вычитание значения слова D110 из значения слова D100 как 4-разрядных двоично-десятичных чисел, результат выводится в слово D120.

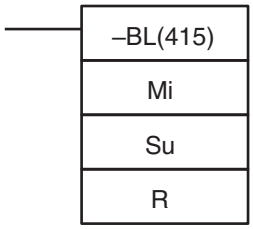


3-10-14 ВЫЧЕСТЬ БЕЗ ПЕРЕНОСА ДВОЙНЫЕ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА: –BL(415)

Назначение

Операция вычитания 8-разрядных (двойное слово) чисел и/или констант в двоично-десятичном формате.

Символ РКС



Mi: Первое слово уменьшаемого

Su: Первое слово вычитаемого

R: Первое слово результата

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	–BL(415)
	Однократное выполнение по положительному фронту	@–BL(415)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

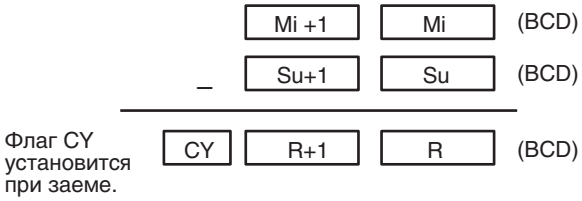
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	Mi	Su	R
Область CIO	CIO 0...CIO 6142		
Рабочая область	W0...W510		
Область битов хранения	H0...H510		
Область вспомогательных битов	A0...A958		A448...A958
Область таймеров	T0000...T4094		
Область счетчиков	C0000...C4094		
Область DM	D0...D32766		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	#00000000...#99999999 (BCD)		---
Регистры данных	---		
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда –BL(415) вычитает значение [Su, Su+1] из значения [Mi, Mi+1] в двоично-десятичном формате и выводит результат в [R, R+1]. Если в результате получается отрицательное число, в [R, R+1] выводится дополнение до 10.



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если Mi и/или Mi +1 не в формате BCD. Включен, если Su и/или Su +1 не в формате BCD. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если результат = 0. Выключен во всех остальных случаях.
Флаг переноса	CY	Включен, если вычитание приводит к заему. Выключен во всех остальных случаях.

Меры предосторожности

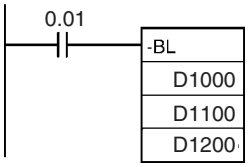
Если [Mi, Mi +1] и/или [Su, Su +1] не является двоично-десятичным значением, будет сгенерирована ошибка и установится флаг ошибки.

Если в результате операции вычитания содержимое [R, R+1] окажется равным 00000000 hex, будет установлен флаг равенства.

Если вычитание приведет к заему, установится флаг переноса.

Примеры

Если в следующем примере бит CIO 0.01 включен, выполняется вычитание значения слов [D1001, D1000] из значения слов [D1101, D1100] как 8-разрядных двоично-десятичных чисел, результат выводится в слова [D1201, D1200].



Если в результате вычитания получается отрицательное число (Mi<Su или [Mi+1, Mi] < [Su+1, Su]), в качестве результата выдается дополнение до 10. При этом устанавливается флаг переноса (CY). Для получения фактического значения из дополнения до 10 требуется программа, вычитающая результат из 0 и использующая флаг переноса (CY) в качестве условия выполнения. Фактически, установка флага переноса означает, что результат вычитания является отрицательным числом.

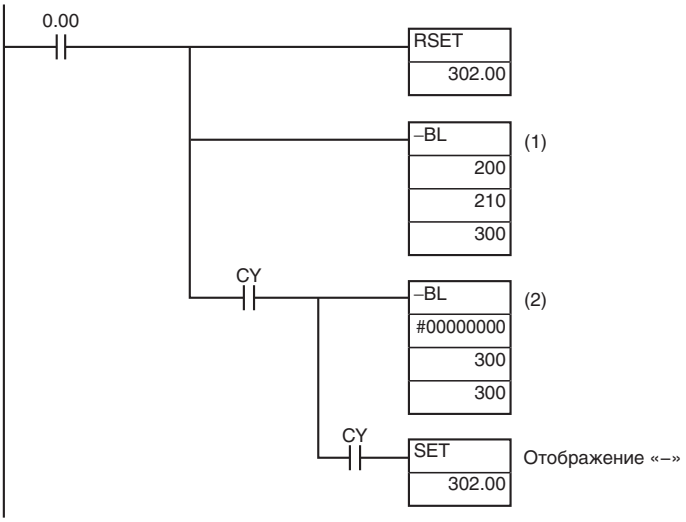
Примечание. **Дополнение до десяти**

Дополнение числа до десяти получают путем вычитания каждого разряда этого числа из 9 и добавления 1 к результату. Например, дополнение до 10 для числа 7556 вычисляется следующим образом: 9999 - 7556 + 1 = 2444. Дополнение до десяти 4-разрядного числа A определяется как: 9999 - A + 1 = B. Фактическое значение из дополнения до десяти B определяется как: A = 10000 - B. Например, если 2444 является дополнением до 10, то фактическое значение: 10000 - 2444 = 7556.

Пример программы

9583960 - 17072641 = -7488681.

В данном примере 8-разрядное двоично-десятичное значение слов [CIO 211, CIO 210] вычитается из 8-разрядного двоично-десятичного значения слов [CIO 201, CIO 200], а 8-разрядный двоично-десятичный результат выводится в слова [CIO 301, CIO 300]. Если результат — отрицательное число, выполняется команда (2) и в слова [CIO 301, CIO 300] выводится фактическое значение.



Вычитание в п. 1

Mi+1: CIO 201 Mi: CIO 200

0 9 5 8 | 3 9 6 0

Su+1: CIO 211 Su: CIO 210

– 1 7 0 7 | 2 6 4 1

09583960 + (100000000 – 17072641)

CY R+1: CIO 301 R+1: CIO 300

1 | 9 2 5 1 | 1 3 1 9

Флаг переноса (CY) включен, поэтому результат вычитается из 0000 0000.

Вычитание в п. 2

0 0 0 0 | 0 0 0 0

Su+1: CIO 301 Su: CIO 300

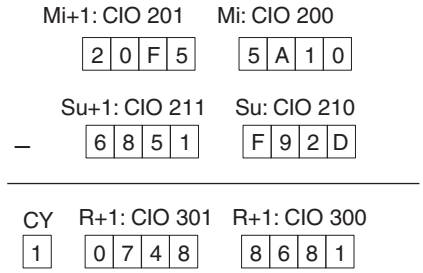
– 9 2 5 1 | 1 3 1 9

00000000 + (100000000 – 92511319)

CY R+1: CIO 301 R+1: CIO 300

1 | 0 7 4 8 | 8 6 8 1

Конечный результат вычитани



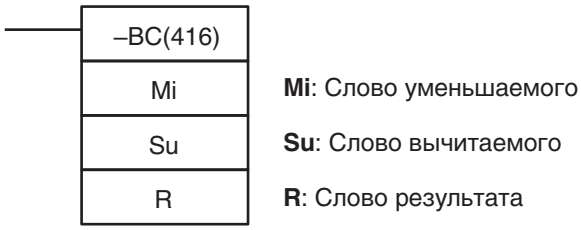
Флаг переноса (CY) будет включен, так что фактическое значение будет равно -7488681. Так как слова [CIO 301, CIO 300] содержат отрицательное значение, флаг CY используется для включения бита CIO 302.00, чтобы указать на это.

3-10-15 ВЫЧЕСТЬ С ПЕРЕНОСОМ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА: –BC(416)

Назначение

Операция вычитания 4-разрядных (одно слово) чисел и/или констант в двоично-десятичном формате с участием флага переноса (CY).

Символ РКС



Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	–BC(416)
	Однократное выполнение по положительному фронту	@–BC(416)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограм-мы	Задачи обработки прерываний
OK	OK	OK	OK

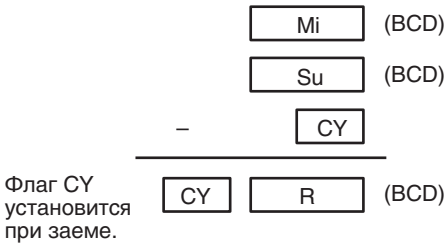
Характеристики операндов

Область	Mi	Su	R
Область CIO	CIO 0...CIO 6143		
Рабочая область	W0...W511		
Область битов хранения	H0...H511		
Область вспомогательных битов	A0...A959		A448...A959
Область таймеров	T0000...T4095		
Область счетчиков	C0000...C4095		

Область	Mi	Su	R
Область DM	D0...D32767		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	#0000...#9999 (BCD)		---
Регистры данных	DR0...DR15		
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда –BC(416) вычитает значение Su и флага CY из значения Mi в двоично-десятичном формате и выводит результат в R. Если в результате получается отрицательное число, в R выводится дополнение до десяти.



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если Mi не в формате BCD. Включен, если Su не в формате BCD. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если результат = 0. Выключен во всех остальных случаях.
Флаг переноса	CY	Включен, если вычитание приводит к заему. Выключен во всех остальных случаях.

Меры предосторожности

Если Mi и/или Su не является двоично-десятичным значением, будет сгенерирована ошибка и установится флаг ошибки.

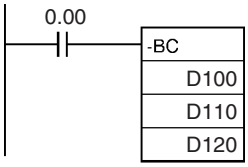
Если в результате операции вычитания содержимое R окажется равным 0000 hex, будет установлен флаг равенства.

Если вычитание приведет к заему, установится флаг переноса.

Примечание. Для того чтобы очистить флаг переноса (CY), выполните команду CLC(041) (Сброс флага переноса).

Примеры

Если в следующем примере бит CIO 0.00 включен, выполняется вычитание значения слова D110 и флага CY из значения слова D100 как 4-разрядных двоично-десятичных чисел, результат выводится в слово D120.

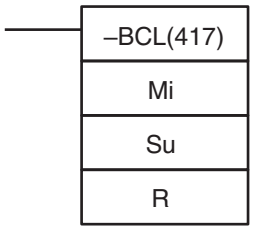


3-10-16 ВЫЧЕСТЬ С ПЕРЕНОСОМ ДВОЙНЫЕ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА: –BCL(417)

Назначение

Операция вычитания 8-разрядных (двойное слово) чисел и/или констант в двоично-десятичном формате с участием флага переноса (CY).

Символ РКС



Mi: Первое слово уменьшаемого

Su: Первое слово вычитаемого

R: Первое слово результата

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	–BCL(417)
	Однократное выполнение по положительному фронту	@–BCL(417)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

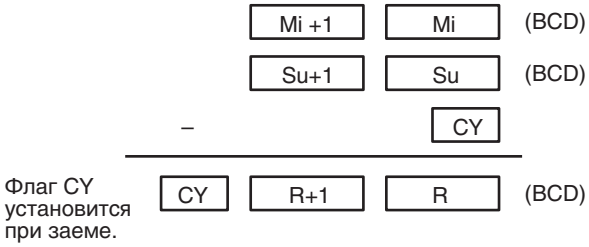
Характеристики операндов

Область	Mi	Su	R
Область CIO	CIO 0...CIO 6142		
Рабочая область	W0...W510		
Область битов хранения	H0...H510		
Область вспомогательных битов	A0...A958		A448...A958
Область таймеров	T0000...T4094		
Область счетчиков	C0000...C4094		
Область DM	D0...D32766		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	#00000000...#99999999 (BCD)		---
Регистры данных	---		

Область	Mi	Su	R
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда –BLC(417) вычитает значение [Su, Su+1 и CY] из значения [Mi, Mi+1] в двоично-десятичном формате и выводит результат в [R, R+1]. Если в результате получается отрицательное число, в [R, R+1] выводится дополнение до 10.



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если Mi и/или Mi +1 не в формате BCD. Включен, если Su и/или Su +1 не в формате BCD. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если результат = 0. Выключен во всех остальных случаях.
Флаг переноса	CY	Включен, если вычитание приводит к заему. Выключен во всех остальных случаях.

Меры предосторожности

Если [Mi, Mi +1] и/или [Su, Su +1] не является двоично-десятичным значением, будет сгенерирована ошибка и установится флаг ошибки.

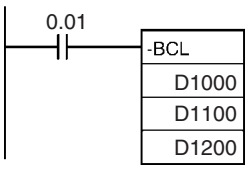
Если в результате операции вычитания содержимое [R, R+1] окажется равным 00000000 hex, будет установлен флаг равенства.

Если вычитание приведет к заему, установится флаг переноса.

Примечание. Для того чтобы очистить флаг переноса (CY), выполните команду CLC(041) (Сброс флага переноса).

Примеры

Если в следующем примере бит CIO 0.01 включен, выполняется вычитание содержимого слов [D1101, D1100 и CY] из содержимого слов [D1001, D1000] как 8-разрядных двоично-десятичных чисел, результат выводится в слова [D1201, D1200].



Если в результате вычитания получается отрицательное число (Mi<Su или [Mi+1, Mi] < [Su+1, Su]), в качестве результата выдается дополнение до 10. При этом устанавливается флаг переноса (CY). Для получения фактического значения из дополнения до 10 требуется программа, вычитающая результат из 0 и использующая флаг переноса (CY) в качестве условия выполнения. Фактически, установка флага переноса означает, что результат вычитания является отрицательным числом.

Примечание. **Дополнение до десяти**

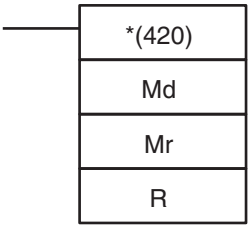
Дополнение числа до десяти получают путем вычитания каждого разряда этого числа из 9 и добавления 1 к результату. Например, дополнение до 10 для числа 7556 вычисляется следующим образом: 9999 - 7556 + 1 = 2444. Дополнение до десяти 4-разрядного числа A определяется как: 9999 - A + 1 = B. Фактическое значение из дополнения до десяти B определяется как: A = 10000 - B. Например, если 2444 является дополнением до 10, то фактическое значение: 10000 - 2444 = 7556.

3-10-17 УМНОЖИТЬ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ: *(420)

Назначение

Операция перемножения 4-разрядных шестнадцатеричных чисел со знаком и/или констант.

Символ РКС



Md: Слово 1–го множителя

Mr: Слово 2–го множителя

R: Слово результата

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	*(420)
	Однократное выполнение по положительному фронту	@*(420)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	Md	Mr	R
Область CIO	CIO 0...CIO 6143		CIO 0...CIO 6142
Рабочая область	W0...W511		W0...W510
Область битов хранения	H0...H511		H0...H510
Область вспомогательных битов	A0...A959		A448...A958
Область таймеров	T0000...T4095		T0000...T4094
Область счетчиков	C0000...C4095		C0000...C4094

Область	Md	Mr	R
Область DM	D0...D32767		D0...D32766
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	#0000...#FFFF (двоичное) &0...&65535 (десятичное без знака) -32768...32767 (десятичное со знаком)		---
Регистры данных	DR0...DR15		---
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда *(420) выполняет умножение Md и Mr как двоичных значений со знаком и выводит результат в [R, R+1].



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Выключен
Флаг равенства	=	Включен, если результат = 0. Выключен во всех остальных случаях.
Флаг отрицательного значения	N	Включен, если старший бит результата = «1». Выключен во всех остальных случаях.

Меры предосторожности

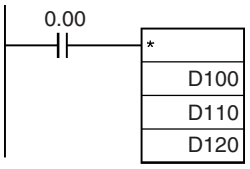
При выполнении команды *(420) флаг ошибки выключается.

Если в результате операции умножения содержимое R окажется равным 0000 hex, будет установлен флаг равенства.

Если в результате операции умножения старший бит [R+1, R] окажется равным «1», будет установлен флаг отрицательного значения.

Примеры

Если в следующем примере бит CIO 0.00 включен, выполняется умножение содержимого слов D100 и D110 как 4-разрядных шестнадцатеричных чисел со знаком, результат выводится в слова [D121, D120].

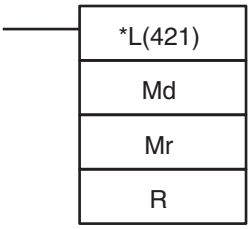


3-10-18 УМНОЖИТЬ ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ: *L(421)

Назначение

Операция перемножения 8-разрядных шестнадцатеричных чисел со знаком и/или констант.

Символ РКС



Md: Первое слово 1–го множителя

Mr: Первое слово 2–го множителя

R: Первое слово результата

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	*L(421)
	Однократное выполнение по положительному фронту	@*L(421)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	**Области пошаговых программ**	**Подпрограммы**	**Задачи обработки прерываний**
OK	OK	OK	OK

Характеристики операндов

Область	**Md**	**Mr**	**R**
Область CIO	CIO 0...CIO 6142		CIO 0...CIO 6140
Рабочая область	W0...W510		W0...W508
Область битов хранения	H0...H510		H0...H508
Область вспомогательных битов	A0...A958		A448...A956
Область таймеров	T0000...T4094		T0000...T4092
Область счетчиков	C0000...C4094		C0000...C4092
Область DM	D0...D32766		D0...D32764
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	#00000000...#FFFFFFFF (двоичное) &0...&4294967295 (десятичное без знака) -2147483648...0...2147483647 (десятичное со знаком)		---
Регистры данных	---		

Область	Md	Mr	R
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда *L(421) выполняет умножение [Md, Md+1] и [Mr, Mr+1] как двоичных значений со знаком и выводит результат в [R, R+1, R+2 и R+3].



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Выключен
Флаг равенства	=	Включен, если результат = 0. Выключен во всех остальных случаях.
Флаг отрицательного значения	N	Включен, если старший бит результата = «1». Выключен во всех остальных случаях.

Меры предосторожности

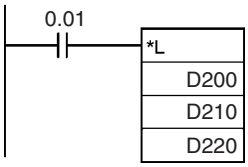
При выполнении команды *L(421) флаг ошибки выключается.

Если в результате операции умножения содержимое [R, R+1, R+2, R+3] окажется равным 0000 0000 0000 0000 hex, будет установлен флаг равенства.

Если в результате операции умножения старший бит R+3 окажется равным «1», будет установлен флаг отрицательного значения.

Примеры

Если в следующем примере бит CIO 0.01 включен, выполняется умножение содержимого слов [D201, D200] и [D211, D210] как 8-разрядных шестнадцатеричных чисел со знаком, результат выводится в слова [D220,...D223].



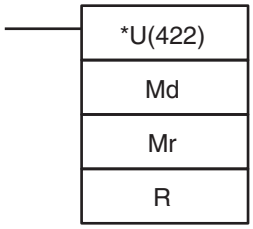
Предварительная версия
Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

3-10-19 УМНОЖИТЬ ДВОИЧНЫЕ СЛОВА БЕЗ ЗНАКА: *U(422)

Назначение

Операция перемножения 4-разрядных шестнадцатеричных чисел без знака и/или констант.

Символ РКС



Md: Слово 1–го множителя

Mr: Слово 2–го множителя

R: Слово результата

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	*U(422)
	Однократное выполнение по положительному фронту	@*U(422)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

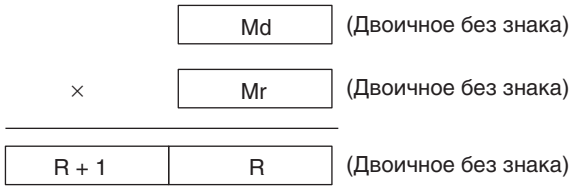
Области программных блоков	**Области пошаговых программ**	**Подпрограммы**	**Задачи обработки прерываний**
OK	OK	OK	OK

Характеристики операндов

Область	**Md**	**Mr**	**R**
Область CIO	CIO 0...CIO 6143		CIO 0...CIO 6142
Рабочая область	W0...W511		W0...W510
Область битов хранения	H0...H511		H0...H510
Область вспомогательных битов	A0...A959		A448...A958
Область таймеров	T0000...T4095		T0000...T4094
Область счетчиков	C0000...C4095		C0000...C4094
Область DM	D0...D32767		D0...D32766
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	#0000...#FFFF (двоичное) &0...&65535 (десятичное без знака)		---
Регистры данных	DR0...DR15		---
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда *U(420) выполняет умножение двоичных значений Md и Mr и выводит результат в [R, R+1].



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Выключен
Флаг равенства	=	Включен, если результат = 0. Выключен во всех остальных случаях.
Флаг отрицательного значения	N	Включен, если старший бит результата = «1». Выключен во всех остальных случаях.

Меры предосторожности

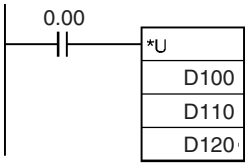
При выполнении команды *U(422) флаг ошибки выключается.

Если в результате операции умножения содержимое [R, R+1] окажется равным 0000 hex, будет установлен флаг равенства.

Если в результате операции умножения старший бит R+1 окажется равным «1», будет установлен флаг отрицательного значения.

Примеры

Если в следующем примере бит CIO 0.00 включен, выполняется умножение содержимого слов D100 и D110 как 4-разрядных шестнадцатеричных чисел без знака, результат выводится в слова [D121, D120].

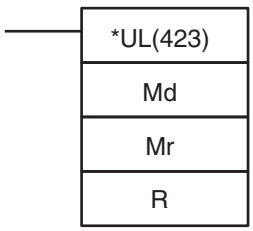


3-10-20 УМНОЖИТЬ ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА БЕЗ ЗНАКА: *UL(423)

Назначение

Операция перемножения 8-разрядных шестнадцатеричных чисел без знака и/или констант.

Символ РКС



Md: Первое слово 1–го множителя

Mr: Первое слово 2–го множителя

R: Первое слово результата

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	*UL(423)
	Однократное выполнение по положительному фронту	@*UL(423)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

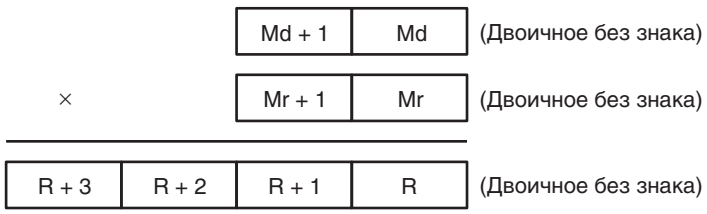
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	Md	Mr	R
Область CIO	CIO 0...CIO 6142		CIO 0...CIO 6140
Рабочая область	W0...W510		W0...W508
Область битов хранения	H0...H510		H0...H508
Область вспомогательных битов	A0...A958		A448...A956
Область таймеров	T0000...T4094		T0000...T4092
Область счетчиков	C0000...C4094		C0000...C4092
Область DM	D0...D32766		D0...D32764
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	#00000000...#FFFFFFFF (двоичное) &0...&4294967295 (десятичное без знака)		---
Регистры данных	---		
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда *UL(423) выполняет умножение [Md, Md+1] и [Mr, Mr+1] как двоичных значений без знака и выводит результат в [R,...R+3].



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Выключен
Флаг равенства	=	Включен, если результат = 0. Выключен во всех остальных случаях.
Флаг отрицательного значения	N	Включен, если старший бит результата = «1». Выключен во всех остальных случаях.

Меры предосторожности

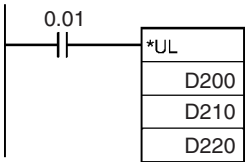
При выполнении команды *UL(423) флаг ошибки выключается.

Если в результате операции умножения содержимое [R,...R+3] окажется равным 0000 hex, будет установлен флаг равенства.

Если в результате операции умножения старший бит [R,...R+3] окажется равным «1», будет установлен флаг отрицательного значения.

Примеры

Если в следующем примере бит CIO 0.01 включен, выполняется умножение содержимого слов [D201, D200] и [D211, D210] как 8-разрядных двоичных чисел без знака, результат выводится в слова [D220,...D223].



3-10-21 УМНОЖИТЬ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА: *B(424)

Назначение

Операция перемножения 4-разрядных (одно слово) чисел и/или констант в двоично-десятичном формате.

Символ РКС



Md: Слово 1–го множителя

Mr: Слово 2–го множителя

R: Слово результата

Варианты выполнения

Варианты выполнения		
	Выполнение в каждом цикле при включенном условии выполнения	*B(424)
	Однократное выполнение по положительному фронту	@*B(424)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	Md	Mr	R
Область CIO	CIO 0...CIO 6143		CIO 0...CIO 6142
Рабочая область	W0...W511		W0...W510
Область битов хранения	H0...H511		H0...H510
Область вспомогательных битов	A0...A959		A448...A958
Область таймеров	T0000...T4095		T0000...T4094
Область счетчиков	C0000...C4095		C0000...C4094
Область DM	D0...D32767		D0...D32766
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	#0000...#9999 (BCD)		---
Регистры данных	DR0...DR15		---
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда *B(424) выполняет умножение двоично-десятичных значений Md и Mr и выводит результат в [R, R+1].



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если Md не в формате BCD. Включен, если Mr не в формате BCD. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если результат = 0. Выключен во всех остальных случаях.

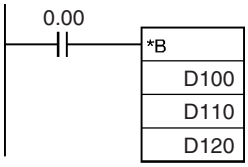
Меры предосторожности

Если Md и/или Mi не является двоично-десятичным значением, будет сгенерирована ошибка и установится флаг ошибки.

Если в результате операции умножения содержимое [R, R+1] окажется равным 0000 hex, будет установлен флаг равенства.

Примеры

Если в следующем примере бит CIO 0.00 включен, выполняется умножение содержимого слов D100 и D110 как 4-разрядных двоично-десятичных чисел, результат выводится в слова [D121, D120].



3-10-22 УМНОЖИТЬ ДВОЙНЫЕ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА: *BL(425)

Назначение

Операция перемножения 8-разрядных (двойное слово) чисел и/или констант в двоично-десятичном формате.

Символ РКС



Md: Первое слово 1–го множителя

Mr: Первое слово 2–го множителя

R: Первое слово результата

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	*BL(425)
	Однократное выполнение по положительному фронту	@*BL(425)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

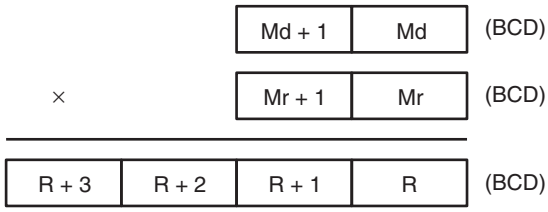
Характеристики операндов

Область	Md	Mr	R
Область CIO	CIO 0...CIO 6142		CIO 0...CIO 6140
Рабочая область	W0...W510		W0...W508
Область битов хранения	H0...H510		H0...H508
Область вспомогательных битов	A0...A958		A448...A956
Область таймеров	T0000...T4094		T0000...T4092
Область счетчиков	C0000...C4094		C0000...C4092
Область DM	D0...D32766		D0...D32764
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	#00000000...#99999999 (BCD)		---
Регистры данных	---		

Область	Md	Mr	R
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда *BL(425) выполняет умножение двоично-десятичных значений [Md, Md+1] и [Mr, Mr+1] и выводит результат в [R,...R+3].



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если Md и/или Md+1 не в формате BCD. Включен, если Mr и/или Mr +1 не в формате BCD. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если результат = 0. Выключен во всех остальных случаях.

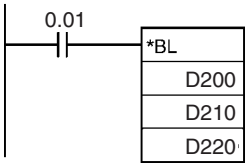
Меры предосторожности

Если [Md, Md+1] и/или [Mr, Mr+1] не является двоично-десятичным значением, будет сгенерирована ошибка и установится флаг ошибки.

Если в результате операции умножения содержимое [R, R+1, R+2, R+3] окажется равным 0000 hex, будет установлен флаг равенства.

Примеры

Если в следующем примере бит CIO 0.01 включен, выполняется умножение содержимого слов [D201, D200] и [D211, D210] как 8-разрядных чисел без знака в двоично-десятичном формате, результат выводится в слова [D220,...D223].



3-10-23 ДЕЛИТЬ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ: /(430)

Назначение

Операция деления 4-разрядных (одно слово) шестнадцатеричных чисел и/или констант со знаком.

Символ РКС



Dd: Слово делимого

Dr: Слово делителя

R: Слово результата

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	/(430)
	Однократное выполнение по положительному фронту	@/(430)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

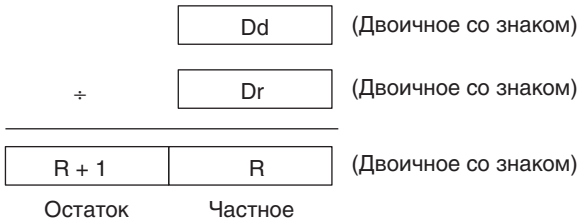
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	Dd	Dr	R
Область CIO	CIO 0...CIO 6143		CIO 0...CIO 6142
Рабочая область	W0...W511		W0...W510
Область битов хранения	H0...H511		H0...H510
Область вспомогательных битов	A0...A959		A448...A958
Область таймеров	T0000...T4095		T0000...T4094
Область счетчиков	C0000...C4095		C0000...C4094
Область DM	D0...D32767		D0...D32766
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	#0000...#FFFF (двоичное) &0...&65535 (десятичное без знака) -32768...0...32767 (десятичное со знаком)	#0001... #FFFF (двоичное) &1...&65535 (десятичное без знака) -32768..-1, 1...32767 (десятичное со знаком)	---
Регистры данных	DR0...DR15		---
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда /(430) выполняет деление двоичного значения со знаком (16 бит) Dd на двоичное значение со знаком (16 бит) Dr и выводит результат в [R, R+1]. Частное записывается в R, а остаток — в R+1.



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если результат = 0. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если в результате деления R = 0. Выключен во всех остальных случаях.
Флаг отрицательного значения	N	Включен, если старший бит результата R = «1». Выключен во всех остальных случаях.

Меры предосторожности

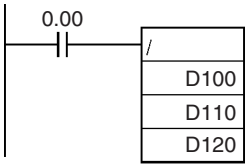
Если слово Dr = 0, будет сгенерирована ошибка и установится флаг ошибки.

Если в результате операции деления содержимое R окажется равным 0000 hex, будет установлен флаг равенства.

Если в результате операции деления старший бит R окажется равным «1», будет установлен флаг отрицательного значения.

Примеры

Если в следующем примере бит CIO 0.00 включен, выполняется деление значения слова D100 на значение слова D110 как 4-разрядных двоичных чисел со знаком, частное выводится в слово D120, остаток — в слово D121.

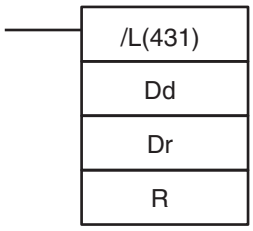


3-10-24 ДЕЛИТЬ ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ: /L(431)

Назначение

Операция деления 8-разрядных (двойное слово) шестнадцатеричных чисел и/или констант со знаком.

Символ РКС



Dd: Первое слово делимого

Dr: Первое слово делителя

R: Первое слово результата

Варианты выполнения

Варианты выполнения		/L(431)
	Выполнение в каждом цикле при включенном условии выполнения	
	Однократное выполнение по положительному фронту	@/L(431)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

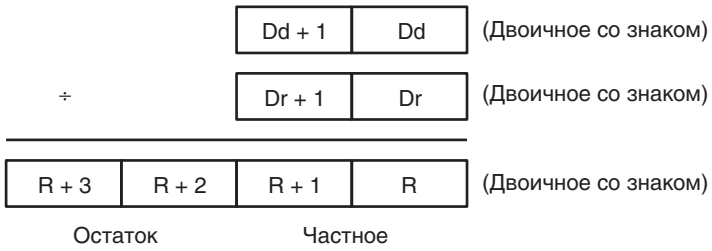
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	Dd	Dr	R
Область CIO	CIO 0...CIO 6142		CIO 0...CIO 6140
Рабочая область	W0...W510		W0...W508
Область битов хранения	H0...H510		H0...H508
Область вспомогательных битов	A0...A958		A448...A956
Область таймеров	T0000...T4094		T0000...T4092
Область счетчиков	C0000...C4094		C0000...C4092
Область DM	D0...D32766		D0...D32764
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	#00000000...#FFFFFFFF (двоичное) &0...&4294967295 (десятичное без знака) -2147483647...2147483647 (десятичное со знаком)	#00000001...#FFFFFFFF (двоичное) &1...&4294967295 (десятичное без знака) -2147483648...-1, 1...2147483647 (десятичное со знаком)	---
Регистры данных	---		
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда /L(431) выполняет деление двоичного значения со знаком [Dd, Dd+1] на двоичное значение со знаком [Dr, Dr+1] и выводит результат в [R, R+1, R+2 и R+3]. Частное выводится в [R, R+1], остаток — в [R+2, R+3].



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если результат = 0. Выключен во всех остальных случаях.

Название	Обозначение	Описание работы
Флаг равенства	=	Включен, если в результате деления [R+1, R] = 0. Выключен во всех остальных случаях.
Флаг отрицательного значения	N	Включен, если старший бит результата [R+1, R] = «1». Выключен во всех остальных случаях.

Меры предосторожности

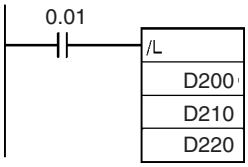
Если остаток от деления [R+3, R+2] = 0, будет установлен флаг ошибки.

Если в результате операции деления содержимое [R+1, R] окажется равным 0000 0000 hex, будет установлен флаг равенства.

Если в результате операции деления старший бит [R+1, R] окажется равным «1», будет установлен флаг отрицательного значения.

Примеры

Если в следующем примере бит CIO 0.01 включен, выполняется деление значения слов [D201, D200] на значение слов [D211, D210] как 8-разрядных шестнадцатеричных чисел со знаком, частное выводится в слова [D221, D220], остаток — в слова [D223, D222].

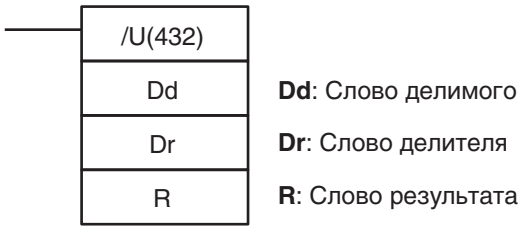


3-10-25 ДЕЛИТЬ ДВОИЧНЫЕ СЛОВА БЕЗ ЗНАКА: /U(432)

Назначение

Операция деления 4-разрядных (одно слово) шестнадцатеричных чисел и/или констант без знака.

Символ РКС



Dd: Слово делимого

Dr: Слово делителя

R: Слово результата

Варианты выполнения

Варианты выполнения		
	Выполнение в каждом цикле при включенном условии выполнения	/U(432)
	Однократное выполнение по положительному фронту	@/U(432)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

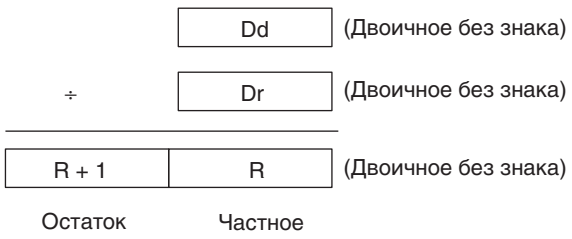
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	Dd	Dr	R
Область CIO	CIO 0...CIO 6143		CIO 0...CIO 6142
Рабочая область	W0...W511		W0...W510
Область битов хранения	H0...H511		H0...H510
Область вспомогательных битов	A0...A959		A448...A958
Область таймеров	T0000...T4095		T0000...T4094
Область счетчиков	C0000...C4095		C0000...C4094
Область DM	D0...D32767		D0...D32766
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	#0000...#FFFF (двоичное) &0...&65535 (десятичное без знака)	#0001... #FFFF (двоичное) &1...&65535 (десятичное без знака)	---
Регистры данных	DR0...15		---
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда /U(432) выполняет деление двоичного значения без знака Dd на двоичное значение без знака Dr и выводит частное в R, а остаток — в R+1.



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если результат = 0. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если в результате деления R = 0. Выключен во всех остальных случаях.
Флаг отрицательного значения	N	Включен, если старший бит результата R = «1». Выключен во всех остальных случаях.

Меры предосторожности

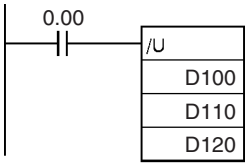
Если в результате операции деления содержимое R+1 окажется равным 0, будет установлен флаг ошибки.

Если в результате операции деления содержимое R окажется равным 0000 hex, будет установлен флаг равенства.

Если в результате операции деления старший бит R окажется равным «1», будет установлен флаг отрицательного значения.

Примеры

Если в следующем примере бит CIO 0.00 включен, выполняется деление значения слова D100 на значение слова D110 как 4-разрядных двоичных чисел без знака, частное будет выведено в слово D120, остаток — в слово D121.

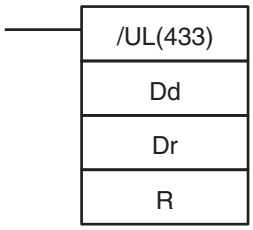


3-10-26 ДЕЛИТЬ ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА БЕЗ ЗНАКА: /UL(433)

Назначение

Операция деления 8-разрядных (двойное слово) шестнадцатеричных чисел и/или констант без знака.

Символ РКС



Dd: Первое слово делимого

Dr: Первое слово делителя

R: Первое слово результата

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	/UL(433)
	Однократное выполнение по положительному фронту	@/UL(433)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

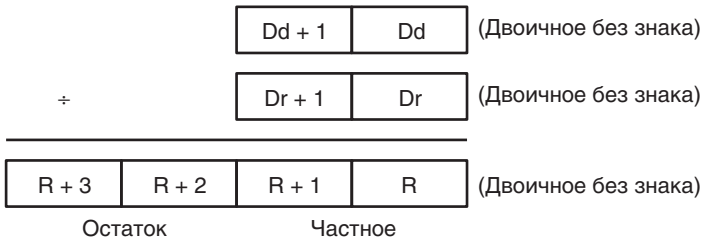
Характеристики операндов

Область	Dd	Dr	R
Область CIO	CIO 0...CIO 6142		CIO 0...CIO 6140
Рабочая область	W0...W510		W0...W508
Область битов хранения	H0...H510		H0...H508
Область вспомогательных битов	A0...A958		A448...A956
Область таймеров	T0000...T4094		T0000...T4092
Область счетчиков	C0000...C4094		C0000...C4092
Область DM	D0...D32766		D0...D32764
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		

Область	Dd	Dr	R
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	#00000000...#FFFFFFFF (двоичное) &0...&4294967295 (десятичное без знака)	#00000001...#FFFFFFFF (двоичное) &1...&4294967295 (десятичное без знака)	---
Регистры данных	---		
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда /UL(433) выполняет деление двоичного значения без знака [Dd, Dd+1] на двоичное значение без знака [Dr, Dr+1] и выводит частное в [R, R+1], а остаток — в [R+2, R+3].



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если результат = 0. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если в результате деления [R+1, R] = 0. Выключен во всех остальных случаях.
Флаг отрицательного значения	N	Включен, если старший бит результата [R+1, R] = «1». Выключен во всех остальных случаях.

Меры предосторожности

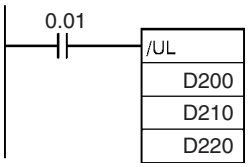
Если слова [Dr, Dr+1] содержат 0, будет включен флаг ошибки.

Если в результате операции деления содержимое [R, R+1] окажется равным 0000 hex, будет установлен флаг равенства.

Если в результате операции деления старший бит R+1 окажется равным «1», будет установлен флаг отрицательного значения.

Примеры

Если в следующем примере бит CIO 0.01 включен, выполняется деление значения слов [D201, D200] на значение слов [D211, D210] как 8-разрядных шестнадцатеричных чисел без знака, частное будет выведено в слова [D221, D220], остаток — в слова [D223, D222].

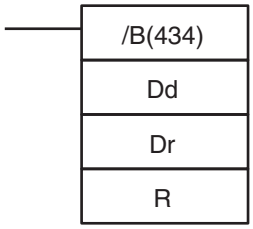


3-10-27 ДЕЛИТЬ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА: /B(434)

Назначение

Операция деления 4-разрядных (одно слово) чисел и/или констант в двоично-десятичном формате.

Символ РКС



Dd: Слово делимого

Dr: Слово делителя

R: Слово результата

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	/B(434)
	Однократное выполнение по положительному фронту	@/B(434)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

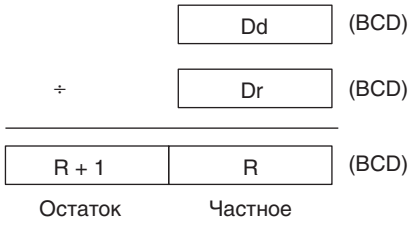
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	Dd	Dr	R
Область CIO	CIO 0...CIO 6143		CIO 0...CIO 6142
Рабочая область	W0...W511		W0...W510
Область битов хранения	H0...H511		H0...H510
Область вспомогательных битов	A0...A959		A448...A958
Область таймеров	T0000...T4095		T0000...T4094
Область счетчиков	C0000...C4095		C0000...C4094
Область DM	D0...D32767		D0...D32766
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	#0000...#9999 (BCD)	#0001... #9999 (BCD)	---
Регистры данных	DR0...DR15		---
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда /B(434) выполняет деление двоично-десятичного значения Dd на двоично-десятичное значение Dr и выводит частное в R, а остаток — в R+1.



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если Dd не в формате BCD. Включен, если Dr не в формате BCD. Включен, если остаток = 0. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если R = 0. Выключен во всех остальных случаях.

Меры предосторожности

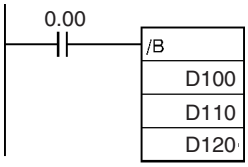
Если Dd или Dr не являются двоично-десятичными значениями или остаток (R+1) = 0, будет сгенерирована ошибка и установится флаг ошибки.

Если в результате операции деления содержимое R окажется равным 0000 hex, будет установлен флаг равенства.

Если в результате операции деления старший бит R окажется равным «1», будет установлен флаг отрицательного значения.

Примеры

Если в следующем примере бит CIO 0.00 включен, выполняется деление значения слова D100 на значение слова D110 как 4-разрядных двоично-десятичных чисел, частное будет выведено в слово D120, остаток — в слово D121.

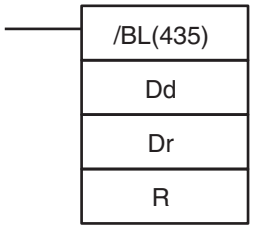


3-10-28 ДЕЛИТЬ ДВОЙНЫЕ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА: /BL(435)

Назначение

Операция деления 8-разрядных (двойное слово) чисел и/или констант в двоично-десятичном формате.

Символ РКС



Dd: Первое слово делимого

Dr: Первое слово делителя

R: Первое слово результата

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	/BL(435)
	Однократное выполнение по положительному фронту	@/BL(435)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

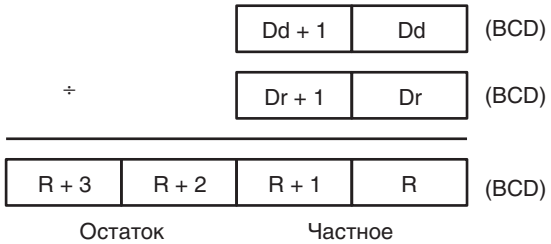
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	Dd	Dr	R
Область CIO	CIO 0...CIO 6142		CIO 0...CIO 6140
Рабочая область	W0...W510		W0...W508
Область битов хранения	H0...H510		H0...H508
Область вспомогательных битов	A0...A958		A448...A956
Область таймеров	T0000...T4094		T0000...T4092
Область счетчиков	C0000...C4094		C0000...C4092
Область DM	D0...D32766		D0...D32764
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	#00000000...#99999999 (BCD)	#00000001...#99999999 (BCD)	---
Регистры данных	---		
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда /BL(435) выполняет деление двоично-десятичного значения [Dd, Dd+1] на двоично-десятичное значение [Dr, Dr+1] и выводит частное в [R, R+1], а остаток — в [R+2, R+3].



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если [Dd, Dd+1] не в формате BCD. Включен, если Dr, Dr +1 не в формате BCD. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если результат = 0. Выключен во всех остальных случаях.

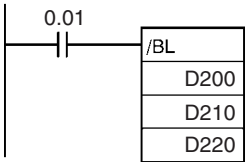
Меры предосторожности

Если [Dd, Dd+1] и/или [Dr, Dr+1] не являются двоично-десятичными значениями или содержимое [Dr, Dr+1] = 0, будет сгенерирована ошибка и установится флаг ошибки.

Если в результате операции деления содержимое [R, R+1] окажется равным 00000000 hex, будет установлен флаг равенства.

Примеры

Если в следующем примере бит CIO 0.01 включен, выполняется деление значения слов [D201, D200] на значение слов [D211, D210] как 8-разрядных двоично-десятичных чисел, частное будет выведено в слова [D221, D220], остаток — в слова [D223, D222].



3-11 Команды преобразования

В настоящем разделе описаны команды, используемые для преобразования данных.

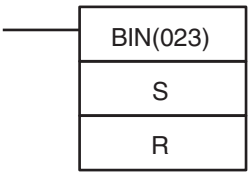
Команда	Мнемоническое обозначение	Код функции	Стр.
BCD В ДВОИЧНОЕ	BIN	023	425
ДВОЙНОЕ BCD В ДВОЙНОЕ ДВОИЧНОЕ	BINL	058	427
ДВОИЧНОЕ В BCD	BCD	024	429
ДВОЙНОЕ ДВОИЧНОЕ В ДВОЙНОЕ BCD	BCDL	059	430
ДОПОЛНЕНИЕ ДО 2	NEG	160	432
ДОПОЛНЕНИЕ ДВОЙНОГО СЛОВА ДО 2	NEGL	161	434
16-БИТОВОЕ ДВОИЧНОЕ СО ЗНАКОМ В 32-БИТОВОЕ	SIGN	600	435
ДЕКОДИРОВАНИЕ ДАННЫХ	MLPX	076	437
КОДИРОВАНИЕ ДАННЫХ	DMPX	077	442
ПРЕОБРАЗОВАНИЕ В ASCII	ASC	086	446
ASCII В HEX	HEX	162	450
СТОЛБЕЦ В СТРОКУ	LINE	063	454
СТРОКУ В СТОЛБЕЦ	COLM	064	456
BCD СО ЗНАКОМ В ДВОИЧНОЕ	BINS	470	459
ДВОЙНОЕ BCD СО ЗНАКОМ В ДВОИЧНОЕ	BISL	472	462
ДВОИЧНОЕ СО ЗНАКОМ В BCD	BCDS	471	465
ДВОЙНОЕ ДВОИЧНОЕ СО ЗНАКОМ В BCD	BDSL	473	468
ПРЕОБРАЗОВАНИЕ КОДА ГРЕЯ	GRY	474	472

3-11-1 BCD В ДВОИЧНОЕ: BIN(023)

Назначение

Преобразование двоично-десятичного значения в двоичное значение.

Символ РКС



S: Исходное слово

R: Слово результата

Варианты выполнения

Варианты выполнения		
	Выполнение в каждом цикле при включенном условии выполнения	BIN(023)
	Однократное выполнение по положительному фронту	@BIN(023)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

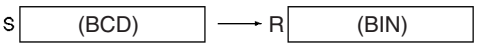
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	S	R
Область CIO	CIO 0...CIO 6143	
Рабочая область	W0...W511	
Область битов хранения	H0...H511	
Область вспомогательных битов	A0...A959	A448...A959
Область таймеров	T0000...T4095	
Область счетчиков	C0000...C4095	
Область DM	D0...D32767	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Постоянные	---	
Регистры данных	DR0...DR15	
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

Описание

Команда BIN(023) преобразует двоично-десятичное значение S в двоичное значение и записывает результат в R.

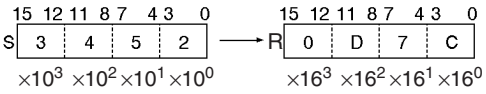


Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если содержимое S не в формате BCD. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если результат = 0000. Выключен во всех остальных случаях.
Флаг отрицательного значения	N	Выключен

Пример

Пример преобразования числа из двоично-десятичного формата в двоичный формат показан на рисунке ниже.

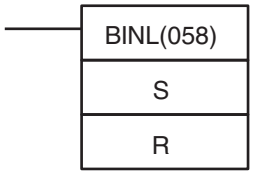


3-11-2 ДВОЙНОЕ BCD В ДВОЙНОЕ ДВОИЧНОЕ: BINL(058)

Назначение

Преобразование 8-разрядного двоично-десятичного значения в 8-разрядное шестнадцатеричное значение (32-битовое двоичное значение).

Символ РКС



S: Первое исходное слово

R: Первое слово результата

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	BINL(058)
	Однократное выполнение по положительному фронту	@BINL(058)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	S	R
Область CIO	CIO 0...CIO 6142	
Рабочая область	W0...W510	
Область битов хранения	H0...H510	
Область вспомогательных битов	A0...A958	A448...A958
Область таймеров	T0000...T4094	
Область счетчиков	C0000...C4094	
Область DM	D0...D32766	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Постоянные	---	
Регистры данных	---	
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

Описание

Команда BINL(058) преобразует 8-разрядное двоично-десятичное значение [S, S+1] в 8-разрядное шестнадцатеричное (32-битовое двоичное) значение и записывает результат в [R, R+1].

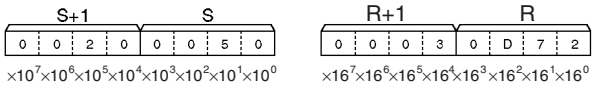


Флаги

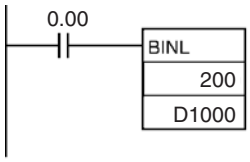
Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если содержимое [S+1, S] не в формате BCD. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если результат = 0. Выключен во всех остальных случаях.
Флаг отрицательного значения	N	Выключен

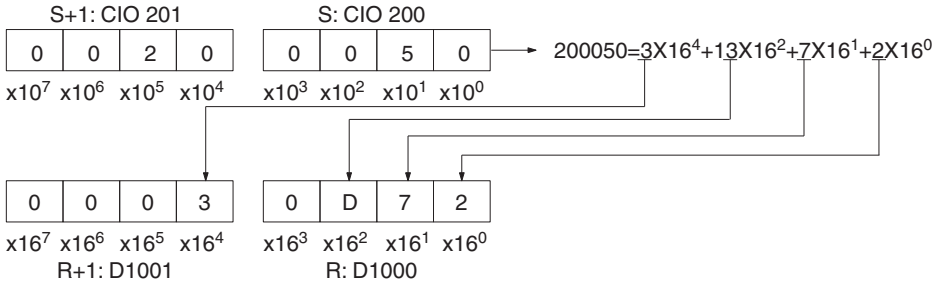
Примеры

Пример преобразования 8-разрядного числа из двоично-десятичного формата в двоичный формат показан на рисунке ниже.



Если в следующем примере бит CIO 0.00 включен, выполняется преобразование 8-разрядного двоично-десятичного значения слов [CIO 201, CIO 200] в шестнадцатеричное значение, результат записывается в слова [D1001, D1000].



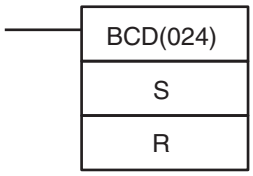


3-11-3 ДВОИЧНОЕ В BCD: BCD(024)

Назначение

Преобразование слова двоичных данных в слово данных в двоично-десятичном формате.

Символ РКС



S: Исходное слово

R: Слово результата

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	BCD(024)
	Однократное выполнение по положительному фронту	@BCD(024)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

S: Исходное слово

Значение S должно принадлежать диапазону 0000...270F hex (0000...9999 десятичн.).

Характеристики операндов

Область	S	R
Область CIO	CIO 0...CIO 6143	
Рабочая область	W0...W511	
Область битов хранения	H0...H511	
Область вспомогательных битов	A0...A959	A448...A959
Область таймеров	T0000...T4095	
Область счетчиков	C0000...C4095	
Область DM	D0...D32767	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Постоянные	---	
Регистры данных	DR0...DR15	
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

Описание

Команда BCD(024) преобразует двоичное значение S в двоично-десятичное значение и записывает результат в R.



Флаги

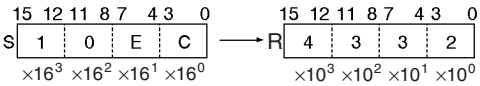
Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если значение S больше 270F (9999 десятичн.) Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если результат = 0000. Выключен во всех остальных случаях.

Меры предосторожности

Значение S не должно превышать 270F (9999 десятичн.).

Пример

Пример преобразования числа из двоичного формата в двоично-десятичный формат показан на рисунке ниже.

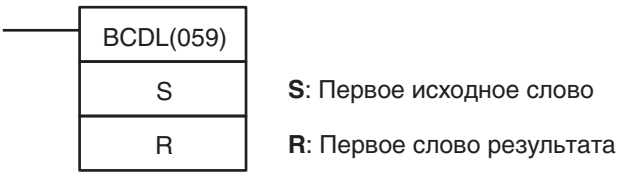


3-11-4 ДВОЙНОЕ ДВОИЧНОЕ В ДВОЙНОЕ BCD: BCDL(059)

Назначение

Преобразование 8-разрядного шестнадцатеричного значения (32-битового двоичного значения) в 8-разрядное двоично-десятичное значение.

Символ РКС



Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	BCDL(059)
	Однократное выполнение по положительному фронту	@BCDL(059)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

S: Первое исходное слово

Значение двойного слова [S+1, S] должно принадлежать диапазону 0000 0000...05F5 E0FF hex (0000 0000...9999 9999 десятичн.).

Характеристики операндов

Область	S	R
Область CIO	CIO 0...CIO 6142	
Рабочая область	W0...W510	
Область битов хранения	H0...H510	
Область вспомогательных битов	A0...A958	A448...A958
Область таймеров	T0000...T4094	
Область счетчиков	C0000...C4094	
Область DM	D0...D32766	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Постоянные	---	
Регистры данных	---	
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

Описание

Команда BCDL(059) преобразует 8-разрядное шестнадцатеричное (32-битовое двоичное) значение [S, S+1] в 8-разрядное двоично-десятичное значение и записывает результат в [R, R+1].



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если значение [S, S+1] больше 05F5 E0FF (9999 9999 десятичн.). Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если результат = 0. Выключен во всех остальных случаях.

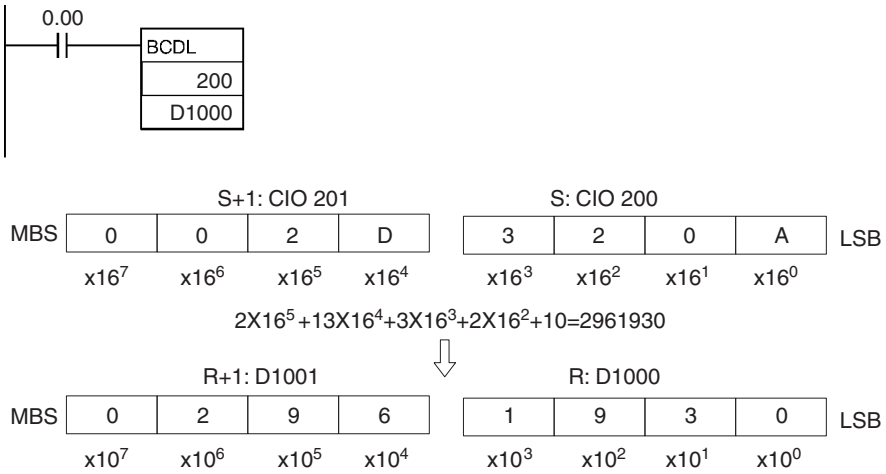
Меры предосторожности

Значение [S+1, S] не должно превышать 05F5 E0FF (9999 9999 десятичн.).

Примеры

Пример преобразования 8-разрядного числа из двоичного формата в двоично-десятичный формат показан на рисунке ниже.

Если в следующем примере бит CIO 0.00 включен, выполняется преобразование шестнадцатеричного значения двойного слова [CIO 201, CIO 200] в значение в двоично-десятичном формате, результат записывается в слова [D1001, D1000].

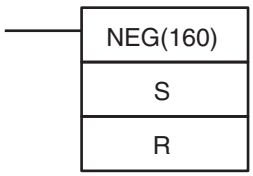


3-11-5 ДОПОЛНЕНИЕ ДО 2: NEG(160)

Назначение

Операция дополнения до 2 слова шестнадцатеричных данных.

Символ РКС



S: Исходное слово

R: Слово результата

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	NEG(160)
	Однократное выполнение по положительному фронту	@NEG(160)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограм-мы	Задачи обработки прерываний
OK	OK	OK	OK

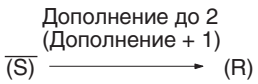
Характеристики операндов

Область	S	R
Область CIO	CIO 0...CIO 6143	
Рабочая область	W0...W511	
Область битов хранения	H0...H511	
Область вспомогательных битов	A0...A959	A448...A959

Область	S	R
Область таймеров	T0000...T4095	
Область счетчиков	C0000...C4095	
Область DM	D0...D32767	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Постоянные	#0000...#FFFF (двоичн.)	---
Регистры данных	DR0...DR15	
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

Описание

Команда NEG(160) рассчитывает дополнение до двух значения S и записывает результат в R. Операция дополнения до 2 состоит в инверсии состояний битов слова S и добавлении 1.



Примечание. Эта операция (инвертирование состояния битов и добавление 1) эквивалентна вычитанию значения S из 0000.

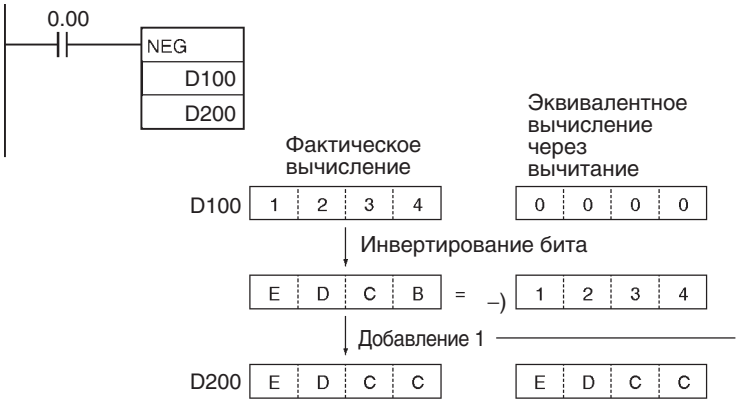
Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Выключен
Флаг равенства	=	Включен, если результат = 0000. Выключен во всех остальных случаях.
Флаг отрицательного значения	N	Включен, если бит 15 результата = «1». Выключен во всех остальных случаях.

Примечание. Для 8000 hex будет получен результат 8000 hex.

Пример

Если в следующем примере бит CIO 0.00 включен, команда NEG(160) вычисляет дополнение до 2 значения слова D100 и записывает результат в слово D200.

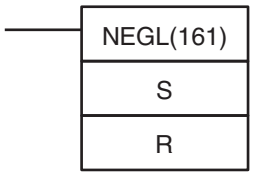


3-11-6 ДОПОЛНЕНИЕ ДВОЙНОГО СЛОВА ДО 2: NEGL(161)

Назначение

Операция дополнения до 2 двойного слова шестнадцатеричных данных.

Символ РКС



S: Первое исходное слово

R: Первое слово результата

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	NEGL(161)
	Однократное выполнение по положительному фронту	@NEGL(161)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	S	R
Область CIO	CIO 0...CIO 6142	
Рабочая область	W0...W510	
Область битов хранения	H0...H510	
Область вспомогательных битов	A0...A958	A448...A958
Область таймеров	T0000...T4094	
Область счетчиков	C0000...C4094	
Область DM	D0...D32766	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Постоянные	#00000000...#FFFFFFFF (двоичное)	---
Регистры данных	---	
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

Описание

Команда NEGL(161) рассчитывает дополнение до двух значения [S+1, S] и записывает результат в [R+1, R]. Операция дополнения до 2 состоит в инверсии состояний битов значения [S+1, S] и добавлении 1.

Дополнение до 2
(Дополнение + 1)

$\overline{(S+1, S)} \longrightarrow (R+1, R)$

Примечание. Эта операция (инвертирование состояния битов и добавление 1) эквивалентна вычитанию значения [S+1, S] из 0000 0000.

Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Выключен
Флаг равенства	=	Включен, если результат = 0000 0000. Выключен во всех остальных случаях.
Флаг отрицательного значения	N	Включен, если бит 15 в R+1 = «1». Выключен во всех остальных случаях.

Примечание. Для 8000 hex будет получен результат 8000 hex.

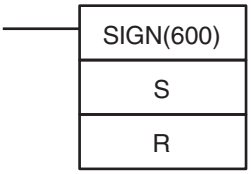
Пример Если в следующем примере бит CIO 0.01 включен, команда NEGL(161) вычисляет дополнение до 2 значения двойного слова [D1001, D1000] и записывает результат в слова [D2001, D2000].



3-11-7 16-БИТОВОЕ ДВОИЧНОЕ СО ЗНАКОМ В 32-БИТОВОЕ: SIGN(600)

Назначение Преобразование 16-битового двоичного значения со знаком в эквивалентное 32-битовое значение.

Символ РКС



S: Исходное слово

R: Первое слово результата

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	SIGN(600)
	Однократное выполнение по положительному фронту	@SIGN(600)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

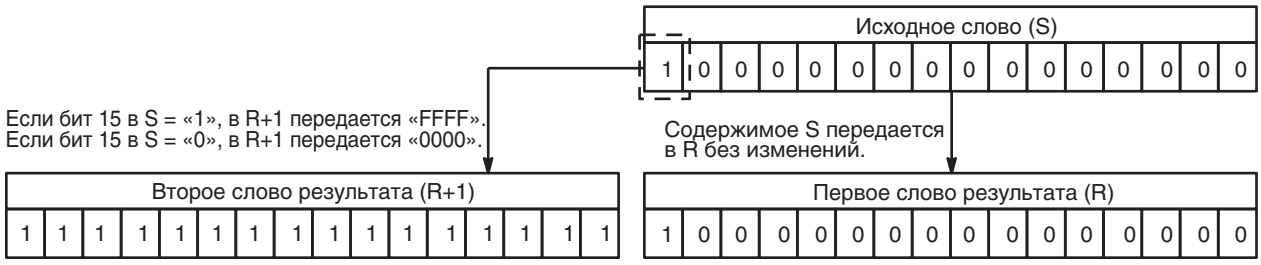
Область	S	R
Область CIO	CIO 0...CIO 6143	CIO 0...CIO 6142
Рабочая область	W0...W511	W0...W510
Область битов хранения	H0...H511	H0...H510
Область вспомогательных битов	A0...A959	A448...A958
Область таймеров	T0000...T4095	T0000...T4094
Область счетчиков	C0000...C4095	C0000...C4094
Область DM	D0...D32767	D0...D32766
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Постоянные	#0000...#FFFF (двоичн.)	---
Регистры данных	DR0...DR15	---
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

Примечание. Слова R и R+1 должны принадлежать одной области.

Описание

Команда SIGN(600) преобразует 16-битовое двоичное значение со знаком S в эквивалентное ему 32-битовое двоичное значение со знаком и записывает результат в [R+1, R].

Преобразование осуществляется путем копирования значения S в R и записи в R+1 значения «FFFF» (если бит 15 слова S содержит «1») или значения «0000» (если бит 15 слова S содержит «0»).

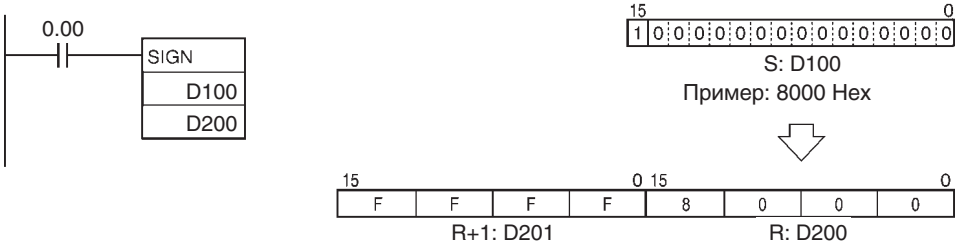


Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Выключен
Флаг равенства	=	Включен, если результат = 0000 0000. Выключен во всех остальных случаях.
Флаг отрицательного значения	N	Включен, если бит 15 в R+1 = «1». Выключен во всех остальных случаях.

Пример

Если в следующем примере бит CIO 0.00 включен, команда SIGN(600) преобразует 16-битовое двоичное значение со знаком слова D100 (8000 hex = -32768 десятичн.) в эквивалентное ему 32-битовое значение (FFFF 8000 hex = -32768 десятичн.) и записывает результат в слова [D201, D200].

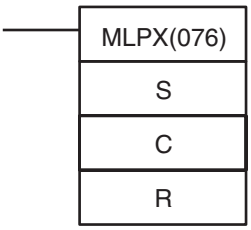


3-11-8 ДЕКОДИРОВАНИЕ ДАННЫХ: MLPX(076)

Назначение

Чтение числового значения в указанном разряде (или байте) входного слова, установка соответствующего бита в слове результата (или в матрице из 16 слов) и сброс всех остальных битов в слове результата (или в матрице из 16-ти слов).

Символ РКС



S: Исходное слово

C: Управляющее слово

R: Первое слово результата

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	MLPX(076)
	Однократное выполнение по положительному фронту	@MLPX(076)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

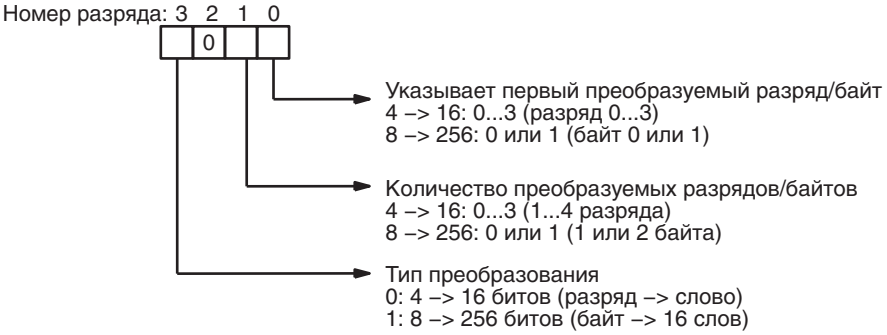
Операнды

S: Исходное слово

Значение исходного слова указывает положение бита или битов, которые должны быть установлены.

C: Управляющее слово

Управляющее слово указывает тип преобразования, выполняемого командой MLPX(076): преобразование 4-битового значения в 16-битовое или 8-битового значения в 256-битовое, количество преобразуемых разрядов или байтов и начальный разряд или байт.



R: Первое слово результата

Первым словом может быть любое из 32 слов результата (нумерация начинается с 1), в зависимости от типа преобразования и количества преобразуемых разрядов/байтов. Слова результата должны принадлежать одной области.

Характеристики операндов

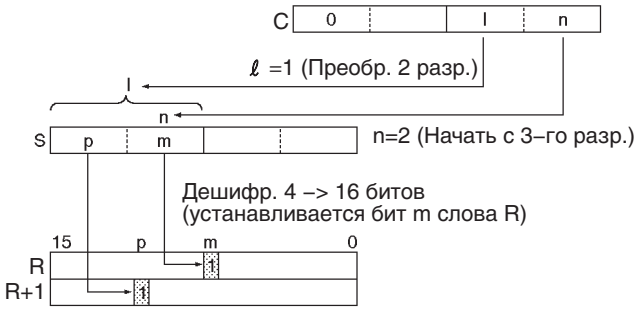
Область	S	C	R
Область CIO	CIO 0...CIO 6143		
Рабочая область	W0...W511		
Область битов хранения	H0...H511		
Область вспомогательных битов	A0...A959		A448...A959
Область таймеров	T0000...T4095		
Область счетчиков	C0000...C4095		
Область DM	D0...D32767		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	---	---	---
Регистры данных	DR0...DR15		---
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда MLPX(076) преобразует 4-битовое значение в 16-битовое или преобразует 8-битовое значение в 256-битовое. Чтобы указать преобразование 4->16 бит, введите «0» в старший разряд слова C. Чтобы указать преобразование 8->256 бит, введите «1» в старший разряд слова C.

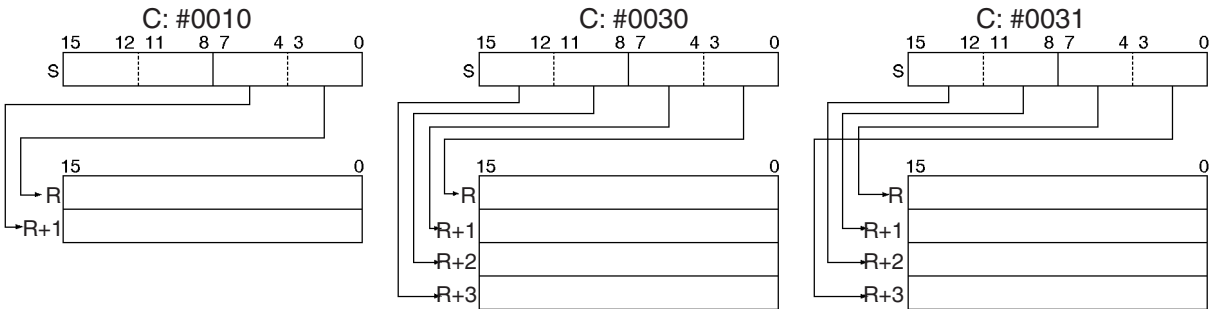
Преобразование из 4-битового в 16-битовое значение (4->16 бит)

Если старший разряд C равен «0», команда MLPX(076) считывает значение указанного разряда S (0...F) и устанавливает соответствующий бит в слове результата. Все прочие биты слова результата будут сброшены. Можно преобразовать до четырех разрядов.



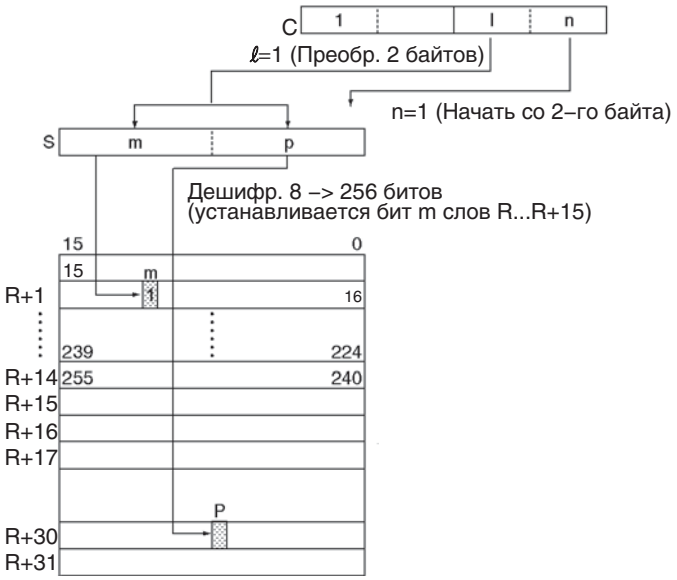
При преобразовании двух или большего числа разрядов команда MLPX(076) считывает разряды слова S справа налево (от младшего к старшему), а в случае необходимости возвращается к младшему разряду после считывания старшего разряда.

На следующем рисунке показано выполнение преобразования 4->16 бит для различных значений слова C.



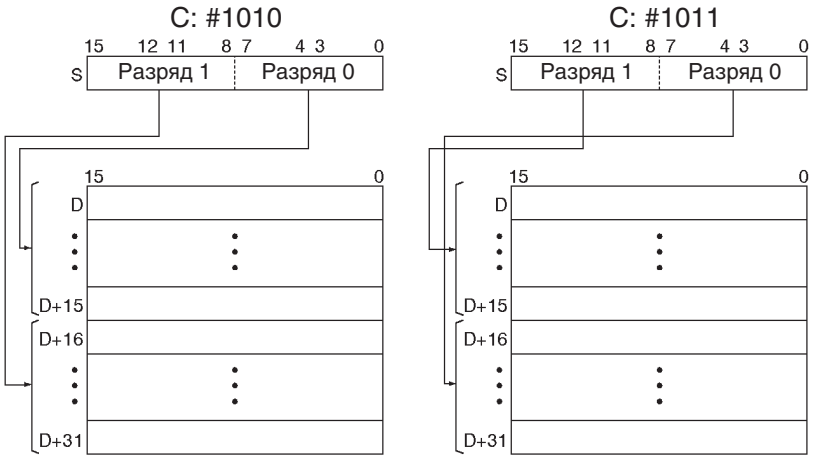
Преобразование из 8-битового в 256-битовое значение

Если старший разряд C равен «1», команда MLPX(076) считывает значение указанного байта слова S (00...FF) и устанавливает соответствующий бит в одном из 16 слов результата. Все прочие биты слова результата будут сброшены. Можно преобразовать до четырех байтов.



При преобразовании двух байтов команда MLPX(076) считывает байты слова S справа налево (от младшего к старшему), а при необходимости возвращается к младшему байту, если в качестве начального байта указан старший байт (байт 1).

На следующем рисунке показано выполнение преобразования 8->256 бит для различных значений слова C.



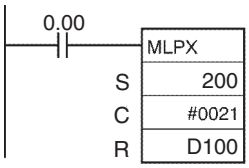
Флаги

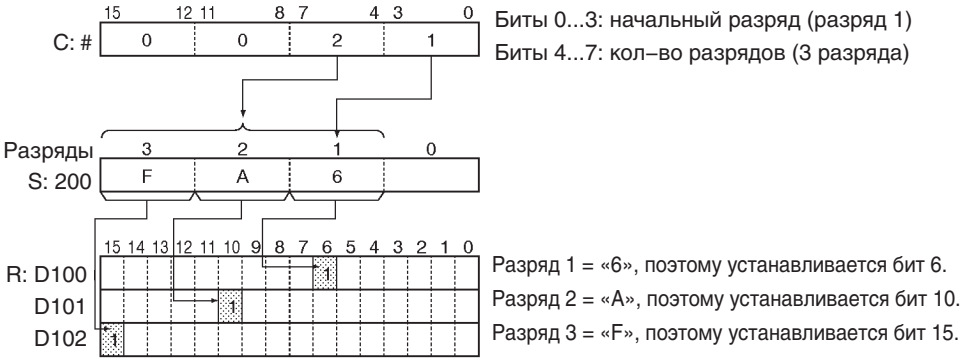
Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если C выходит за указанные диапазоны. Выключен во всех остальных случаях.

Примеры

Преобразование из 4-битового в 16-битовое значение (4->16 бит)

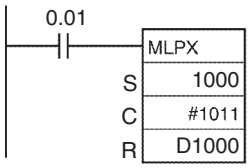
Если в следующем примере бит CIO 0.00 включен, команда MLPX(076) преобразует 3 разряда слова CIO 200, начиная со второго разряда, в соответствии со значением слова C (#0021). Будут установлены соответствующие биты слов D100...D102.

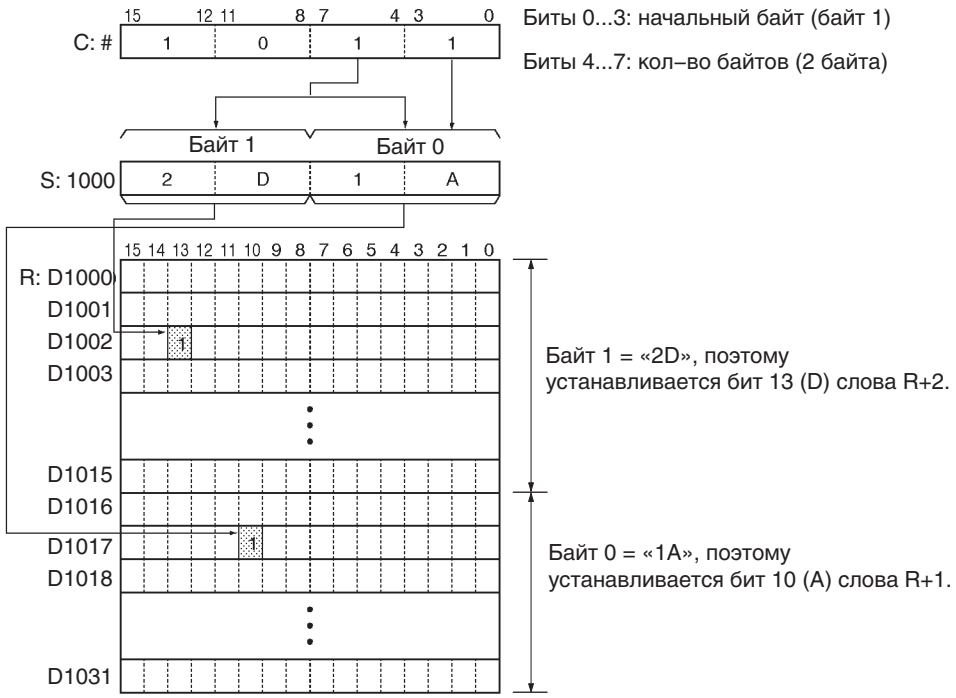




Преобразование из 8-битового в 256-битовое значение (8->256 бит)

Если в следующем примере бит CIO 0.01 включен, команда MLPX(076) преобразует 2 байта слова S, начиная с первого байта (старший байт), в соответствии со значением слова C (#1011). Будут установлены соответствующие биты слов D1000...D1015 и D1016...D1031 .



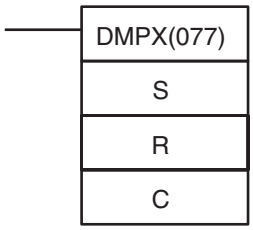


3-11-9 КОДИРОВАНИЕ ДАННЫХ: DMPX(077)

Назначение

Определение позиции первого или последнего включенного бита внутри входного слова (или матрицы из 16 слов) и запись номера позиции в указанный разряд (или байт) слова результата.

Символ РКС



S: Первое исходное слово

R: Слово результата

C: Управляющее слово

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	DMPX(077)
	Однократное выполнение по положительному фронту	@DMPX(077)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

S: Первое исходное слово

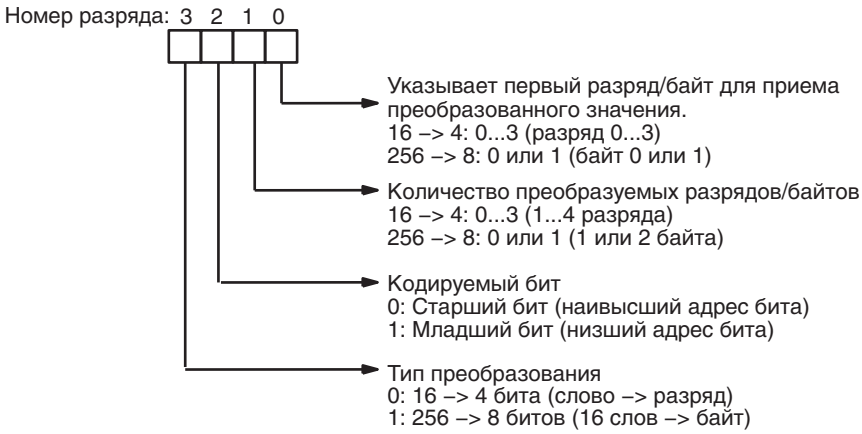
Первым словом может быть любое из 32 исходных слов (нумерация начинается с 1), в зависимости от типа преобразования и количества преобразуемых разрядов/байтов. Исходные слова должны принадлежать одной области.

R: Слово результата

Положения включенных битов в исходных словах записываются в разряды/байты слова R, начиная с указанного начального разряда/байта.

C: Управляющее слово

Управляющее слово указывает: тип преобразования, выполняемого командой DMPX(077) (преобразование 4-битового значения в 16-битовое или 8-битового значения в 256-битовое); какой установленный бит кодируется (старший или младший); количество преобразуемых разрядов или байтов; начальный разряд или байт диапазона, в который записывается результат.



Характеристики операндов

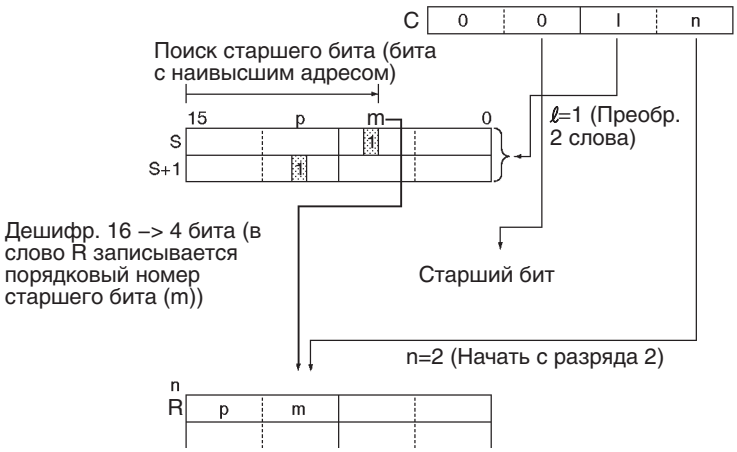
Область	S	R	C
Область CIO	CIO 0...CIO 6143		
Рабочая область	W0...W511		
Область битов хранения	H0...H511		
Область вспомогательных битов	A0...A959	A448...A959	A0...A959
Область таймеров	T0000...T4095		
Область счетчиков	C0000...C4095		
Область DM	D0...D32767		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	---	---	Только указанные значения
Регистры данных	---	DR0...DR15	
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда DMPX(077) преобразует 16-битовое значение в 4-битовое или преобразует 256-битовое значение в 8-битовое. Чтобы указать преобразование 16->4 бита, введите «0» в старший разряд слова C. Чтобы указать преобразование 256->8 битов, введите «1» в старший разряд слова C.

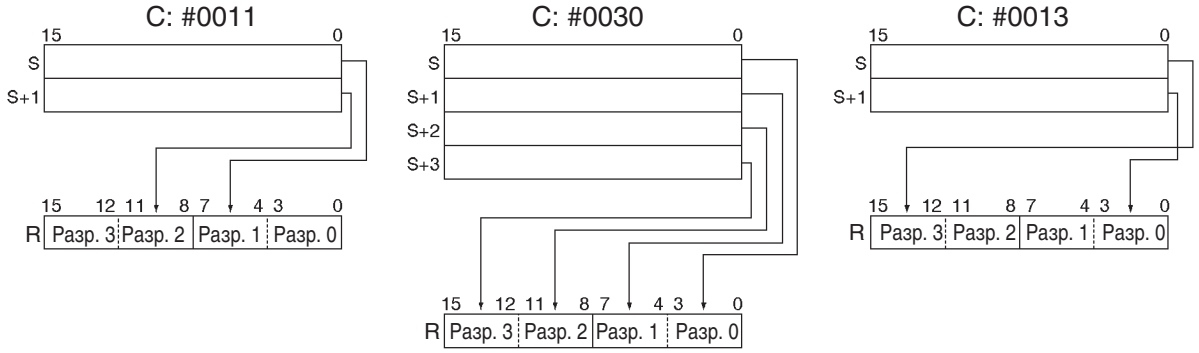
Преобразование из 16-битового в 4-битовое значение

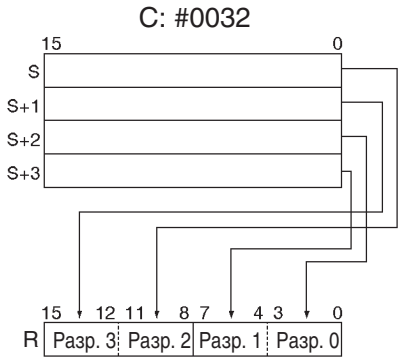
Если четвертый (старший разряд) слова C равен «0», команда DMPX(077) определяет положение старшего или младшего включенных битов максимум в 4 исходных словах и записывает их позиции в R, начиная с указанного разряда. (Для поиска самого старшего установленного бита введите «0» в третий разряд C. Для поиска самого младшего установленного бита введите «1» в третий разряд C.)



При преобразовании двух или большего числа разрядов команда DMPX(077) считывает разряды R справа налево (от младшего к старшему), а в случае необходимости возвращается к младшему разряду после считывания старшего разряда.

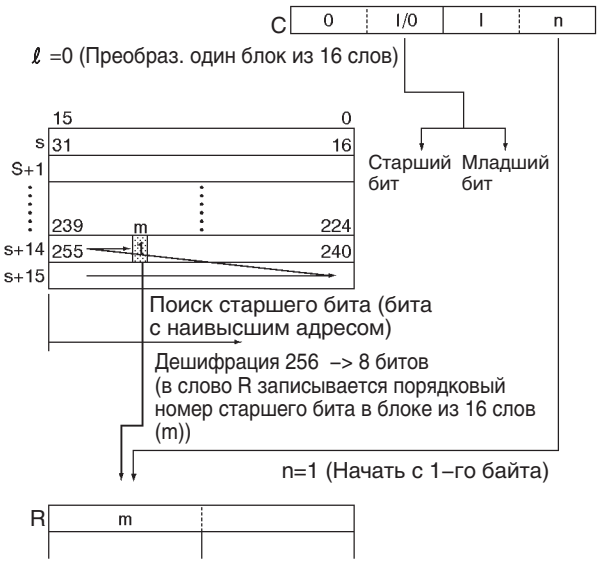
На следующем рисунке показано выполнение преобразования 16->4 бит для различных значений слова C.





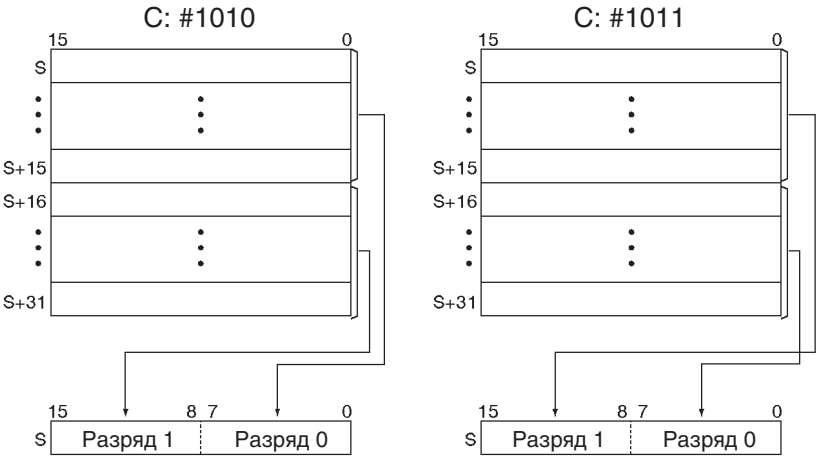
Преобразование из 256-битового в 8-битовое значение

Если четвертый (старший) разряд слова C равен «1», команда DMPX(077) определяет положение старших (с наивысшим адресом) или младших (с низшим адресом) включенных битов в одной или двух группах из 16 исходных слов. Номера позиций этих битов записываются в R, начиная с указанного байта. (Для поиска самого старшего установленного бита введите «0» в третий разряд C. Для поиска самого младшего установленного бита введите «1» в третий разряд C.)



При преобразовании двух байтов команда DMPX(077) записывает значения в байты R справа налево, а при необходимости возвращается к младшему байту, если в качестве начального байта указан старший байт (байт 1).

На следующем рисунке показано выполнение преобразования 256->8 бит для различных значений слова C.



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если любое из исходных слов содержит значение 0000 hex (т. е. отсутствует бит, который нужно закодировать). Включен, если C выходит за указанные диапазоны. Выключен во всех остальных случаях.

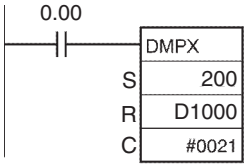
Меры предосторожности

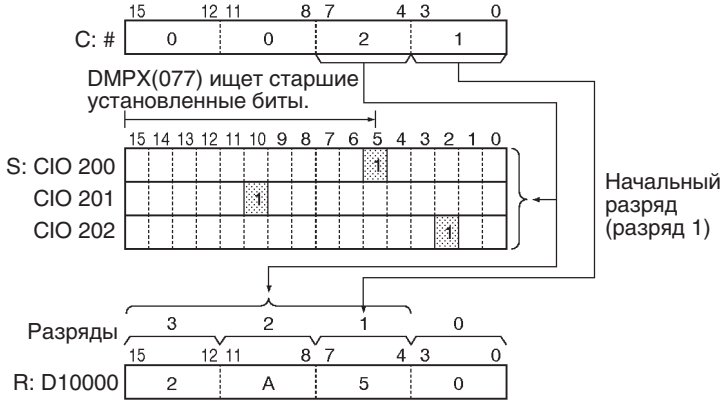
Если преобразуемые данные содержат 0000 hex, но имеются другие данные, которые нужно закодировать, разбейте преобразуемые данные на части и примените команду DMPX(077) несколько раз.

```
DMPX(077)  D0    D100  #0300
DMPX(077)  D0    D100  #0000
DMPX(077)  D1    D100  #0001
DMPX(077)  D2    D100  #0002
DMPX(077)  D3    D100  #0003
```

Примеры

Если в следующем примере бит CIO 0.00 включен, команда DMPX(077) находит старшие включенные биты слов CIO 200...C202 и записывает номера позиций этих битов в три разряда слова R, начиная со второго разряда, в соответствии со значением слова C (#0021).



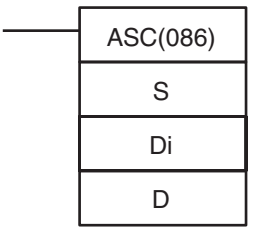


3-11-10 ПРЕОБРАЗОВАНИЕ В ASCII: ASC(086)

Назначение

Преобразование 4-битовых шестнадцатеричных разрядов входного слова в эквивалентные 8-битовые коды ASCII.

Символ РКС



S: Исходное слово

Di: Указатель разрядов

D: Первое слово назначения

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	ASC(086)
	Однократное выполнение по положительному фронту	@ASC(086)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

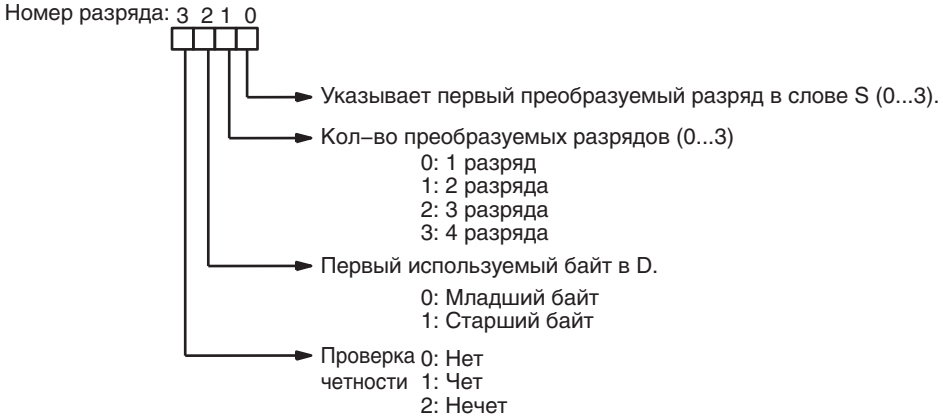
Операнды

S: Исходное слово

Можно преобразовать до 4 разрядов исходного слова. Разряды нумеруются от 0 до 3, справа налево.

Di: Указатель разрядов

Указатель разрядов содержит ряд параметров преобразования. Его содержание показано на следующем рисунке.



D: Первое слово назначения

Выходные коды ASCII записываются в слово или слова назначения, начиная с указанного байта D. Если выполняется преобразование четырех разрядов и в качестве первого байта D указан самый старший байт, для записи результата потребуются три слова (D...D+3). Все слова назначения должны принадлежать одной и той же области данных.

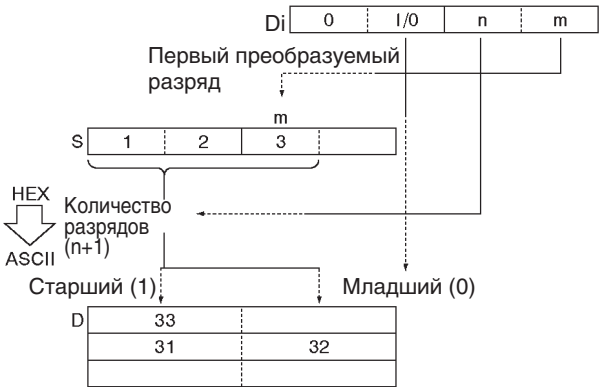
Остальные байты слова или слов назначения, в которые не были записаны данные ASCII, остаются неизменными.

Характеристики операндов

Область	S	Di	D
Область CIO	CIO 0...CIO 6143		
Рабочая область	W0...W511		
Область битов хранения	H0...H511		
Область вспомогательных битов	A0...A959		A448...A959
Область таймеров	T0000...T4095		
Область счетчиков	C0000...C4095		
Область DM	D0...D32767		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	---	---	---
Регистры данных	DR0...DR15		---
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда ASC(086) воспринимает значение S как четыре шестнадцатеричных разряда, преобразует указанный разряд или разряды S в 8-битовые коды эквивалентных ASCII-символов и записывает эти коды в слово или слова назначения, начиная с указанного байта слова D.



Проверка четности

Для кодов ASCII можно выбрать проверку на четность или нечетность с целью контроля ошибок при передаче данных. Старший бит каждого кода ASCII будет автоматически скорректирован, в зависимости от выбранного типа проверки: проверка четности, проверка нечетности или без проверки.

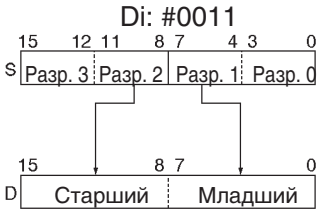
Если выбран тип «без проверки» (0), старший бит всегда будет равен «0». Если выбрана проверка четности (1), старший бит изменяется так, чтобы общее количество включенных битов было четным. Если выбрана проверка нечетности (2), старший бит изменяется так, чтобы общее количество включенных битов было нечетным. Состояние бита проверки четности не влияет на значение кода ASCII.

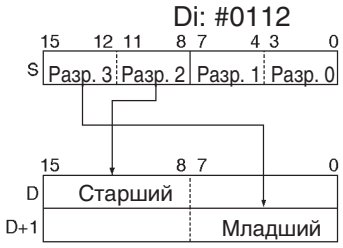
Примеры проверки на четность:
С поправкой на проверку четности эквивалентом «31» (00110001) в ASCII будет «B1» (10110001: бит проверки на четность установлен, чтобы общее количество включенных битов было четным); эквивалентом «36» (00110110) в ASCII будет «36» (00110110: бит проверки на четность остается выключенным, так как общее количество включенных битов четное).

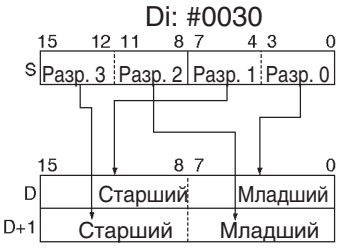
Примеры проверки на нечетность:
С поправкой на проверку нечетности эквивалентом «36» (00110110) в ASCII будет «B6» (10110110: бит проверки на четность установлен, чтобы общее количество включенных битов было нечетным); эквивалентом «46» (01000110) в ASCII будет «46» (01000110: бит проверки на четность остается выключенным, так как общее количество включенных битов нечетное).

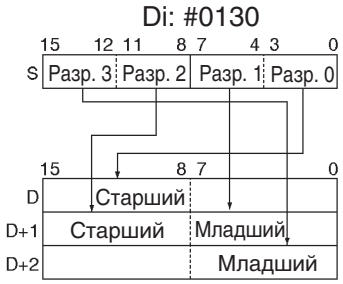
Примеры для различных значений Di

При преобразовании двух или большего числа разрядов команда ASC(086) считывает байты слова S справа налево, а в случае необходимости возвращается к младшему байту после считывания старшего байта. На следующем рисунке показаны примеры преобразования для различных значений слова Di.







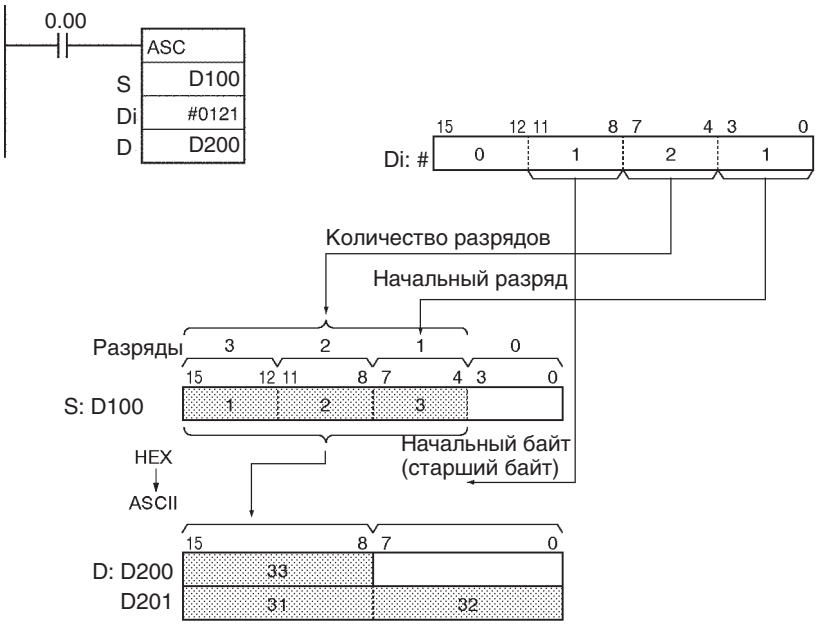


Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если содержимое Di не находится в пределах указанных диапазонов. Выключен во всех остальных случаях.

Пример

Если в следующем примере бит CIO 0.00 включен, команда ASC(086) преобразует три разряда шестнадцатеричного слова D100 (начиная с разряда 1) в эквивалентные коды ASCII и записывает эти коды в слова D200 и D201, начиная со старшего байта слова D200. В этом случае указатель разрядов #0121 указывает на то, что проверка четности не выполняется, начальным байтом для записи является старший байт, считываются 3 байта, начальным разрядом для чтения является разряд 1.

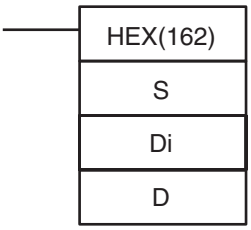


3-11-11 ASCII В HEX: HEX(162)

Назначение

Преобразование максимум 4 байтов исходного слова, содержащих коды ASCII, в эквивалентные шестнадцатеричные значения и запись этих значений в соответствующие разряды указанного слова назначения.

Символ РКС



S: Первое исходное слово

Di: Указатель разрядов

D: Адресуемое слово

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	HEX(162)
	Однократное выполнение по положительному фронту	@HEX(162)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

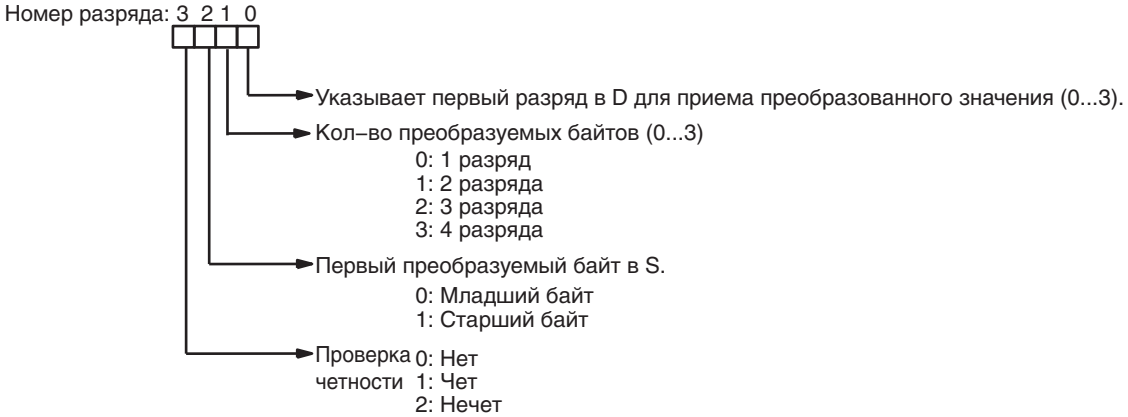
Операнды

S: Первое исходное слово

Содержимое исходных слов воспринимается как коды ASCII. Можно использовать до трех исходных слов. (Три исходных слова потребуются, если выполняется преобразование 4 байтов и в качестве начального байта S выбран самый старший байт.) Исходные слова должны принадлежать одной области.

Di: Указатель разрядов

Указатель разрядов содержит ряд параметров преобразования. Его содержание показано на следующем рисунке.



D: Слово назначения

Полученные значения шестнадцатеричных разрядов записываются в слово D справа налево, начиная с указанного начального разряда. Остальные разряды слова назначения, в которые не были записаны результаты преобразования, остаются неизменными.

Характеристики операндов

Область	S	Di	D
Область CIO	CIO 0...CIO 6143		
Рабочая область	W0...W511		
Область битов хранения	H0...H511		
Область вспомогательных битов	A0...A959		A448...A959
Область таймеров	T0000...T4095		
Область счетчиков	C0000...C4095		
Область DM	D0...D32767		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	---	Только указанные значения	---
Регистры данных	---	DR0...DR15	---
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда HEX(162) воспринимает содержимое байтов исходного слова или слов как коды ASCII-символов, соответствующих шестнадцатеричным цифрам (0...9 и A...F), преобразует указанное количество байтов в эквивалентные шестнадцатеричные значения и записывает эти значения в слово назначения, начиная с указанного разряда.

Если какой-либо из байтов исходного слова (слов) содержит значение, не являющееся ASCII-кодом цифры или буквы шестнадцатеричной системы счисления, будет сгенерирована ошибка. В следующей таблице показаны шестнадцатеричные цифры и соответствующие им ASCII-коды (без битов проверки четности).

Флаги

Шестнадцатеричные цифры (4 бита)	Эквивалентный код ASCII (2 шестнадц. разряда)
0...9	30...39
A...F	41...46

На следующей схеме показано выполнение команды HEX(162) при Di=0021.



Проверка четности

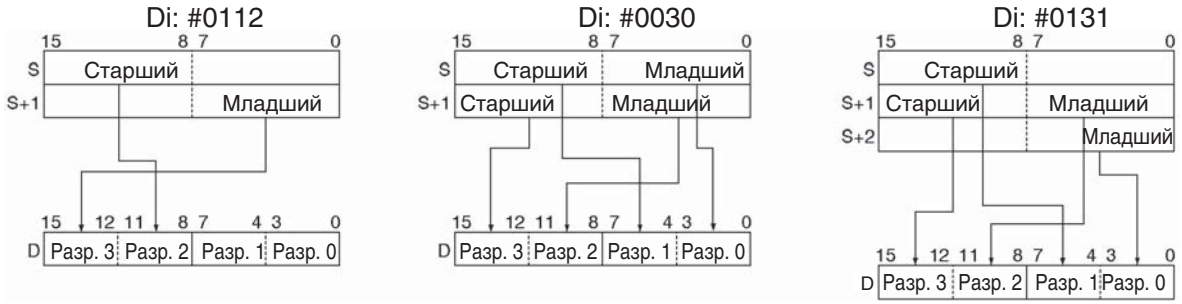
Для кодов ASCII можно выбрать проверку на четность или нечетность с целью контроля ошибок при передаче данных. Старший бит каждого байта служит битом проверки четности. Если выбран тип «без проверки», бит проверки четности всегда должен имеет значение «0». Если выбрана проверка четности, состояние бита проверки четности должно быть таким, чтобы общее количество включенных битов было четным. Если выбрана проверка нечетности, состояние бита проверки четности должно быть таким, чтобы общее количество включенных битов было нечетным.

В следующей таблице описана работа команды HEX(162) при разных типах проверки четности.

Значение параметра проверки четности (старший разряд Di)	Работа команды HEX(162)
Без проверки (0)	Команда HEX(162) выполняется только в том случае, если бит проверки четности каждого байта равен «0». Если бит проверки четности не равен «0», будет сгенерирована ошибка.
Проверка четности (1)	Команда HEX(162) выполняется только в том случае, если количество включенных битов в каждом байте четное. Если какой-либо байт содержит нечетное количество включенных битов, будет сгенерирована ошибка.
Проверка нечетности (2)	Команда HEX(162) выполняется только в том случае, если количество включенных битов в каждом байте нечетное. Если какой-либо байт содержит четное количество включенных битов, будет сгенерирована ошибка.

Примеры для различных значений Di

При преобразовании двух или большего числа байтов команда HEX(162) записывает преобразованные разряды в слово назначения справа налево, а в случае необходимости возвращается к младшему разряду после считывания старшего разряда. На следующем рисунке показаны примеры преобразования для различных значений слова Di.



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен при ошибке проверки четности в ASCII-данных. Включен, если ASCII-данные исходных слов не соответствуют никаким шестнадцатеричным цифрам. Включен, если содержимое Di не находится в пределах указанных диапазонов. Выключен во всех остальных случаях.

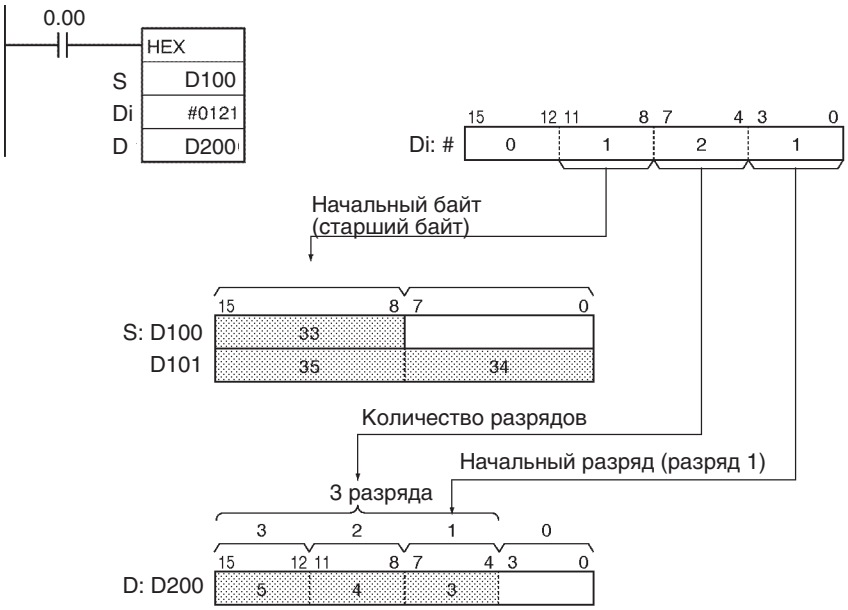
Меры предосторожности

Если в ASCII-данных имеется ошибка проверки четности, или ASCII-данные исходных слов не соответствуют никаким шестнадцатеричным цифрам или содержимое Di выходит за пределы указанного диапазона: возникнет ошибка и будет установлен флаг ошибки.

Примеры

Если в следующем примере бит CIO 0.00 включен, команда HEX(162) преобразует содержащиеся в словах D100 и D101 коды ASCII в соответствии с настройками указателя разрядов. (Di=#0121 указывает на то, что проверка четности не выполняется, начальным байтом для чтения является старший байт, считываются 3 байта, начальным разрядом для записи является разряд 1.)

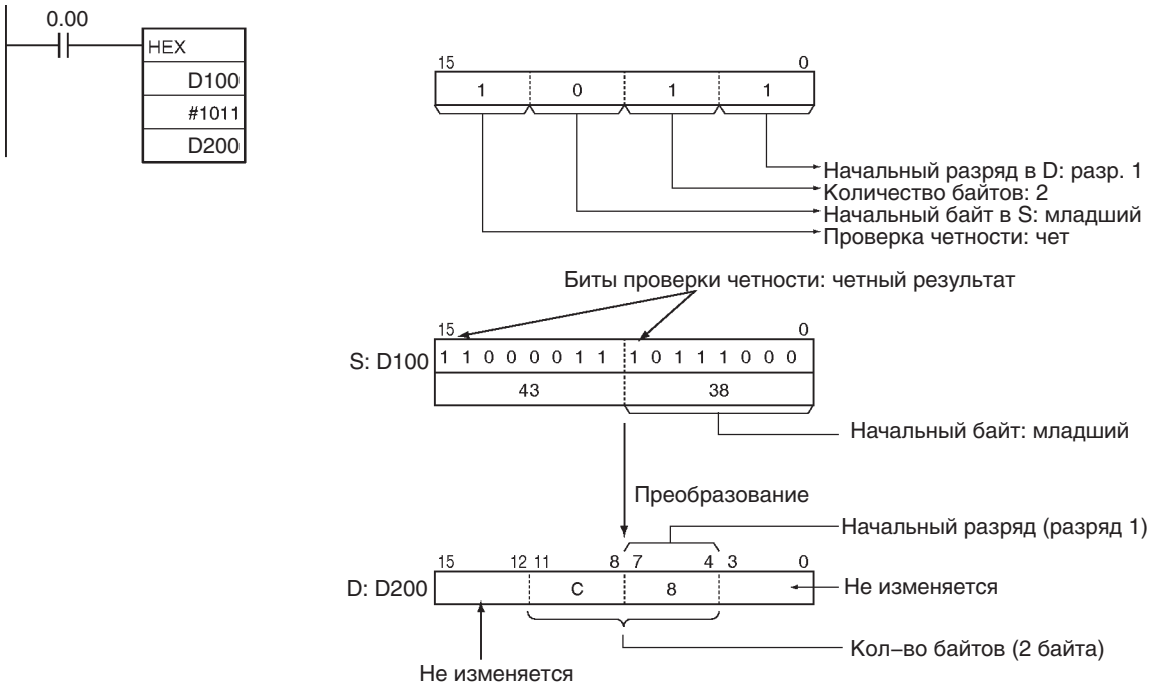
Команда HEX(162) преобразует три байта с кодами ASCII (3 символа), начиная со старшего байта слова D100, в эквивалентные шестнадцатеричные значения и записывает эти значения в слово D200, начиная с разряда 1.



Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

Если в следующем примере бит CIO 0.00 включен, команда HEX(162) преобразует содержащиеся в слове D100 коды ASCII, начиная с младшего байта, и записывает 16-ричные эквиваленты в слово D200, начиная с разряда 1.

Настройка указателя разрядов #1011 указывает на то, что выполняется проверка четности, начальным байтом для чтения является младший байт, считываются 2 байта, начальным разрядом для записи является разряд 1.)

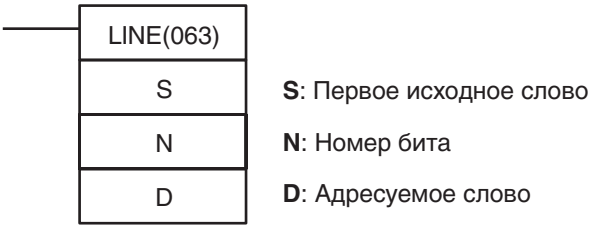


3-11-12 СТОЛБЕЦ В СТРОКУ: LINE(063)

Назначение

Преобразование столбца битов матрицы из 16 слов (биты, находящиеся в одной и той же позиции в 16 следующих друг за другом словах) в 16 битов слова назначения.

Символ РКС



Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	LINE(063)
	Однократное выполнение по положительному фронту	@LINE(063)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограм-мы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

S: Первое исходное слово

Указывает первое исходное слово. Слова S и S +15 должны принадлежать одной области.

N: Номер бита

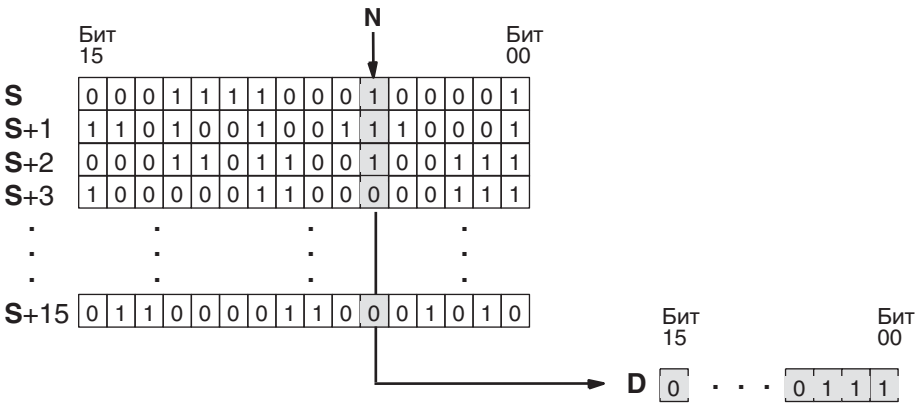
Указывает номер позиции считываемого бита (0000...000F или &0...&15) в исходных словах.

Характеристики операндов

Область	S	N	D
Область CIO	CIO 0...CIO 6128	CIO 0...CIO 6143	
Рабочая область	W0...W496	W0...W511	
Область битов хранения	H0...H496	H0...H511	
Область вспомогательных битов	A0...A944	A0...A959	A448...A959
Область таймеров	T0000...T4080	T0000...T4095	
Область счетчиков	C0000...C4080	C0000...C4095	
Область DM	D0...D32752	D0...D32767	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	---	#0000...#000F (двоичный) или &0...&15	---
Регистры данных	---	DR0...DR15	
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда LINE(063) копирует 16 битов с номером бита N из 16 слов диапазона S...S+15 в слово назначения D. Бит N слова S+m копируется в бит m слова D, т. е. бит N слова S копируется в бит 00 слова D, а бит N слова S+15 копируется в бит 15 слова D.

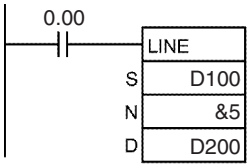


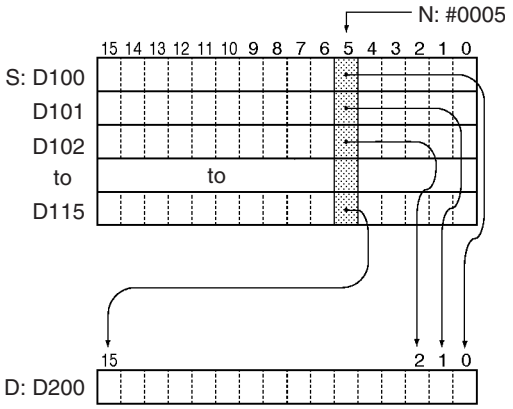
Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если значение N выходит за допустимый диапазон 0000...000F. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если D = 0000 после выполнения команды. Выключен во всех остальных случаях.

Пример

Если в следующем примере бит CIO 0.00 включен, команда LINE(063) копирует бит 5 из слов D100...D115 в 16 битов слова D200.



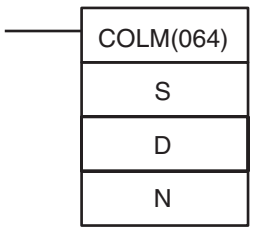


3-11-13 СТРОКУ В СТОЛБЕЦ: COLM(064)

Назначение

Преобразование 16 битов исходного слова в столбец битов матрицы из 16 слов назначения (биты, находящиеся в одной и той же позиции в 16 следующих друг за другом словах).

Символ РКС



S: Исходное слово

D: Первое слово назначения

N: Номер бита

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	COLM(064)
	Однократное выполнение по положительному фронту	@COLM(064)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограм-мы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

D: Первое слово назначения

Указывает первое слово назначения. Слова D и D +15 должны принадлежать одной области.

N: Номер бита

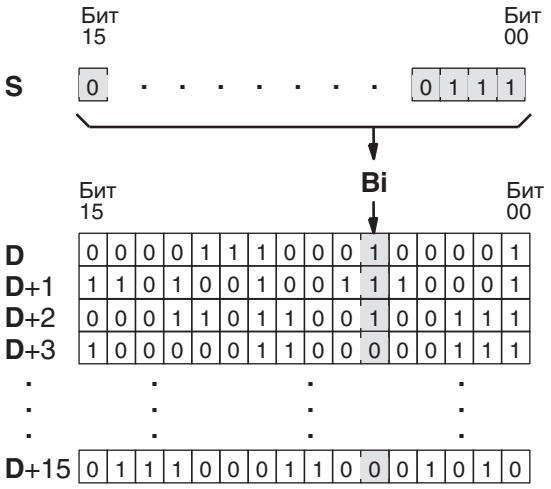
Указывает номер позиции битов (0000...000F или &0...&15), в которые будет записано значение исходного слова.

Характеристики операндов

Область	S	D	N
Область CIO	CIO 0...CIO 6143	CIO 0...CIO 6128	CIO 0...CIO 6143
Рабочая область	W0...W511	W0...W496	W0...W511
Область битов хранения	H0...H511	H0...H496	H0...H511
Область вспомогательных битов	A0...A959	A448...A944	A0...A959
Область таймеров	T0000...T4095	T0000...T4080	T0000...T4095
Область счетчиков	C0000...C4095	C0000...C4080	C0000...C4095
Область DM	D0...D32767	D0...D32752	D0...D32767
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	#0000...#FFFF (двоичн.)	---	#0000...#000F (двоичный) или &0...&15
Регистры данных	DR0...DR15	---	DR0...DR15
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда COLM(064) копирует 16 битов из слова S в 16 битов с номером N матрицы из 16 слов D...D+15. Бит m слова S копируется в бит N слова D+m, т. е. бит 00 слова S копируется в бит N слова D, а бит 15 слова S копируется в бит N слова D+15.

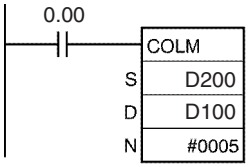


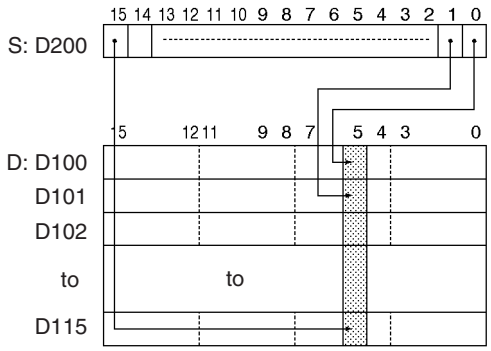
Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если значение N выходит за допустимый диапазон 0000...000F. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если в результате выполнения бит N равен «0» во всех 16 словах D...D+15. Выключен во всех остальных случаях.

Пример

Если в следующем примере бит CIO 0.00 включен, команда COLM(064) копирует 16 битов слова D200 (биты 00...15) в бит 5 слов D100...D115.





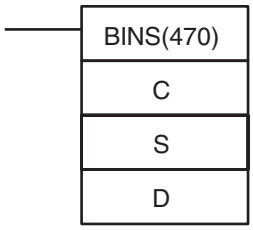
Предварительная версия
Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

3-11-14 BCD СО ЗНАКОМ В ДВОИЧНОЕ: BINS(470)

Назначение

Преобразование одного слова данных в двоично-десятичном формате со знаком в одно слово двоичных данных со знаком.

Символ РКС



C: Управляющее слово

S: Исходное слово

D: Адресуемое слово

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	BINS(470)
	Однократное выполнение по положительному фронту	@BINS(470)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

C: Управляющее слово

Указывает формат записи двоично-десятичного значения со знаком. «C» должно находиться в пределах от 0000 до 0003.

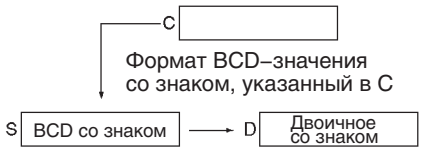
Характеристики операндов

Область	C	S	D
Область CIO	CIO 0...CIO 6143		
Рабочая область	W0...W511		
Область битов хранения	H0...H511		
Область вспомогательных битов	A0...A959		A448...A959
Область таймеров	T0000...T4095		
Область счетчиков	C0000...C4095		
Область DM	D0...D32767		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	#0000...#0003 (двоичн.)	---	
Регистры данных	DR0...DR15		

Область	C	S	D
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда BINS(470) преобразует двоично-десятичное значение со знаком в двоичное значение со знаком. Сначала проверяется, соответствуют ли двоично-десятичное значение со знаком (S) и его формат настройкам управляющего слова (C). Если исходные данные соответствуют настройкам, двоично-десятичное значение со знаком слова S преобразуется в двоичное значение, результат выводится в слово D. Если исходные данные не соответствуют настройкам, устанавливается флаг ошибки, команда не выполняется.



Если преобразованное значение окажется отрицательным, оно будет представлено как дополнение до двух, при этом будет установлен флаг отрицательного значения. Для вычисления модуля отрицательного двоичного числа можно использовать команду NEG(160). Подробное описание смотрите в *3-11-5 ДОПОЛНЕНИЕ ДО 2: NEG(160)432*.

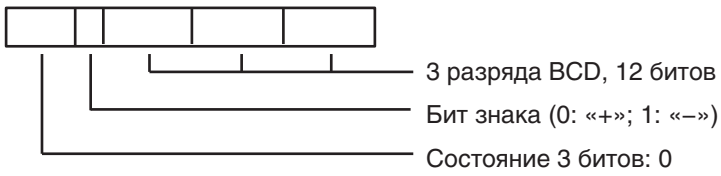
Значение «-0» в исходных данных будет восприниматься как «0» и не вызовет ошибку. Кроме того, состояние битов 13...15 не проверяется, если C = 0000.

Примечание.

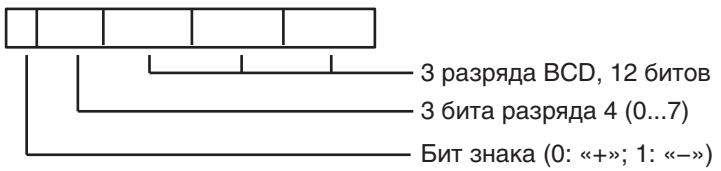
Некоторые специальные модули ввода/вывода выдают значения со знаком в двоично-десятичном формате. Преобразование таких значений в двоичные значения со знаком с помощью команды BINS(470) облегчает дальнейшие вычисления.

Формат записи двоично-десятичного значения со знаком определяется управляющим словом следующим образом.

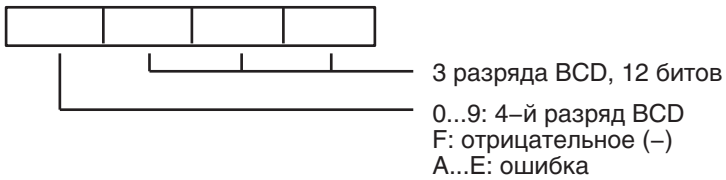
C = 0000 (диапазон входных значений: -999...999 BCD)



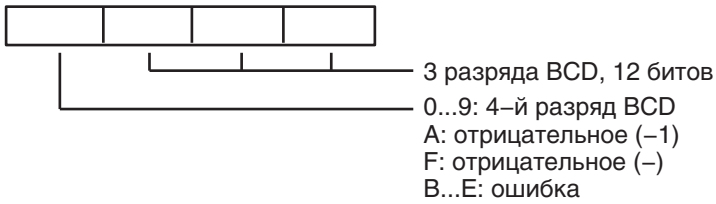
C = 0001 (диапазон входных значений: -7999...7999 BCD)



C = 0002 (диапазон входных значений: -999...9999 BCD)



C = 0003 (диапазон входных значений: -1999...9999 BCD)



В следующей таблице перечислены возможные двоично-десятичные значения и соответствующие им двоичные значения со знаком для каждого формата записи двоично-десятичного значения со знаком.

Значение	BCD-значения со знаком	Двоичные значения со знаком
C=0000	-999...-1 и 0...999	FC19...FFFF и 0000...03E7
C=0001	-7999...-1 и 0...7999	E0C1...FFFF и 0000...1F3F
C=0002	-999...-1 и 0...9999	FC19...FFFF и 0000...270F
C=0003	-1999...-1 и 0...9999	F831...FFFF и 0000...270F

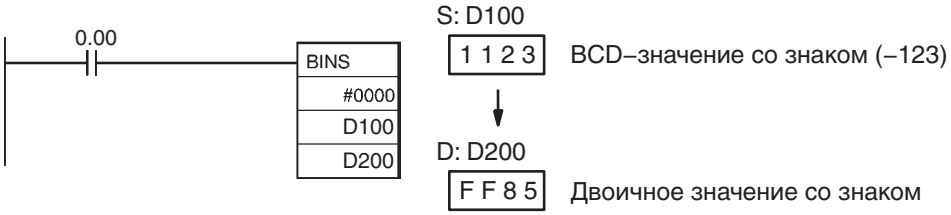
Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если значение C выходит за допустимый диапазон 0000...0003. Включен, если C = 0002 и старший разряд S = A...E. Включен, если C = 0003 и старший разряд S = B...E. Включен, если содержимое S не в формате BCD. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если D = 0000 после выполнения команды. Выключен во всех остальных случаях.
Флаг отрицательного значения	N	Включен, если после выполнения команды бит 15 слова D = «1». Выключен во всех остальных случаях.

Примеры

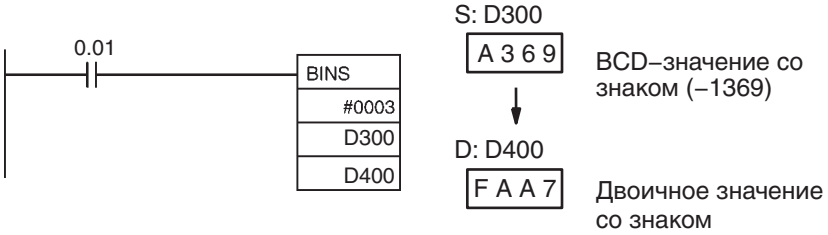
Формат BCD 0 (C=#0000)

Если в следующем примере бит CIO 0.00 включен, выполняется проверка соответствия формата и значения BCD-данных со знаком слова D100 формату, указанному в управляющем слове (0000). Исходные данные соответствуют настройкам, двоично-десятичное значение со знаком слова D100 преобразуется в двоичное значение со знаком и записывается в слово D200.



Формат BCD 0 (C=#0003)

Если в следующем примере бит CIO 0.01 включен, выполняется проверка соответствия формата и значения BCD-данных со знаком слова D300 формату, указанному в управляющем слове (0003). Исходные данные соответствуют настройкам, двоично-десятичное значение со знаком слова D300 преобразуется в двоичное значение со знаком и записывается в слово D400.

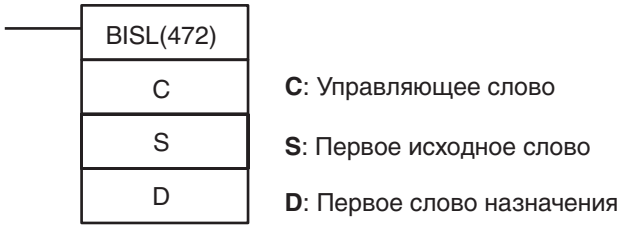


3-11-15 ДВОЙНОЕ BCD СО ЗНАКОМ В ДВОИЧНОЕ: BISL(472)

Назначение

Преобразование двойного слова данных в двоично-десятичном формате со знаком в двойное слово двоичных данных со знаком.

Символ РКС



Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	BISL(472)
	Однократное выполнение по положительному фронту	@BISL(472)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

C: Управляющее слово

Указывает формат записи двоично-десятичного значения со знаком. «C» должно находиться в пределах от 0000 до 0003.

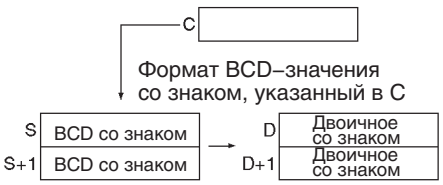
Характеристики операндов

Область	C	S	D
Область CIO	CIO 0...CIO 6143	CIO 0...CIO 6142	
Рабочая область	W0...W511	W0...W510	
Область битов хранения	H0...H511	H0...H510	
Область вспомогательных битов	A0...A959	A0...A958	A448...A958

Область	C	S	D
Область таймеров	T0000...T4095	T0000...T4094	
Область счетчиков	C0000...C4095	C0000...C4094	
Область DM	D0...D32767	D0...D32766	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	#0000...#0003 (двоичн.)	---	
Регистры данных	DR0...DR15	---	
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда BISL(472) преобразует 8-разрядное двоично-десятичное значение со знаком [S+1, S] в 8-разрядное двоичное значение со знаком и записывает результат в [D+1, D]. Сначала проверяется, соответствуют ли двоично-десятичное значение со знаком двойного слова [S+1, S] и его формат настройкам управляющего слова (C). Если исходные данные соответствуют настройкам, двоично-десятичное значение со знаком двойного слова [S+1, S] преобразуется в двоичное значение, результат выводится в слова [D+1, D]. Если исходные данные не соответствуют настройкам, устанавливается флаг ошибки, команда не выполняется.

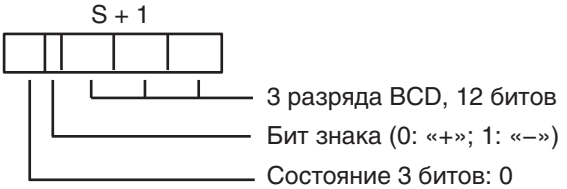


Если преобразованное значение окажется отрицательным, оно будет представлено как дополнение до двух, при этом будет установлен флаг отрицательного значения. Для вычисления модуля отрицательного двоичного числа длиной в два слова можно использовать команду NEGL(161). Подробное описание смотрите в *3-11-6 ДОПОЛНЕНИЕ ДВОЙНОГО СЛОВА ДО 2: NEGL(161)*.

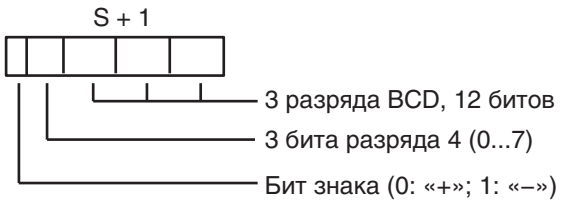
Значение «-0» в исходных данных будет восприниматься как «0» и не вызовет ошибку. Кроме того, состояние битов 13...15 слова S+1 не проверяется, если C = 0000.

Формат записи двоично-десятичного значения со знаком определяется управляющим словом следующим образом.

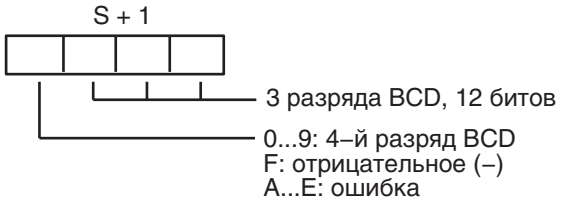
C = 0000 (диапазон входных значений: -999 9999...999 9999 BCD)



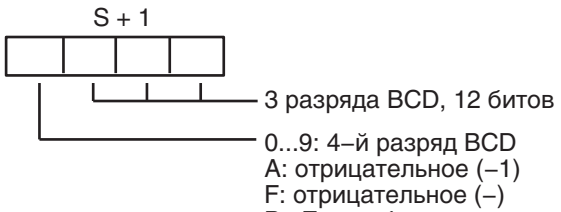
C = 0001 (диапазон входных значений: -7999 9999...7999 9999 BCD)



C = 0002 (диапазон входных значений: -999 9999...9999 9999 BCD)



C = 0003 (диапазон входных значений: –1999 9999...9999 9999 BCD)



В следующей таблице перечислены возможные двоично-десятичные значения и соответствующие им двоичные значения со знаком для каждого формата записи двоично-десятичного значения со знаком.

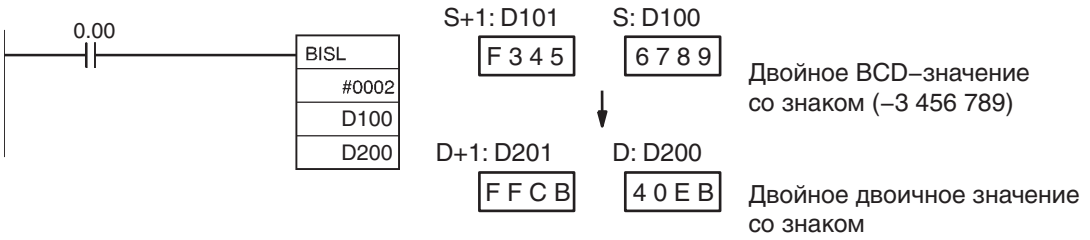
Значение	BCD-значения со знаком	Двоичные значения со знаком
C=0000	-999 9999...-1	FF67 6981...FFFF FFFF
	0...999 9999	0000 0000...0098 967F
C=0001	-7999 9999...-1	FB3B 4C01...FFFF FFFF
	0...7999 9999	0000 0000...04C4 B3FF
C=0002	-999 9999...-1	FF67 6981...FFFF FFFF
	0...9999 9999	0000 0000...05F5 E0FF
C=0003	-1999 9999...-1	FECE D301...FFFF FFFF
	0...9999 9999	0000 0000...05F5 E0FF

Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если значение C выходит за допустимый диапазон 0000...0003. Включен, если C = 0002 и старший разряд S+1 = A...E. Включен, если C = 0003 и старший разряд S+1 = B...E. Включен, если содержимое [S+1, S] не в формате BCD. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если после выполнения команды двойное слово [D, D+1] содержит значение 0000 0000. Выключен во всех остальных случаях.
Флаг отрицательного значения	N	Включен, если после выполнения команды бит 15 слова D+1 = «1». Выключен во всех остальных случаях.

Пример

Если в следующем примере бит CIO 0.00 включен, выполняется проверка соответствия формата и значения BCD-данных со знаком двойного слова [D101, D100] формату, указанному в управляющем слове (0002). Исходные данные соответствуют настройкам, двоично-десятичное значение со знаком двойного слова [D101, D100] преобразуется в двоичное значение со знаком и записывается в двойное слово [D201, D200].

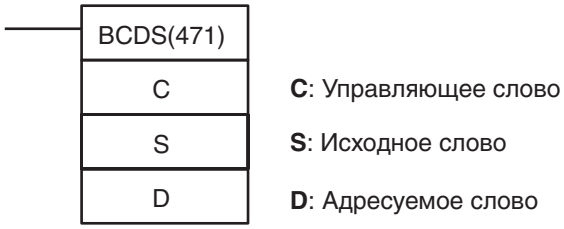


3-11-16 ДВОИЧНОЕ СО ЗНАКОМ В BCD: BCDS(471)

Назначение

Преобразование одного слова двоичных данных со знаком в одно слово в двоично-десятичном формате со знаком.

Символ РКС



Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	BCDS(471)
	Однократное выполнение по положительному фронту	@BCDS(471)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнд

C: Управляющее слово

Указывает формат записи двоично-десятичного значения со знаком. «C» должно находиться в пределах от 0000 до 0003.

S: Исходное слово

Содержит преобразуемое двоичное значение со знаком. Значение S должно принадлежать диапазону допустимых двоично-десятичных значений, который соответствует формату, заданному словом C.

Значение	Допустимые значения S
C=0000	FC19...FFFF или 0000...03E7
C=0001	E0C1...FFFF или 0000...1F3F
C=0002	FC19...FFFF или 0000...270F
C=0003	F831...FFFF или 0000...270F

D: Слово назначения

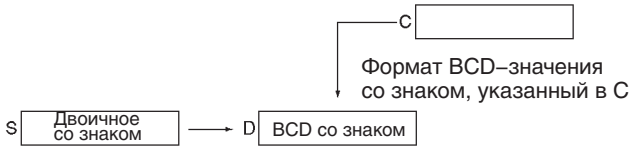
Содержит результат преобразования: двоично-десятичное значение со знаком. Форматы записи двоично-десятичных значений подробно поясняются в подразделе «Описание» ниже.

Характеристики операндов

Область	C	S	D
Область CIO	CIO 0...CIO 6143		
Рабочая область	W0...W511		
Область битов хранения	H0...H511		
Область вспомогательных битов	A0...A959		A448...A959
Область таймеров	T0000...T4095		
Область счетчиков	C0000...C4095		
Область DM	D0...D32767		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	#0000...#0003 (двоичн.)	---	
Регистры данных	DR0...DR15		
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...1-2048...+2047 ,IR5 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда BCDS(471) преобразует двоичное значение со знаком в двоично-десятичное значение со знаком. Сначала выполняется проверка, попадает ли двоичное значение со знаком (S) в допустимый диапазон двоично-десятичных значений со знаком, соответствующий формату, который указан в управляющем слове (C). Если исходные данные соответствуют настройкам, двоичное значение со знаком слова S преобразуется в двоично-десятичное значение со знаком, которое выводится в слово D. Если исходные данные не соответствуют настройкам, устанавливается флаг ошибки, команда не выполняется.



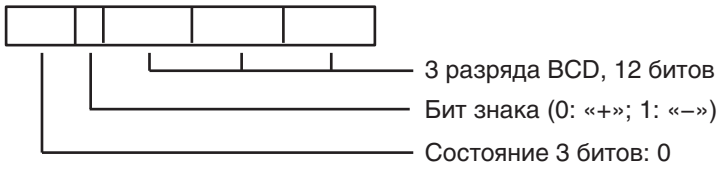
Примечание.

(1) Значение «-0» в исходных данных будет восприниматься как «0» и не вызовет ошибку.

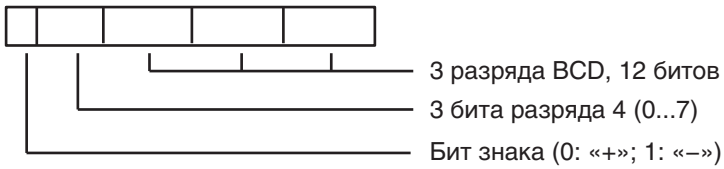
(2) Некоторые специальные модули ввода/вывода принимают данные только в формате двоично-десятичных значений со знаком. Перед выводом данных в такие модули можно использовать команду BCDS(471) для преобразования двоичных значений со знаком в двоично-десятичные значения.

Используемый для результата формат записи двоично-десятичного значения со знаком определяется управляющим словом следующим образом.

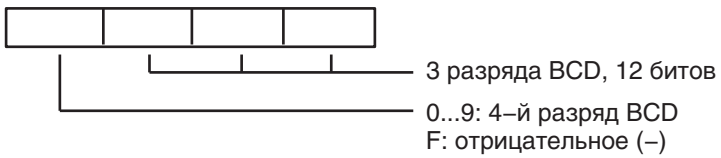
C = 0000 (диапазон выходных значений: -999...999 BCD)



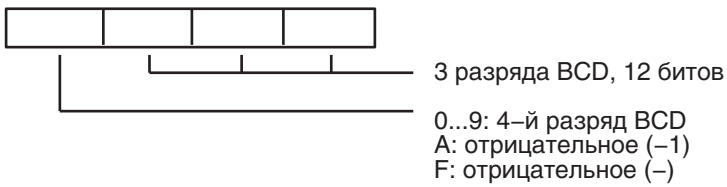
C = 0001 (диапазон выходных значений: -7999...7999 BCD)



C = 0002 (диапазон выходных значений: -999...9999 BCD)



C = 0003 (диапазон выходных значений: -1999...9999 BCD)



В следующей таблице показаны допустимые значения двоичных значений со знаком для каждого диапазона двоично-десятичных значений со знаком. Если входное значение выходит за допустимый диапазон значений, соответствующий указанному формату записи двоично-десятичного значения со знаком, возникнет ошибка.

Значение	Двоичные значения со знаком	BCD-значения со знаком
C=0000	FC19...FFFF и 0000...03E7	-999...-1 и 0...999
C=0001	E0C1...FFFF и 0000...1F3F	-7999...-1 и 0...7999
C=0002	FC19...FFFF и 0000...270F	-999...-1 и 0...9999
C=0003	F831...FFFF и 0000...270F	-1999...-1 и 0...9999

Флаги

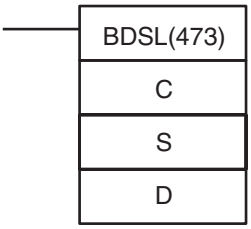
Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если значение C выходит за допустимый диапазон 0000...0003. Включен, если C = 0000, а исходное значение выходит за допустимый диапазон (FC19...FFFF или 0000...03E7). Включен, если C = 0001, а исходное значение выходит за допустимый диапазон (E0C1...FFFF или 0000...1F3F). Включен, если C = 0002, а исходное значение выходит за допустимый диапазон (FC19...FFFF или 0000...270F). Включен, если C = 0003, а исходное значение выходит за допустимый диапазон (F831...FFFF или 0000...270F). Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если D = 0000 после выполнения команды. Выключен во всех остальных случаях.
Флаг отрицательного значения	N	Включен, если C = 0000 или 0001 и после выполнения команды бит знака результата = «1». Включен, если C = 0002 и старший разряд результата = «F». Включен, если C = 0003 и старший разряд результата = «A» или «F». Выключен во всех остальных случаях.

3-11-17 ДВОЙНОЕ ДВОИЧНОЕ СО ЗНАКОМ В ДВОЙНОЕ BCD: BDSL(473)

Назначение

Преобразование двойного слова двоичных данных со знаком в двойное слово данных в двоично-десятичном формате со знаком.

Символ РКС



C: Управляющее слово

S: Первое исходное слово

D: Первое слово назначения

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	BDSL(473)
	Однократное выполнение по положительному фронту	@BDSL(473)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Операнды

C: Управляющее слово

Указывает формат записи двоично-десятичного значения со знаком. «C» должно находиться в пределах от 0000 до 0003.

S: Первое исходное слово

Исходное двойное слово [S+1, S] содержит преобразуемое двоичное значение со знаком. Значение [S+1, S] должно принадлежать диапазону допустимых двоично-десятичных значений, который соответствует формату, заданному словом C.

Значение	Допустимые значения [S+1, S]
C=0000	FF67 6981...FFFF FFFF... 0000 0000...0098 967F
C=0001	FB3B 4C01...FFFF FFFF...0000 0000...04C4 B3FF
C=0002	FF67 6981...FFFF FFFF... 0000 0000...05F5 E0FF
C=0003	FECE D301...FFFF FFFF...0000 0000...05F5 E0FF

D: Первое слово назначения

Слова назначения D+1 и D — это двойное слово, которое содержит результат преобразования, т. е. двоично-десятичное значение со знаком. Форматы записи двоично-десятичных значений подробно поясняются в подразделе «Описание» ниже.

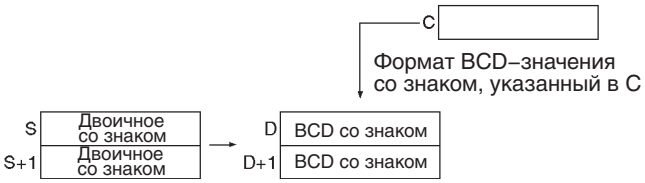
Характеристики операндов

Область	C	S	D
Область CIO	CIO 0...CIO 6143	CIO 0...CIO 6142	
Рабочая область	W0...W511	W0...W510	
Область битов хранения	H0...H511	H0...H510	
Область вспомогательных битов	A0...A959	A0...A958	A448...A958
Область таймеров	T0000...T4095	T0000...T4094	
Область счетчиков	C0000...C4095	C0000...C4094	
Область DM	D0...D32767	D0...D32766	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	#0000...#0003 (двоичн.)	---	
Регистры данных	DR0...DR15	---	
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда BDSL(473) преобразует 8-разрядное двоичное значение со знаком в 8-разрядное двоично-десятичное значение со знаком. Сначала выполняется проверка, попадает ли двоичное значение со знаком двойного слова [S+1, S] в допустимый диапазон двоично-десятичных значений со знаком, соответствующий формату, который указан в управляющем слове (C). Если исходные данные соответствуют

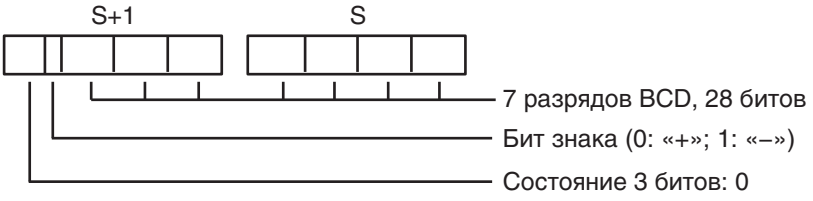
настройкам, 8-разрядное двоичное значение со знаком [S+1, S] преобразуется в 8-разрядное двоично-десятичное значение со знаком, которое записывается в слова [D+1, D]. Если исходные данные не соответствуют настройкам, устанавливается флаг ошибки, команда не выполняется.



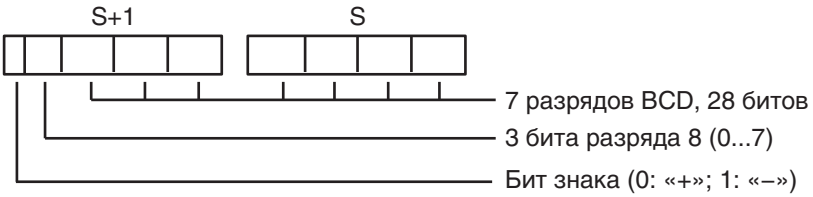
Примечание. Значение «-0» в исходных данных будет восприниматься как «0» и не вызовет ошибку.

Используемый для результата формат записи двоично-десятичного значения со знаком определяется управляющим словом следующим образом.

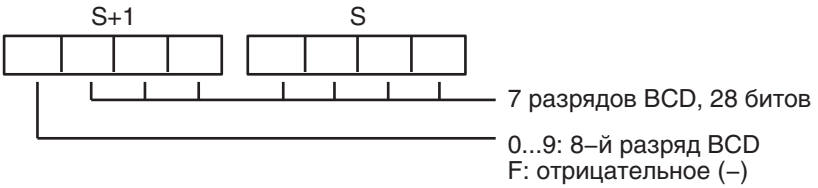
C = 0000 (диапазон выходных значений: -999 9999...999 9999 BCD)



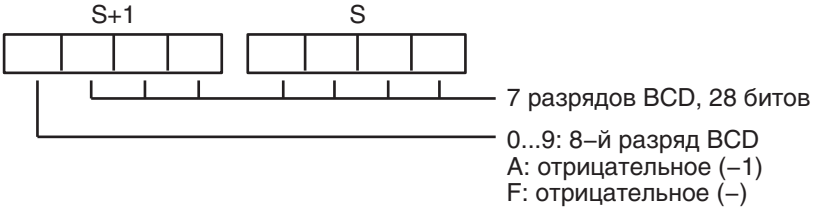
C = 0001 (диапазон выходных значений: -7999 9999...7999 9999 BCD)



C = 0002 (диапазон выходных значений: -999 9999...9999 9999 BCD)



C = 0003 (диапазон выходных значений: -1999 9999...9999 9999 BCD)



В следующей таблице показаны допустимые 8-разрядные двоичные значения со знаком для каждого диапазона двоично-десятичных значений со знаком. Если входное значение выходит за допустимый

диапазон значений, соответствующий указанному формату записи двоично-десятичного значения со знаком, возникнет ошибка.

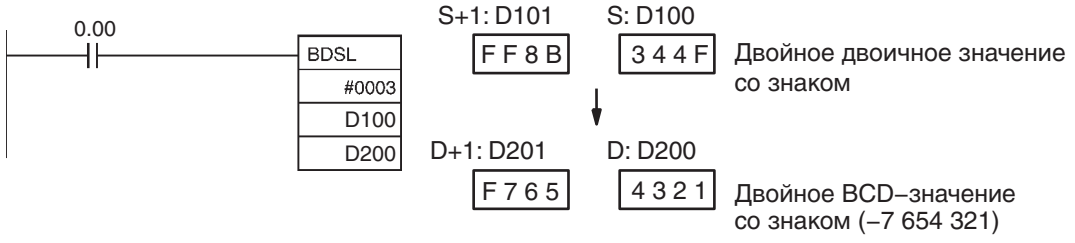
Значение	Двоичные значения со знаком	BCD-значения со знаком
C=0000	FF67 6981...FFFF FFFF	-999 9999...-1
	0000 0000...0098 967F	0...999 9999
C=0001	FB3B 4C01...FFFF FFFF	-7999 9999...-1
	0000 0000...04C4 B3FF	0...7999 9999
C=0002	FF67 6981...FFFF FFFF	-999 9999...-1
	0000 0000...05F5 E0FF	0...9999 9999
C=0003	FECE D301...FFFF FFFF	-1999 9999...-1
	0000 0000...05F5 E0FF	0...9999 9999

Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если значение C выходит за допустимый диапазон 0000...0003. Включен, если C = 0000, а исходное значение выходит за диапазон: FF67 6981...FFFF FFFF или 0000 0000...0098 967F. Включен, если C = 0001, а исходное значение выходит за диапазон: FB3B 4C01...FFFF FFFF или 0000 0000...04C4 B3FF. Включен, если C = 0002, а исходное значение выходит за диапазон: FF67 6981...FFFF FFFF или 0000 0000...05F5 E0FF. Включен, если C = 0003, а исходное значение выходит за диапазон: FECE D301...FFFF FFFF или 0000 0000...05F5 E0FF. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если после выполнения команды двойное слово [D, D+1] содержит значение 0000 0000. Выключен во всех остальных случаях.
Флаг отрицательного значения	N	Включен, если C = 0000 или 0001 и после выполнения команды бит знака результата = «1». Включен, если C = 0002 и старший разряд результата = «F». Включен, если C = 0003 и старший разряд результата = «A» или «F». Выключен во всех остальных случаях.

Пример

Если в следующем примере бит CIO 0.00 включен, выполняется проверка соответствия 8-разрядного двоичного значения со знаком [D101, D100] формату, указанному в управляющем слове (0003). Исходные данные соответствуют настройкам, двоичное значение со знаком двойного слова [D101, D100] преобразуется в двоично-десятичное значение со знаком и записывается в слова [D201, D200].

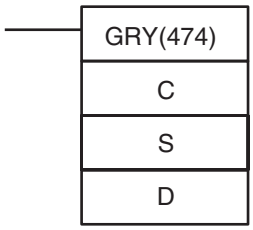


3-11-18 ПРЕОБРАЗОВАНИЕ КОДА ГРЕЯ: GRY(474)

Назначение

Преобразование кода Грэя, содержащегося в указанном слове, в стандартный двоичный формат, в двоично-десятичный формат или в значение угла с указанным разрешением.

Символ РКС



C: Первое управляющее слово

S: Исходное слово

D: Первое слово назначения

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	GRY(474)
	Однократное выполнение по положительному фронту	@GRY(474)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

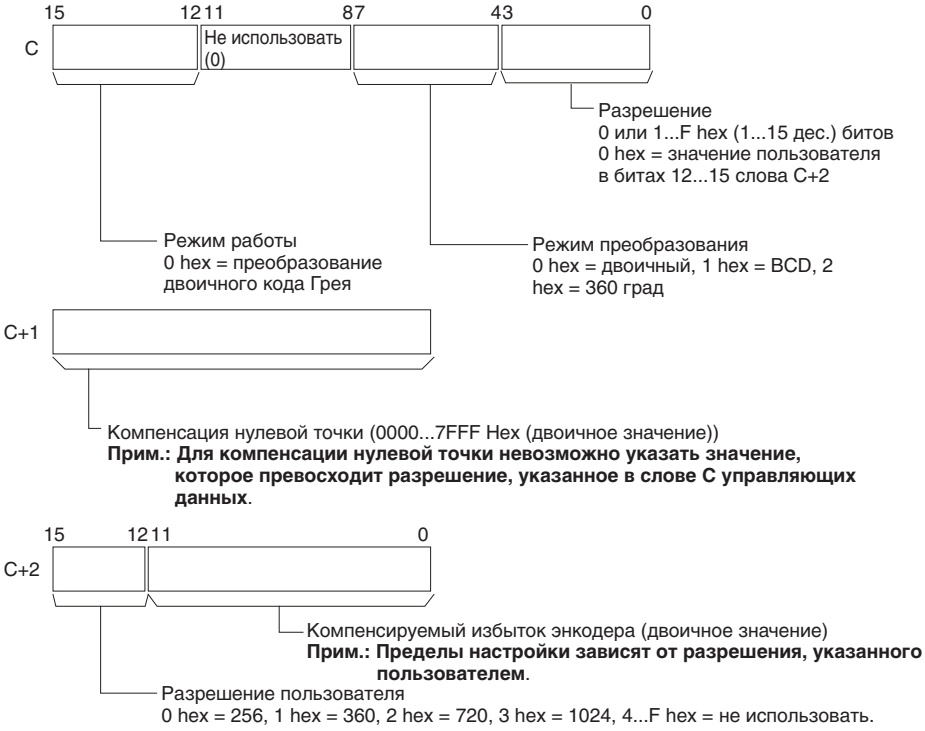
Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

C: Управляющее слово

Указывает параметры преобразования, как показано ниже.



Прим.: Показанная выше настройка действительна, если в битах 00...03 слова C задано разрешение 0 hex.

S: Исходное слово

Содержит преобразуемые данные, закодированные кодом Грея. Значение исходного слова не должно приводить к превышению числа битов, которое определяется разрешением, указанным в битах 00...03 слова C. Все биты вне диапазона, определяемого разрешением, игнорируются. Например, если указано разрешение 08 hex, а слово S содержит FFFF hex, в качестве кода Грея воспринимается значение 00FF hex.

S	

D: Первое слово назначения

Слова назначения D+1 и D — это двойное слово, которое содержит результаты преобразования кода Грея с разрешением, указанным в битах 00...03 управляющего слова C, и типом преобразования, указанным в битах 04...07 управляющего слова C. Старшее слово выводится в слово D+1, младшее слово выводится в слово D. Диапазоны выходных значений приведены ниже:

Двоичный формат: 0000 0000...0000 7FFF hex
Формат BCD:0000 0000...0003 2767
Режим 360°:0000 0000...0000 3599
(0,0°...359,9° с шагом 0,1°, BCD)

D	Младшее слово
D+1	Старшее слово

Характеристики операндов

Область	C	S	D
Область CIO	CIO 0...CIO 6141	CIO 0...CIO 6143	CIO 0...CIO 6142
Рабочая область	W0...W509	W0...W511	W0...W510
Область битов хранения	H0...H509	H0...H511	H0...H510
Область вспомогательных битов	A0...A957	A0...A959	A448...A958
Область таймеров	T0000...T4093	T0000...T4095	T0000...T4094
Область счетчиков	C0000...C4093	C0000...C4095	C0000...C4094
Область DM	D0...D32765	D0...D32767	D0...D32766
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	---	#0000...#FFFF (двоичн.)	---
Регистры данных	---	DR0...DR15	---
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда GRY(474) преобразует код Грея, указанный операндом S, в значение в указанном формате, и записывает результат в [D, D+1]. Разрешение и выходной формат (двоичный, двоично-десятичный или 360°) указываются в операнде C..

Метод преобразования	Описание
Двоичный формат	Код Грея преобразуется в двоичное значение в диапазоне 0000 0000...0000 7FFF hex. Применяются сдвиг нулевой точки и компенсация излишка, после чего результат выводится в слова D и D+1.
Формат BCD	Значение двоичного кода Грея преобразуется в двоично-десятичное значение. Применяются сдвиг нулевой точки и компенсация излишка, код Грея преобразуется в значение в двоично-десятичном формате в диапазоне 0000 0000... 0003 2767, результат записывается в слова D и D+1.
Формат 360°	Значение двоичного кода Грея преобразуется в двоично-десятичное значение. Применяются сдвиг нулевой точки и компенсация излишка, код Грея преобразуется в значение угла в диапазоне 0000 0000 и 0000 3599 (0,0°...359,9° с шагом 0,1°), а затем результат записывается в слова D и D+1.

Примечание.

(1) Команда GRY(474) обычно используется, когда к модулю входов постоянного тока подключен абсолютный энкодер с параллельным интерфейсом (2^n), кодирующий данные кодом Грея.

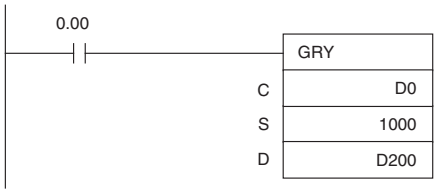
(2) Если в операнде S указано слово, отведенное для модуля ввода, входные данные, преобразуемые командой GRY(474), будут соответствовать коду Грея, полученному в предыдущем цикле модуля ЦПУ, т. е. будут отставать на время выполнения одного цикла.

Флаги

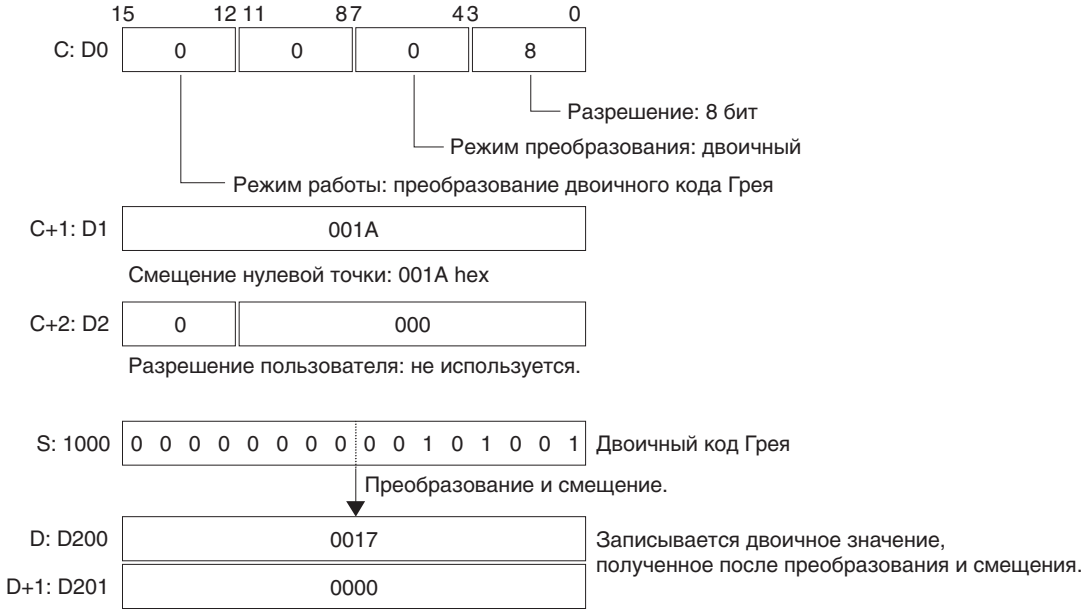
Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если биты 12...15 слова C не равны 0 hex (режим = преобразование кода Грея). Включен, если сдвиг нулевой точки, заданный в C+1, не соответствует заданному разрешению (в том числе разрешению, указанному пользователем). Включен, если биты 04...07 слова C не равны 0 hex (= двоичный формат), 1 hex (= формат BCD) или 2 hex (= формат 360°). Включен, если указанный компенсируемый избыток превышает указанное пользователем разрешение, а биты 00...03 слова C равны 0 hex (= указанное пользователем разрешение). Включен, если результирующее двоичное значение меньше компенсируемого излишка, а биты 00...03 слова C равны 0 hex (= указанное пользователем разрешение). Включен, если результирующее двоичное значение меньше разрешения, а биты 00...03 слова C равны 0 hex (= указанное пользователем разрешение). Выключен во всех остальных случаях.
Флаг равенства	=	Выключен во всех случаях.
Флаг отрицательного значения	N	Выключен во всех случаях.

Примеры

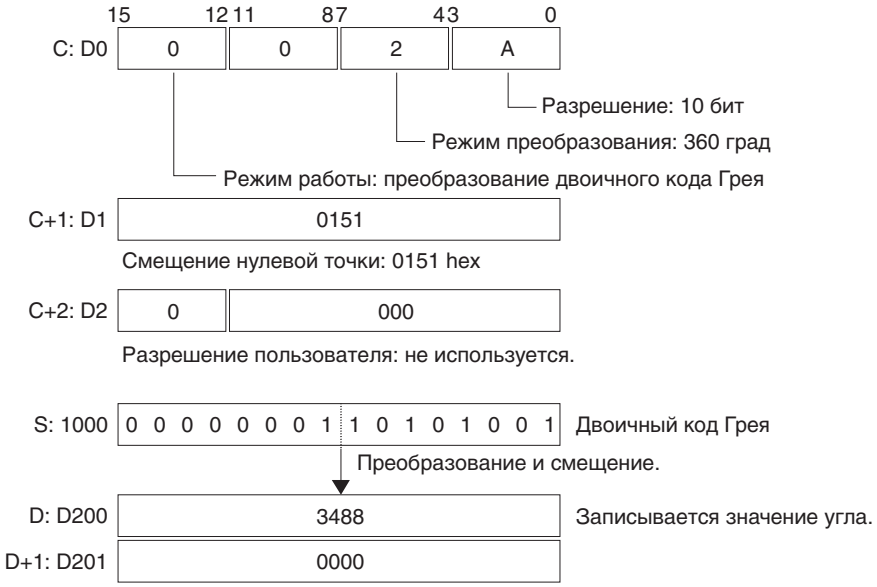
Если в следующем примере бит CIO 0.00 включен, содержащийся в слове CIO 1000 код Грея преобразуется в соответствии с содержанием управляющих слов D0...D2, результат выводится в слово D200.



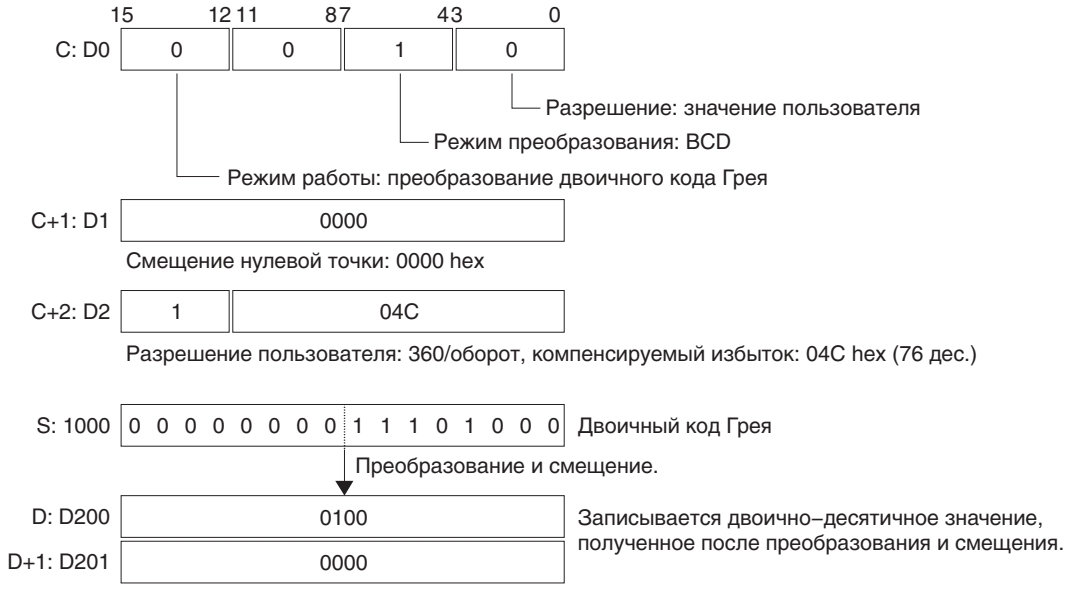
Пример 1: преобразование в двоичное значение с разрешением 8 бит и сдвигом нулевой точки на 001A Hex



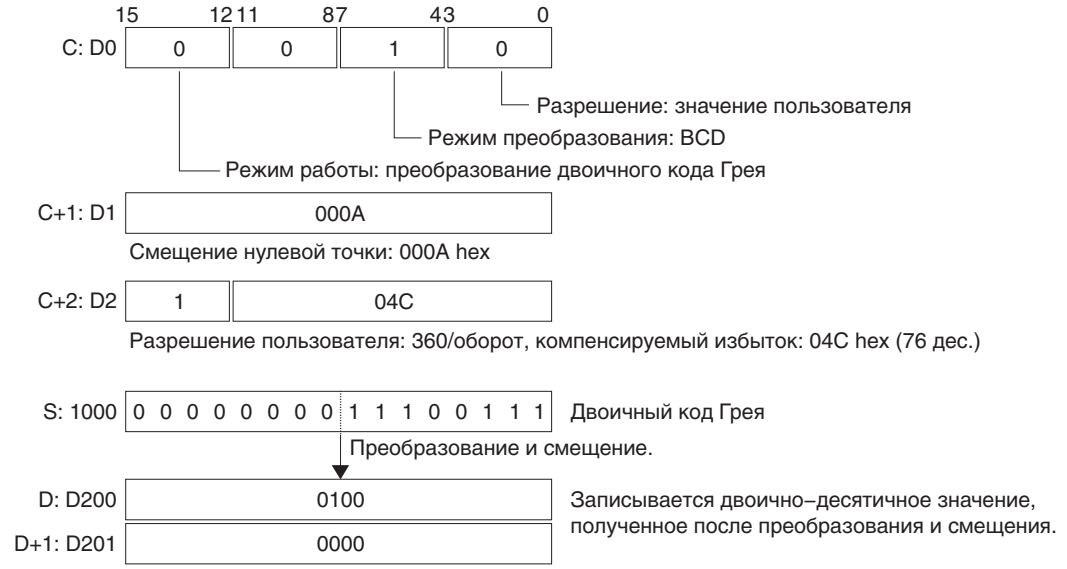
Пример 2: Преобразование в значение угла с разрешением 10 бит и сдвигом нулевой точки на 0151 Hex



Пример 3: преобразование в двоично-десятичное значение для абсолютного энкодера E6C2-AG5C компании Omron (разрешение: 360/оборот, компенсируемый избыток: 76) со сдвигом нулевой точки на 0000 Hex



Пример 4: Преобразование в двоично-десятичное значение для абсолютного энкодера E6C2-AG5C компании Omron (разрешение: 360/оборот, компенсируемый избыток: 76) со сдвигом нулевой точки на 000A Hex



3-12 Логические команды

В данном разделе описаны команды, выполняющие логические операции над словами данных.

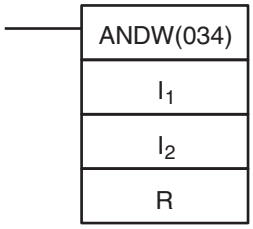
Команда	Мнемоническое обозначение	Код функции	Стр.
ЛОГИЧЕСКОЕ И	ANDW	034	477
ДВОЙНОЕ ЛОГИЧЕСКОЕ И	ANDL	610	478
ЛОГИЧЕСКОЕ ИЛИ	ORW	035	480
ДВОЙНОЕ ЛОГИЧЕСКОЕ ИЛИ	ORWL	611	482
ИСКЛЮЧАЮЩЕЕ ИЛИ	XORW	036	483
ДВОЙНОЕ ИСКЛЮЧАЮЩЕЕ ИЛИ	XORL	612	485
ИСКЛЮЧАЮЩЕЕ НЕ-ИЛИ	XNRW	037	487
ДВОЙНОЕ ИСКЛЮЧАЮЩЕЕ НЕ-ИЛИ	XNRL	613	488
ДОПОЛНЕНИЕ	Общий (COM)	029	490
ДВОЙНОЕ ДОПОЛНЕНИЕ	COML	614	492

3-12-1 ЛОГИЧЕСКОЕ И: ANDW(034)

Назначение

Операция «логическое И» над соответствующими битами одиночных слов данных и/или констант.

Символ РКС



I1: Вход 1

I2: Вход 2

R: Слово результата

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	ANDW(034)
	Однократное выполнение по положительному фронту	@ANDW(034)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	I_1	I_2	R
Область CIO	CIO 0...CIO 6143		
Рабочая область	W0...W511		
Область битов хранения	H0...H511		
Область вспомогательных битов	A0...A959		A448...A959
Область таймеров	T0000...T4095		
Область счетчиков	C0000...C4095		
Область DM	D0...D32767		

Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

Область	I_1	I_2	R
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767		
Постоянные	#0000...#FFFF (двоичн.)		---
Регистры данных	DR0...DR15		
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда ANDW(034) выполняет операцию «логическое И» над данными, указанными в словах I_1 и I_2, и выводит результат в слово R.

- Выполняется побитовое логическое умножение слов I_1 и I_2 .
- Если соответствующие биты в I_1 и I_2 одновременно содержат «1», в соответствующий бит слова R будет выведено значение «1», если какой-либо из них содержит «0», в соответствующий бит слова R будет выведено значение «0».

I_1, I_2 → R

I_1	I_2	R
1	1	1
1	0	0
0	1	0
0	0	0

Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	ВЫКЛ
Флаг равенства	=	Включен, если результат = 0. Выключен во всех остальных случаях.
Флаг «Меньше нуля»	N	Включен, если старший бит результата (R) = «1». Выключен во всех остальных случаях.

Меры предосторожности

При выполнении команды ANDW(034) сбрасывается флаг ошибки.

Если в результате операции «И» содержимое R окажется равным 0000 hex, будет установлен флаг равенства.

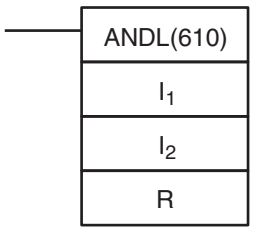
Если в результате операции «И» старший бит R окажется равным «1», будет установлен флаг отрицательного значения.

3-12-2 ДВОЙНОЕ ЛОГИЧЕСКОЕ И: ANDL(610)

Назначение

Операция «логическое И» над соответствующими битами двойных слов данных и/или констант.

Символ РКС



I_1: Вход 1

I_2: Вход 2

R: Слово результата

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	ANDL(610)
	Однократное выполнение по положительному фронту	@ANDL(610)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	I_1	I_2	R
Область CIO	CIO 0...CIO 6142		
Рабочая область	W0...W510		
Область битов хранения	H0...H510		
Область вспомогательных битов	A0...A958		A448...A958
Область таймеров	T0000...T4094		
Область счетчиков	C0000...C4094		
Область DM	D0...D32766		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767		
Постоянные	#00000000...#FFFFFFFF (двоичное)		---
Регистры данных	---		
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда ANDL(610) выполняет операцию «логическое И» над данными, указанными в словах I_1, I_1+1 и I_2, I_2+1, и выводит результат в слова R, R+1.

(I_1, I_1+1), (I_2, I_2+1) → (R, R+1)

I_1, I_1+1	I_2, I_2+1	R, R+1
1	1	1
1	0	0
0	1	0
0	0	0

Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	ВЫКЛ
Флаг равенства	=	Включен, если результат = 0. Выключен во всех остальных случаях.
Флаг «Меньше нуля»	N	Включен, если старший бит результата (R) = «1». Выключен во всех остальных случаях.

Меры предосторожности

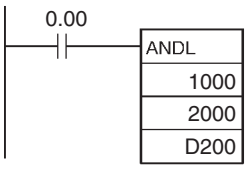
При выполнении команды ANDL(610) сбрасывается флаг ошибки.

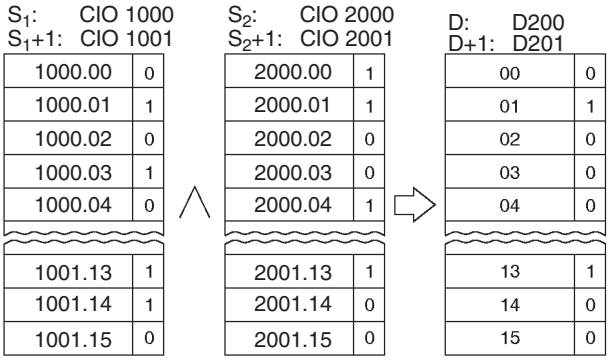
Если в результате операции «И» содержимое R, R+1 окажется равным 00000000 hex, будет установлен флаг равенства.

Если в результате операции «И» старший бит R+1 окажется равным «1», будет установлен флаг отрицательного значения.

Примеры

Если установлено условие выполнения CIO 0.00, будет выполнено логическое умножение соответствующих битов слов CIO 1001, CIO 1000 и CIO 2001, CIO 2000, результаты будут выведены в соответствующие биты слов D201 и D200.





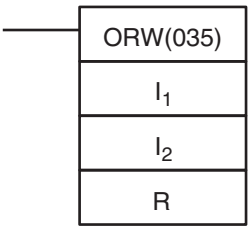
Прим.: Вертикальная стрелка обозначает логическое «И».

3-12-3 ЛОГИЧЕСКОЕ ИЛИ: ORW(035)

Назначение

Операция «логическое ИЛИ» над соответствующими битами одиночных слов данных и/или констант.

Символ РКС



I1: Вход 1

I2: Вход 2

R: Слово результата

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	ORW(035)
	Однократное выполнение по положительному фронту	@ORW(035)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	I_1	I_2	R
Область CIO	CIO 0...CIO 6143		
Рабочая область	W0...W511		
Область битов хранения	H0...H511		
Область вспомогательных битов	A0...A959		A448...A959
Область таймеров	T0000...T4095		
Область счетчиков	C0000...C4095		
Область DM	D0...D32767		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767		
Постоянные	#0000...#FFFF (двоичн.)		---
Регистры данных	DR0...DR15		
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда ORW(035) выполняет операцию «логическое ИЛИ» над данными, указанными в словах I_1 и I_2, и выводит результат в слово R.

- Выполняется побитовое логическое сложение слов I_1 и I_2 .
- Если хотя бы один из соответствующих битов в I_1 и I_2 содержит 1, в соответствующий бит слова R будет выведено значение «1», если оба они содержат 0, в соответствующий бит слова R будет выведено значение «0».

$I_1 + I_2 \rightarrow R$

I_1	I_2	R
1	1	1
1	0	1
0	1	1
0	0	0

Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	ВЫКЛ
Флаг равенства	=	Включен, если результат = 0. Выключен во всех остальных случаях.
Флаг «Меньше нуля»	N	Включен, если старший бит результата (R) = «1». Выключен во всех остальных случаях.

Меры предосторожности

При выполнении ORW(035) сбрасывается флаг ошибки.

Если в результате операции «ИЛИ» содержимое R окажется равным 0000 hex, будет установлен флаг равенства.

Если в результате операции «ИЛИ» старший бит R окажется равным «1», будет установлен флаг отрицательного значения.

3-12-4 ДВОЙНОЕ ЛОГИЧЕСКОЕ ИЛИ: ORWL(611)

Назначение

Операция «логическое ИЛИ» над соответствующими битами двойных слов данных и/или констант.

Символ РКС



Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	ORWL(611)
	Однократное выполнение по положительному фронту	@ORWL(611)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	I_1	I_2	R
Область CIO	CIO 0...CIO 6142		
Рабочая область	W0...W510		
Область битов хранения	H0...H510		
Область вспомогательных битов	A0...A958		A448...A958
Область таймеров	T0000...T4094		
Область счетчиков	C0000...C4094		
Область DM	D0...D32766		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767		
Постоянные	#00000000...#FFFFFFFF (двоичное)		---
Регистры данных	---		
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда ORWL(611) выполняет операцию «логическое ИЛИ» над данными, указанными в словах I_1, I_1+1, I_2 и I_2 +1, как над двойными словами данных, и выводит результат в слова R, R+1.

- Если хотя бы один из соответствующих битов слов I_1, I_1+1, I_2 и I_2 +1 содержит значение «1», в соответствующий бит слов R, R+1 будет выведено значение «1». Если оба соответствующих бита содержат значение «0», в соответствующий бит слов R, R+1 будет выведено значение «0».

(I_1, I_1+1) + (I_2, I_2+1) → (R, R+1)

I_1, I_1+1	I_2, I_2+1	R, R+1
1	1	1
1	0	1
0	1	1
0	0	0

Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	ВЫКЛ
Флаг равенства	=	Включен, если результат = 0. Выключен во всех остальных случаях.
Флаг «Меньше нуля»	N	Включен, если старший бит результата (R) = «1». Выключен во всех остальных случаях.

Меры предосторожности

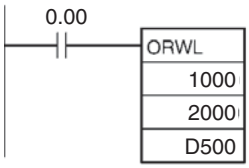
При выполнении команды ORWL(611) сбрасывается флаг ошибки.

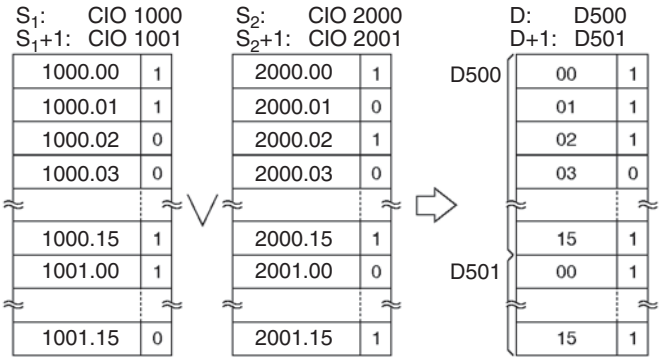
Если в результате операции «ИЛИ» содержимое R, R+1 окажется равным 00000000 hex, будет установлен флаг равенства.

Если в результате операции «ИЛИ» старший бит R+1 окажется равным «1», будет установлен флаг отрицательного значения.

Примеры

Если установлено условие выполнения CIO 0.00, будет выполнено логическое сложение соответствующих битов слов CIO 1001, CIO 1000 и CIO 2001, CIO 2000, результаты будут выведены в соответствующие биты слов D501 и D500.





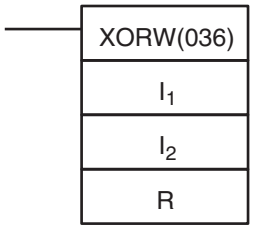
Прим.: Вертикальная стрелка обозначает логическое «ИЛИ».

3-12-5 ИСКЛЮЧАЮЩЕЕ ИЛИ: XORW(036)

Назначение

Операция «исключающее ИЛИ» над соответствующими битами одиночных слов данных и/или констант.

Символ РКС



I1: Вход 1

I2: Вход 2

R: Слово результата

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	XORW(036)
	Однократное выполнение по положительному фронту	@XORW(036)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	I_1	I_2	R
Область CIO	CIO 0...CIO 6143		
Рабочая область	W0...W511		
Область битов хранения	H0...H511		
Область вспомогательных битов	A0...A959		A448...A959
Область таймеров	T0000...T4095		
Область счетчиков	C0000...C4095		
Область DM	D0...D32767		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767		
Постоянные	#0000...#FFFF (двоичн.)		---
Регистры данных	DR0...DR15		
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда XORW(036) выполняет операцию «исключающее ИЛИ» над данными, указанными в словах I_1 и I_2, и выводит результат в слово R.

- Выполняется побитовое сложение по модулю два слов I_1 и I_2 .
- Если значения соответствующих битов слов I_1 и I_2 не равны, в соответствующий бит слова R выводится значение «1», если они равны — в соответствующий бит слова R выводится 0.

$I_1, I_2 + I_1, I_2 \rightarrow R$

I_1	I_2	R
1	1	0
1	0	1
0	1	1
0	0	0

Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	ВЫКЛ
Флаг равенства	=	Включен, если результат = 0. Выключен во всех остальных случаях.
Флаг «Меньше нуля»	N	Включен, если старший бит результата (R) = «1». Выключен во всех остальных случаях.

Меры предосторожности

При выполнении XORW(036) сбрасывается флаг ошибки.

Если в результате операции «ИЛИ» содержимое R окажется равным 0000 hex, будет установлен флаг равенства.

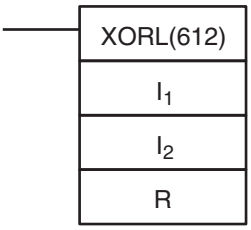
Если в результате операции «ИЛИ» старший бит R окажется равным «1», будет установлен флаг отрицательного значения.

3-12-6 ДВОЙНОЕ ИСКЛЮЧАЮЩЕЕ ИЛИ: XORL(612)

Назначение

Операция «исключающее ИЛИ» над соответствующими битами двойных слов данных и/или констант.

Символ РКС



I_1: Вход 1

I_2: Вход 2

R: Слово результата

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	XORL(612)
	Однократное выполнение по положительному фронту	@XORL(612)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	I_1	I_2	R
Область CIO	CIO 0...CIO 6142		
Рабочая область	W0...W510		
Область битов хранения	H0...H510		

Область	I_1	I_2	R
Область вспомогательных битов	A0...A958		A448...A958
Область таймеров	T0000...T4094		
Область счетчиков	C0000...C4094		
Область DM	D0...D32766		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767		
Постоянные	#00000000...#FFFFFFFF (двоичное)		---
Регистры данных	---		
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда XORL(612) выполняет операцию «исключающее ИЛИ» над данными, указанными в словах I_1, I_1+1, I_2 и I_2 +1, как над двойными словами данных, и выводит результат в слова R, R+1.

- Если значения соответствующих битов слов I_1, I_1+1, I_2 и I_2 +1 не равны, в соответствующий бит слов R, R+1 будет выведено значение «1». Если они равны, в соответствующий бит слов R, R+1 будет выведено значение «0».

$(I_1, I_1+1), \overline{(I_2, I_2+1)} + \overline{(I_1, I_1+1)}, (I_2, I_2+1) \rightarrow (R, R+1)$

I_1, I_1+1	I_2, I_2+1	R, R+1
1	1	0
1	0	1
0	1	1
0	0	0

Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	ВЫКЛ
Флаг равенства	=	Включен, если результат = 0. Выключен во всех остальных случаях.
Флаг «Меньше нуля»	N	Включен, если старший бит результата (R) = «1». Выключен во всех остальных случаях.

Меры предосторожности

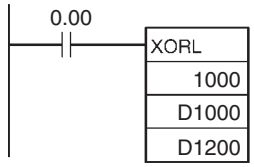
При выполнении команды XORL(612) сбрасывается флаг ошибки.

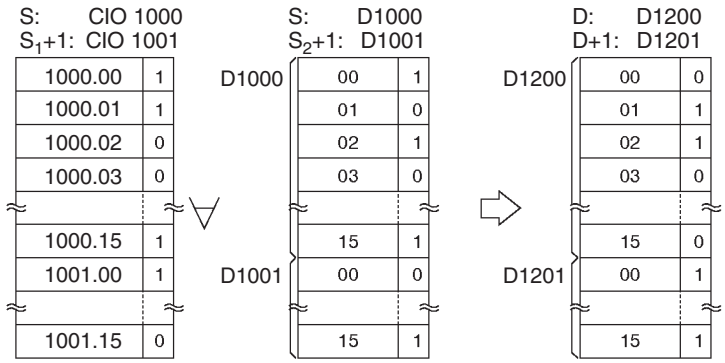
Если в результате операции «исключающее ИЛИ» содержимое R, R+1 окажется равным 00000000 hex, будет установлен флаг равенства.

Если в результате операции «исключающее ИЛИ» старший бит R+1 окажется равным «1», будет установлен флаг отрицательного значения.

Примеры

Если установлено условие выполнения CIO 0.00, будет выполнено сложение по модулю 2 соответствующих битов слов CIO 1001, CIO 1000 и D1001, D1000, результаты будут выведены в соответствующие биты слов D1201 и D1200.





Прим.: Символ обозначает логическое «исключающее ИЛИ».

3-12-7 ИСКЛЮЧАЮЩЕЕ ИЛИ-НЕ: XNRW(037)

Назначение

Операция «исключающее ИЛИ-НЕ» над соответствующими битами одиночных слов данных и/или констант.

Символ РКС



Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	XNRW(037)
	Однократное выполнение по положительному фронту	@XNRW(037)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	I_1	I_2	R
Область CIO	CIO 0...CIO 6143		
Рабочая область	W0...W511		
Область битов хранения	H0...H511		
Область вспомогательных битов	A0...A959		A448...A959
Область таймеров	T0000...T4095		
Область счетчиков	C0000...C4095		
Область DM	D0...D32767		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		

Область	I_1	I_2	R
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767		
Постоянные	#0000...#FFFF (двоичн.)		---
Регистры данных	DR0...DR15		
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда XNRW(037) выполняет операцию «исключающее ИЛИ-НЕ» над данными, указанными в словах I_1 и I_2, и выводит результат в слово R.

- К соответствующим битам слов I_1 и I_2 последовательно применяется операция ИЛИ-НЕ.
- Если значения соответствующих битов слов I_1 и I_2 не равны, в соответствующий бит слова R выводится значение «0», если они равны — в соответствующий бит слова R выводится 1.

$I_1, I_2 + \overline{I_1}, \overline{I_2} \rightarrow R$

I_1	I_2	R
1	1	1
1	0	0
0	1	0
0	0	1

Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	ВЫКЛ
Флаг равенства	=	Включен, если результат = 0. Выключен во всех остальных случаях.
Флаг «Меньше нуля»	N	Включен, если старший бит результата (R) = «1». Выключен во всех остальных случаях.

Меры предосторожности

При выполнении XNRW(037) выключается флаг ошибки.

Если в результате операции «ИЛИ-НЕ» содержимое R окажется равным 0000 hex, будет установлен флаг равенства.

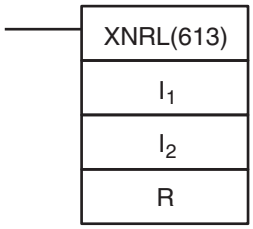
Если в результате операции «ИЛИ-НЕ» старший бит R окажется равным «1», будет установлен флаг отрицательного значения.

3-12-8 ДВОЙНОЕ ИСКЛЮЧАЮЩЕЕ ИЛИ-НЕ: XNRL(613)

Назначение

Операция «исключающее ИЛИ-НЕ» над соответствующими битами двойных слов данных и/или констант.

Символ РКС



I_1: Вход 1

I_2: Вход 2

R: Слово результата

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	XNRL(613)
	Однократное выполнение по положительному фронту	@XNRL(613)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	I_1	I_2	R
Область CIO	CIO 0...CIO 6142		
Рабочая область	W0...W510		
Область битов хранения	H0...H510		
Область вспомогательных битов	A0...A958		A448...A958
Область таймеров	T0000...T4094		
Область счетчиков	C0000...C4094		
Область DM	D0...D32766		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767		
Постоянные	#00000000...#FFFFFFFF (двоичное)		---
Регистры данных	---		
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда XNRL(613) выполняет операцию «исключающее ИЛИ-НЕ» над данными, указанными в словах I_1, I_1+1, I_2 и I_2 +1, и выводит результат в слова R, R+1.

- Если значения соответствующих битов слов I_1, I_1+1, I_2 и I_2 +1 не равны, в соответствующий бит слов R, R+1 будет выведено значение «0». Если они равны, в соответствующий бит слов R, R+1 будет выведено значение «1».

$(I_1, I_1+1), (I_2, I_2+1) + \overline{(I_1, I_1+1)}, \overline{(I_2, I_2+1)} \rightarrow (R, R+1)$

I_1, I_1+1	I_2, I_2+1	R, R+1
1	1	1
1	0	0
0	1	0
0	0	1

Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	ВЫКЛ
Флаг равенства	=	Включен, если результат = 0. Выключен во всех остальных случаях.
Флаг «Меньше нуля»	N	Включен, если старший бит результата (R) = «1». Выключен во всех остальных случаях.

Меры предосторожности

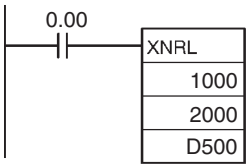
При выполнении команды XNRL(613) выключается флаг ошибки.

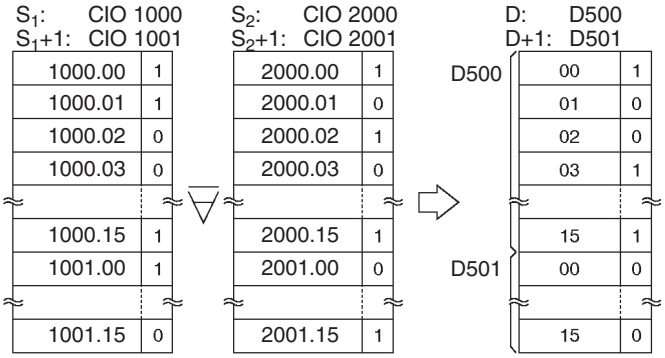
Если в результате операции «исключающее ИЛИ-НЕ» содержимое R, R+1 окажется равным 00000000 hex, будет установлен флаг равенства.

Если в результате операции «исключающее ИЛИ-НЕ» старший бит R+1 окажется равным «1», будет установлен флаг отрицательного значения.

Примеры

Если установлено условие выполнения CIO 0.00, к соответствующим битам слов CIO 1001, CIO 1000 и CIO 2001, CIO 2000 будет применена операция «исключающее ИЛИ-НЕ», результаты будут выведены в соответствующие биты слов D501 и D500.





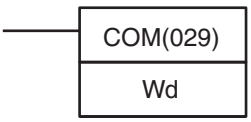
Прим.: Символ обозначает логическое «исключающее НЕ–ИЛИ».

3-12-9 ДОПОЛНЕНИЕ: COM(029)

Назначение

Сброс всех установленных битов и установка всех сброшенных битов в слове Wd.

Символ РКС



Wd: Слово

Варианты выполнения

Варианты выполнения		
	Выполнение в каждом цикле при включенном условии выполнения	COM(029)
	Однократное выполнение по положительному фронту	@COM(029)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	Wd
Область CIO	CIO 0...CIO 6143
Рабочая область	W0...W511
Область битов хранения	H0...H511
Область вспомогательных битов	A448...A959
Область таймеров	T0000...T4095
Область счетчиков	C0000...C4095
Область DM	D0...D32767
Косвенные адреса DM в двоичном формате	@ D0...@ D32767
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767
Постоянные	---
Регистры данных	DR0...DR15
Регистры указателей	---
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15

Описание

Команда COM(029) изменяет состояние каждого бита слова Wd на противоположное.

$\overline{Wd} \rightarrow Wd$: $1 \rightarrow 0$ и $0 \rightarrow 1$

Примечание. Обратите внимание, что при использовании команды COM состояние каждого бита будет изменяться в каждом цикле, в котором будет включено условие выполнения.

Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	ВЫКЛ
Флаг равенства	=	Включен, если результат = 0. Выключен во всех остальных случаях.
Флаг «Меньше нуля»	N	Включен, если старший бит результата (R) = «1». Выключен во всех остальных случаях.

Меры предосторожности

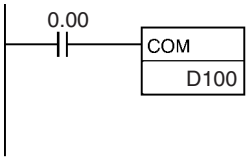
При выполнении COM(029) сбрасывается флаг ошибки.

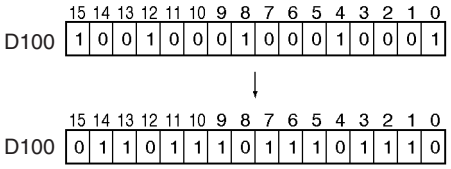
Если в результате операции COM содержимое R окажется равным 0000 hex, будет установлен флаг равенства.

Если в результате операции COM старший бит R окажется равным «1», будет установлен флаг отрицательного значения.

Примеры

Если в следующем примере бит CIO 0.00 включен, состояние каждого бита слова D100 будет изменено на противоположное.





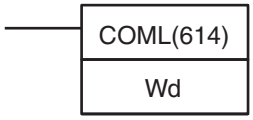
Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

3-12-10 ДВОЙНОЕ ДОПОЛНЕНИЕ: COML(614)

Назначение

Сброс всех установленных битов и установка всех сброшенных битов слов Wd и Wd+1.

Символ РКС



Wd: Слово

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	COML(614)
	Однократное выполнение по положительному фронту	@COML(614)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	Wd
Область CIO	CIO 0...CIO 6142
Рабочая область	W0...W510
Область битов хранения	H0...H510
Область вспомогательных битов	A448...A958
Область таймеров	T0000...T4094
Область счетчиков	C0000...C4094
Область DM	D0...D32766
Косвенные адреса DM в двоичном формате	@ D0...@ D32767
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767
Постоянные	---
Регистры данных	---
Регистры указателей	---
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15

Описание

Команда COML(614) изменяет состояние каждого бита слов Wd и Wd+1 на противоположное.

(Wd+1, Wd)→(Wd+1, Wd)

Примечание. Обратите внимание, что при использовании команды COM состояние каждого бита будет изменяться в каждом цикле, в котором будет включено условие выполнения.

Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	ВЫКЛ
Флаг равенства	=	Включен, если результат = 0. Выключен во всех остальных случаях.
Флаг «Меньше нуля»	N	Включен, если старший бит результата (R) = «1». Выключен во всех остальных случаях.

Меры предосторожности

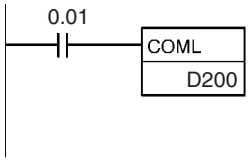
При выполнении команды COML(614) выключается флаг ошибки.

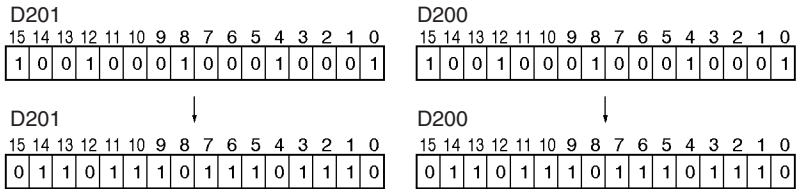
Если в результате операции COML содержимое R, R+1 окажется равным 00000000 hex, будет установлен флаг равенства.

Если в результате операции COML старший бит R+1 окажется равным «1», будет установлен флаг отрицательного значения.

Примеры

Если в следующем примере бит CIO 0.01 включен, состояние каждого бита слов D201 и D200 будет изменено на противоположное.





3-13 Специальные математические команды

В данном разделе описаны команды, используемые для специальных математических вычислений.

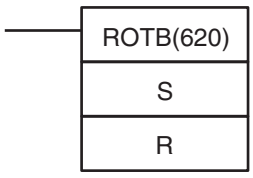
Команда	Мнемоническое обозначение	Код функции	Стр.
КВАДРАТНЫЙ КОРЕНЬ ИЗ ДВОИЧНОГО ЧИСЛА	ROTB	620	493
КВАДРАТНЫЙ КОРЕНЬ ИЗ ДВОИЧНО-ДЕСЯТИЧНОГО ЧИСЛА	ROOT	072	495
МАТЕМАТИЧЕСКАЯ ФУНКЦИЯ	APR	069	499
ДЕЛЕНИЕ С ПЛАВАЮЩЕЙ ЗАПЯТОЙ	FDIV	079	510
ПОДСЧЕТ БИТОВ	BCNT	067	514

3-13-1 КВАДРАТНЫЙ КОРЕНЬ ИЗ ДВОИЧНОГО ЧИСЛА: ROTB(620)

Назначение

Вычисление квадратного корня из 32-битового двоичного содержимого со знаком (положительного значения) указанных слов и запись целой части результата в указанное слово результата.

Символ РКС



S: Первое исходное слово

R: Слово результата

Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	ROTB(620)
	Однократное выполнение по положительному фронту	@ROTB(620)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

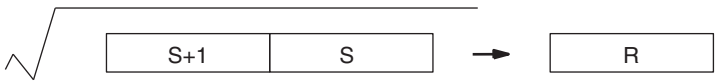
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	S	R
Область CIO	CIO 0...CIO 6142	CIO 0...CIO 6143
Рабочая область	W0...W510	W0...W511
Область битов хранения	H0...H510	H0...H511
Область вспомогательных битов	A0...A958	A448...A959
Область таймеров	T0000...T4094	T0000...T4095
Область счетчиков	C0000...C4094	C0000...C4095
Область DM	D0...D32766	D0...D32767
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767	
Постоянные	#00000000...#FFFFFFFF (двоичное)	---
Регистры данных	DR0...DR15	
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

Описание

Команда ROTB(620) вычисляет квадратный корень 32-битового двоичного числа в словах S+1 и S и выводит целую часть результата в слово R. Дробная часть исключается.



Двоичное значение (32 бита) Двоичное значение (16 бит)

Диапазон допустимых значений слов [S+1, S]: 0000 0000...3FFF FFFF. Если указано число от 4000 0000 до 7FFF FFFF, при расчете квадратного корня оно воспринимается как 3FFF FFFF. Если содержимое входных слов больше 7FFF FFFF, т. е. если бит 15 слова S+1 содержит значение «1», будет сгенерирована ошибка.

Флаги

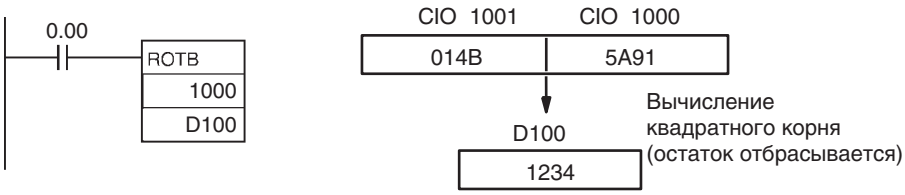
Название	Обозначение	Действие
Флаг ошибки	ER	Включен, если бит 15 слова S+1 содержит «1». Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если результат = 0000. Выключен во всех остальных случаях.
Флаг переполнения	OF	Включен, если содержимое слов [S+1, S] принадлежит диапазону 4000 0000...7FFF FFFF. Выключен во всех остальных случаях.
Флаг потери значимости	UF	ВЫКЛ
Флаг отрицательного значения	N	ВЫКЛ

Меры предосторожности

Содержимое слов [S+1, S] должно быть меньше 8000 0000.
Все операнды данной команды (S+1, S и R) воспринимаются как двоичные значения. Если входные данные указаны в двоично-десятичном формате, воспользуйтесь командой ROOT(072).

Пример

По включению бита CIO 0.00 команда ROTB(620) вычисляет квадратный корень из данных в словах CIO 1001 и CIO 1000 и записывает целую часть результата в слово D100.



3-13-2 КВАДРАТНЫЙ КОРЕНЬ ИЗ ДВОИЧНО-ДЕСЯТИЧНОГО ЧИСЛА: ROOT(072)

Назначение

Вычисление квадратного корня из 8-разрядного двоично-десятичного числа и запись целой части результата в указанное слово результата.

Символ РКС



S: Первое исходное слово

R: Слово результата

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	ROOT(072)
	Однократное выполнение по положительному фронту	@ROOT(072)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

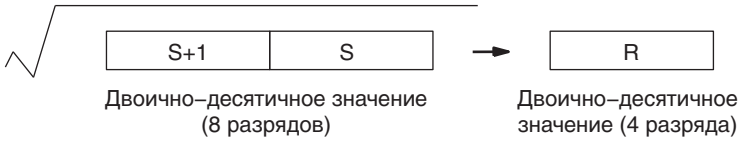
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	S	R
Область CIO	CIO 0...CIO 6142	CIO 0...CIO 6143
Рабочая область	W0...W510	W0...W511
Область битов хранения	H0...H510	H0...H511
Область вспомогательных битов	A0...A958	A448...A959
Область таймеров	T0000...T4094	T0000...T4095
Область счетчиков	C0000...C4094	C0000...C4095
Область DM	D0...D32766	D0...D32767
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767	
Постоянные	#00000000...#99999999 (BCD)	---
Регистры данных	---	DR0...DR15
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

Описание

Команда ROOT(072) вычисляет квадратный корень 8-разрядного двоично-десятичного числа в словах S+1 и S и выводит целую часть результата в слово R. Дробная часть исключается.



Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	Включен, если данные в словах S+1 и S заданы в формате, отличном от BCD. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если результат = 0000. Выключен во всех остальных случаях.

Меры предосторожности

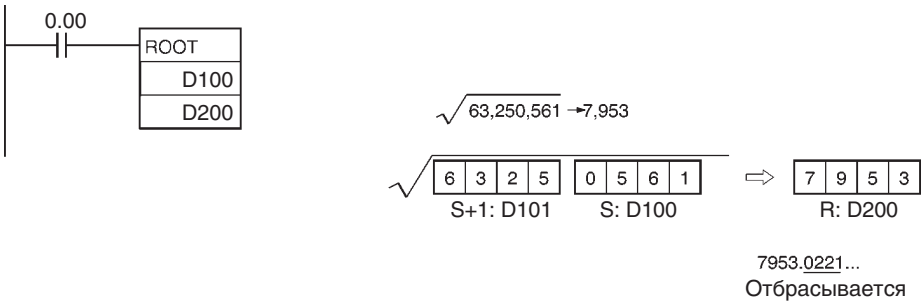
Все операнды данной команды (S+1, S и R) воспринимаются как данные в формате BCD. Если входные данные указаны в двоичном формате, воспользуйтесь командой ROTB(620).

Примеры

Квадратный корень из 8-разрядного числа

Если в следующем примере установлен бит CIO 0.00, команда ROOT(072) вычисляет квадратный корень из данных в словах D101 и D100 и записывает целую часть результата в слово D200.

Примечание. Дробная часть (разряды после запятой) 8-разрядных чисел отбрасывается.



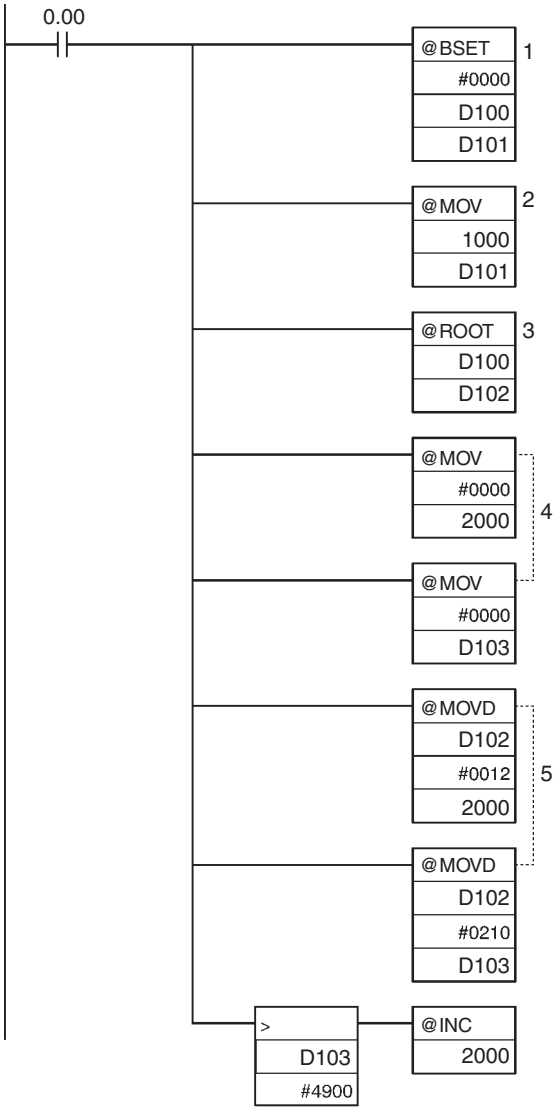
Квадратный корень из 4-разрядного числа

В следующем примере показано извлечение квадратного корня из 4-разрядного числа и округление результата. Приведенная в данном примере программа рассчитывает квадратный корень 4-разрядного числа в слове CIO 1000, округляет результат и записывает его в CIO 2000. (По сути 4-разрядное число умножается на 10000 (100^2), а результат делится на 100, что повышает точность вычислений в 100 раз.)

Примечание. Дробная часть 4-разрядных чисел округляется.

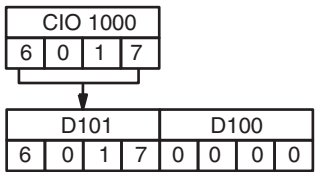
$\sqrt{6017}$=77.56...→78

Значение округляется до целого.

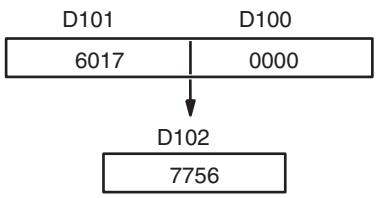


1,2,3... 1. В слова, которые будут использоваться в качестве входных (D101 и D100), записывается значение 0000 0000.

2. Из слова CIO 1000 в слово D101 копируется 4-разрядное значение.



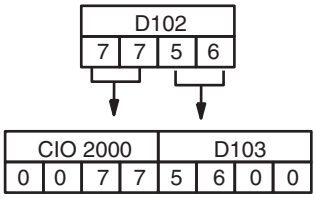
3. Команда ROOT(072) вычисляет квадратный корень из D101 и D100 и записывает результат в слово D102.



$\sqrt{60,170,000}$ = 7, 756,932…

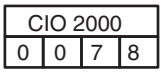
Вычисление квадратного корня (остаток отбрасывается)

4. Слово D103 и слово результата CIO 2000 обнуляются (=0000 0000).
5. Результат расчета квадратного корня делится на 100, целая часть записывается в CIO 2000, остаток записывается в D103.



6. Если содержимое слова D103 больше 4900, CIO 2000 увеличивается на 1. В этом случае результат равен 78.

5600 > 4900

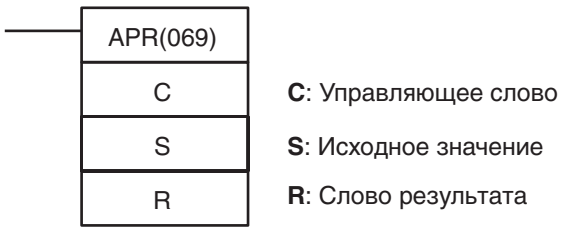


3-13-3 МАТЕМАТИЧЕСКИЕ ФУНКЦИИ: APR(069)

Назначение

Вычисление синуса, косинуса или выполнение линейной экстраполяции входных данных.
Функция линейной экстраполяции позволяет получить линейную аппроксимацию любой зависимости между X и Y.

Символ РКС



Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	APR(069)
	Однократное выполнение по положительному фронту	@APR(069)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

Синус (C = 0000 Hex)

Операнд	Значение	Диапазон значений
C	0000 hex	---
S	0000...0900 (BCD)	0°...90°
D	0000...9999 (BCD)	0,0000...0,9999
	9999 (BCD)	1,0000

Косинус (C = 0001 Hex)

Операнд	Значение	Диапазон значений
C	0001 hex	---
S	0000...0900 (BCD)	0°...90°

Операнд	Значение	Диапазон значений
D	0000...9999 (BCD)	0,0000...0,9999
	9999 (BCD)	1,0000

Линейная экстраполяция (C = адрес области данных)

Операнд	Значение	Диапазон значений
C	Адрес области данных	---
S	16-битовое BCD-значение без знака	0000...9999
	16-битовое двоичное значение без знака	0...65535
	16-битовое двоичное значение со знаком[1]	-32 768...32 767
	32-битовое двоичное значение со знаком[1]	-2 147 483 648...2 147 483 647
	Значение с плавающей запятой[1]	$-\infty$, $-3{,}402823 \times 10^{38}$...$-1{,}175494 \times 10^{-38}$ $1{,}175494 \times 10^{-38}$...$3{,}402823 \times 10^{38}$, $+\infty$
D	16-битовое BCD-значение без знака	0000...9999
	16-битовое двоичное значение без знака	0...65535
	16-битовое двоичное значение со знаком[1]	-32 768...32 767
	32-битовое двоичное значение со знаком[1]	-2 147 483 648...2 147 483 647
	Значение с плавающей запятой[1]	$-\infty$, $-3{,}402823 \times 10^{38}$...$-1{,}175494 \times 10^{-38}$, $1{,}175494 \times 10^{-38}$...$3{,}402823 \times 10^{38}$, $+\infty$

Примечание. Если в C указан адрес слова, команда APR(069) выполняет экстраполяцию значения Y по значению X в S на основании координат (узлов), ранее записанных в таблицу, начинающуюся с C. Более подробная информация приведена в пункте *Описание* ниже.

Характеристики операндов

Область	C	S	R
Область CIO	CIO 0...CIO 6143		
Рабочая область	W0...W511		
Область битов хранения	H0...H511		
Область вспомогательных битов	A0...A959		A448...A959
Область таймеров	T0000...T4095		
Область счетчиков	C0000...C4095		
Область DM	D0...D32767		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767		
Постоянные	Только указанные значения		---
Регистры данных	---	DR0...DR15	

Область	C	S	R
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Работа команды APR(069) зависит от управляющего слова C. Если C содержит значение 0000 или 0001, команда APR(069) вычисляет косинус или синус S, при этом S указывается в десятых долях градуса.

Если в C указан адрес слова, команда APR(069) выполняет экстраполяцию значения Y в точке S на основании координат (узлов), ранее записанных в таблицу, начинающуюся с C.

Синус (C=0000)

Если значение C равно «0000», команда APR(069) вычисляет SIN(S) и записывает результат в R. Диапазон значений S: 0000...0900 BCD (0,0°...90,0°), диапазон значений R: 0000...9999 BCD (0,0000...0,9999). Результат содержит только четыре разряда после десятичной запятой, остальные разряды отбрасываются.

Косинус (C=0001)

Если значение C равно 0001, команда APR(069) вычисляет COS(S) и записывает результат в R. Диапазон значений S: 0000...0900 BCD (0,0°...90,0°), диапазон значений R: 0000...9999 BCD (0,0000...0,9999). Результат содержит только четыре разряда после десятичной запятой, остальные разряды отбрасываются.

Линейная экстраполяция

Чтобы выполнить линейную экстраполяцию с помощью команды APR(069), в C нужно указать адрес слова.

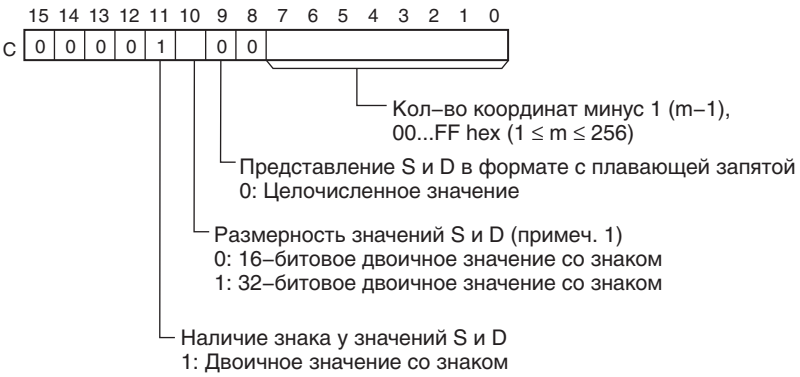
В слове C указывается: количество координат в таблице данных, начиная с C+2; форма входных данных и формат (двоичный или BCD).

Целочисленное значение без знака (двоичное или двоично-десятичное)

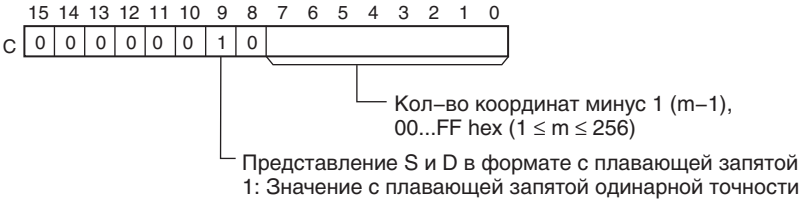


Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

Целочисленное значение со знаком (двоичное)



Значение с плавающей запятой одинарной точности



Если используются 16-битовые двоичные или двоично-десятичные значения, координаты хранятся в словах C+1...C+2m+2. Если используются 32-битовые двоичные значения или значения с плавающей запятой, координаты хранятся в словах C+1...C+4m+4.

Биты 00...07 содержат количество (в двоичном формате) координат узлов минус 1, то есть m-1. Биты 08...12 не используются. Бит 13 указывает, чему равно f(x) (f(S) или f(X_m-S)): «0» — f(x)=f(S), «1» — f(x)=f(X_m-S). Бит 14 указывает на формат выходных данных (двоичный или BCD): «0» — двоичный формат, «1» — BCD-формат. Бит 15 указывает на формат входных данных (двоичный или BCD): «0» — двоичный формат, «1» — BCD-формат.

16-битовое двоичное (со знаком или без) или 16-битовое BCD значение

C+1	X0 (*1)
C+2	Y0
C+3	X1
C+4	Y1
C+5	X2
C+6	Y2
	Xn
	Yn
C+ (2m+1)	Xm
C+ (2m+2)	Ym

Прим.:Если данные ввода/вывода в словах S и D (бит 11 слова C = 0) содержат значения со знаком, в слово C+1 должно быть записано X_m (максимальное значение X в таблице).

32-битовое двоичное значение со знаком

C+1	X0 (16 младших битов)
C+2	X0 (16 старших битов)
C+3	Y0 (16 младших битов)
C+4	Y0 (16 старших битов)
C+5	X1 (16 младших битов)
C+6	X1 (16 старших битов)
C+7	Y1 (16 младших битов)
C+8	Y1 (16 старших битов)
...	...
C+ (4n+1)	Xn (16 младших битов)
C+ (4n+2)	Xn (16 старших битов)
C+ (4n+3)	Yn (16 младших битов)
C+ (4n+4)	Yn (16 старших битов)
...	...
C+ (4m+1)	Xm (16 младших битов)
C+ (4m+2)	Xm (16 старших битов)
C+ (4m+3)	Ym (16 младших битов)
C+ (4m+4)	Ym (16 старших битов)

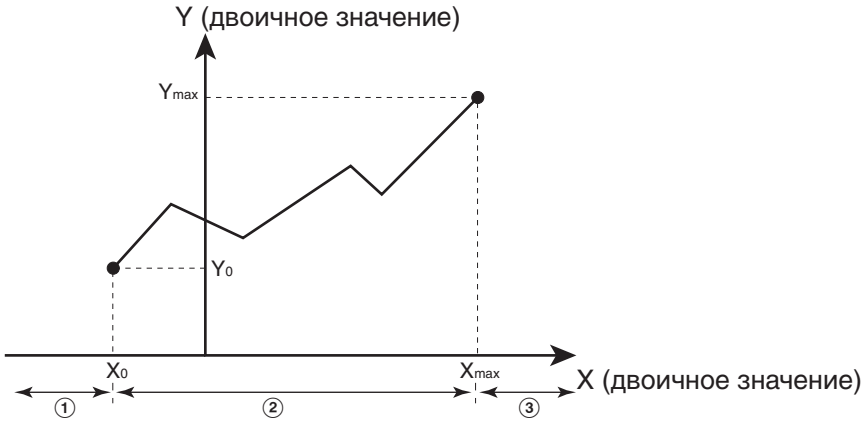
Значение с плавающей запятой

C+1	X0 (16 младших битов)
C+2	X0 (16 старших битов)
C+3	Y0 (16 младших битов)
C+4	Y0 (16 старших битов)
C+5	X1 (16 младших битов)
C+6	X1 (16 старших битов)
C+7	Y1 (16 младших битов)
C+8	Y1 (16 старших битов)
...	...
C+ (4n+1)	Xn (16 младших битов)
C+ (4n+2)	Xn (16 старших битов)
C+ (4n+3)	Yn (16 младших битов)
C+ (4n+4)	Yn (16 старших битов)
...	...
C+ (4m+1)	Xm (16 младших битов)
C+ (4m+2)	Xm (16 старших битов)
C+ (4m+3)	Ym (16 младших битов)
C+ (4m+4)	Ym (16 старших битов)

Примечание. Координаты X должны быть указаны в порядке возрастания:
$X_1 < X_2 < ... < X_m$.
Вводите все значения (X_n, Y_n) в двоичном формате, даже если в управляющем слове C указан другой формат данных.

Работа функции линейной экстраполяции

Команда APR(069) обрабатывает входные данные, указанные в S, на основании следующего выражения и координат (X_n, Y_n), указанных в таблице, начиная с C+1. Результат выводится в слово (слова) назначения, заданное операндом D.



1. Для $S < X_0$
   Преобразованное значение = Y_0
2. Для $X_0 \leq S \leq X_{max}$, если $X_n < S < X_{n+1}$
   Преобразованное значение = $Y_n + [\{Y_n + 1 - Y_n\}/\{X_n + 1 - X_n\}] \times [$*Входные данные* $S - X_n\}$



3. $X_{max} < S$
   Преобразованное значение = Y_{max}

В таблице координат может храниться до 256 узлов, начиная с C+1. Можно использовать следующие пять типов данных ввода/вывода:

- 16-битовое BCD-значение без знака;
- 16-битовое двоичное значение без знака;
- 16-битовое двоичное значение со знаком;

- 32-битовое двоичное значение со знаком;
- значение с плавающей запятой одинарной точности.

Задание формата данных в управляющем слове C

- 16-битовое BCD значение без знака
  Входные данные и/или выходные данные могут быть 16-битовыми BCD значениями без знака. Кроме того, можно выбрать, какими значениями должна оперировать функция линейной экстраполяции: непосредственно значениями операнда S или значениями X_m-S (где X_m — максимальное значение координаты X в таблице координат).

Название параметра	Бит слова C	Настройка
Формат входных значений (S)	15	0: Двоичный 1: BCD
Формат выходных значений (D)	14	0: Двоичный 1: BCD
Форма входных значений	13	0: Оперировать значениями S 1: Оперировать значениями X_m-S
Наличие знака у значений S и D	11	0: Значение без знака
Размерность значений S и D	10	Не действ. (фикс. 16 бит)
Представление с плавающей запятой	09	0: Целочисленное значение

- 16-битовое двоичное значение без знака
  Входные данные и/или выходные данные могут быть 16-битовыми двоичными значениями без знака. Кроме того, можно выбрать, какими значениями должна оперировать функция линейной экстраполяции: непосредственно значениями операнда S или значениями X_m-S (где X_m — максимальное значение координаты X в таблице координат).

Название параметра	Бит слова C	Настройка
Формат входных значений (S)	15	0: Двоичный 1: BCD
Формат выходных значений (D)	14	0: Двоичный 1: BCD
Форма входных значений	13	0: Оперировать значениями S 1: Оперировать значениями X_m-S
Наличие знака у значений S и D	11	0: Значение без знака
Размерность значений S и D	10	Не действ. (фикс. 16 бит)
Представление с плавающей запятой	09	0: Целочисленное значение

- 16-битовое двоичное значение со знаком

Название параметра	Бит слова C	Настройка
Формат входных значений (S)	15	0: Двоичный
Формат выходных значений (D)	14	0: Двоичный
Форма входных значений	13	0
Наличие знака у значений S и D	11	1: Значение со знаком
Размерность значений S и D	10	0: 16-битовое двоичное значение со знаком
Представление с плавающей запятой	09	0: Целочисленное значение

• 32-битовое двоичное значение со знаком

Название параметра	Бит слова C	Настройка
Формат входных значений (S)	15	0: Двоичный
Формат выходных значений (D)	14	0: Двоичный
Форма входных значений	13	0
Наличие знака у значений S и D	11	1: Значение со знаком
Размерность значений S и D	10	1: 32-битовое двоичное значение со знаком
Представление с плавающей запятой	09	0: Целочисленное значение

Примечание. Если параметр «Размерность значений S и D» в бите 10 слова C имеет значение «1» и в качестве S используется 16-битовая константа, перед выполнением линейной экстраполяции входные данные будут преобразованы в 32-битовое двоичное значение со знаком.

• Значение с плавающей запятой

Название параметра	Бит слова C	Настройка
Формат входных значений (S)	15	0: Двоичный
Формат выходных значений (D)	14	0: Двоичный
Форма входных значений	13	0
Наличие знака у значений S и D	11	0
Размерность значений S и D	10	0
Представление с плавающей запятой	09	1: Значение с плавающей запятой

Примечание. Если параметр «Представление с плавающей запятой» в бите 09 слова C имеет значение «1», слово S не может содержать константу.

Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	Включен, если значение C является константой и превышает 0001. Включен, если C является адресом слова, но координаты X указаны не в порядке возрастания ($X_1 \leq X_2 \leq ... \leq X_m$). Включен, если C является адресом слова, в битах 9, 11 и 15 C указано, что входные данные представлены в формате BCD, но значение S указано в формате, отличном от BCD. Включен, если C является адресом слова, но бит 9 слова C указывает, что используются значения с плавающей запятой, а S является константой длиной в одно слово. Включен, если C содержит значение 0000 или 0001, а S не является значением BCD в диапазоне 0000...0900. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если результат = 0000. Выключен во всех остальных случаях.
Флаг отрицательного значения	N	Включен, если установлен бит 15 слова R. Выключен во всех остальных случаях.

Меры предосторожности

На самом деле результатом вычисления SIN(90°) и COS(0°) будет 1, но в слово R выводится значение 9999 (0,9999).

Если значение C является константой и превышает 0001, будет сгенерирована ошибка.

Если задана линейная экстраполяция, но координаты X указаны не в порядке возрастания ($X_1 < X_2 < ... < X_m$), будет сгенерирована ошибка.

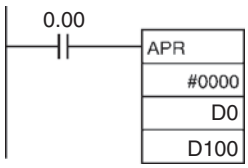
Если задана линейная экстраполяция и в качестве формата входных данных указан формат BCD (установлен бит 15 слова C), но значение S указано в формате, отличном от BCD, будет сгенерирована ошибка.

Если задана тригонометрическая функция (C=0000 или 0001), но значение S не является BCD значением в диапазоне 0000...0900, будет сгенерирована ошибка.

Примеры

Синус (C: #0000)

В следующем примере показано использование функции APR(069) для вычисления синуса 30°.



Исходное значение

S: D0			
0	10^1	10^0	10^{-1}
0	3	0	0

Установка входного значения (в разряде десятков) (0000...0900, BCD)

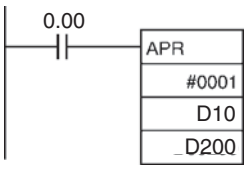
Результат

R: D100			
10^{-1}	10^{-2}	10^{-3}	10^{-4}
5	0	0	0

Значение результата содержит четыре значащих разряда, 5–й и более старшие разряды игнорируются (0000...9999, BCD)

Косинус (C: #0001)

В следующем примере показано использование функции APR(069) для вычисления косинуса 30°.
(SIN(30) = 0,8660)



Исходное значение

S: D10			
0	10^1	10^0	10^{-1}
0	3	0	0

Установка входного значения (в разряде десятков) (0000...0900, BCD)

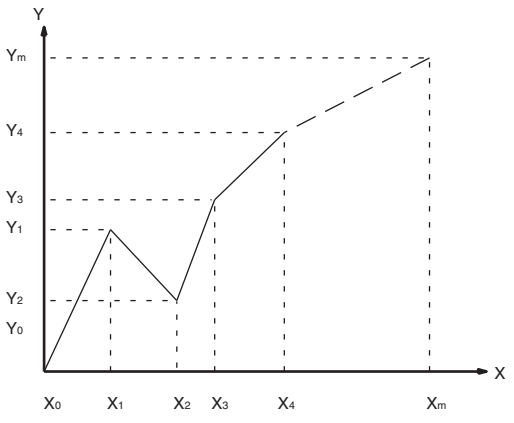
Результат

R: D200			
10^{-1}	10^{-2}	10^{-3}	10^{-4}
8	6	6	0

Значение результата содержит четыре значащих разряда, 5–й и более старшие разряды игнорируются (0000...9999, BCD)

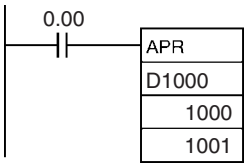
Линейная экстраполяция (C: Адрес слова)
Использование 16-битового двоично-десятичного или двоичного значения без знака

Команда APR(069) обрабатывает входные данные, указанные в S, на основании управляющих данных в слове C и координат, указанных в таблице, начиная с C+1. Результат выводится в слово D.

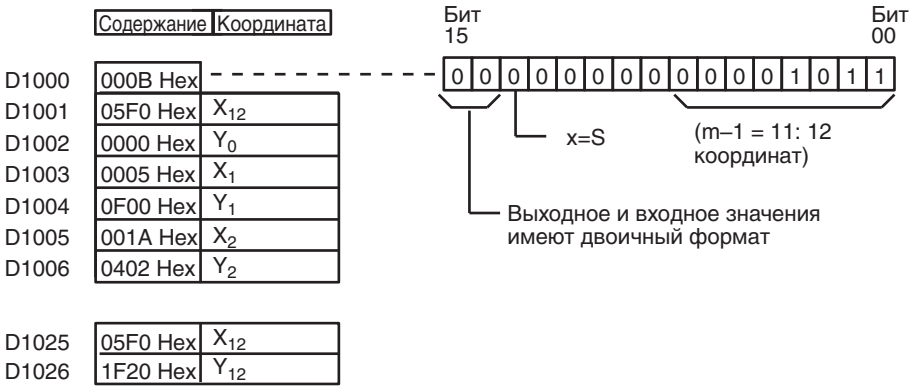


Слово	Координата
C+1	Xm (максимальное значение X)
C+2	Y_0
C+3	X_1
C+4	Y_1
C+5	X_2
C+6	Y_2
↓	↓
C+(2m+1)	X_m (максимальное значение X)
C+(2m+2)	Y_m

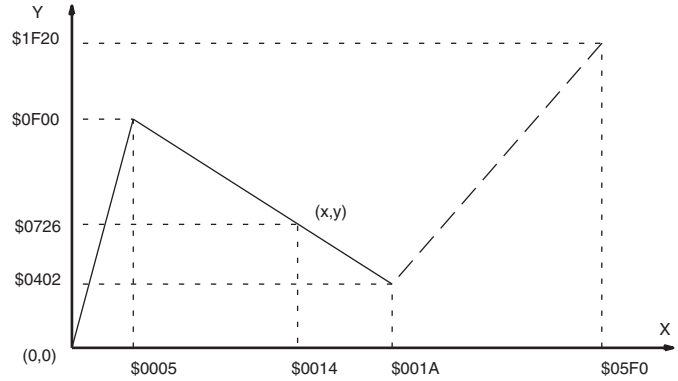
- $Y_n = f(X_n)$, $Y_0 = f(X_0)$
- X_{n-1} всегда должно быть меньше X_n.
- Все значения (X_n, Y_n) следует вводить в двоичном формате.



В данном примере показана реализация линейной экстраполяции по 12 координатам. Задан непрерывный блок данных D1000...D1026 (C...C + (2 × 12 + 2)). Входные данные команда считывает из слова CIO 1000, результат выводится в слово CIO 1001.



В данном случае входное слово CIO 1000 содержит значение 0014, значение f(0014) = 0726 выводится в R, CIO 1001.



Ниже приведен расчет линейной экстраполяции.

$$Y = 0F00 + \frac{0402 - 0F00}{001A - 0005} \times (0014 - 0015)$$

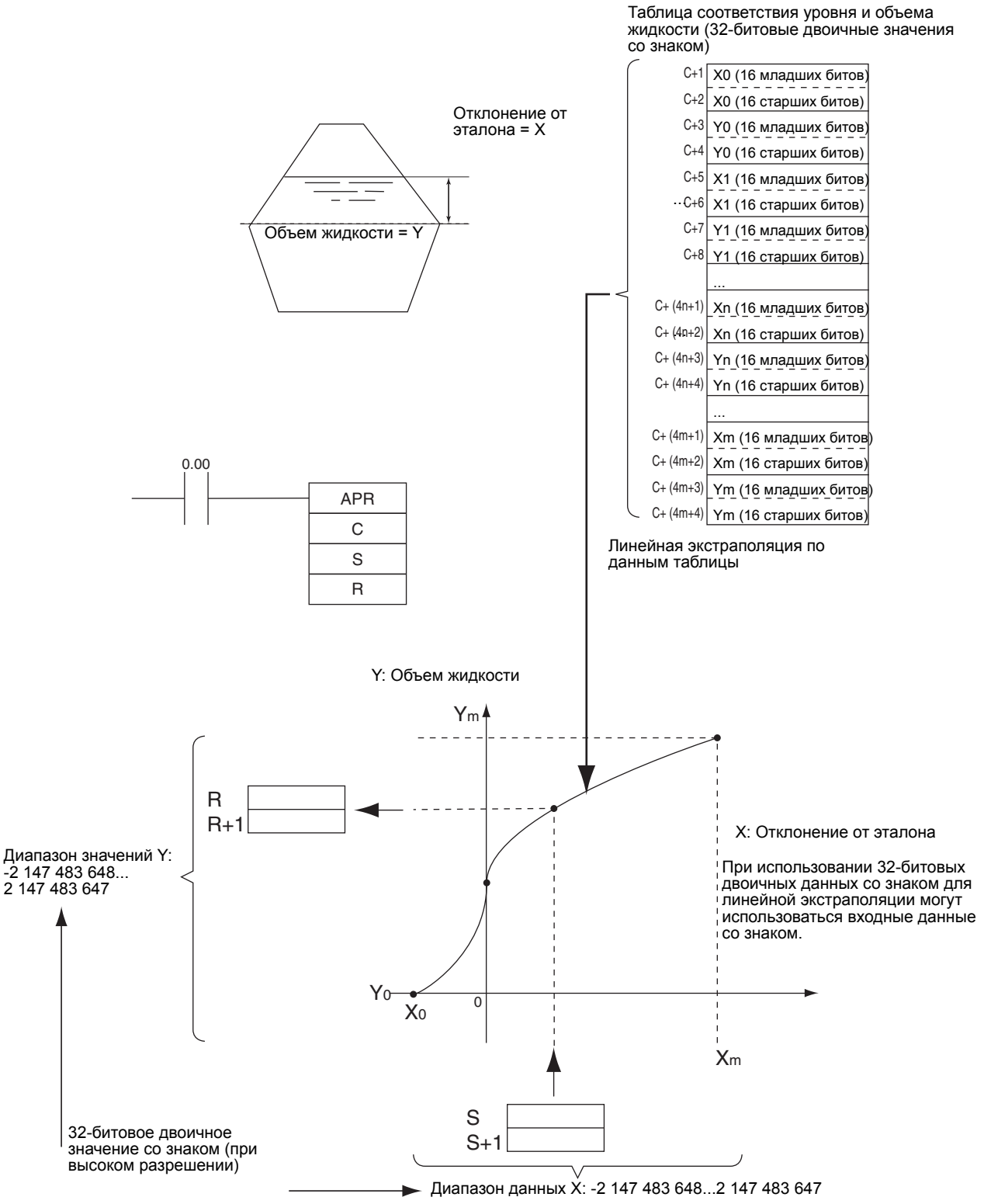
$$= 0F00 - (0086 \times 000F)$$

$$= 0726$$

Все значения являются шестнадцатеричными (hex).

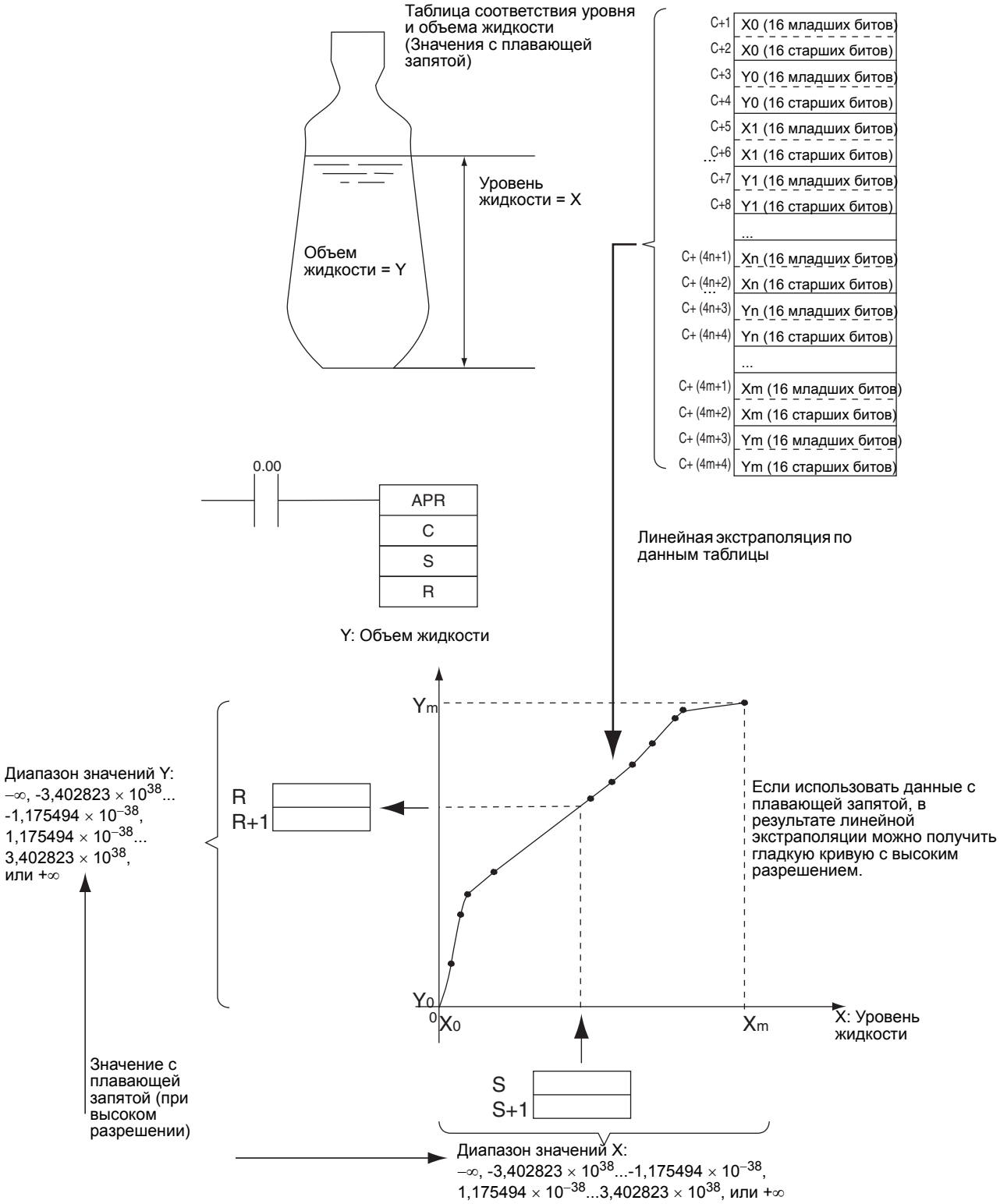
Линейная экстраполяция (C: Адрес слова)
Использование 32-битового двоично-десятичного или двоичного значения со знаком

Ниже приведен пример применения команды APR(069) для расчета объема жидкости в резервуаре на основании известной текущей высоты уровня жидкости и данных о форме резервуара.



Линейная экстраполяция (C: Адрес слова)
Использование данных с плавающей запятой

Ниже приведен пример применения команды APR(069) для расчета объема жидкости в резервуаре на основании известной текущей высоты уровня жидкости и данных о форме резервуара.

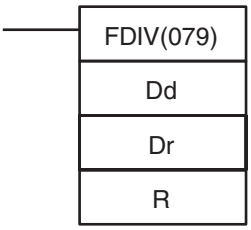


3-13-4 ДЕЛЕНИЕ ЧИСЕЛ С ПЛАВАЮЩЕЙ ЗАПЯТОЙ: FDIV(079)

Назначение

Деление одного 7-разрядного числа с плавающей запятой на другое. Для чисел с плавающей запятой используется экспоненциальное представление (7-разрядная мантисса и 1-разрядный показатель).

Символ РКС



Dd: Первое слово делимого

Dr: Первое слово делителя

R: Первое слово результата

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	FDIV(079)
	Однократное выполнение по положительному фронту	@FDIV(079)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

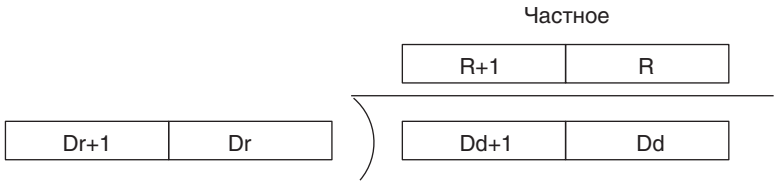
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	Dd	Dr	R
Область CIO	CIO 0...CIO 6142		
Рабочая область	W0...W510		
Область битов хранения	H0...H510		
Область вспомогательных битов	A0...A958		A448...A958
Область таймеров	T0000...T4094		
Область счетчиков	C0000...C4094		
Область DM	D0...D32766		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767		
Постоянные	---		
Регистры данных	---		
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда FDIV(079) делит значение с плавающей запятой в словах Dd и Dd+1 на значение, указанное в словах Dr и Dr+1, и записывает результат в слова R и R+1.



При задании значений с плавающей запятой младшие семь разрядов используются для хранения мантиссы, а старший разряд — для показателя, как показано на схеме ниже. Старший разряд может содержать значение от 0 до F; положительный показатель может иметь значение от 0 до 7, отрицательный показатель может иметь значение от 8 до F (0...-7). Младшие 7 разрядов должны содержать BCD значение.



Ниже приведены еще два примера значений с плавающей запятой:

6123 4567: $0{,}1234567 \times 10^{6}$ (6 = 0110 двоичн.)

B123 4567: $0{,}1234567 \times 10^{-3}$ (B = 1011 двоичн.)

В следующей таблице приведены максимальные и минимальные допустимые значения.

Предел	8-разр. шестнадцатеричное	С плавающей запятой
Максимальное значение	7999 9999	$0{,}9999999 \times 10^{7}$
Минимальное значение (делимое и делитель)	F000 0001	$0{,}0000001 \times 10^{-7}$
Минимальное значение (результат)	F100 0000	$0{,}1000000 \times 10^{-7}$

Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	Включен, если мантисса (младшие 7 разрядов) в словах Dd+1 и Dd содержит значение в формате, отличном от BCD. Включен, если мантисса (младшие 7 разрядов) в словах Dr+1 и Dr содержит значение в формате, отличном от BCD. Включен, если делитель (Dr+1 и Dr) равен 0. Включен, если результат не принадлежит диапазону $0{,}1000000 \times 10^{-7}$...$0{,}9999999 \times 10^{7}$. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если результат = 0. Выключен во всех остальных случаях.

Меры предосторожности

Результат выводится в формате значения с плавающей запятой, т. е. содержит 7 значащих разрядов. Восьмой разряд и все последующие опускаются.

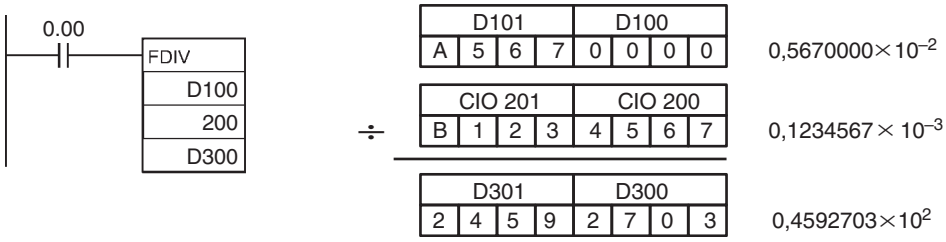
Результат должен принадлежать диапазону

$0{,}1000000 \times 10^{-7}$...$0{,}9999999 \times 10^{7}$.

Примеры

Обычное деление чисел с плавающей запятой

Если в следующем примере установлен бит CIO 0.00, команда FDIV(079) делит число с плавающей запятой в словах D101 и D100 на число с плавающей запятой в словах CIO 201 и CIO 200 и записывает результат в слова D301 и D300.

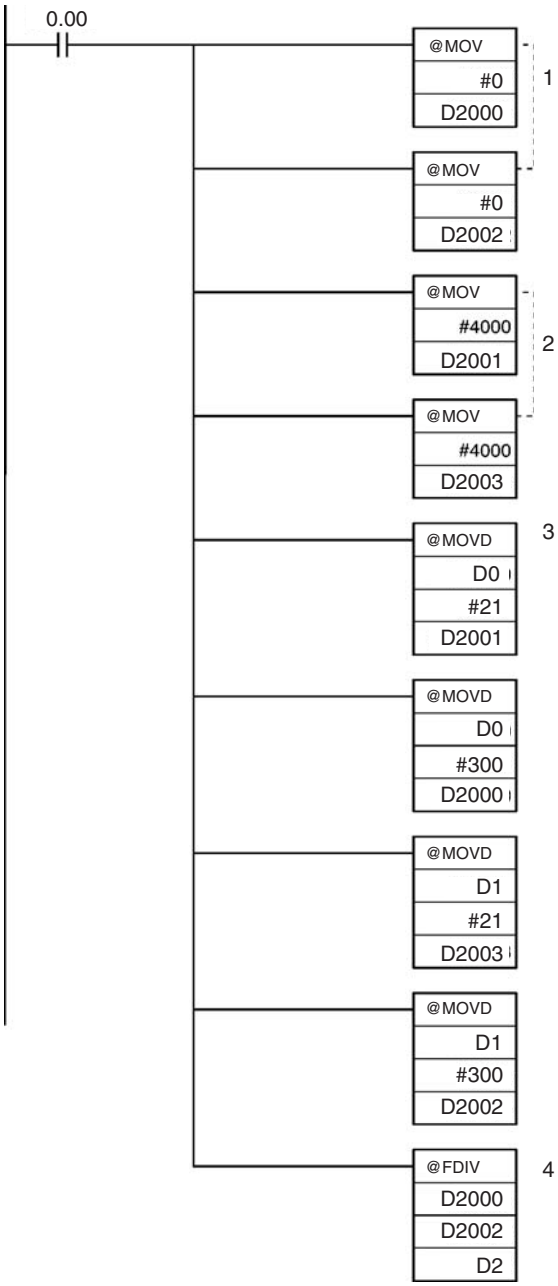


Предварительная версия
Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

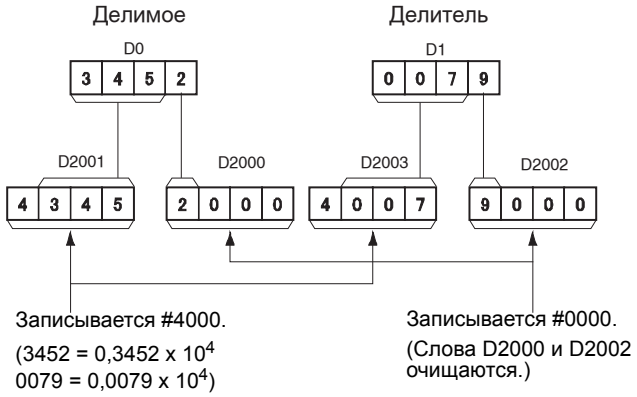
Деление двух BCD значений как чисел с плавающей запятой

В данном примере 4-разрядное BCD значение в D0 делится на 4-разрядное BCD значение в D1 и результат с плавающей запятой записывается в слова D2 и D3.

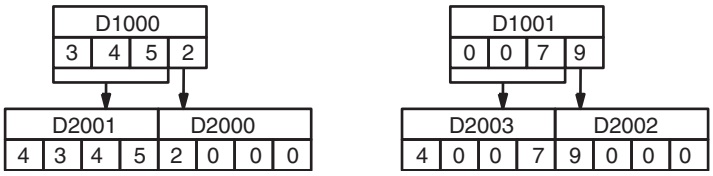
Для выполнения деления с плавающей запятой BCD значение в D0 преобразуется в формат числа с плавающей запятой и записывается в слова D2001 и D2000, а значение BCD в слове D1 преобразуется в формат числа с плавающей запятой и записывается в слова D2003 и D2002.



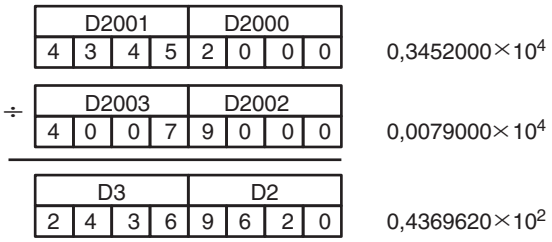
1,2,3... 1. В слова D2000 и D2002 записывается значение 0000, а в слова D2001 и D2003 — значение 4000.



2. Команда MOVD(083) используется для записи разрядов исходных входных слов в нужные разряды значения с плавающей запятой длиной в два слова.



3. Команда FDIV(079) делит число с плавающей запятой в словах D2001 и D2000 на значение с плавающей запятой в словах D2003 и D2002.

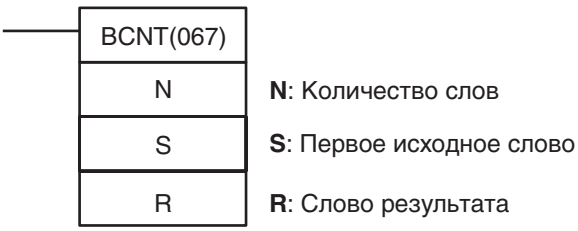


3-13-5 ПОДСЧЕТ БИТОВ: BCNT(067)

Назначение

Определяет общее количество включенных битов в указанном слове или словах.

Символ РКС



Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	BCNT(067)
	Однократное выполнение по положительному фронту	@BCNT(067)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

N: Количество слов

Количество слов должно принадлежать диапазону 0001...FFFF (1...65 535 слов).

S: Первое исходное слово

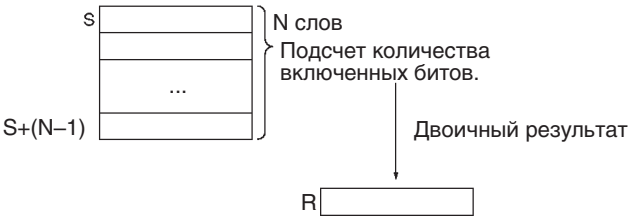
S и S+(N-1) должны принадлежать одной области.

Характеристики операндов

Область	N	S	R
Область CIO	CIO 0...CIO 6143		
Рабочая область	W0...W511		
Область битов хранения	H0...H511		
Область вспомогательных битов	A0...A959		A448...A959
Область таймеров	T0000...T4095		
Область счетчиков	C0000...C4095		
Область DM	D0...D32767		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767		
Постоянные	#0001...#FFFF (двоичный) или &1...&65 535	---	
Регистры данных	DR0...DR15	---	DR0...DR15
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда BCNT(067) подсчитывает суммарное количество включенных битов во всех словах диапазона S...S+(N-1) и записывает результат в операнд R.

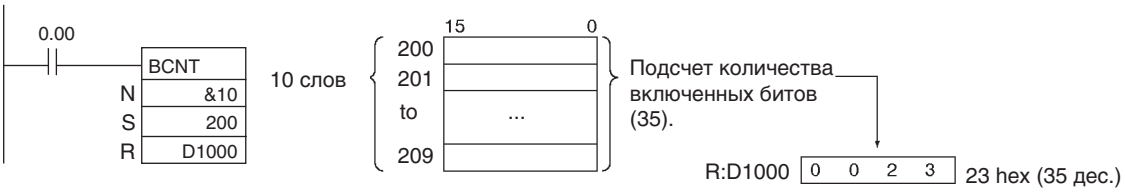


Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	Включен, если N равно «0000». Включен, если результат превышает FFFF. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если результат = 0000. Выключен во всех остальных случаях.

Меры предосторожности Если N=0000 или результат превышает FFFF, будет сгенерирована ошибка.

Пример Если в следующем примере установлен бит CIO 0.00, команда BCNT(067) подсчитывает суммарное количество включенных битов в 10 словах диапазона CIO 200...CIO 209 и записывает результат в слово D1000.



3-14 Команды математических операций над числами с плавающей запятой

Команды математических операций преобразуют значения и выполняют арифметические операции над числами с плавающей запятой.

Команда	Мнемоническое обозначение	Код функции	Стр.
ПЛАВ. ЗАПЯТАЯ В 16-БИТ	FIX	450	522
ПЛАВ. ЗАПЯТАЯ В 32-БИТ	FIXL	451	524
16-БИТ В ПЛАВ. ЗАПЯТАЯ	FLT	452	525
32-БИТ В ПЛАВ. ЗАПЯТАЯ	FLTL	453	527
СЛОЖЕНИЕ С ПЛАВ. ЗАПЯТОЙ	+F	454	528
ВЫЧИТАНИЕ С ПЛАВ. ЗАПЯТОЙ	–F	455	530
УМНОЖЕНИЕ С ПЛАВ. ЗАПЯТОЙ	∗F	456	532
ДЕЛЕНИЕ С ПЛАВ. ЗАПЯТОЙ	/F	457	534
ГРАДУСЫ В РАДИАНЫ	RAD	458	536
РАДИАНЫ В ГРАДУСЫ	DEG	459	538
СИНУС	SIN	460	540
КОСИНУС	COS	461	542
ТАНГЕНС	TAN	462	543
АРКСИНУС	ASIN	463	545
АРККОСИНУС	ACOS	464	547
АРКТАНГЕНС	ATAN	465	549
КВАДРАТНЫЙ КОРЕНЬ	SQRT	466	551
ЭКСПОНЕНТА	EXP	467	553
ЛОГАРИФМ	LOG	468	555
ВОЗВЕДЕНИЕ В СТЕПЕНЬ	PWR	840	557

Подробную информацию о командах для чисел с плавающей запятой двойной точности см. в разделе *3-15-21 Входные команды для чисел с плавающей запятой двойной точности*.

Команда	Мнемоническое обозначение	Код функции	Стр.
Символьные команды сравнения чисел с плавающей запятой одинарной точности	LD, AND, OR + =F, <>F, <F, <=F, >F или >=F	329...334	559
ПЛАВ. ЗАПЯТАЯ В ASCII	FSTR	448	563
ASCII В ПЛАВ. ЗАПЯТАЯ	FVAL	449	568

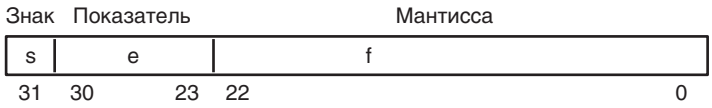
Формат данных

Для представления вещественного числа в формате с плавающей запятой используются знак, показатель степени и мантисса. Для представления вещественного значения в формате с плавающей запятой используется следующая формула.

Вещественное число = $(-1)^s \, 2^{e-127} \, (1.f)$

s: знак
e: показатель степени
f: мантисса

Формат представления чисел с плавающей запятой описан в стандарте IEEE754. Значение в формате с плавающей запятой представляется с помощью 32 битов следующим образом:



Данные	Кол-во битов	Содержание
s: знак	1	0: «+»; 1: «-»
e: показатель степени	8	Показатель степени (e) может принимать значения от 0 до 255. Фактическое значение показателя рассчитывается путем вычитания 127 из e и лежит в диапазоне -127...128. При этом значениям «e=0» и «e=255» соответствуют особые состояния.
f: мантисса	23	Мантисса двоичного значения с плавающей запятой удовлетворяет неравенству 2,0 > 1, f ≥1,0.

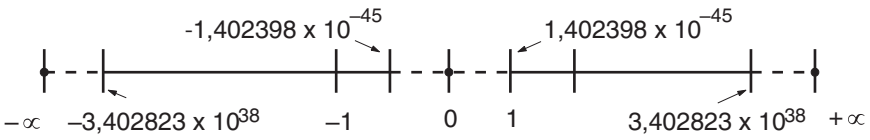
Количество разрядов

Количество значимых разрядов двоичного значения с плавающей запятой ограничивается 24 битами (около 7 десятичных знаков).

Значение с плавающей запятой

В формате с плавающей запятой могут быть выражены следующие значения:

- $-\infty$
- $-3{,}402823 \times 10^{38} \leq$ значение $\leq -1{,}402398 \times 10^{-45}$
- 0
- $1{,}402398 \times 10^{-45} \leq$ значение $\leq 3{,}402823 \times 10^{38}$
- $+\infty$
- NaN («не число»)



Особые состояния

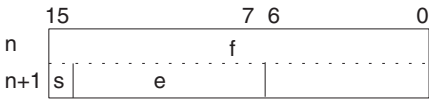
Для особых состояний, которыми являются NaN, $\pm\infty$ и 0, используются следующие форматы записи:

NaN*: e = 255, f $\neq$ 0
+∞: e = 255, f = 0, s= 0
–∞: e = 255, f = 0, s= 1
0: e = 0

*NaN («не число») не является каким-либо числом с плавающей запятой. Возможность получения значения NaN в результате выполнения команд вычислений с плавающей запятой исключена.

Запись данных с плавающей запятой

Если в окне редактирования памяти ввода/вывода CX-Programmer выбран формат с плавающей запятой, числа, введенные в стандартном десятичном формате, автоматически приводятся к описанному выше формату с плавающей запятой (в соответствии со стандартом IEEE754) и записываются в память ввода/вывода. Данные, записанные в формате IEEE754, для отображения на дисплее автоматически приводятся к стандартному десятичному формату.



Для чтения и записи значений с плавающей запятой пользователю не обязательно разбираться в тонкостях формата IEEE754. Достаточно помнить, что каждое значение с плавающей запятой занимает два слова.

Типы чисел в формате с плавающей запятой

Могут использоваться следующие типы чисел с плавающей запятой.

Мантисса (f)	Показатель (e)		
	0	Не «0» и не все «1»	Все «1» (255)
0	0	Нормализованное число	Бесконечность
Не «0»	Ненормализованное число		NaN

Примечание. Ненормализованным называется число, абсолютное значение которого настолько мало, что не может быть выражено как нормализованное число. Ненормализованные числа содержат меньшее количество значащих разрядов. Если результатом вычисления оказывается ненормализованное число (включая промежуточные результаты), количество значащих разрядов числа сокращается.

Нормализованные числа

Нормализованные числа используются для представления вещественных чисел. Для положительного числа бит знака равен «0», для отрицательного числа бит знака равен «1».

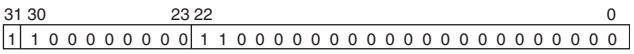
Показатель степени (e) может принимать значения от 1 до 254, при этом фактическое значение показателя на 127 меньше, т. е. изменяется от –126 до 127.

Мантисса (f) может принимать значения от 0 до 2^{33} - 1, при этом предполагается, что у фактической мантиссы бит 2^{33} = 1 и двоичная запятая располагается сразу после этого бита.

Для нормализованных чисел используется следующая форма представления:

$(-1)^{(\text{знак s})} \times 2^{(\text{показатель e}) - 127} \times (1 + \text{мантисса} \times 2^{-23})$

Пример



Знак: -
Показатель: 128 - 127 = 1
Мантисса: $1 + (2^{22} + 2^{21}) \times 2^{-23} = 1 + (2^{-1} + 2^{-2}) = 1 + 0,75 = 1,75$
Значение: $-1,75 \times 2^{1} = -3,5$

Ненормализованные числа

Ненормализованными числами выражаются вещественные числа, имеющие очень маленькое абсолютное значение. Для положительного числа бит знака равен «0», для отрицательного числа бит знака равен «1».

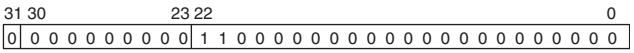
Показатель степени (e) равен «0», т. е. фактическое значение показателя: -126.

Мантисса (f) может принимать значения от 1 до $2^{33} - 1$, при этом предполагается, что у фактической мантиссы бит $2^{33} = 0$ и двоичная запятая располагается сразу после этого бита.

Для ненормализованных чисел используется следующая форма представления:

$(-1)^{(\text{знак s})} \times 2^{-126} \times (\text{мантисса} \times 2^{-23})$

Пример



Знак: -
Показатель: -126
Мантисса: $0 + (2^{22} + 2^{21}) \times 2^{-23} = 0 + (2^{-1} + 2^{-2}) = 0 + 0,75 = 0,75$
Значение: $-0,75 \times 2^{-126}$

Нуль

Значения +0,0 и -0,0 могут быть выражены путем записи «0» в бит знака для положительного нуля и «1» — для отрицательного. Показатель степени и мантисса будут одновременно равны «0». Числа +0,0 и -0,0 соответствуют 0,0. Информацию о том, на что влияет знак нуля, смотрите в разделе *Результаты арифметических операций с плавающей запятой* ниже.

Бесконечность

Значения $+\infty$ и $-\infty$ могут быть выражены путем записи «0» в бит знака для положительной бесконечной величины и «1» — для отрицательной. Показатель степени равен 255 ($2^8 - 1$), мантисса равна «0».

NaN

Состояние NaN («не число») создается, когда результат вычислений, например 0,0/0,0, ∞/∞ или $\infty-\infty$, не является действительным числом или бесконечностью. Показатель степени равен 255 ($2^8 - 1$), мантисса не равна «0».

Примечание. Предписаний в отношение знака NaN или значения поля мантиссы не существует (за исключением того, что мантисса не должна быть равна «0»).

Результаты арифметических операций с плавающей запятой

Округление результатов

Если операция над числами с плавающей запятой дает результат, для точного представления которого требуется больше значащих разрядов, чем допускают внутренние вычислительные операции, результат округляется по одному из следующих правил.

Если результат близок к какому-то из двух внутренних выражений для числа с плавающей запятой, используется это выражение. Если результат находится посередине между двумя внутренними

выражениями числа с плавающей запятой, он округляется таким образом, чтобы последний разряд мантиссы содержал «0».

Переполнение, потеря значимости и недопустимые операции

Если происходит переполнение, в качестве результата выдается положительная или отрицательная бесконечность, в зависимости от знака результата. Если происходит потеря значимости (отрицательное переполнение), в качестве результата выдается положительный или отрицательный ноль, в зависимости от знака результата.

Результатом недопустимой операции является состояние NaN («не число»). К недопустимым операциям относятся: сложение бесконечности с числом противоположного знака, вычитание бесконечности из числа с противоположным знаком, умножение ноля на бесконечность, деление ноля на ноль или деление бесконечности на бесконечность.

Если переполнение происходит при преобразовании числа из формата с плавающей запятой в целочисленный формат, результат преобразования может быть неверным.

Правила выполнения операций над специальными значениями

При выполнении операций над специальными значениями (ноль, бесконечность и NaN) применяются следующие правила.

- Сложение положительного ноля с отрицательным нолем дает положительный ноль.
- Вычитание ноля из ноля с таким же знаком дает положительный ноль.
- Если любым из операндов является NaN, результатом также будет NaN.
- В операциях сравнения положительный ноль и отрицательный ноль считаются равными друг другу.
- Результатом операции сравнения или определения равенства над одним или несколькими состояниями NaN всегда является «истина» для команды != и «ложь» для всех остальных команд.

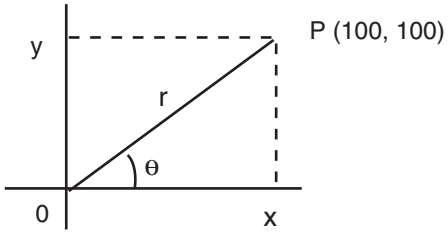
Результаты вычислений с плавающей запятой

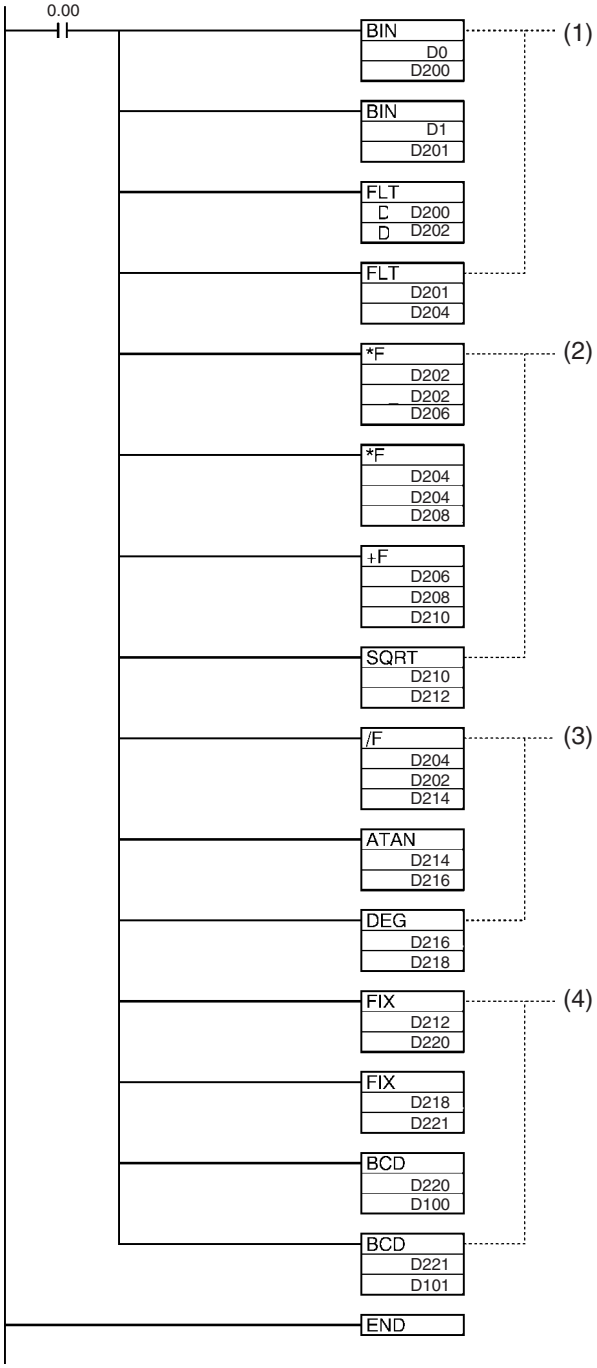
Если результат по модулю превосходит максимальное значение, которое может быть представлено в формате с плавающей запятой, включается флаг переполнения и в качестве результата выдается $\pm\infty$. При положительном знаке результата выдается $+\infty$; при отрицательном выдается $-\infty$.

Флаг равенства устанавливается, только если в конце вычислений и показатель (e), и мантисса (f) равны нулю. В качестве результата вычислений также будет выдан ноль, если абсолютное значение результата меньше минимального значения, которое может быть выражено числом с плавающей запятой. В этом случае устанавливается флаг потери значимости.

Пример

В данном примере программы координаты по осям X и Y (x, y) заданы 4-разрядными двоично-десятичными значениями в словах D0 и D1. Программа рассчитывает расстояние от начала координат (r) и угол (θ, заданный в градусах) и записывает полученные значения в D100 и D101. В результате отбрасываются все цифры справа от десятичной запятой.





Расчет

Расстояние $r = \sqrt{x^2 + y^2}$

Угол $= \tan^{-1}\left(\frac{y}{x}\right)$

Пример

Расстояние $r = \sqrt{100^2 + 100^2} = 141.4214$

Угол $= \tan^{-1}\left(\frac{100}{100}\right) \times \left(\frac{180}{\pi}\right) = 45.0$

Содержание DM

D0 [#0100] (BCD) x → D100 [0 1 4 1] (BCD) r

D1 [#0100] (BCD) y → D101 [0 0 4 5] (BCD)

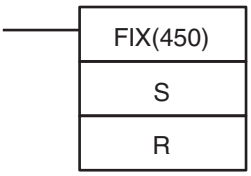
1. Этот сегмент программы приводит двоично-десятичные значения к формату с плавающей запятой.
   a. В качестве рабочей области используется область данных, начинающаяся со слова D200.
   b. Сначала используется команда BIN(023) для временного преобразования двоично-десятичных значений в двоичные, затем команда FLT(452), приводящая двоичные значения к формату с плавающей запятой.
   c. Значение x, приведенное к формату с плавающей запятой, выводится в D203 и D202.
   d. Значение y, приведенное к формату с плавающей запятой, выводится в D205 и D204.
2. При расчете расстояния r команды математических операций с плавающей запятой используются для вычисления квадратного корня из x^2+y^2. Результат записывается в D213 и D212 в формате с плавающей запятой.
3. При расчете угла θ команды математических операций используются для вычисления $\tan^{-1}$ (y/x). Команда ATAN(465) выводит результат в радианах, поэтому используется команда DEG(459) для преобразования результата в градусы. Результат записывается в D219 и D218 в формате с плавающей запятой.
4. Значения с плавающей запятой преобразуются обратно в двоично-десятичный формат.
   a. Сначала используется команда FIX(450) для временного преобразования значений с плавающей запятой в двоичный формат, а затем команда BCD(024), приводящая двоичные значения к двоично-десятичному формату.
   b. Расстояние r выводится в D100.
   c. Угол θ выводится в D101.

3-14-1 ПЛАВ. ЗАПЯТАЯ В 16-БИТ: FIX(450)

Назначение

Преобразование 32-битового значения с плавающей запятой в 16-битовое двоичное значение со знаком и запись результата в указанное слово результата.

Символ РКС



S: Первое исходное слово

R: Слово результата

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	FIX(450)
	Однократное выполнение по положительному фронту	@FIX(450)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

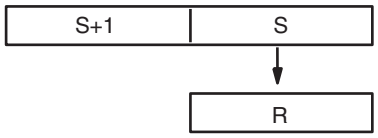
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	S	R
Область CIO	CIO 0...CIO 6142	CIO 0...CIO 6143
Рабочая область	W0...W510	W0...W511
Область битов хранения	H0...H510	H0...H511
Область вспомогательных битов	A0...A958	A448...A959
Область таймеров	T0000...T4094	T0000...T4095
Область счетчиков	C0000...C4094	C0000...C4095
Область DM	D0...D32766	D0...D32767
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Постоянные	#00000000...#FFFFFFFF (двоичное)	---
Регистры данных	---	DR0...DR15
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

Описание

Команда FIX(450) преобразует целую часть 32-битового значения с плавающей запятой в [S+1, S] (в формате IEEE754) в 16-битовое двоичное значение со знаком и записывает результат в R.



Значение с плавающей запятой (32 бит)

Двоичное значение со знаком (16 бит)

Преобразуется только целая часть числа с плавающей запятой, дробная часть отбрасывается. Целая часть числа с плавающей запятой должна находиться в диапазоне от -32768 до 32767.

Примеры преобразования:
Значение с плавающей запятой 3,5 преобразуется в 3.
Значение с плавающей запятой -3,5 преобразуется в -3.

Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если содержимое [S+1, S] не является числом (NaN). Включен, если целая часть значения [S+1, S] находится за пределами диапазона -32 768...32 767. Выключен во всех остальных случаях.

Название	Обозначение	Описание работы
Флаг равенства	=	Включен, если результат = 0000. Выключен во всех остальных случаях.
Флаг отрицательного значения	N	Включен, если бит 15 результата = «1». Выключен во всех остальных случаях.

Меры предосторожности

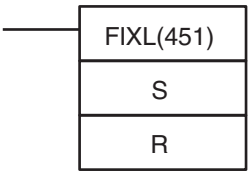
Операнд [S+1, S] должен содержать значение с плавающей запятой, целая часть которого принадлежит диапазону -32 768...32 767.

3-14-2 ПЛАВ. ЗАПЯТАЯ В 32-БИТ: FIXL(451)

Назначение

Преобразование 32-битового значения с плавающей запятой в 32-битовое двоичное значение со знаком и запись результата в указанные слова.

Символ РКС



S: Первое исходное слово

R: Первое слово результата

Варианты выполнения

Варианты выполнения		
	Выполнение в каждом цикле при включенном условии выполнения	FIXL(451)
	Однократное выполнение по положительному фронту	@FIXL(451)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

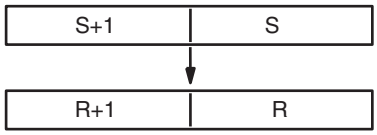
Характеристики операндов

Область	S	R
Область CIO	CIO 0...CIO 6142	
Рабочая область	W0...W510	
Область битов хранения	H0...H510	
Область вспомогательных битов	A0...A958	A448...A958
Область таймеров	T0000...T4094	
Область счетчиков	C0000...C4094	
Область DM	D0...D32766	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Постоянные	#00000000...#FFFFFFFF (двоичное)	---
Регистры данных	---	

Область	S	R
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –()IR15	

Описание

Команда FIXL(451) преобразует целую часть 32-битового значения с плавающей запятой [S+1, S] (в формате IEEE754) в 32-битовое двоичное значение со знаком и записывает результат в [R+1, R].



Значение с плавающей запятой (32 бит)

Двоичное значение со знаком (32 бит)

Преобразуется только целая часть числа с плавающей запятой, дробная часть отбрасывается. (Целая часть числа с плавающей запятой должна находиться в диапазоне от -2 147 483 648 до 2 147 483 647.)

Примеры преобразования:
Значение с плавающей запятой 2 147 483 640,5 преобразуется в 2 147 483 640.
Значение с плавающей запятой -214 748 340,5 преобразуется в -214 748 340.

Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если содержимое [S+1, S] не является числом (NaN). Включен, если целая часть [S+1, S] находится за пределами диапазона -2 147 483 648...2 147 483 647. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если результат = 0000 0000. Выключен во всех остальных случаях.
Флаг отрицательного значения	N	Включен, если бит 15 в слове R+1 находится в состоянии «1» после выполнения команды. Выключен во всех остальных случаях.

Меры предосторожности

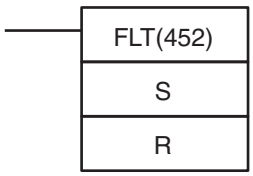
Содержимое [S+1, S] должно быть значением с плавающей запятой, целая часть которого принадлежит диапазону
-2 147 483 648...2 147 483 647.

3-14-3 16-БИТ В ПЛАВ. ЗАПЯТАЯ: FLT(452)

Назначение

Преобразование 16-битового двоичного значения со знаком в эквивалентное 32-битовое значение с плавающей запятой и запись результата в указанные слова.

Символ РКС



S: Исходное слово

R: Первое слово результата

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	FLT(452)
	Однократное выполнение по положительному фронту	@FLT(452)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

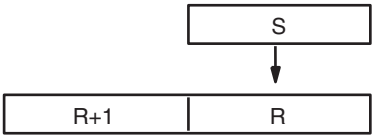
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	S	R
Область CIO	CIO 0...CIO 6143	CIO 0...CIO 6142
Рабочая область	W0...W511	W0...W510
Область битов хранения	H0...H511	H0...H510
Область вспомогательных битов	A0...A959	A448...A958
Область таймеров	T0000...T4095	T0000...T4094
Область счетчиков	C0000...C4095	C0000...C4094
Область DM	D0...D32767	D0...D32766
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Постоянные	#0000...#FFFF (двоичн.)	---
Регистры данных	DR0...DR15	---
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

Описание

Команда FLT(452) преобразует 16-битовое двоичное значение со знаком в слове S в 32-битовое значение с плавающей запятой (в формате IEEE754) и записывает результат в [R+1, R]. В слове результата после десятичной запятой добавляется 0.



Двоичное значение со знаком (16 бит)

Значение с плавающей запятой (32 бит)

В операнде S можно указывать только значения, принадлежащие диапазону -32768...32767. Для преобразования двоичных значений со знаком, выходящих за пределы этого диапазона, используется команда FLTL(453).

Примеры преобразования:
Двоичное значение со знаком 3 преобразуется в 3,0.
Двоичное значение со знаком -3 преобразуется в -3,0.

Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Выключен
Флаг равенства	=	Включен, если и показатель, и мантисса результата равны «0». Выключен во всех остальных случаях.
Флаг отрицательного значения	N	Включен, если результат отрицателен. Выключен во всех остальных случаях.

Меры предосторожности

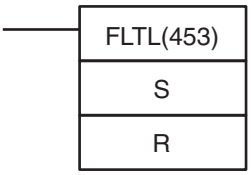
S должно содержать двоичное значение со знаком, десятичный эквивалент которого должен находиться в диапазоне от -32768 до 32767.

3-14-4 32-БИТ В ПЛАВ. ЗАПЯТАЯ: FLTL(453)

Назначение

Преобразование 32-битового двоичного значения со знаком в 32-битовое значение с плавающей запятой и запись результата в указанные слова.

Символ РКС



S: Первое исходное слово

R: Первое слово результата

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	FLTL(453)
	Однократное выполнение по положительному фронту	@FLTL(453)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

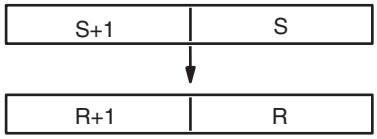
Характеристики операндов

Область	S	R
Область CIO	CIO 0...CIO 6142	
Рабочая область	W0...W510	
Область битов хранения	H0...H510	
Область вспомогательных битов	A0...A958	A448...A958
Область таймеров	T0000...T4094	
Область счетчиков	C0000...C4094	
Область DM	D0...D32766	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Постоянные	#00000000...#FFFFFFFF (двоичное)	---
Регистры данных	---	

Область	S	R
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

Описание

Команда FLTL(453) преобразует 32-битовое двоичное значение со знаком в [S+1, S] в 32-битовое значение с плавающей запятой (в формате IEEE754) и записывает результат в [R+1, R]. В слове результата после десятичной запятой добавляется 0.



Двоичное значение со знаком (32 бит)

Значение с плавающей запятой (32 бит)

S+1 и S должны содержать или указывать на двоичное значение со знаком в диапазоне от -2 147 483 648 до 2 147 483 647. Значение в формате с плавающей запятой содержит 24 значащих двоичных разряда (бита). В случае использования команды FLTL(453) для преобразования значения больше 16 777 215 (самое большое число, которое можно представить 24 битами) результат будет неточным.

Примеры преобразования:

Двоичное значение со знаком 16 777 215 преобразуется в 16 777 215,0.
Двоичное значение со знаком -16 777 215 преобразуется в -15 777 215,0.

Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Выключен
Флаг равенства	=	Включен, если и показатель, и мантисса результата равны «0». Выключен во всех остальных случаях.
Флаг отрицательного значения	N	Включен, если результат отрицателен. Выключен во всех остальных случаях.

Меры предосторожности

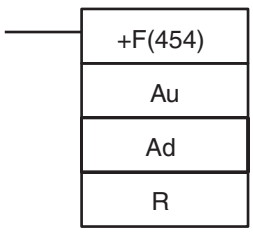
Если абсолютное значение числа больше, чем 16 777 215 (максимальное значение, которое может быть выражено 24 битами), результат преобразования такого числа будет неточным.

3-14-5 СЛОЖЕНИЕ С ПЛАВАЮЩЕЙ ЗАПЯТОЙ: +F(454)

Назначение

Сложение двух 32-битовых чисел с плавающей запятой и запись результата в указанные слова.

Символ РКС



Au: Первое слово 1–го слагаемого

AD: Первое слово 2–го слагаемого

R: Первое слово результата

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	+F(454)
	Однократное выполнение по положительному фронту	@+F(454)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

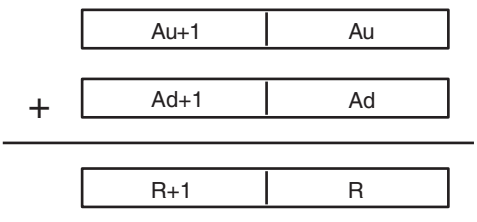
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	Au	Ad	R
Область CIO	CIO 0...CIO 6142		
Рабочая область	W0...W510		
Область битов хранения	H0...H510		
Область вспомогательных битов	A0...A958		A448...A958
Область таймеров	T0000...T4094		
Область счетчиков	C0000...C4094		
Область DM	D0...D32766		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	#00000000...#FFFFFFFF (двоичное)		---
Регистры данных	---		
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда +F(454) добавляет 32-битовое значение с плавающей запятой в [Ad+1, Ad] к 32-битовому значению с плавающей запятой в [Au+1, Au] и записывает результат в [R+1, R]. (Значения с плавающей запятой должны быть указаны в формате IEEE754.)



Если результат по модулю превосходит максимальное значение, которое может быть представлено в формате с плавающей запятой, включается флаг переполнения и в качестве результата выдается $\pm\infty$.

Если абсолютное значение результата меньше минимального значения, которое может быть выражено числом с плавающей запятой, включается флаг потери значимости и в качестве результата выдается «0».

В следующей таблице приведены результаты выполнения сложения для разных комбинаций первого и второго слагаемых.

	Слагаемое 1				
Слагаемое 2	**0**	**Число**	**+∞**	**–∞**	**NaN**
0	0	Число	+∞	–∞	
Число	Число	См. примеч. 1	+∞	–∞	
+∞	+∞	+∞	+∞	См. примеч. 2	
–∞	–∞	–∞	См. примеч. 2	–∞	
NaN	См. примеч. 2				

Примечание.

(1) Результатом может быть ноль (включая потерю значимости), числовое значение, +∞ или -∞.

(2) Устанавливается флаг ошибки, команда не выполняется.

Флаги

Название	**Обозначение**	**Описание работы**
Флаг ошибки	ER	Включен, если значение первого или второго слагаемого не опознано как значение с плавающей запятой. Включен, если значение первого или второго слагаемого не является числом (NaN). Включен, если складываются +∞ и –∞. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если и показатель, и мантисса результата равны «0». Выключен во всех остальных случаях.
Флаг переполнения	OF	Включен, если абсолютное значение результата слишком велико и не может быть выражено 32-битовым числом с плавающей запятой.
Флаг потери значимости	UF	Включен, если абсолютное значение результата слишком мало и не может быть выражено 32-битовым числом с плавающей запятой.
Флаг отрицательного значения	N	Включен, если результат отрицателен. Выключен во всех остальных случаях.

Меры предосторожности

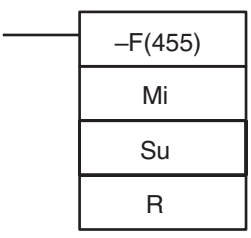
Значения первого [Au+1, Au] и второго [Ad+1, Ad] слагаемых должны быть представлены в формате с плавающей запятой в соответствии с IEEE754.

3-14-6 ВЫЧИТАНИЕ С ПЛАВАЮЩЕЙ ЗАПЯТОЙ: –F(455)

Назначение

Вычитание одного 32-битового числа с плавающей запятой из другого и запись результата в указанные слова.

Символ РКС



Mi: Первое слово уменьшаемого

Su: Первое слово вычитаемого

R: Первое слово результата

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	–F(455)
	Однократное выполнение по положительному фронту	@–F(455)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

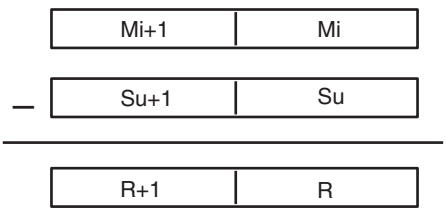
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	Mi	Su	R
Область CIO	CIO 0...CIO 6142		
Рабочая область	W0...W510		
Область битов хранения	H0...H510		
Область вспомогательных битов	A0...A958		A448...A958
Область таймеров	T0000...T4094		
Область счетчиков	C0000...C4094		
Область DM	D0...D32766		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	#00000000...#FFFFFFFF (двоичное)		---
Регистры данных	---		
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда –F(455) вычитает 32-битовое число с плавающей запятой в [Su+1, Su] из 32-битового числа с плавающей запятой в [Mi+1, Mi] и записывает результат в [R+1, R]. (Значения с плавающей запятой должны быть указаны в формате IEEE754.)



Если результат по модулю превосходит максимальное значение, которое может быть представлено в формате с плавающей запятой, включается флаг переполнения и в качестве результата выдается $\pm\infty$.

Если абсолютное значение результата меньше минимального значения, которое может быть выражено числом с плавающей запятой, включается флаг потери значимости и в качестве результата выдается «0».

В следующей таблице приведены результаты выполнения вычитания для разных комбинаций уменьшаемого и вычитаемого.

	Уменьшаемое				
Вычитаемое	**0**	**Число**	**+∞**	**–∞**	**NaN**
0	0	Число	+∞	–∞	
Число	Число	См. примеч. 1	+∞	–∞	
+∞	–∞	–∞	См. примеч. 2	–∞	
–∞	+∞	+∞	+∞	См. примеч. 2	
NaN					См. примеч. 2

Примечание.

(1) Результатом может быть ноль (включая потерю значимости), числовое значение, +∞ или -∞.

(2) Устанавливается флаг ошибки, команда не выполняется.

Флаги

Название	**Обозначение**	**Описание работы**
Флаг ошибки	ER	Включен, если значение уменьшаемого или вычитаемого не опознано как значение с плавающей запятой. Включен, если значение уменьшаемого или вычитаемого не является числом (NaN). Включен, если +∞ вычитается из +∞. Включен, если -∞ вычитается из -∞. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если и показатель, и мантисса результата равны «0». Выключен во всех остальных случаях.
Флаг переполнения	OF	Включен, если абсолютное значение результата слишком велико и не может быть выражено 32-битовым числом с плавающей запятой.
Флаг потери значимости	UF	Включен, если абсолютное значение результата слишком мало и не может быть выражено 32-битовым числом с плавающей запятой.
Флаг отрицательного значения	N	Включен, если результат отрицателен. Выключен во всех остальных случаях.

Меры предосторожности

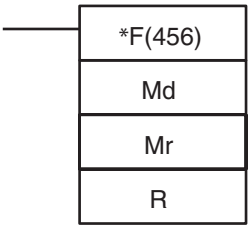
Значения уменьшаемого [Mi+1, Mi] и вычитаемого [Su+1, Su] должны быть представлены в формате с плавающей запятой в соответствии с IEEE754.

3-14-7 УМНОЖЕНИЕ С ПЛАВАЮЩЕЙ ЗАПЯТОЙ: ∗F(456)

Назначение

Умножение двух 32-битовых чисел с плавающей запятой и запись результата в указанные слова.

Символ РКС



Md: Первое слово умножаемого

Mr: Первое слово множителя

R: Первое слово результата

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	*F(456)
	Однократное выполнение по положительному фронту	@*F(456)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

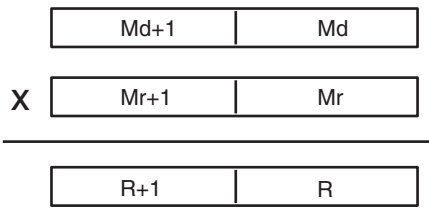
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	Md	Mr	R
Область CIO	CIO 0...CIO 6142		
Рабочая область	W0...W510		
Область битов хранения	H0...H510		
Область вспомогательных битов	A0...A958		A448...A958
Область таймеров	T0000...T4094		
Область счетчиков	C0000...C4094		
Область DM	D0...D32766		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	#00000000...#FFFFFFFF (двоичное)		---
Регистры данных	---		
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда *F(456) умножает 32-битовое число с плавающей запятой в [Md+1, Md] на 32-битовое число с плавающей запятой в [Mr+1, Mr] и записывает результат в [R+1, R]. (Значения с плавающей запятой должны быть указаны в формате IEEE754.)



Умножаемое
(с плавающей запятой, 32 бит)

Множитель
(с плавающей запятой, 32 бит)

Результат
(с плавающей запятой, 32 бит)

Если результат по модулю превосходит максимальное значение, которое может быть представлено в формате с плавающей запятой, включается флаг переполнения и в качестве результата выдается ±∞.

Если абсолютное значение результата меньше минимального значения, которое может быть выражено числом с плавающей запятой, включается флаг потери значимости и в качестве результата выдается «0».

В следующей таблице приведены результаты выполнения умножения для разных комбинаций умножаемого и множителя.

	Умножаемое				
Множитель	**0**	**Число**	**+∞**	**–∞**	**NaN**
0	0	0	См. примеч. 2	См. примеч. 2	
Число	0	См. примеч. 1	+/–∞	+/–∞	
+∞	См. примеч. 2	+/–∞	+∞	–∞	
–∞	См. примеч. 2	+/–∞	–∞	+∞	
NaN	См. примеч. 2				

Примечание.

(1) Результатом может быть ноль (включая потерю значимости), числовое значение, +∞ или -∞.

(2) Устанавливается флаг ошибки, команда не выполняется.

Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если значение умножаемого или множителя не опознано как значение с плавающей запятой. Включен, если значение умножаемого или множителя не является числом (NaN). Включен, если +∞ умножается на 0. Включен, если -∞ умножается на 0. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если и показатель, и мантисса результата равны «0». Выключен во всех остальных случаях.
Флаг переполнения	OF	Включен, если абсолютное значение результата слишком велико и не может быть выражено 32-битовым числом с плавающей запятой.
Флаг потери значимости	UF	Включен, если абсолютное значение результата слишком мало и не может быть выражено 32-битовым числом с плавающей запятой.
Флаг отрицательного значения	N	Включен, если результат отрицателен. Выключен во всех остальных случаях.

Меры предосторожности

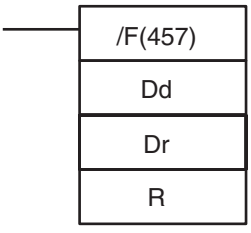
Значения умножаемого [Md+1, Md] и множителя [Mr+1, Mr] должны быть представлены в формате с плавающей запятой в соответствии с IEEE754

3-14-8 ДЕЛЕНИЕ С ПЛАВАЮЩЕЙ ЗАПЯТОЙ: /F(457)

Назначение

Деление одного 32-битового числа с плавающей запятой на другое и запись результата в указанные слова.

Символ РКС



Dd: Первое слово делимого

Dr: Первое слово делителя

R: Первое слово результата

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	/F(457)
	Однократное выполнение по положительному фронту	@/F(457)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

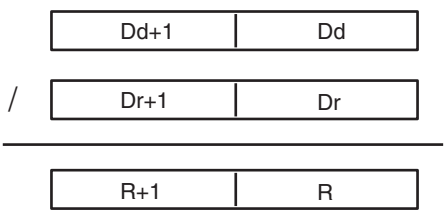
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	Dd	Dr	R
Область CIO	CIO 0...CIO 6142		
Рабочая область	W0...W510		
Область битов хранения	H0...H510		
Область вспомогательных битов	A0...A958		A448...A958
Область таймеров	T0000...T4094		
Область счетчиков	C0000...C4094		
Область DM	D0...D32766		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	#00000000...#FFFFFFFF (двоичное)		---
Регистры данных	---		
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда /F(457) делит 32-битовое число с плавающей запятой в [Dd+1, Dd] на 32-битовое число с плавающей запятой в [Dr+1, Dr] и записывает результат в [R+1, R]. (Значения с плавающей запятой должны быть указаны в формате IEEE754.)



Если результат по модулю превосходит максимальное значение, которое может быть представлено в формате с плавающей запятой, включается флаг переполнения и в качестве результата выдается $\pm\infty$.

Если абсолютное значение результата меньше минимального значения, которое может быть выражено числом с плавающей запятой, включается флаг потери значимости и в качестве результата выдается «0».

В следующей таблице приведены результаты выполнения деления для разных комбинаций делимого и делителя.

	Делимое				
Делитель	**0**	**Число**	**+∞**	**–∞**	**NaN**
0	См. примеч. 3	+/–∞	+∞	–∞	
Число	0	См. примеч. 1	+/–∞	+/–∞	
+∞	0	См. примеч. 2	См. примеч. 3	См. примеч. 3	
–∞	0	См. примеч. 2	См. примеч. 3	См. примеч. 3	
NaN	См. примеч. 3				

Примечание.

(1) Результатом может быть ноль (включая потерю значимости), числовое значение, +∞ или -∞.

(2) При потере значимости результатом будет «0».

(3) Устанавливается флаг ошибки, команда не выполняется.

Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если значение делимого или делителя не опознано как значение с плавающей запятой. Включен, если значение делимого или делителя не является числом (NaN). Включен, если и делимое и делитель оба равны «0». Включен, если и делимое и делитель оба равны +∞ или -∞. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если и показатель, и мантисса результата равны «0». Выключен во всех остальных случаях.
Флаг переполнения	OF	Включен, если абсолютное значение результата слишком велико и не может быть выражено 32-битовым числом с плавающей запятой.
Флаг потери значимости	UF	Включен, если абсолютное значение результата слишком мало и не может быть выражено 32-битовым числом с плавающей запятой.
Флаг отрицательного значения	N	Включен, если результат отрицателен. Выключен во всех остальных случаях.

Меры предосторожности

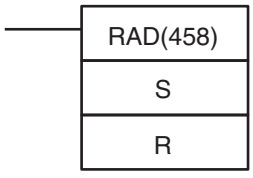
Значения делимого [Dd+1, Dd] и делителя [Dr+1, Dr] должны быть представлены в формате с плавающей запятой в соответствии с IEEE754.

3-14-9 ГРАДУСЫ В РАДИАНЫ: RAD(458)

Назначение

Преобразование 32-битового числа с плавающей запятой из градусов в радианы и запись результата в указанные слова.

Символ РКС



S: Первое исходное слово

R: Первое слово результата

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	RAD(458)
	Однократное выполнение по положительному фронту	@RAD(458)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

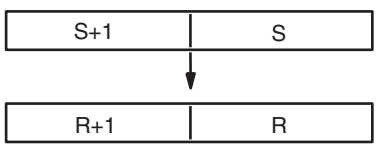
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	S	R
Область CIO	CIO 0...CIO 6142	
Рабочая область	W0...W510	
Область битов хранения	H0...H510	
Область вспомогательных битов	A0...A958	A448...A958
Область таймеров	T0000...T4094	
Область счетчиков	C0000...C4094	
Область DM	D0...D32766	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Постоянные	#00000000...#FFFFFFFF (двоичное)	---
Регистры данных	---	
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

Описание

Команда RAD(458) преобразует 32-битовое число с плавающей запятой в [S+1, S] из градусов в радианы и записывает результат в [R+1, R]. (Значения с плавающей запятой должны быть указаны в формате IEEE754.)



Исходное значение
(градусы, с плавающей запятой, 32 бит)

Результат
(радианы, с плавающей запятой, 32 бит)

Преобразование из градусов в радианы производится по следующей формуле:

Значение в градусах $\times \pi/180$ = значение в радианах

Если результат по модулю превосходит максимальное значение, которое может быть представлено в формате с плавающей запятой, включается флаг переполнения и в качестве результата выдается $\pm\infty$.

Если абсолютное значение результата меньше минимального значения, которое может быть выражено числом с плавающей запятой, включается флаг потери значимости и в качестве результата выдается «0».

Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если входное значение не опознано как значение с плавающей запятой. Включен, если входное значение не является числом (NaN). Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если и показатель, и мантисса результата равны «0». Выключен во всех остальных случаях.
Флаг переполнения	OF	Включен, если абсолютное значение результата слишком велико и не может быть выражено 32-битовым числом с плавающей запятой.
Флаг потери значимости	UF	Включен, если абсолютное значение результата слишком мало и не может быть выражено 32-битовым числом с плавающей запятой.
Флаг отрицательного значения	N	Включен, если результат отрицателен. Выключен во всех остальных случаях.

Меры предосторожности

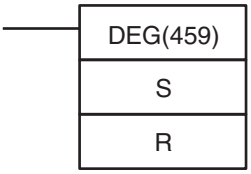
Входное значение в [S+1, S] должно быть представлено в формате с плавающей запятой в соответствии с IEEE754.

3-14-10 РАДИАНЫ В ГРАДУСЫ: DEG(459)

Назначение

Преобразование 32-битового числа с плавающей запятой из радианов в градусы и запись результата в указанные слова.

Символ РКС



S: Первое исходное слово

R: Первое слово результата

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	DEG(459)
	Однократное выполнение по положительному фронту	@DEG(459)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

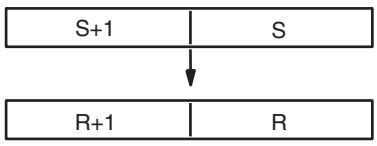
Характеристики операндов

Область	S	R
Область CIO	CIO 0...CIO 6142	
Рабочая область	W0...W510	

Область	S	R
Область битов хранения	H0...H510	
Область вспомогательных битов	A0...A958	A448...A958
Область таймеров	T0000...T4094	
Область счетчиков	C0000...C4094	
Область DM	D0...D32766	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Постоянные	#0000000...#FFFFFFFF (двоичн.)	---
Регистры данных	---	
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

Описание

Команда DEG(459) преобразует 32-битовое число с плавающей запятой в [S+1, S] из радианов в градусы и записывает результат в [R+1, R]. (Значения с плавающей запятой должны быть указаны в формате IEEE754.)



Исходное значение
(радианы, с плавающей запятой, 32 бит)

Результат
(градусы, с плавающей запятой, 32 бит)

Преобразование из радиан в градусы производится по следующей формуле:

Значение в радианах $\times$ 180/π = значение в градусах

Если результат по модулю превосходит максимальное значение, которое может быть представлено в формате с плавающей запятой, включается флаг переполнения и в качестве результата выдается $\pm\infty$.

Если абсолютное значение результата меньше минимального значения, которое может быть выражено числом с плавающей запятой, включается флаг потери значимости и в качестве результата выдается «0».

Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если входное значение не опознано как значение с плавающей запятой. Включен, если входное значение не является числом (NaN). Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если и показатель, и мантисса результата равны «0». Выключен во всех остальных случаях.
Флаг переполнения	OF	Включен, если абсолютное значение результата слишком велико и не может быть выражено 32-битовым числом с плавающей запятой.

Название	Обозначение	Описание работы
Флаг потери значимости	UF	Включен, если абсолютное значение результата слишком мало и не может быть выражено 32-битовым числом с плавающей запятой.
Флаг отрицательного значения	N	Включен, если результат отрицателен. Выключен во всех остальных случаях.

Меры предосторожности

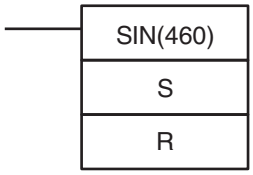
Входное значение в [S+1, S] должно быть представлено в формате с плавающей запятой в соответствии с IEEE754.

3-14-11 СИНУС: SIN(460)

Назначение

Расчет синуса от 32-битового числа с плавающей запятой (в радианах) и запись результата в указанные слова.

Символ РКС



S: Первое исходное слово

R: Первое слово результата

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	SIN(460)
	Однократное выполнение по положительному фронту	@SIN(460)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

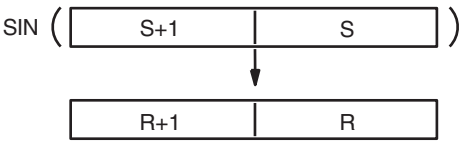
Характеристики операндов

Область	S	R
Область CIO	CIO 0...CIO 6142	
Рабочая область	W0...W510	
Область битов хранения	H0...H510	
Область вспомогательных битов	A0...A958	A448...A958
Область таймеров	T0000...T4094	
Область счетчиков	C0000...C4094	
Область DM	D0...D32766	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Постоянные	#00000000...#FFFFFFFF (двоичное)	---
Регистры данных	---	

Область	S	R
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

Описание

Команда SIN(460) рассчитывает синус угла (в радианах), представленного в виде 32-битового числа с плавающей запятой в [S+1, S], и записывает результат в [R+1, R].
(Значения с плавающей запятой должны быть указаны в формате IEEE754.)

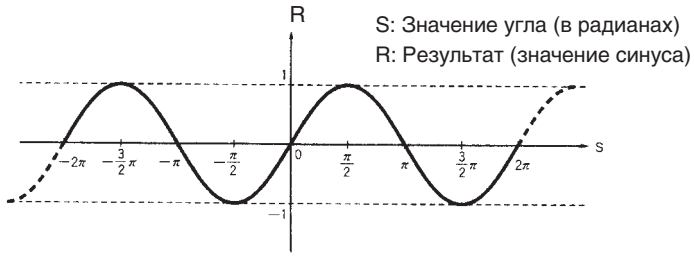


Исходное значение
(с плавающей запятой, 32 бит)

Результат
(с плавающей запятой, 32 бит)

В S+1 и S следует указать требуемое значение угла в радианах (от -65 535 до 65 535). Если значение угла будет выходить за диапазон -65 535...65 535, произойдет ошибка и команда выполнена не будет. Информацию о преобразовании из градусов в радианы см. в разделе *3-14-9 ГРАДУСЫ В РАДИАНЫ: RAD(458)*.

На следующем графике отображена взаимосвязь между величиной угла и результатом.



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если входное значение не является числом (NaN). Включен, если входное значение по модулю превосходит 65 535. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если и показатель, и мантисса результата равны «0». Выключен во всех остальных случаях.
Флаг переполнения	OF	Выключен
Флаг потери значимости	UF	Выключен
Флаг отрицательного значения	N	Включен, если результат отрицателен. Выключен во всех остальных случаях.

Меры предосторожности

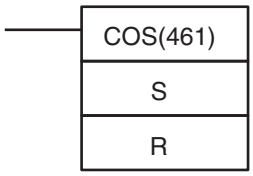
Входное значение в [S+1, S] должно быть представлено в формате с плавающей запятой в соответствии с IEEE754.

3-14-12 КОСИНУС: COS(461)

Назначение

Расчет косинуса от 32-битового числа с плавающей запятой (в радианах) и запись результата в указанные слова.

Символ РКС



S: Первое исходное слово

R: Первое слово результата

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	COS(461)
	Однократное выполнение по положительному фронту	@COS(461)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

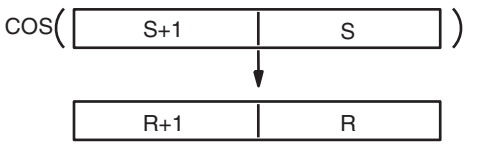
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	S	R
Область CIO	CIO 0...CIO 6142	
Рабочая область	W0...W510	
Область битов хранения	H0...H510	
Область вспомогательных битов	A0...A958	A448...A958
Область таймеров	T0000...T4094	
Область счетчиков	C0000...C4094	
Область DM	D0...D32766	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Постоянные	#00000000...#FFFFFFFF (двоичное)	---
Регистры данных	---	
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

Описание

Команда COS(461) рассчитывает косинус угла (в радианах), представленного в виде 32-битового числа с плавающей запятой в [S+1, S], и записывает результат в [R+1, R].
(Значения с плавающей запятой должны быть указаны в формате IEEE754.)

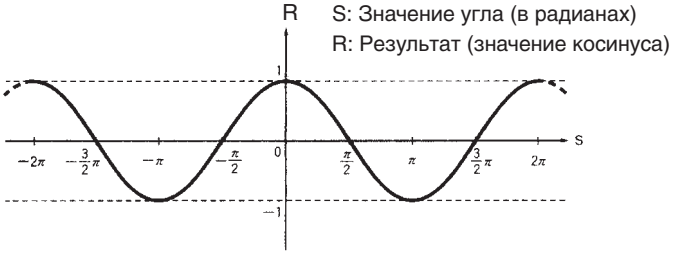


Исходное значение
(с плавающей запятой, 32 бит)

Результат
(с плавающей запятой, 32 бит)

В S+1 и S следует указать требуемое значение угла в радианах (от -65 535 до 65 535). Если значение угла будет выходить за диапазон -65 535...65 535, произойдет ошибка и команда выполнена не будет. Информацию о преобразовании из градусов в радианы см. в разделе *3-14-9 ГРАДУСЫ В РАДИАНЫ: RAD(458)*.

На следующем графике отображена взаимосвязь между величиной угла и результатом.



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если входное значение не является числом (NaN). Включен, если входное значение по модулю превосходит 65 535. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если и показатель, и мантисса результата равны «0». Выключен во всех остальных случаях.
Флаг переполнения	OF	Выключен
Флаг потери значимости	UF	Выключен
Флаг отрицательного значения	N	Включен, если результат отрицателен. Выключен во всех остальных случаях.

Меры предосторожности

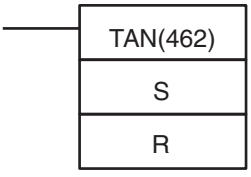
Входное значение в [S+1, S] должно быть представлено в формате с плавающей запятой в соответствии с IEEE754.

3-14-13 ТАНГЕНС: TAN(462)

Назначение

Расчет тангенса от 32-битового числа с плавающей запятой (в радианах) и запись результата в указанные слова.

Символ РКС



S: Первое исходное слово

R: Первое слово результата

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	TAN(462)
	Однократное выполнение по положительному фронту	@TAN(462)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

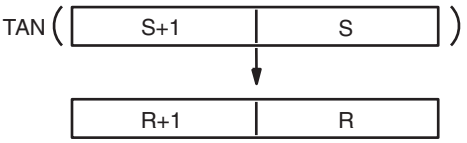
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	S	R
Область CIO	CIO 0...CIO 6142	
Рабочая область	W0...W510	
Область битов хранения	H0...H510	
Область вспомогательных битов	A0...A958	A448...A958
Область таймеров	T0000...T4094	
Область счетчиков	C0000...C4094	
Область DM	D0...D32766	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Постоянные	#00000000...#FFFFFFFF (двоичное)	---
Регистры данных	---	
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

Описание

Команда TAN(462) рассчитывает тангенс угла (в радианах), представленного в виде 32-битового числа с плавающей запятой в [S+1, S], и записывает результат в [R+1, R].
(Значения с плавающей запятой должны быть указаны в формате IEEE754.)



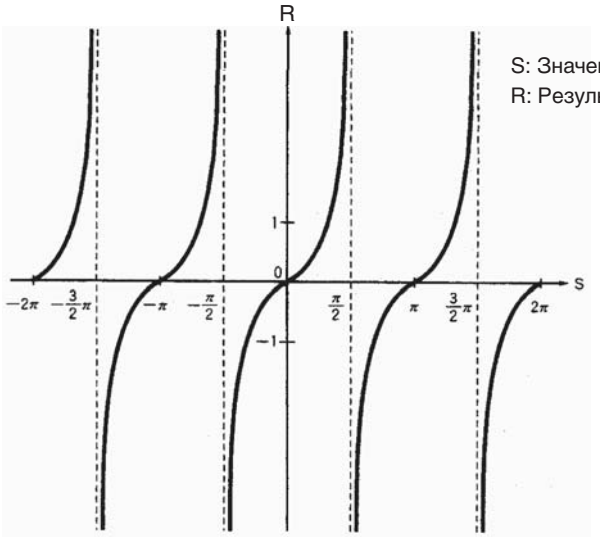
Исходное значение
(с плавающей запятой, 32 бит)

Результат
(с плавающей запятой, 32 бит)

В S+1 и S следует указать требуемое значение угла в радианах (от -65 535 до 65 535). Если значение угла будет выходить за диапазон -65 535...65 535, произойдет ошибка и команда выполнена не будет. Информацию о преобразовании из градусов в радианы см. в разделе *3-14-9 ГРАДУСЫ В РАДИАНЫ: RAD(458)*.

Если результат по модулю превосходит максимальное значение, которое может быть представлено в формате с плавающей запятой, включается флаг переполнения и в качестве результата выдается ±∞.

На следующем графике отображена взаимосвязь между величиной угла и результатом.



S: Значение угла (в радианах)
R: Результат (значение тангенса)

Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если входное значение не является числом (NaN). Включен, если входное значение по модулю превосходит 65 535. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если и показатель, и мантисса результата равны «0». Выключен во всех остальных случаях.
Флаг переполнения	OF	Выключен
Флаг потери значимости	UF	Выключен
Флаг отрицательного значения	N	Включен, если результат отрицателен. Выключен во всех остальных случаях.

Меры предосторожности

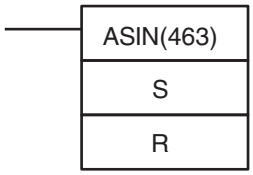
Входное значение в [S+1, S] должно быть представлено в формате с плавающей запятой в соответствии с IEEE754.

3-14-14 АРКСИНУС: ASIN(463)

Назначение

Расчет арксинуса от 32-битового числа с плавающей запятой (в радианах) и запись результата в указанные слова. (Функция арксинуса является обратной по отношению к функции синуса. Она возвращает значение угла, которому соответствует указанное значение синуса от -1 до 1.)

Символ РКС



S: Первое исходное слово

R: Первое слово результата

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	ASIN(463)
	Однократное выполнение по положительному фронту	@ASIN(463)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

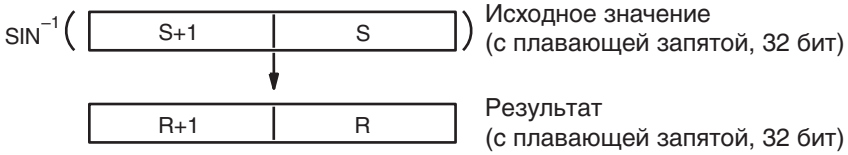
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	S	R
Область CIO	CIO 0...CIO 6142	
Рабочая область	W0...W510	
Область битов хранения	H0...H510	
Область вспомогательных битов	A0...A958	A448...A958
Область таймеров	T0000...T4094	
Область счетчиков	C0000...C4094	
Область DM	D0...D32766	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Постоянные	#00000000...#FFFFFFFF (двоичное)	---
Регистры данных	---	
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

Описание

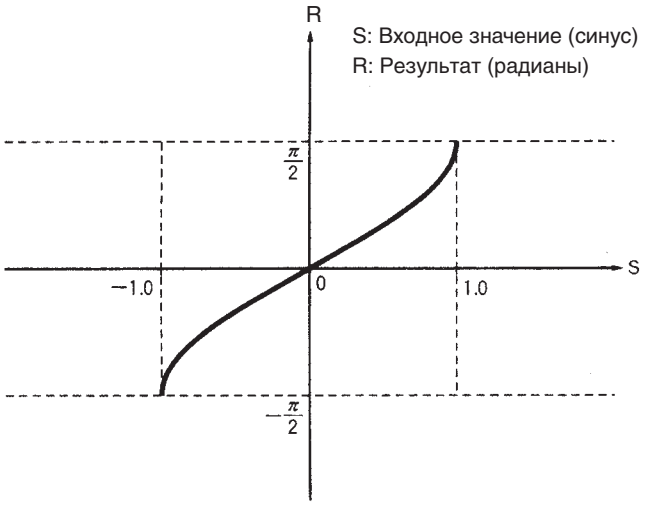
Команда ASIN(463) рассчитывает арксинус угла (в радианах), представленного в виде 32-битового числа с плавающей запятой в [S+1, S], и записывает результат в [R+1, R].
(Значения с плавающей запятой должны быть указаны в формате IEEE754.)



Входное значение должно находиться в диапазоне от -1,0 до 1,0. Если входное значение по модулю превосходит 1,0, произойдет ошибка и команда выполнена не будет.

В качестве результата в слова [R+1, R] выводится значение угла (в радианах) в диапазоне от $-\pi/2$ до $\pi/2$.

На следующем графике отображена взаимосвязь между входным значением и результатом.



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если входное значение не опознано как значение с плавающей запятой. Включен, если входное значение не является числом (NaN). Включен, если входное значение по модулю превосходит 1,0. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если и показатель, и мантисса результата равны «0». Выключен во всех остальных случаях.
Флаг переполнения	OF	Выключен
Флаг потери значимости	UF	Выключен
Флаг отрицательного значения	N	Включен, если результат отрицателен. Выключен во всех остальных случаях.

Меры предосторожности

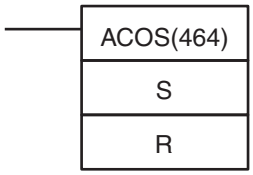
Входное значение в [S+1, S] должно быть представлено в формате с плавающей запятой в соответствии с IEEE754.

3-14-15 АРККОСИНУС: ACOS(464)

Назначение

Расчет арккосинуса от 32-битового числа с плавающей запятой (в радианах) и запись результата в указанные слова. (Функция арккосинуса является обратной по отношению к функции косинуса. Она возвращает значение угла, которому соответствует указанное значение косинуса от -1 до 1.)

Символ РКС



S: Первое исходное слово

R: Первое слово результата

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	ACOS(464)
	Однократное выполнение по положительному фронту	@ACOS(464)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

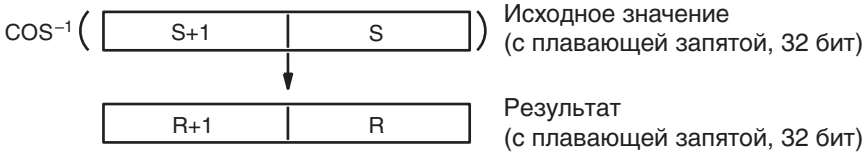
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	S	R
Область CIO	CIO 0...CIO 6142	
Рабочая область	W0...W510	
Область битов хранения	H0...H510	
Область вспомогательных битов	A0...A958	A448...A958
Область таймеров	T0000...T4094	
Область счетчиков	C0000...C4094	
Область DM	D0...D32766	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Постоянные	#00000000...#FFFFFFFF (двоичное)	---
Регистры данных	---	
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

Описание

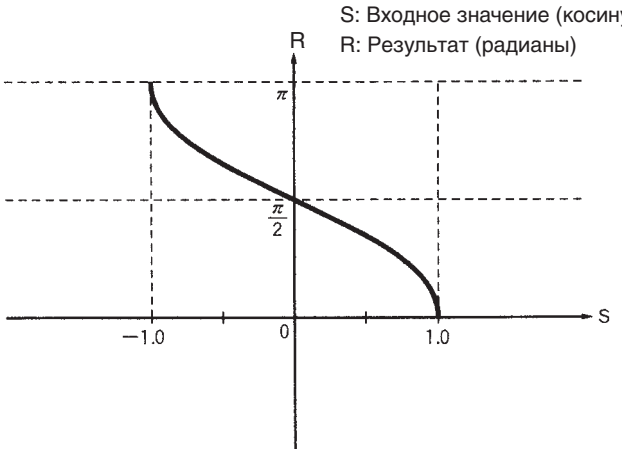
Команда ACOS(464) рассчитывает арккосинус угла (в радианах), представленного в виде 32-битового числа с плавающей запятой в [S+1, S], и записывает результат в [R+1, R].
(Значения с плавающей запятой должны быть указаны в формате IEEE754.)



Входное значение должно находиться в диапазоне от -1,0 до 1,0. Если входное значение по модулю превосходит 1,0, произойдет ошибка и команда выполнена не будет.

Команда возвращает значение угла (в радианах) в диапазоне от 0 до π и записывает его в [R+1, R].

На следующем графике отображена взаимосвязь между входным значением и результатом.



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если входное значение не опознано как значение с плавающей запятой. Включен, если входное значение не является числом (NaN). Включен, если входное значение по модулю превосходит 1,0. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если и показатель, и мантисса результата равны «0». Выключен во всех остальных случаях.
Флаг переполнения	OF	Выключен
Флаг потери значимости	UF	Выключен
Флаг отрицательного значения	N	Включен, если результат отрицателен. Выключен во всех остальных случаях.

Меры предосторожности

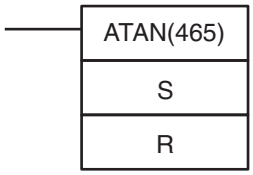
Входное значение в [S+1, S] должно быть представлено в формате с плавающей запятой в соответствии с IEEE754.

3-14-16 АРКТАНГЕНС: ATAN(465)

Назначение

Расчет арктангенса от 32-битового числа с плавающей запятой (в радианах) и запись результата в указанные слова. (Функция арктангенса является обратной по отношению к функции тангенса. Она возвращает значение угла, которому соответствует указанное значение тангенса.)

Символ РКС



S: Первое исходное слово

R: Первое слово результата

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	ATAN(465)
	Однократное выполнение по положительному фронту	@ATAN(465)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

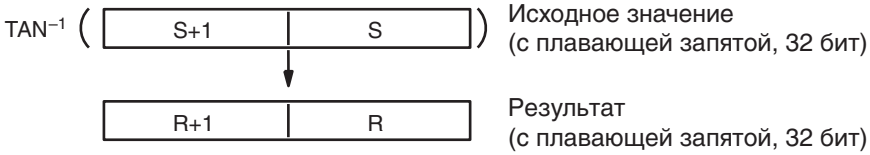
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	S	R
Область CIO	CIO 0...CIO 6142	
Рабочая область	W0...W510	
Область битов хранения	H0...H510	
Область вспомогательных битов	A0...A958	A448...A958
Область таймеров	T0000...T4094	
Область счетчиков	C0000...C4094	
Область DM	D0...D32766	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Постоянные	#00000000...#FFFFFFFF (двоичное)	---
Регистры данных	---	
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

Описание

Команда ATAN(465) рассчитывает арктангенс угла (в радианах), представленного в виде 32-битового числа с плавающей запятой в [S+1, S], и записывает результат в [R+1, R].
(Значения с плавающей запятой должны быть указаны в формате IEEE754.)

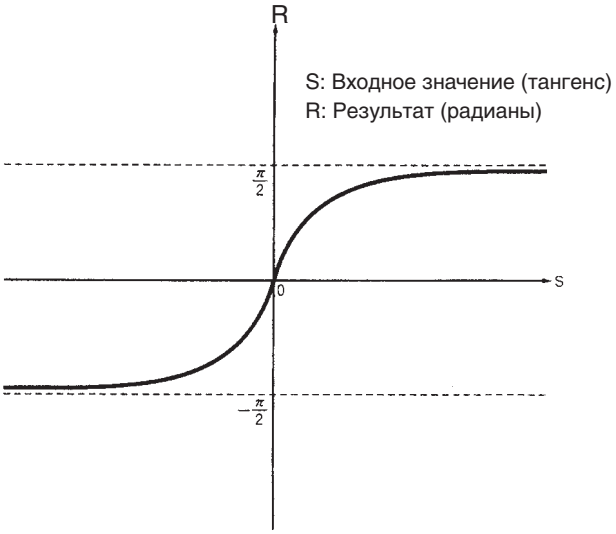


Исходное значение
(с плавающей запятой, 32 бит)

Результат
(с плавающей запятой, 32 бит)

В качестве результата в слова [R+1, R] выводится значение угла (в радианах) в диапазоне от -π/2 до π/2.

На следующем графике отображена взаимосвязь между входным значением и результатом.



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если входное значение не опознано как значение с плавающей запятой. Включен, если входное значение не является числом (NaN). Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если и показатель, и мантисса результата равны «0». Выключен во всех остальных случаях.
Флаг переполнения	OF	Выключен
Флаг потери значимости	UF	Выключен
Флаг отрицательного значения	N	Включен, если результат отрицателен. Выключен во всех остальных случаях.

Меры предосторожности

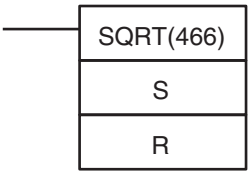
Входное значение в [S+1, S] должно быть представлено в формате с плавающей запятой в соответствии с IEEE754.

3-14-17 КВАДРАТНЫЙ КОРЕНЬ: SQRT(466)

Назначение

Расчет квадратного корня от 32-битового числа с плавающей запятой и запись результата в указанные слова.

Символ РКС



S: Первое исходное слово

R: Первое слово результата

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	SQRT(466)
	Однократное выполнение по положительному фронту	@SQRT(466)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

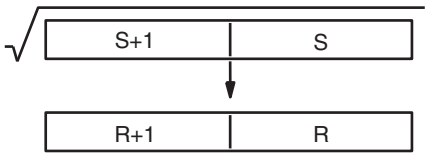
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	S	R
Область CIO	CIO 0...CIO 6142	
Рабочая область	W0...W510	
Область битов хранения	H0...H510	
Область вспомогательных битов	A0...A958	A448...A958
Область таймеров	T0000...T4094	
Область счетчиков	C0000...C4094	
Область DM	D0...D32766	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Постоянные	#00000000...#FFFFFFFF (двоичное)	---
Регистры данных	---	
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

Описание

Команда SQRT(466) рассчитывает квадратный корень от 32-битового числа с плавающей запятой в [S+1, S] и записывает результат в [R+1, R]. (Значения с плавающей запятой должны быть указаны в формате IEEE754.)



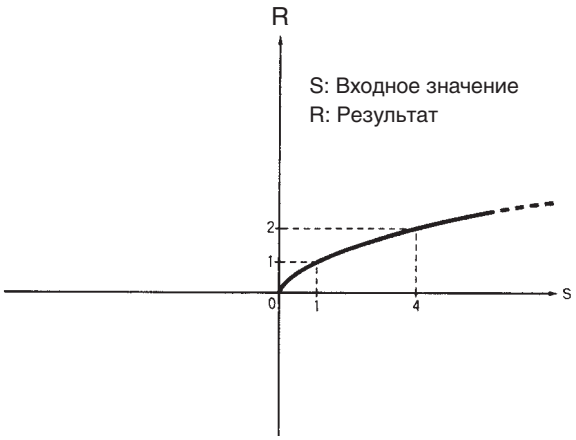
Исходное значение
(с плавающей запятой, 32 бит)

Результат
(с плавающей запятой, 32 бит)

Входное значение должно быть положительным числом. Если оно будет отрицательным, произойдет ошибка и команда выполнена не будет.

Если результат по модулю превосходит максимальное значение, которое может быть представлено в формате с плавающей запятой, включается флаг переполнения и в качестве результата выдается $\pm\infty$.

На следующем графике отображена взаимосвязь между входным значением и результатом.



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если входное значение не опознано как значение с плавающей запятой. Включен, если входное значение отрицательно. Включен, если входное значение не является числом (NaN). Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если и показатель, и мантисса результата равны «0». Выключен во всех остальных случаях.
Флаг переполнения	OF	Включен, если абсолютное значение результата слишком велико и не может быть выражено 32-битовым числом с плавающей запятой.
Флаг потери значимости	UF	Выключен
Флаг отрицательного значения	N	Выключен

Меры предосторожности

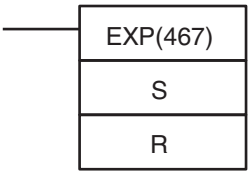
Входное значение в [S+1, S] должно быть представлено в формате с плавающей запятой в соответствии с IEEE754.

3-14-18 ЭКСПОНЕНТА: EXP(467)

Назначение

Расчет натуральной экспоненты (по основанию «e») от 32-битового числа с плавающей запятой и запись результата в указанные слова.

Символ РКС



S: Первое исходное слово

R: Первое слово результата

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	EXP(467)
	Однократное выполнение по положительному фронту	@EXP(467)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

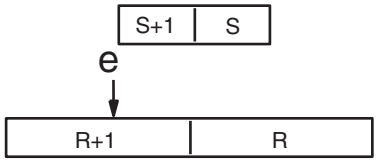
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	S	R
Область CIO	CIO 0...CIO 6142	
Рабочая область	W0...W510	
Область битов хранения	H0...H510	
Область вспомогательных битов	A0...A958	A448...A958
Область таймеров	T0000...T4094	
Область счетчиков	C0000...C4094	
Область DM	D0...D32766	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Постоянные	#00000000...#FFFFFFFF (двоичное)	---
Регистры данных	---	
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

Описание

Команда EXP(467) рассчитывает натуральную экспоненту (по основанию «e») от 32-битового числа с плавающей запятой в [S+1, S] и записывает результат в [R+1, R]. Другими словами, команда EXP(467) рассчитывает e^x (x = входное значение) и записывает результат в [R+1, R].



Исходное значение
(с плавающей запятой, 32 бит)

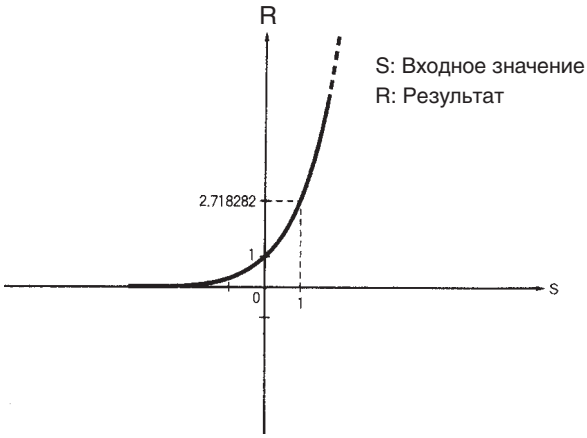
Результат
(с плавающей запятой, 32 бит)

Если результат по модулю превосходит максимальное значение, которое может быть представлено в формате с плавающей запятой, включается флаг переполнения и в качестве результата выдается $\pm\infty$.

Если абсолютное значение результата меньше минимального значения, которое может быть выражено числом с плавающей запятой, включается флаг потери значимости и в качестве результата выдается «0».

Примечание. Постоянная «e» = 2,718282.

На следующем графике отображена взаимосвязь между входным значением и результатом.



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если входное значение не опознано как значение с плавающей запятой. Включен, если входное значение не является числом (NaN). Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если и показатель, и мантисса результата равны «0». Выключен во всех остальных случаях.
Флаг переполнения	OF	Включен, если абсолютное значение результата слишком велико и не может быть выражено 32-битовым числом с плавающей запятой.
Флаг потери значимости	UF	Включен, если абсолютное значение результата слишком мало и не может быть выражено 32-битовым числом с плавающей запятой.
Флаг отрицательного значения	N	Выключен

Меры предосторожности

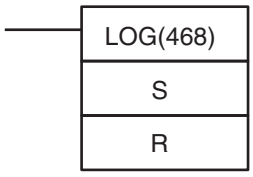
Входное значение в [S+1, S] должно быть представлено в формате с плавающей запятой в соответствии с IEEE754.

3-14-19 ЛОГАРИФМ: LOG(468)

Назначение

Расчет натурального логарифма (по основанию «e») от 32-битового числа с плавающей запятой и запись результата в указанные слова.

Символ РКС



S: Первое исходное слово

R: Первое слово результата

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	LOG(468)
	Однократное выполнение по положительному фронту	@LOG(468)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

Доступные области программы

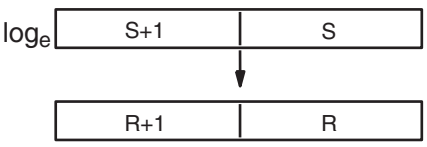
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	S	R
Область CIO	CIO 0...CIO 6142	
Рабочая область	W0...W510	
Область битов хранения	H0...H510	
Область вспомогательных битов	A0...A958	A448...A958
Область таймеров	T0000...T4094	
Область счетчиков	C0000...C4094	
Область DM	D0...D32766	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Постоянные	#00000000...#FFFFFFFF (двоичное)	---
Регистры данных	---	
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

Описание

Команда LOG(468) рассчитывает натуральный логарифм (по основанию «e») от 32-битового числа с плавающей запятой в [S+1, S] и записывает результат в [R+1, R].



Исходное значение (с плавающей запятой, 32 бит)

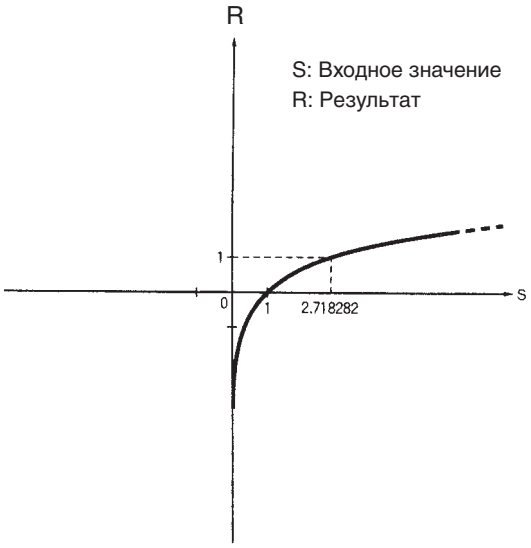
Результат (с плавающей запятой, 32 бит)

Входное значение должно быть положительным числом. Если оно будет отрицательным, произойдет ошибка и команда выполнена не будет.

Если результат по модулю превосходит максимальное значение, которое может быть представлено в формате с плавающей запятой, включается флаг переполнения и в качестве результата выдается $\pm\infty$.

Примечание. Постоянная «e» = 2,718282.

На следующем графике отображена взаимосвязь между входным значением и результатом.



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если входное значение не опознано как значение с плавающей запятой. Включен, если входное значение отрицательно. Включен, если входное значение не является числом (NaN). Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если и показатель, и мантисса результата равны «0». Выключен во всех остальных случаях.
Флаг переполнения	OF	Включен, если абсолютное значение результата слишком велико и не может быть выражено 32-битовым числом с плавающей запятой.
Флаг потери значимости	UF	Выключен
Флаг отрицательного значения	N	Включен, если результат отрицателен. Выключен во всех остальных случаях.

Меры предосторожности

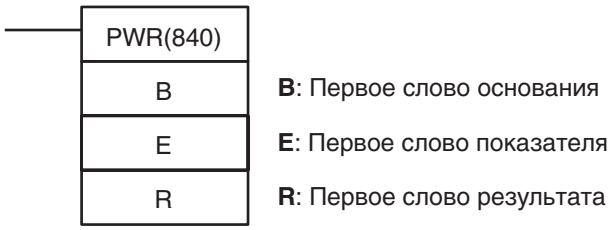
Входное значение в [S+1, S] должно быть представлено в формате с плавающей запятой в соответствии с IEEE754.

3-14-20 ВОЗВЕДЕНИЕ В СТЕПЕНЬ: PWR(840)

Назначение

Возведение 32-битового числа с плавающей запятой в степень, определяемую другим 32-битовым числом с плавающей запятой.

Символ РКС



Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	PWR(840)
	Однократное выполнение по положительному фронту	@PWR(840)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

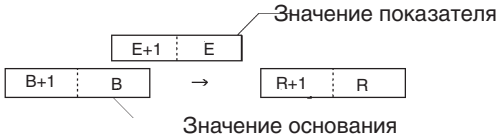
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	B	E	R
Область CIO	CIO 0...CIO 6142		
Рабочая область	W0...W510		
Область битов хранения	H0...H510		
Область вспомогательных битов	A0...A958		A448...A958
Область таймеров	T0000...T4094		
Область счетчиков	C0000...C4094		
Область DM	D0...D32766		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	#00000000...#FFFFFFFF (двоичное)		---
Регистры данных	---		
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда PWR(840) возводит 32-битовое число с плавающей запятой в [B+1, B] в степень 32-битового числа с плавающей запятой в [E+1, E]. Другими словами, команда PWR(840) рассчитывает X^Y (X = [B+1, B]; Y = [E+1, E]).



Например, если основание степени [B+1, B] равно 3,1, а показатель степени [E+1, E] равен 3, результатом выполнения команды будет значение $3{,}1^3$ или 29,791.

Если результат по модулю превосходит максимальное значение, которое может быть представлено в формате с плавающей запятой, включается флаг переполнения.

Если абсолютное значение результата меньше минимального значения, которое может быть выражено числом с плавающей запятой, включается флаг потери значимости.

Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если значение основания [B+1, B] или показателя степени [E+1, E] не опознано как число с плавающей запятой. Включен, если значение основания [B+1, B] или показателя степени [E+1, E] не является числом (NaN). Включен, если основание степени [B+1, B] равно нулю, а показатель степени [E+1, E] меньше 0 (деление на 0). Включен, если основание степени [B+1, B] меньше нуля, а показатель степени [E+1, E] не является целым числом (корень из отрицательного числа). Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если и показатель, и мантисса результата равны «0». Выключен во всех остальных случаях.
Флаг переполнения	OF	Включен, если абсолютное значение результата слишком велико и не может быть выражено 32-битовым числом с плавающей запятой.
Флаг потери значимости	UF	Включен, если абсолютное значение результата слишком мало и не может быть выражено 32-битовым числом с плавающей запятой.
Флаг отрицательного значения	N	Включен, если результат отрицателен. Выключен во всех остальных случаях.

Меры предосторожности

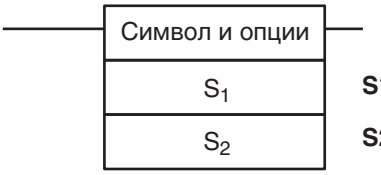
Основание [B+1, B] и показатель степени [E+1, E] должны быть представлены в формате с плавающей запятой в соответствии с IEEE754.

3-14-21 Команды сравнения чисел с плавающей запятой одинарной точности

Назначение

Эти входные команды сравнения служат для сравнения двух значений с плавающей запятой одинарной точности (32-битовых констант в формате IEEE754 и/или содержимого указанных слов) и включают условие выполнения, если выполнено условие сравнения.

Символ РКС



S1: Сравниваемое значение 1

S2: Сравниваемое значение 2

Варианты выполнения

Варианты выполнения	Включение условия выполнения в каждом цикле, в котором результат сравнения = ИСТИНА.	Входная команда сравнения
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

Характеристики операндов

Область	S_1	S_2
Область CIO	CIO 0...CIO 6142	
Рабочая область	W0...W510	
Область битов хранения	H0...H510	
Область вспомогательных битов	A0...A958	
Область таймеров	T0000...T4094	
Область счетчиков	C0000...C4094	
Область DM	D0...D32766	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Постоянные	#00000000...#FFFFFFFF (двоичное)	
Регистры данных	---	
Регистры указателей	IR0...IR15 (только для значений без знака)	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

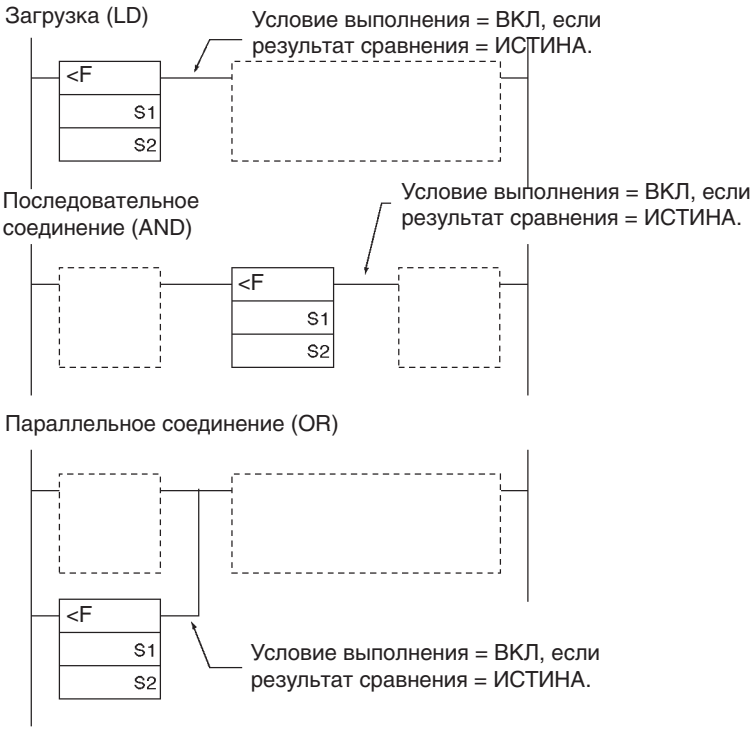
Описание

Входная команда сравнения сравнивает значения, указанные в S_1 и S_2, как значения с плавающей запятой одинарной точности (32-битовые числа в формате IEEE754) и включает условие выполнения, если выполнено условие сравнения. После записи значений в слова S_1 и S_2 укажите первое из двух слов, содержащих 32-битовое значение. Также можно ввести значение с плавающей запятой в формате 8-разрядной шестнадцатеричной константы.

Применение команд в программе

Входные команды сравнения используются в программе так же, как команды LD, AND и OR. Они управляют выполнением следующих за ними команд.

Тип ввода	Описание работы
LD	Команда может присоединяться непосредственно к левой шине.
AND	Команда не может присоединяться непосредственно к левой шине.
ИЛИ	Команда может присоединяться непосредственно к левой шине.



Варианты применения

Доступны шесть видов сравнения и три способа ввода команды: всего 18 комбинаций.

Символ		Вариант (формат данных)
=	(равно)	F: Значение с плавающей запятой одинарной точности
< >	(не равно)	
<	(меньше)	
<=	(меньше или равно)	
>	(больше)	
>=	(больше или равно)	

Краткий обзор входных команд сравнения

В приведенной ниже таблице перечислены коды функций, мнемонические коды, названия и назначение 18 входных команд сравнения чисел с плавающей запятой одинарной точности.
($C1=S_1+1$, S_1, а $C2=S_2+1$, S_2.)

Код	Мнемоническое обозначение	Название	Описание
329	LD=F	ЗАГРУЗКА С ПЛАВАЮЩЕЙ ЗАПЯТОЙ РАВНО	Истина, если C1 = C2
	AND=F	И С ПЛАВАЮЩЕЙ ЗАПЯТОЙ РАВНО	
	OR=F	ИЛИ С ПЛАВАЮЩЕЙ ЗАПЯТОЙ РАВНО	
330	LD<>F	ЗАГРУЗКА С ПЛАВАЮЩЕЙ ЗАПЯТОЙ НЕ РАВНО	Истина, если C1 ≠ C2
	AND<>F	И С ПЛАВАЮЩЕЙ ЗАПЯТОЙ НЕ РАВНО	
	OR<>F	ИЛИ С ПЛАВАЮЩЕЙ ЗАПЯТОЙ НЕ РАВНО	
331	LD<F	ЗАГРУЗКА С ПЛАВАЮЩЕЙ ЗАПЯТОЙ МЕНЬШЕ	Истина, если C1 < C2
	AND<F	И С ПЛАВАЮЩЕЙ ЗАПЯТОЙ МЕНЬШЕ	
	OR<F	ИЛИ С ПЛАВАЮЩЕЙ ЗАПЯТОЙ МЕНЬШЕ	

Код	Мнемоническое обозначение	Название	Описание
332	LD<=F	ЗАГРУЗКА С ПЛАВАЮЩЕЙ ЗАПЯТОЙ МЕНЬШЕ ИЛИ РАВНО	Истина, если $C1 \leq C2$
	AND<=F	И С ПЛАВАЮЩЕЙ ЗАПЯТОЙ МЕНЬШЕ ИЛИ РАВНО	
	OR<=F	ИЛИ С ПЛАВАЮЩЕЙ ЗАПЯТОЙ МЕНЬШЕ ИЛИ РАВНО	
333	LD>F	ЗАГРУЗКА С ПЛАВАЮЩЕЙ ЗАПЯТОЙ БОЛЬШЕ	Истина, если $C1 > C2$
	AND>F	И С ПЛАВАЮЩЕЙ ЗАПЯТОЙ БОЛЬШЕ	
	OR>F	ИЛИ С ПЛАВАЮЩЕЙ ЗАПЯТОЙ БОЛЬШЕ	
334	LD>=F	ЗАГРУЗКА С ПЛАВАЮЩЕЙ ЗАПЯТОЙ БОЛЬШЕ ИЛИ РАВНО	Истина, если $C1 \geq C2$
	AND>=F	И С ПЛАВАЮЩЕЙ ЗАПЯТОЙ БОЛЬШЕ ИЛИ РАВНО	
	OR>=F	ИЛИ С ПЛАВАЮЩЕЙ ЗАПЯТОЙ БОЛЬШЕ ИЛИ РАВНО	

Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если S_1+1, S_1 или S_2+1, S_2 не является допустимым числом с плавающей запятой (NaN). Включен, если значение S_1+1, S_1 или S_2+1, S_2 равно $+\infty$. Включен, если значение S_1+1, S_1 или S_2+1, S_2 равно $-\infty$. Выключен во всех остальных случаях.
Флаг «Больше»	>	Включен, если $(S_1+1, S_1) > (S_2+1, S_2)$. Выключен во всех остальных случаях.
Флаг «Больше или равно»	> =	Включен, если $(S_1+1, S_1) \geq (S_2+1, S_2)$. Выключен во всех остальных случаях.
Флаг «Равно»	=	Включен, если $(S_1+1, S_1) = (S_2+1, S_2)$. Выключен во всех остальных случаях.
Флаг «Не равно»	=	Включен, если $(S_1+1, S_1) \neq (S_2+1, S_2)$. Выключен во всех остальных случаях.
Флаг «Меньше»	<	Включен, если $(S_1+1, S_1) < (S_2+1, S_2)$. Выключен во всех остальных случаях.
Флаг «Меньше или равно»	< =	Включен, если $(S_1+1, S_1) \leq (S_2+1, S_2)$. Выключен во всех остальных случаях.
Флаг отрицательного значения	N	Не изменяется

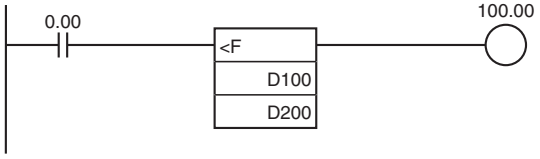
Меры предосторожности

Входная команда сравнения не может быть последней командой цепи. Между ней и правой шиной должна находиться как минимум еще одна команда.

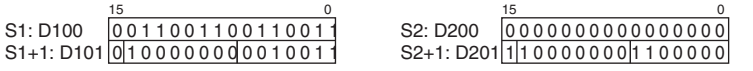
Пример

И С ПЛАВАЮЩЕЙ ЗАПЯТОЙ МЕНЬШЕ: AND<F(331)

Если в следующем примере бит CIO 0.00 включен, значение с плавающей запятой в D100 и D101 сравнивается со значением с плавающей запятой в D200 и D201. Если содержимое слов D100 и D101 меньше содержимого слов D200 и D201, включается бит CIO 100.00 и начинает выполняться следующая строка. Если содержимое слов D100 и D101 больше или равно содержимому D200 и D201, переход к следующей строке не происходит.



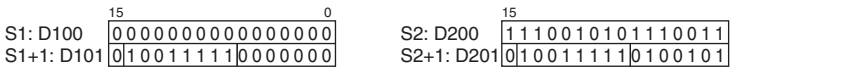
Сравнение: ПЛАВ. ЗАПЯТАЯ МЕНЬШЕ (<F)



Десятичное значение: 2,3 Десятичное значение: –3,5

2.3>-3.5

Не создает состояние “1” (ВКЛ).



Десятичное значение: 4 294 967 296 Десятичное значение: 5 566 555 656

4294967296<5566555656

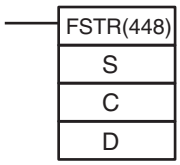
Создает состояние “1” (ВКЛ).

3-14-22 ПЛАВАЮЩАЯ ЗАПЯТАЯ В ASCII: FSTR(448)

Назначение

Представление 32-битового числа с плавающей запятой (в формате IEEE754) в десятичной или экспоненциальной записи и преобразование его в текстовую строку ASCII.

Символ РКС



S: Первое исходное слово
C: Первое управляющее слово
D: Первое слово назначения

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	FSTR(448)
	Однократное выполнение по положительному фронту	@FSTR(448)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

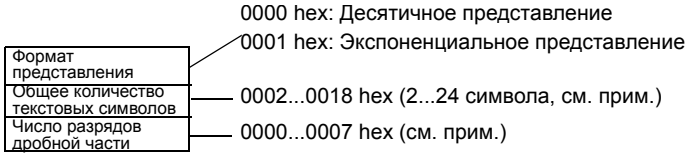
Характеристики операндов

Область	S	C	D
Область CIO	CIO 0...CIO 6142	CIO 0...CIO 6141	CIO 0...CIO 6143
Рабочая область	W0...W510	W0...W509	W0...W511
Область битов хранения	H0...H510	H0...H509	H0...H511
Область вспомогательных битов	A0...A958	A0...A957	A448...A959
Область таймеров	T0000...T4094	T0000...T4093	T0000...T4095
Область счетчиков	C0000...C4094	C0000...C4093	C0000...C4095
Область DM	D0...D32766	D0...D32765	D0...D32767
Косвенные адреса DM в двоичном формате	@ D0... @ D32767	@ D0... @ D32767	@ D0... @ D32767
Косвенные адреса DM в BCD-формате	*D0...*D32767	*D0...*D32767	*D0...*D32767
Постоянные	#00000000...#FFF FFFFF (двоичное)	---	
Регистры данных	---		
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –()IR15 ,IR0...,IR15		

Описание

Команда FSTR(448) представляет 32-битовое число с плавающей запятой в словах [S+1, S] (формат IEEE754) в десятичной или экспоненциальной записи в соответствии со значениями управляющих слов C...C+2, преобразует число в текстовую строку ASCII и записывает результат в указанные слова, начиная с D.

На следующем рисунке показано содержимое 3 управляющих слов.

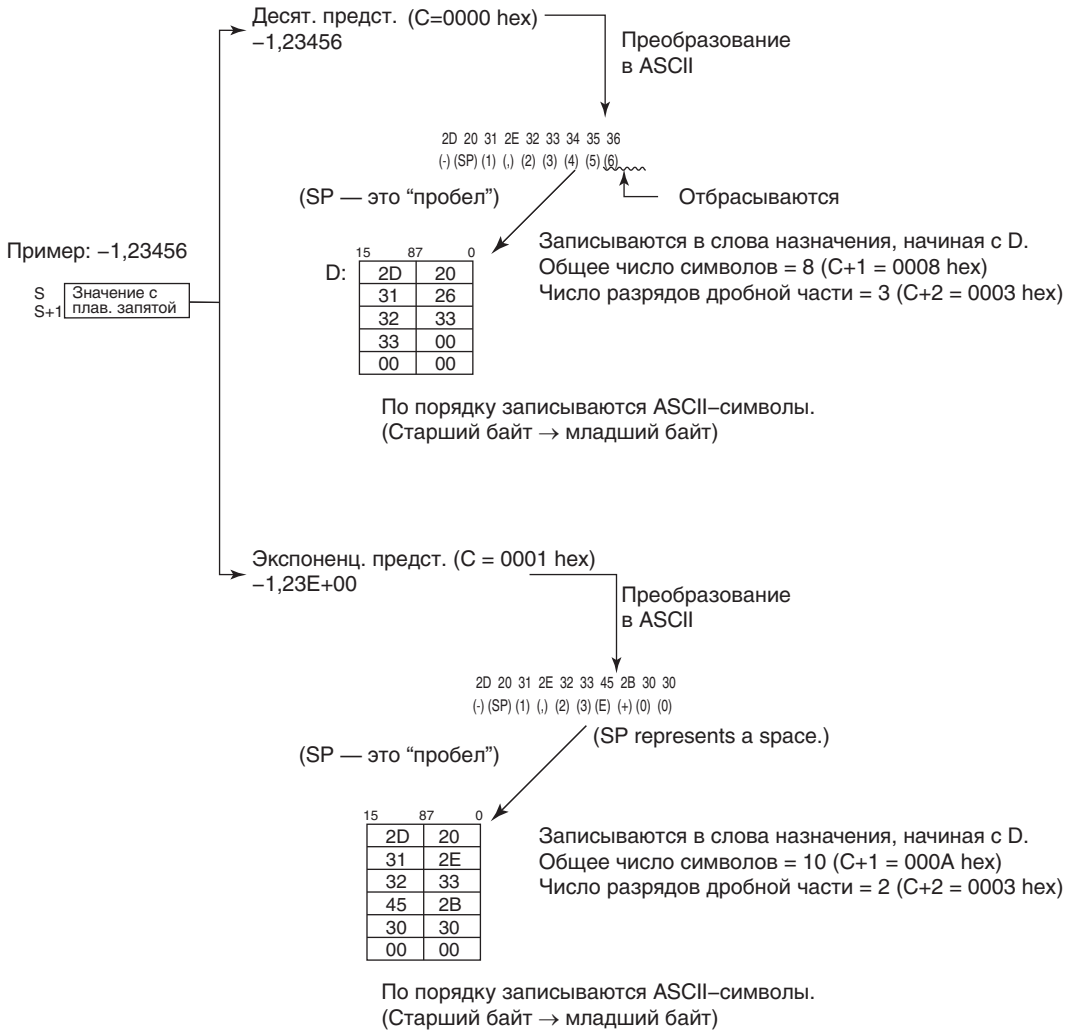


Примечание. Общее количество символов и количество разрядов дробной части ограничены.
См. *Ограничение числа символов ASCII* на стр. 566.

- В зависимости от содержимого слова C, определяющего формат представления, используется десятичное или экспоненциальное представление значения слов [S+1, S].
  - Десятичное представление
    Действительное число записывается в виде целой и дробной частей.
    Пример: 124.56
  - Экспоненциальное представление
    Действительное число записывается в виде целой, дробной и экспоненциальной частей.
    Пример: 1.2456E-2 (1.2456×10^{-2})
- Содержимое слова C+1 (Общее количество текстовых символов) определяет количество символов ASCII после преобразования, включая символы знака и десятичной точки, цифры и пробелы.

• Содержимое слова C+2 (Число разрядов дробной части) определяет количество разрядов (символов) после десятичной точки.

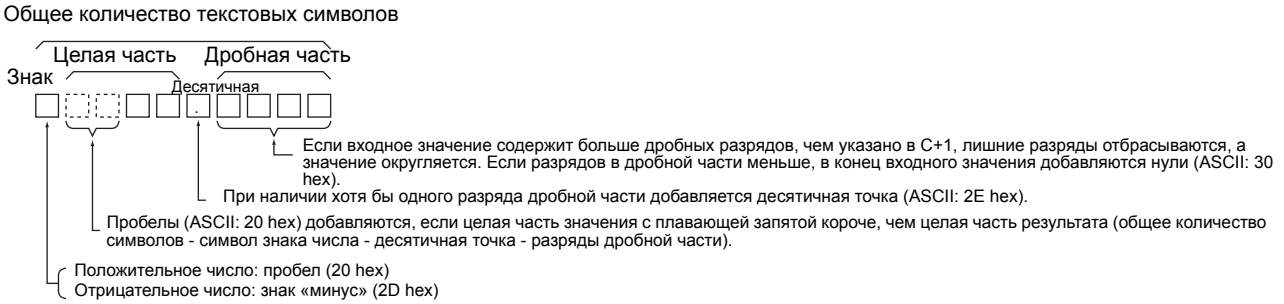
Текстовая строка ASCII записывается в D и следующие за ним слова в таком порядке: старший байт D, младший байт D, старший байт D+1, младший байт D+1 и т. д.



Хранение символов ASCII

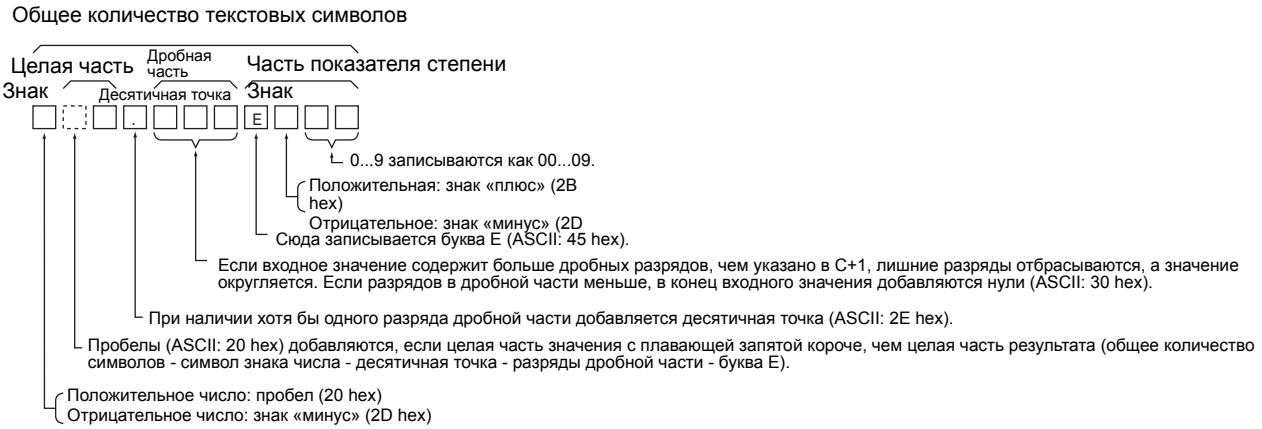
После преобразования числа с плавающей запятой в текстовую строку ASCII символы ASCII записываются в слова назначения, начиная с D, как показано на следующем рисунке. Для записи значений в десятичном и экспоненциальном представлениях используются разные методы.

Десятичное представление (C=0000 hex)



Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

Экспоненциальное представление (C=0001 hex)



Примечание. В конце текстовой строки ASCII в качестве кода завершения добавляются один или два байта нулей.
Общее количество текстовых символов нечетное: после текстовой строки ASCII записывается значение 00 hex.
Общее количество текстовых символов четное: после текстовой строки ASCII записывается значение 0000 hex.

Ограничение числа символов ASCII

Количество символов ASCII в результате преобразования ограничено. Если количество символов превышает допустимый максимум, устанавливается флаг ошибки.

1. Ограничение общего числа символов ASCII
   a. Десятичное представление (C = 0000 hex)
      - При отсутствии дробной части (C+2 = 0000 hex):
        2 ≤ Общее число символов ≤ 24
      - При наличии дробной части (C+2 = 0001...0007 hex):
        (разряды дробной части + 3) ≤ Общее число символов ≤ 24

   b. Экспоненциальное представление (C = 0001 hex)
      - При отсутствии дробной части (C+2 = 0000 hex):
        6 ≤ Общее число символов ≤ 24
      - При наличии дробной части (C+2 = 0001...0007 hex):
        (разряды дробной части + 7) ≤ Общее число символов ≤ 24
2. Ограничение общего числа разрядов целой части
   a. Десятичное представление (C = 0000 hex)
      - При отсутствии дробной части (C+2 = 0000 hex):
        1 ≤ Общее число разрядов целой части ≤ 24
      - При наличии дробной части (C+2 = 0001...0007 hex):
        1 ≤ Число разрядов целой части ≤ (24 - число разрядов дробной части - 2)

   b. Экспоненциальное представление (C = 0001 hex)
      1 разряд (фиксированный)
3. Ограничение общего числа разрядов дробной части
   a. Десятичное представление (C = 0000 hex)
      - Разряды дробной части ≤ 7
      - Также: разряды дробной части ≤ (общее число символов ASCII - 3)

   b. Экспоненциальное представление (C = 0001 hex)
      - Разряды дробной части ≤ 7
      - Также: разряды дробной части ≤ (общее число символов ASCII - 3)

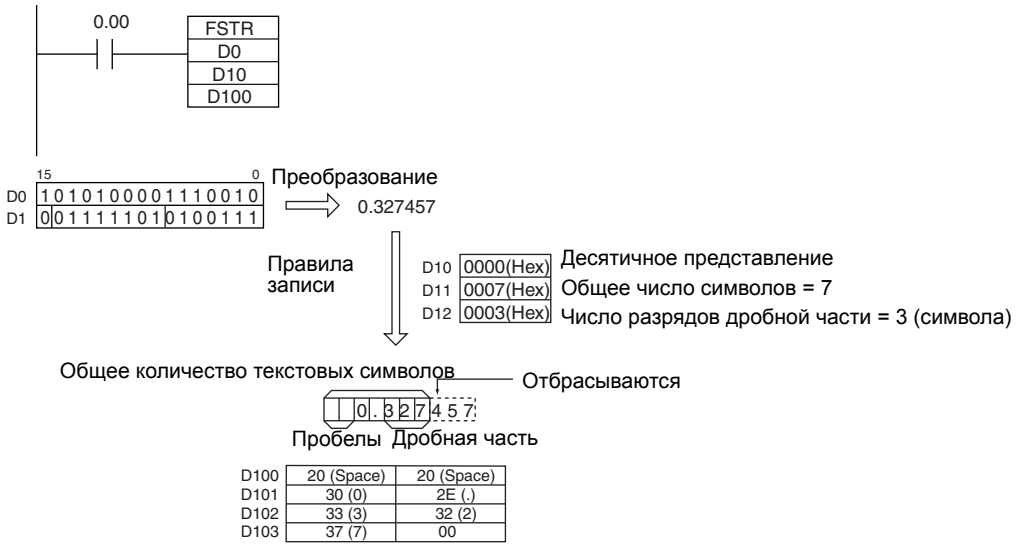
Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если содержимое [S+1, S] не является допустимым числом с плавающей запятой (NaN). Включен, если содержимое [S+1, S] равно +∞ или –∞. Включен, если содержимое управляющего слова C, определяющего формат представления, отлично от 0000 и 0001. Включен, если содержимое управляющего слова C+1, определяющего общее количество текстовых символов, выходит за пределы допустимого диапазона. (См. подробнее в п. *1. Ограничение общего числа символов ASCII* выше). Включен, если содержимое управляющего слова C+2, определяющего число разрядов дробной части, выходит за пределы допустимого диапазона. (См. подробнее в п. *3. Ограничение общего числа разрядов дробной части* выше). Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если результат преобразования = 0. Выключен во всех остальных случаях.

Примеры

Преобразование в текст ASCII в десятичном представлении

Если в следующем примере бит CIO 0.00 включен, команда FSTR(448) преобразует значение с плавающей запятой в D1 и D0 в текст ASCII в десятичном представлении и записывает текст ASCII в слова назначения, начиная с D100. Содержимое управляющих слов (D10...D12) определяет формат представления (десятичное представление, 7 текстовых символов, 3 разряда дробной части).

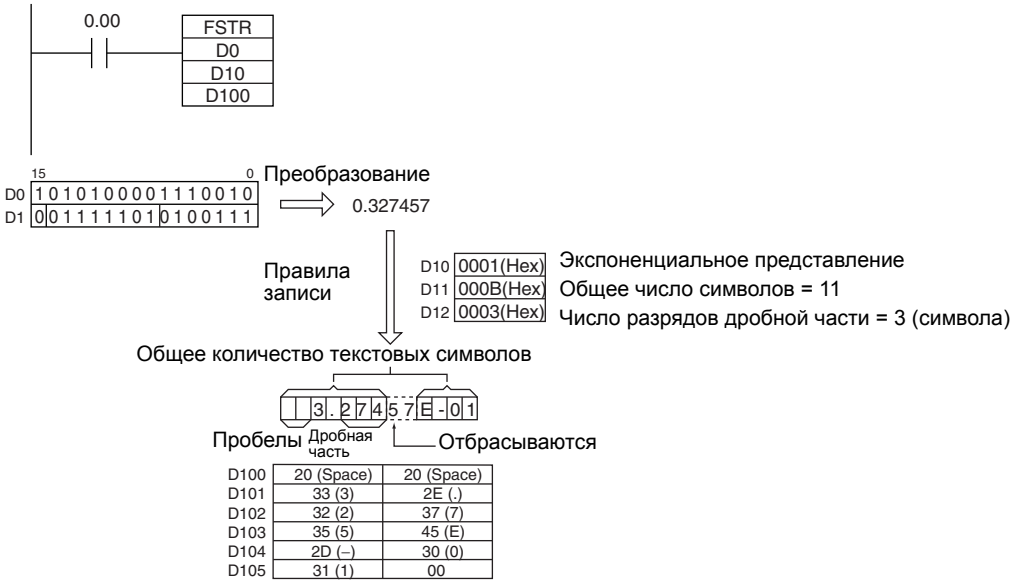


Преобразование в текст ASCII в экспоненциальном представлении

Если в следующем примере бит CIO 0.00 включен, команда FSTR(448) преобразует значение с плавающей запятой в D1 и D0 в текст ASCII в экспоненциальном представлении и записывает текст ASCII в слова назначения, начиная с D100. Содержимое управляющих слов (D10...D12)

Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

определяет формат представления (экспоненциальное представление, 11 текстовых символов, 3 разряда дробной части).

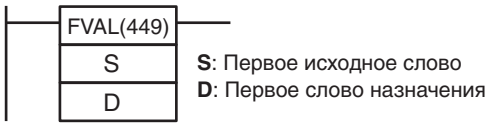


3-14-23 ASCII В ПЛАВАЮЩАЯ ЗАПЯТАЯ: FVAL(449)

Назначение

Преобразование текста ASCII, представляющего число с плавающей запятой (в десятичном или экспоненциальном представлении), в 32-битовое число с плавающей запятой (в формате IEEE754) и запись числа с плавающей запятой в указанные слова.

Символ РКС



Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	FVAL(449)
	Однократное выполнение по положительному фронту	@FVAL(449)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	S	D
Область CIO	CIO 0...CIO 6143	CIO 0...CIO 6142
Рабочая область	W0...W511	W0...W510
Область битов хранения	H0...H511	H0...H510
Область вспомогательных битов	A0...A959	A448...A958

Область	S	D
Область таймеров	T0000...T4095	T0000...T4094
Область счетчиков	C0000...C4095	C0000...C4094
Область DM	D0...D32767	D0...D32766
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Постоянные	---	
Регистры данных	---	
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –()IR15 ,IR0...,IR15	

Описание

Команда FVAL(449) преобразует указанное число в формате текстовой строки в кодировке ASCII (начиная со слова S) в 32-битовое число с плавающей запятой (в формате IEEE754) и записывает результат в слова назначения, начиная с D.

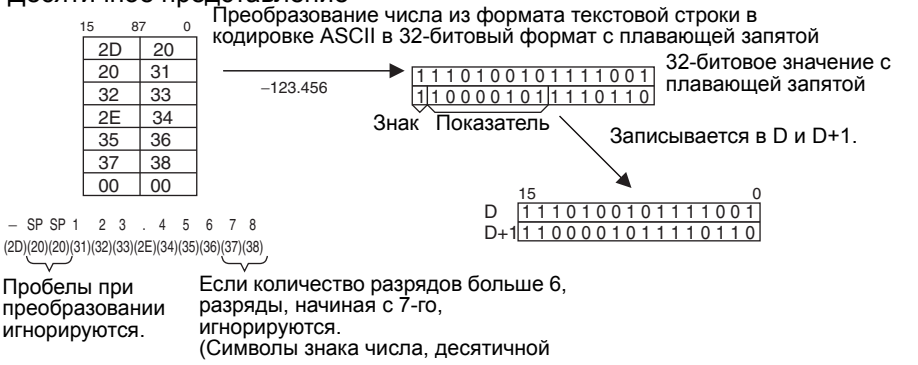
Команда FVAL(449) может преобразовать текст ASCII в десятичном или экспоненциальном представлении, если он соответствует следующим условиям:

- Десятичное представление
  Действительные числа, представленные в виде целой и дробной частей.
  Пример: 124,56
- Экспоненциальное представление
  Действительные числа, представленные в виде целой, дробной и экспоненциальной частей.
  Пример: 1.2456E-2 (1.2456×10^{-2})

Формат представления (десятичное или экспоненциальное) определяется автоматически.

Тестовая строка ASCII должна храниться в S и следующих за ним словах в таком порядке: старший байт S, младший байт S, старший байт S+1, младший байт S+1 и т. д.

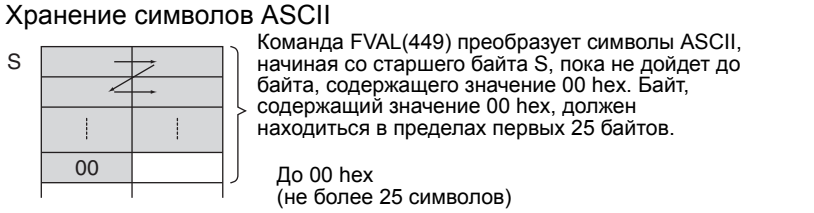
Десятичное представление

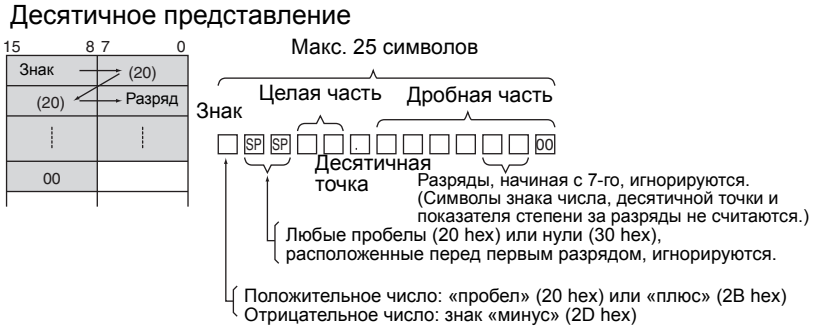


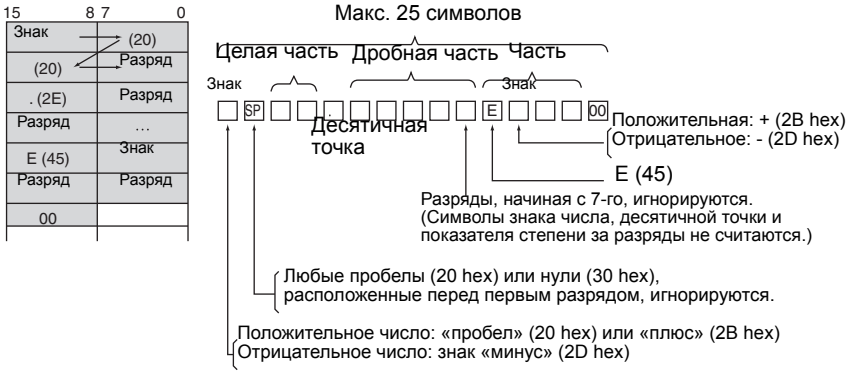


Хранение символов ASCII

На следующем рисунке показано преобразование числа в формате текстовой строки в кодировке ASCII в число с плавающей запятой. К числам, хранящихся в десятичном и экспоненциальном представлении, применяются разные методы преобразования.







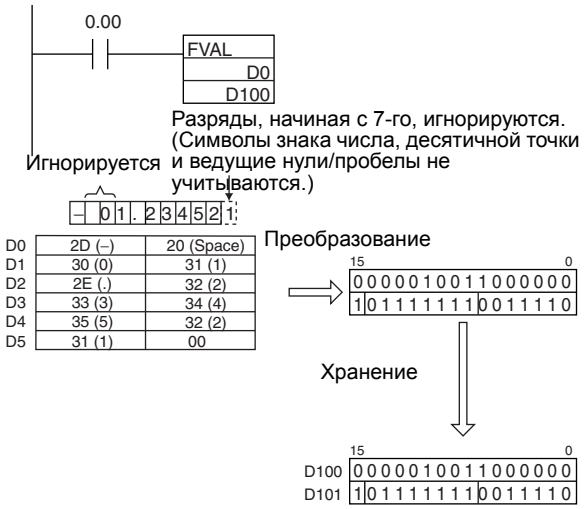
Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если разряды (целой и дробной части) исходных данных, начиная с S, не содержат значений из диапазона 30...39 hex (0...9). Включен, если значение двух первых разрядов экспоненциальной части отлично от 45 и 2B hex (E+) или 45 и 2D hex (E-). Включен, если в исходных данных присутствуют две экспоненциальные части или более. Включен, если результат преобразования равен +∞ или −∞. Включен, если в текстовой строке присутствует символ «0». Включен, если байт, содержащий значение 00 hex, не был найден в пределах первых 25 символов. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если результат преобразования = 0. Выключен во всех остальных случаях.

Примеры

Преобразование текста ASCII в десятичном представлении в значение с плавающей запятой

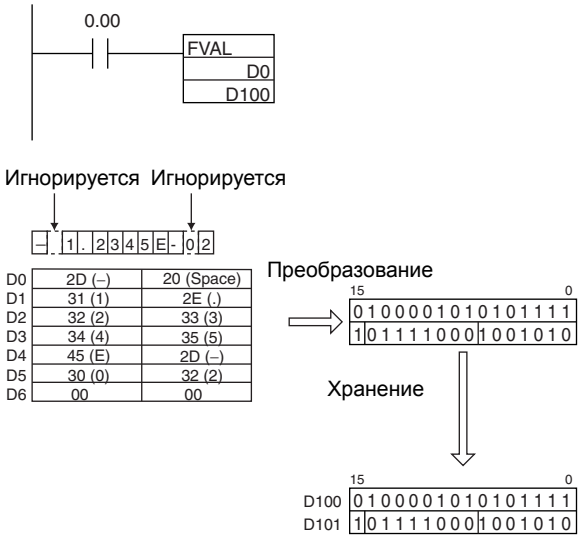
Если в следующем примере бит CIO 0.00 включен, команда FVAL(449) преобразует число в формате текстовой строки ASCII в десятичном представлении, указанное в исходных словах, начиная с D0, в число с плавающей запятой и записывает результат в слова назначения D100 и D101.



Преобразование текста ASCII в экспоненциальном представлении

Если в следующем примере бит CIO 0.00 включен, команда FVAL(449) преобразует число в формате текстовой строки ASCII в экспоненциальном представлении, указанное в исходных словах,

начиная с D0, в число с плавающей запятой и записывает результат в слова назначения D100 и D101.



3-15 Команды для чисел с плавающей запятой двойной точности

Команды для чисел с плавающей запятой двойной точности преобразуют значения и выполняют арифметические операции с числами с плавающей запятой двойной точности.

Команда	Мнемоническое обозначение	Код функции	Стр.
ПЛАВ. ЗАПЯТАЯ ДВОЙНОЙ ТОЧНОСТИ В 16-БИТ	FIXD	841	579
ПЛАВ. ЗАПЯТАЯ ДВОЙНОЙ ТОЧНОСТИ В 32-БИТ	FIXLD	842	581
16-БИТ В ПЛАВ. ЗАПЯТАЯ ДВОЙНОЙ ТОЧНОСТИ	DBL	843	582
32-БИТ В ПЛАВ. ЗАПЯТАЯ ДВОЙНОЙ ТОЧНОСТИ	DBLL	844	584
СЛОЖЕНИЕ С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ	+D	845	585
ВЫЧИТАНИЕ С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ	–D	846	587
УМНОЖЕНИЕ С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ	*D	847	589
ДЕЛЕНИЕ С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ	/D	848	591
ГРАДУСЫ В РАДИАНЫ ДВОЙНОЙ ТОЧНОСТИ	RADD	849	593
РАДИАНЫ В ГРАДУСЫ ДВОЙНОЙ ТОЧНОСТИ	DEGD	850	595
СИНУС ДВОЙНОЙ ТОЧНОСТИ	SIND	851	597
КОСИНУС ДВОЙНОЙ ТОЧНОСТИ	COSD	852	598
ТАНГЕНС ДВОЙНОЙ ТОЧНОСТИ	TAND	853	600
АРКСИНУС ДВОЙНОЙ ТОЧНОСТИ	ASIND	854	602
АРККОСИНУС ДВОЙНОЙ ТОЧНОСТИ	ACOSD	855	604
АРКТАНГЕНС ДВОЙНОЙ ТОЧНОСТИ	ATAND	856	606
КВАДРАТНЫЙ КОРЕНЬ ДВОЙНОЙ ТОЧНОСТИ	SQRTD	857	608
ЭКСПОНЕНТА ДВОЙНОЙ ТОЧНОСТИ	EXPD	858	610
ЛОГАРИФМ ДВОЙНОЙ ТОЧНОСТИ	LOGD	859	612
ВОЗВЕДЕНИЕ В СТЕПЕНЬ ДВОЙНОЙ ТОЧНОСТИ	PWRD	860	614
Символьные команды сравнения чисел с плавающей запятой двойной точности	LD, AND, OR + =D, <>D, <D, <=D, >D или >=D	335...340	616

Формат данных

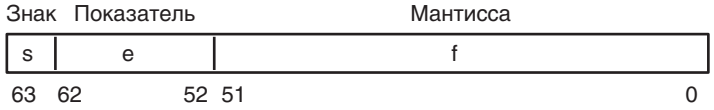
Для представления вещественного числа в формате с плавающей запятой используются знак, показатель степени и мантисса. Для представления вещественного значения в формате с плавающей запятой используется следующая формула.

Вещественное число = $(-1)^s\ 2^{e-1023}\ (1.f)$

s: знак
e: показатель степени
f: мантисса

Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

Формат представления чисел с плавающей запятой описан в стандарте IEEE754. Значение в формате с плавающей запятой представляется с помощью 32 битов следующим образом:



Данные	Кол-во битов	Содержание
s: знак	1	0: «+»; 1: «-»
e: показатель степени	11	Показатель степени (e) может принимать значения от 0 до 2047. Фактическое значение показателя рассчитывается путем вычитания 1023 из «e» и лежит в диапазоне -1023...1024. При этом значениям «e=0» и «e=2047» соответствуют особые состояния.
f: мантисса	52	Мантисса двоичного значения с запятой удовлетворяет неравенству 2,0 > 1,f ≥1,0.

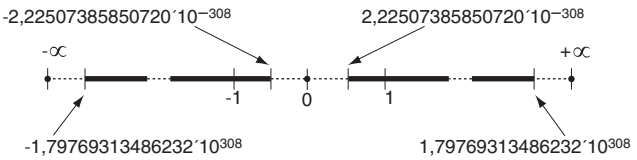
Количество разрядов

Количество значимых разрядов двоичного значения с плавающей запятой ограничивается 53 битами (около 15 десятичных знаков).

Значение с плавающей запятой

В формате с плавающей запятой могут быть выражены следующие значения:

- $-\infty$
- $-1{,}79769313486232 \times 10^{308} \leq$ значение $\leq -2{,}22507385850720 \times 10^{-308}$
- 0
- $2{,}22507385850720 \times 10^{-308} \leq$ значение $\leq 1{,}79769313486232 \times 10^{308}$
- $+\infty$
- NaN («не число»)



Особые состояния

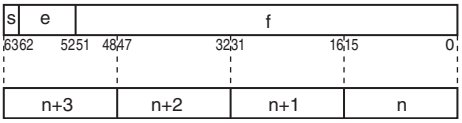
Для особых состояний, которыми являются NaN, $\pm\infty$ и 0, используются следующие форматы записи:

NaN*: e = 1024 и f ≠ 0
+∞: e = 1024, f = 0 и s= 0
–∞: e = 1024, f = 0 и s= 1
0: e = 0 и f = 0

*NaN («не число») не является каким-либо числом с плавающей запятой. Возможность получения значения NaN в результате выполнения команд вычислений с плавающей запятой двойной точности исключена.

Запись данных с плавающей запятой

Если в окне редактирования памяти ввода/вывода CX-Programmer выбран формат с плавающей запятой двойной точности, числа, введенные в стандартном десятичном формате, автоматически приводятся к описанному выше формату с плавающей запятой двойной точности (в соответствии со стандартом IEEE754) и записываются в память ввода/вывода. Данные, записанные в формате IEEE754, для отображения на дисплее автоматически приводятся к стандартному десятичному формату.



Для чтения и записи значений с плавающей запятой двойной точности пользователю не обязательно разбираться в тонкостях формата IEEE754. Достаточно помнить, что каждое значение с плавающей запятой двойной точности занимает четыре слова.

Типы чисел в формате с плавающей запятой

Могут использоваться следующие типы чисел с плавающей запятой.

Мантисса (f)	Показатель (e)		
	0	Не «0» и не все «1» (1024)	Все «1» (1024)
0	0	Нормализованное число	Бесконечность
Не «0»	Ненормализованное число		NaN

Примечание. Ненормализованным называется число, абсолютное значение которого настолько мало, что не может быть выражено как нормализованное число. Ненормализованные числа содержат меньшее количество значащих разрядов. Если результатом вычисления оказывается ненормализованное число (включая промежуточные результаты), количество значащих разрядов числа сокращается.

Нормализованные числа

Нормализованные числа используются для представления вещественных чисел. Для положительного числа бит знака равен «0», для отрицательного числа бит знака равен «1».

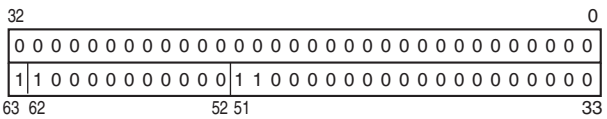
Показатель степени (e) может принимать значения от 1 до 2046, при этом фактическое значение показателя на 1023 меньше, т. е. изменяется от -1022 до 1023.

Мантисса (f) может принимать значения от 0 до $(2^{52} - 1)$, при этом предполагается, что у фактической мантиссы бит $2^{52} = 1$ и двоичная запятая располагается сразу после этого бита.

Для нормализованных чисел используется следующая форма представления:

$(-1)^{(\text{знак s})} \times 2^{(\text{показатель e})-1023} \times (1 + \text{мантисса} \times 2^{-52})$

Пример



Знак: -

Показатель: 1024 - 1023 = 1

Мантисса: $1 + (2^{51} + 2^{50}) \times 2^{-52} = 1 + (2^{-1} + 2^{-2}) = 1 + (0{,}75) = 1{,}75$

Значение: $-1{,}75 \times 2^{1} = -3{,}5$

Ненормализованные числа

Ненормализованными числами выражаются вещественные числа, имеющие очень маленькое абсолютное значение. Для положительного числа бит знака равен «0», для отрицательного числа бит знака равен «1».

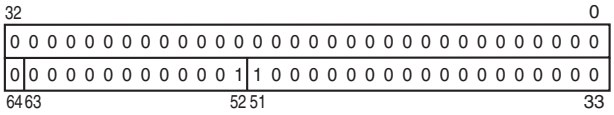
Показатель степени (e) равен «0», т. е. фактическое значение показателя: -1022.

Мантисса (f) может принимать значения от 1 до (2^{52} - 1), при этом предполагается, что у фактической мантиссы бит 2^{52} = 0 и двоичная запятая располагается сразу после этого бита.

Для ненормализованных чисел используется следующая форма представления:

$(-1)^{(знак\ s)} \times 2^{-1022} \times (мантисса \times 2^{-52})$

Пример



Знак: -

Показатель: -1022

Мантисса: $0 + (2^{51} + 2^{50}) \times 2^{-52} = 0 + (2^{-1} + 2^{-2}) = 0 + (0,75) = 0,75$

Значение: $-0,75 \times 2^{-1022} = 1,668805 \times 10^{-308}$

Нуль

Значения +0,0 и -0,0 могут быть выражены путем записи «0» в бит знака для положительного нуля и «1» — для отрицательного. Показатель степени и мантисса будут одновременно равны «0». Числа +0,0 и -0,0 эквивалентны 0,0. Информацию о том, на что влияет знак нуля, смотрите в разделе *Результаты арифметических операций с плавающей запятой* ниже.

Бесконечность

Значения +∞ и -∞ могут быть выражены путем записи «0» в бит знака для положительной бесконечной величины и «1» — для отрицательной. Показатель степени равен 2047 (2^{11} - 1), мантисса равна «0».

NaN

Состояние NaN («не число») создается, когда результат вычислений, например 0,0/0,0, ∞/∞ или ∞-∞, не является действительным числом или бесконечностью. Показатель степени равен 255 (2^8 - 1), мантисса не равна «0».

Примечание. Предписаний в отношение знака NaN или значения поля мантиссы не существует (за исключением того, что мантисса не должна быть равна «0»).

Результаты арифметических операций с плавающей запятой

Округление результатов

Если операция над числами с плавающей запятой дает результат, для точного представления которого требуется больше значащих разрядов, чем допускают внутренние вычислительные операции, результат округляется по одному из следующих правил.

Если результат близок к какому-то из двух внутренних выражений для числа с плавающей запятой, используется это выражение. Если результат находится посередине между двумя внутренними выражениями числа с плавающей запятой, он округляется таким образом, чтобы последний разряд мантиссы содержал «0».

Переполнение, потеря значимости и недопустимые операции

Если происходит переполнение, в качестве результата выдается положительная или отрицательная бесконечность, в зависимости от знака результата. Если происходит потеря значимости (отрицательное переполнение), в качестве результата выдается положительный или отрицательный ноль, в зависимости от знака результата.

Результатом недопустимой операции является состояние NaN («не число»). К недопустимым операциям относятся: сложение бесконечности с числом противоположного знака, вычитание бесконечности из числа с противоположным знаком, умножение ноля на бесконечность, деление ноля на ноль или деление бесконечности на бесконечность.

Если переполнение происходит при преобразовании числа из формата с плавающей запятой в целочисленный формат, результат преобразования может быть неверным.

Правила выполнения операций над специальными значениями

При выполнении операций над специальными значениями (ноль, бесконечность и NaN) применяются следующие правила.

- Сложение положительного ноля с отрицательным нолем дает положительный ноль.
- Вычитание ноля из ноля с таким же знаком дает положительный ноль.
- Если любым из операндов является NaN, результатом также будет NaN.
- В операциях сравнения положительный ноль и отрицательный ноль считаются равными друг другу.
- Результатом операции сравнения или определения равенства над одним или несколькими состояниями NaN всегда является «истина» для команды != и «ложь» для всех остальных команд.

Результаты вычислений с плавающей запятой двойной точности

Если результат по модулю превосходит максимальное значение, которое может быть представлено в формате с плавающей запятой, включается флаг переполнения и в качестве результата выдается $\pm\infty$. При положительном знаке результата выдается $+\infty$; при отрицательном выдается $-\infty$.

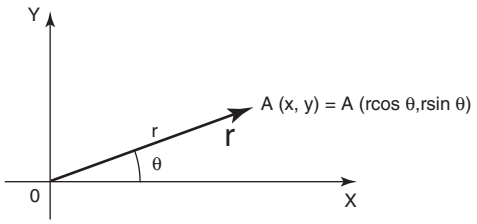
Флаг равенства устанавливается, только если в конце вычислений и показатель (e), и мантисса (f) равны нулю. В качестве результата вычислений также будет выдан ноль, если абсолютное значение результата меньше минимального значения, которое может быть выражено числом с плавающей запятой. В этом случае устанавливается флаг потери значимости.

Сравнение вычислений одинарной и двойной точности

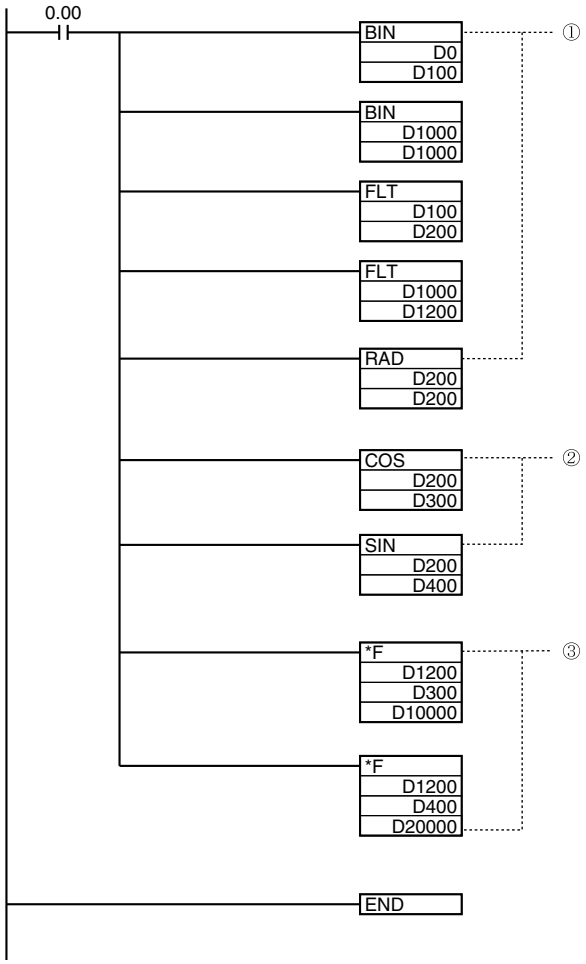
Разница между вычислениями одинарной и двойной точности демонстрируется на примере преобразования полярных координат указанного вектора в декартовы координаты A (x,y).

$$r = re^{j\left(\frac{\pi}{360}\right)\theta}$$

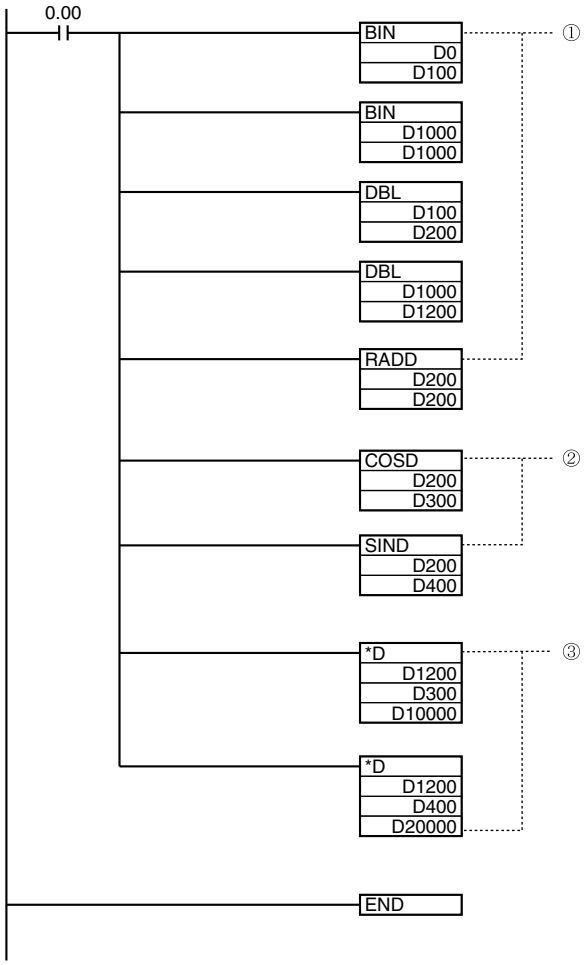
В данном примере из D0 считывается 4-разрядное двоично-десятичное значение угла (θ, в градусах), а из D1000 — 4-разрядное двоично-десятичное значение расстояния (r).



• Лестничная диаграмма для вычислений одинарной точности



• Лестничная диаграмма для вычислений двойной точности



1. Этот сегмент программы преобразует двоично-десятичное значение в значение с плавающей запятой одинарной точности (32 бита, формат IEEE754).
   a. Команды BIN(023) приводят двоично-десятичные значения к двоичному формату, а команды FLT(452) приводят двоичные значения к формату с плавающей запятой одинарной точности.
   b. Значение угла θ в формате с плавающей запятой выводится в слова D200 и D201.
   c. Команда RAD(458) преобразует значение угла в D200 и D201 в радианы.
   d. Значение радиуса r в формате с плавающей запятой выводится в слова D1200 и D1201.
2. Этот сегмент программы вычисляет sin θ и cos θ как значения с плавающей запятой одинарной точности.
   a. Значение cos θ выводится в слова D300 и D301.
   b. Значение sin θ выводится в слова D400 и D401.
3. Этот сегмент программы вычисляет x ($r \times \cos \theta$) и y ($r \times \sin \theta$).
   a. Значение x ($r \times \cos \theta$) выводится в слова D10000 и D10001.
   b. Значение y ($r \times \sin \theta$) выводится в слова D20000 и D20001.

Координата	Число с плавающей запятой	Действительное число
x	4116 59CF	3,4202015399933
y	405A E495	9,3969259262085

1. Этот сегмент программы преобразует двоично-десятичное значение в значение с плавающей запятой двойной точности (64 бита, формат IEEE754).
   a. Команды BIN(023) приводят двоично-десятичные значения к двоичному формату, а команды DBL(843) приводят двоичные значения к формату с плавающей запятой двойной точности.
   b. Значение угла θ в формате с плавающей запятой выводится в слова D200...D203.
   c. Команда RADD(849) преобразует значение угла в словах D200...D203 в радианы.
   d. Значение радиуса r в формате с плавающей запятой выводится в слова D1200...D1203.
2. Этот сегмент программы вычисляет sin θ и cos θ как значения с плавающей запятой двойной точности.
   a. Значение cos θ выводится в слова D300...D303.
   b. Значение sin θ выводится в слова D400...D403.
3. Этот сегмент программы вычисляет x ($r \times \cos \theta$) и y ($r \times \sin \theta$).
   a. Значение x ($r \times \cos \theta$) выводится в слова D10000...D10003.
   b. Значение y ($r \times \sin \theta$) выводится в слова D20000...D20003.

Координата	Число с плавающей запятой	Действительное число
x	4022 CB39 E973 5C32	3,4202014332567
y	400B 5C92 91AC 8EEB	9,3969262078591

Сравнение результатов вычислений

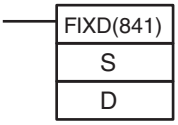
Из сравнения результатов в формате действительных чисел становится очевидным, что вычисления двойной точности позволяют получить более точный результат.

3-15-1 ПЛАВ. ЗАПЯТАЯ ДВОЙНОЙ ТОЧНОСТИ В 16-БИТ: FIXD(841)

Назначение

Преобразование указанного 64-битового значения с плавающей запятой двойной точности в 16-битовое двоичное значение со знаком и запись результата в указанное слово.

Символ РКС



S: Первое исходное слово
D: Слово назначения

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	FIXD(841)
	Однократное выполнение по положительному фронту	@FIXD(841)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

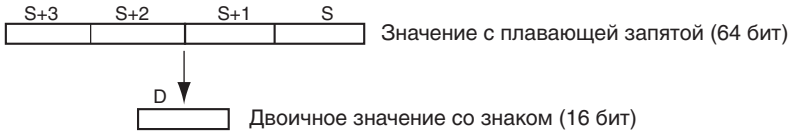
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	S	D
Область CIO	CIO 0...CIO 6140	CIO 0...CIO 6143
Рабочая область	W0...W508	W0...W511
Область битов хранения	H0...H508	H0...H511
Область вспомогательных битов	A0...A956	A448...A959
Область таймеров	T0000...T4092	T0000...T4095
Область счетчиков	C0000...C4092	C0000...C4095
Область DM	D0...D32764	D0...D32767
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Постоянные	---	
Регистры данных	---	DR0...DR15
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

Описание

Команда FIXD(841) преобразует целую часть 64-битового числа с плавающей запятой двойной точности в словах S...S+3 (в формате IEEE754) в 16-битовое двоичное число со знаком и записывает результат в D.



Преобразуется только целая часть числа с плавающей запятой, дробная часть отбрасывается. Целая часть числа с плавающей запятой должна находиться в диапазоне от -32768 до 32767.

Примеры преобразования:
Значение с плавающей запятой 3,5 преобразуется в 3.
Значение с плавающей запятой -3,5 преобразуется в -3.

Флаги

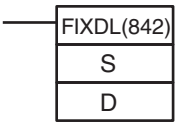
Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если входное значение (S...S+3) не является числом (NaN). Включен, если целая часть входного значения (S...S+3) находится за пределами диапазона -32768...32767. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если результат = 0000. Выключен во всех остальных случаях.
Флаг отрицательного значения	N	Включен, если бит 15 результата = «1». Выключен во всех остальных случаях.

3-15-2 ПЛАВ. ЗАПЯТАЯ ДВОЙНОЙ ТОЧНОСТИ В 32-БИТ: FIXLD(842)

Назначение

Преобразование указанного 64-битового значения с плавающей запятой двойной точности в 32-битовое двоичное значение со знаком и запись результата в указанные слова.

Символ РКС



S: Первое исходное слово
D: Первое слово назначения

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	FIXLD(842)
	Однократное выполнение по положительному фронту	@FIXLD(842)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

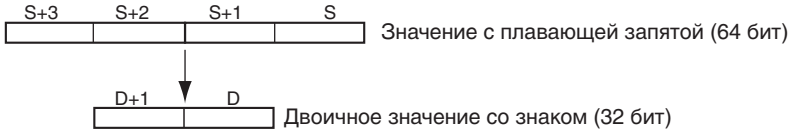
Характеристики операндов

Область	S	D
Область CIO	CIO 0...CIO 6140	CIO 0...CIO 6142
Рабочая область	W0...W508	W0...W510
Область битов хранения	H0...H508	H0...H510
Область вспомогательных битов	A0...A956	A448...A958
Область таймеров	T0000...T4092	T0000...T4094
Область счетчиков	C0000...C4092	C0000...C4094
Область DM	D0...D32764	D0...D32766
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Постоянные	---	
Регистры данных	---	

Область	S	D
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

Описание

Команда FIXLD(842) преобразует целую часть 64-битового числа с плавающей запятой двойной точности в словах S...S+3 (в формате IEEE754) в 32-битовое двоичное число со знаком и записывает результат в D+1 и D.



Преобразуется только целая часть числа с плавающей запятой, дробная часть отбрасывается. (Целая часть числа с плавающей запятой должна находиться в диапазоне от -2 147 483 648 до 2 147 483 647.)

Примеры преобразования:
Значение с плавающей запятой 2 147 483 640,5 преобразуется в 2 147 483 640.
Значение с плавающей запятой -2 147 483 640,5 преобразуется в -2 147 483 640.

Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если входное значение в словах S...S+3 не является числом (NaN). Включен, если целая часть значения в словах S...S+3 находится за пределами диапазона -2 147 483 648... 2 147 483 647. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если результат = 0000 0000. Выключен во всех остальных случаях.
Флаг отрицательного значения	N	Включен, если после выполнения команды бит 15 слова D+1 = «1». Выключен во всех остальных случаях.

Меры предосторожности

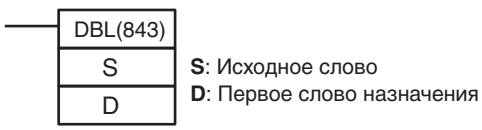
Содержимое слов S...S+3 должно быть значением с плавающей запятой, целая часть которого принадлежит диапазону -2 147 483 648...2 147 483 647.

3-15-3 16-БИТ В ПЛАВ. ЗАПЯТАЯ ДВОЙНОЙ ТОЧНОСТИ: DBL(843)

Назначение

Преобразование 16-битового двоичного значения со знаком в эквивалентное 64-битовое значение с плавающей запятой двойной точности и запись результата в указанные слова назначения.

Символ РКС



Предварительная версия
Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	DBL(843)
	Однократное выполнение по положительному фронту	@DBL(843)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

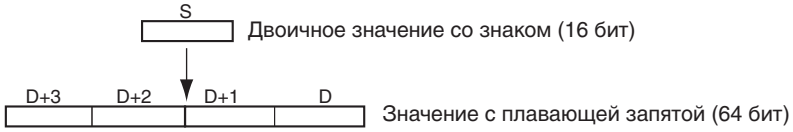
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	S	D
Область CIO	CIO 0...CIO 6143	CIO 0...CIO 6140
Рабочая область	W0...W511	W0...W508
Область битов хранения	H0...H511	H0...H508
Область вспомогательных битов	A0...A959	A448...A956
Область таймеров	T0000...T4095	T0000...T4092
Область счетчиков	C0000...C4095	C0000...C4092
Область DM	D0...D32767	D0...D32764
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Постоянные	#0000...#FFFF (двоичн.)	---
Регистры данных	DR0...DR15	---
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

Описание

Команда DBL(843) преобразует 16-битовое двоичное значение со знаком (S) в 64-битовое значение с плавающей запятой двойной точности в формате IEEE754 и записывает результат в слова D...D+3. В слове результата после десятичной запятой добавляется 0.



В операнде S можно указывать только значения, принадлежащие диапазону -32768...32767. Для преобразования двоичных значений со знаком, выходящих за пределы этого диапазона, используется команда DBLL(844).

Примеры преобразования:
Двоичное значение со знаком 3 преобразуется в 3,0.
Двоичное значение со знаком -3 преобразуется в -3,0.

Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Выключен
Флаг равенства	=	Включен, если и показатель, и мантисса результата равны «0». Выключен во всех остальных случаях.
Флаг отрицательного значения	N	Включен, если результат отрицателен. Выключен во всех остальных случаях.

Меры предосторожности

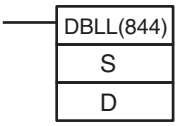
S должно содержать двоичное значение со знаком, десятичный эквивалент которого должен находиться в диапазоне от -32768 до 32767.

3-15-4 32-БИТ В ПЛАВ. ЗАПЯТАЯ ДВОЙНОЙ ТОЧНОСТИ: DBLL(844)

Назначение

Преобразование 32-битового двоичного значения со знаком в эквивалентное 64-битовое значение с плавающей запятой двойной точности и запись результата в указанные слова назначения.

Символ РКС



S: Первое исходное слово
D: Первое слово назначения

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	DBLL(844)
	Однократное выполнение по положительному фронту	@DBLL(844)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

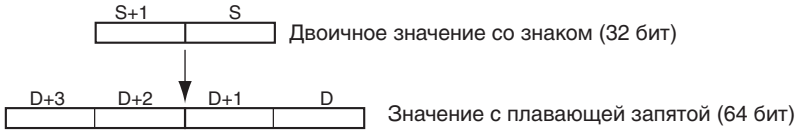
Характеристики операндов

Область	S	D
Область CIO	CIO 0...CIO 6142	CIO 0...CIO 6140
Рабочая область	W0...W510	W0...W508
Область битов хранения	H0...H510	H0...H508
Область вспомогательных битов	A0...A958	A448...A956
Область таймеров	T0000...T4094	T0000...T4092
Область счетчиков	C0000...C4094	C0000...C4092
Область DM	D0...D32766	D0...D32764
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Постоянные	#00000000...#FFFFFFFF (двоичное)	---
Регистры данных	---	

Область	S	D
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

Описание

Команда DBLL(844) преобразует 32-битовое двоичное значение со знаком в S+1 и S в 64-битовое значение с плавающей запятой двойной точности в формате IEEE754 и записывает результат в слова D...D+3. В слове результата после десятичной запятой добавляется 0.



S+1 и S должны содержать или указывать на двоичное значение со знаком в диапазоне от -2 147 483 648 до 2 147 483 647. Значение в формате с плавающей запятой содержит 24 значащих двоичных разряда (бита). В случае использования команды DBLL(844) для преобразования значения больше 16 777 215 (самого большого числа, которое можно представить 24 битами) результат будет неточным.

Примеры преобразования:

Двоичное значение со знаком 16 777 215 преобразуется в 16 777 215,0.
Двоичное значение со знаком -16 777 215 преобразуется в -15 777 215,0.

Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Выключен
Флаг равенства	=	Включен, если и показатель, и мантисса результата равны «0». Выключен во всех остальных случаях.
Флаг отрицательного значения	N	Включен, если результат отрицателен. Выключен во всех остальных случаях.

Меры предосторожности

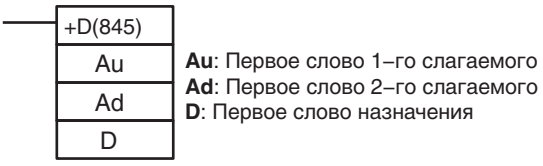
Если абсолютное значение числа больше, чем 16 777 215 (максимальное значение, которое может быть выражено 24 битами), результат преобразования такого числа будет неточным.

3-15-5 СЛОЖЕНИЕ С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ: +D(845)

Назначение

Сложение указанных чисел с плавающей запятой двойной точности (каждое по 64 бита) и запись результата в указанные слова.

Символ РКС



Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	+D(845)
	Однократное выполнение по положительному фронту	@+D(845)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

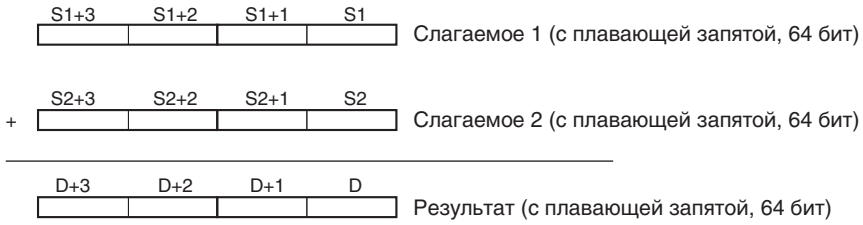
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	Au	Ad	D
Область CIO	CIO 0...CIO 6140		
Рабочая область	W0...W508		
Область битов хранения	H0...H508		
Область вспомогательных битов	A0...A956		A448...A956
Область таймеров	T0000...T4092		
Область счетчиков	C0000...C4092		
Область DM	D0...D32764		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	---		
Регистры данных	---		
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда +D(845) добавляет 64-битовое число с плавающей запятой двойной точности в словах Ad...Ad+3 к 64-битовому числу с плавающей запятой двойной точности в словах Au...Au+3 и записывает результат в слова D...D+3. (Значения с плавающей запятой должны быть указаны в формате IEEE754.)



Если результат по модулю превосходит максимальное значение, которое может быть представлено в формате с плавающей запятой, включается флаг переполнения и в качестве результата выдается $\pm\infty$.

Если абсолютное значение результата меньше минимального значения, которое может быть выражено числом с плавающей запятой, включается флаг потери значимости и в качестве результата выдается «0».

В следующей таблице приведены результаты выполнения сложения для разных комбинаций первого и второго слагаемых.

	Слагаемое 1				
Слагаемое 2	**0**	**Число**	**+∞**	**–∞**	**NaN**
0	0	Число	+∞	–∞	
Число	Число	См. примеч. 1	+∞	–∞	
+∞	+∞	+∞	+∞	См. примеч. 2	
–∞	–∞	–∞	См. примеч. 2	–∞	
NaN					См. примеч. 2

Примечание.
(1) Результатом может быть ноль (включая потерю значимости), числовое значение, +∞ или -∞.
(2) Устанавливается флаг ошибки, команда не выполняется.

Флаги

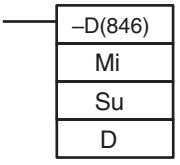
Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если значение первого или второго слагаемого не опознано как значение с плавающей запятой. Включен, если значение первого или второго слагаемого не является числом (NaN). Включен, если +∞ складывается с –∞. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если и показатель, и мантисса результата равны «0». Выключен во всех остальных случаях.
Флаг переполнения	OF	Включен, если абсолютное значение результата слишком велико и не может быть выражено числом с плавающей запятой двойной точности.
Флаг потери значимости	UF	Включен, если абсолютное значение результата слишком мало и не может быть выражено числом с плавающей запятой двойной точности.
Флаг отрицательного значения	N	Включен, если результат отрицателен. Выключен во всех остальных случаях.

Меры предосторожности
Значения первого (Au...Au+3) и второго (Ad...Ad+3) слагаемых должны быть представлены в формате с плавающей запятой в соответствии с IEEE754.

3-15-6 ВЫЧИТАНИЕ С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ: –D(846)

Назначение
Вычитание одного 64-битового числа с плавающей запятой двойной точности из другого и запись результата в указанные слова.

Символ РКС



Mi: Первое слово уменьшаемого
Su: Первое слово вычитаемого
D: Первое слово назначения

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	–D(846)
	Однократное выполнение по положительному фронту	@–D(846)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

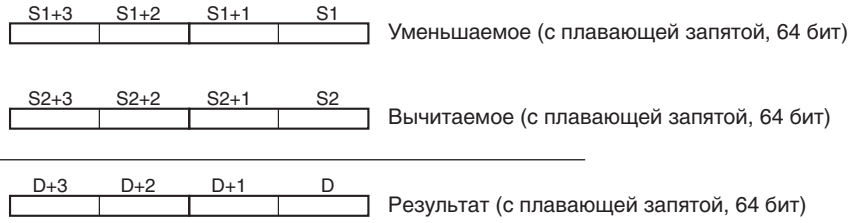
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	Mi	Su	D
Область CIO	CIO 0...CIO 6140		
Рабочая область	W0...W508		
Область битов хранения	H0...H508		
Область вспомогательных битов	A0...A956		A448...A956
Область таймеров	T0000...T4092		
Область счетчиков	C0000...C4092		
Область DM	D0...D32764		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	---		
Регистры данных	---		
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда –D(846) вычитает 64-битовое число с плавающей запятой двойной точности в словах Su...Su+3 из 64-битового числа с плавающей запятой двойной точности в словах Mi...Mi+3 и записывает результат в слова D...D+3. (Значения с плавающей запятой должны быть указаны в формате IEEE754.)



Если результат по модулю превосходит максимальное значение, которое может быть представлено в формате с плавающей запятой, включается флаг переполнения и в качестве результата выдается $\pm\infty$.

Если абсолютное значение результата меньше минимального значения, которое может быть выражено числом с плавающей запятой, включается флаг потери значимости и в качестве результата выдается «0».

В следующей таблице приведены результаты выполнения вычитания для разных комбинаций уменьшаемого и вычитаемого.

	Уменьшаемое				
Вычитаемое	**0**	**Число**	**+∞**	**–∞**	**NaN**
0	0	Число	+∞	–∞	
Число	Число	См. примеч. 1	+∞	–∞	
+∞	–∞	–∞	См. примеч. 2	–∞	
–∞	+∞	+∞	+∞	См. примеч. 2	
NaN					См. примеч. 2

Примечание.

(1) Результатом может быть ноль (включая потерю значимости), числовое значение, +∞ или -∞.

(2) Устанавливается флаг ошибки, команда не выполняется.

Флаги

Название	**Обозначение**	**Описание работы**
Флаг ошибки	ER	Включен, если значение уменьшаемого или вычитаемого не опознано как значение с плавающей запятой. Включен, если значение уменьшаемого или вычитаемого не является числом (NaN). Включен, если +∞ вычитается из +∞. Включен, если -∞ вычитается из -∞. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если и показатель, и мантисса результата равны «0». Выключен во всех остальных случаях.
Флаг переполнения	OF	Включен, если абсолютное значение результата слишком велико и не может быть выражено числом с плавающей запятой двойной точности.
Флаг потери значимости	UF	Включен, если абсолютное значение результата слишком мало и не может быть выражено числом с плавающей запятой двойной точности.
Флаг отрицательного значения	N	Включен, если результат отрицателен. Выключен во всех остальных случаях.

Меры предосторожности

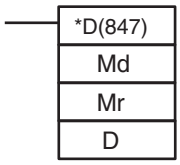
Значения уменьшаемого (Mi...Mi+3) и вычитаемого (Su...Su+3) должны быть представлены в формате с плавающей запятой в соответствии с IEEE754.

3-15-7 УМНОЖЕНИЕ С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ: ∗D(847)

Назначение

Умножение указанных чисел с плавающей запятой двойной точности (каждое по 64 бита) и запись результата в указанные слова.

Символ РКС



Md: Первое слово умножаемого
Mr: Первое слово множителя
D: Первое слово назначения

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	*D(847)
	Однократное выполнение по положительному фронту	@*D(847)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

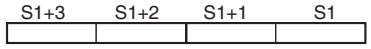
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

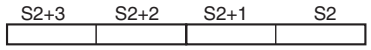
Область	Md	Mr	D
Область CIO	CIO 0...CIO 6140		
Рабочая область	W0...W508		
Область битов хранения	H0...H508		
Область вспомогательных битов	A0...A956		A448...A956
Область таймеров	T0000...T4092		
Область счетчиков	C0000...C4092		
Область DM	D0...D32764		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	---		
Регистры данных	---		
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

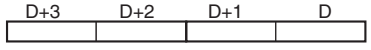
Команда ∗D(847) умножает 64-битовое число с плавающей запятой двойной точности в словах Md...Md+3 на 64-битовое число с плавающей запятой двойной точности в словах Mr...Mr+3 и записывает результат в слова D...D+3. (Значения с плавающей запятой должны быть указаны в формате IEEE754.)



Умножаемое (с плавающей запятой, 64 бит)



Множитель (с плавающей запятой, 64 бит)



Результат (с плавающей запятой, 64 бит)

Если результат по модулю превосходит максимальное значение, которое может быть представлено в формате с плавающей запятой, включается флаг переполнения и в качестве результата выдается ±∞.

Если абсолютное значение результата меньше минимального значения, которое может быть выражено числом с плавающей запятой, включается флаг потери значимости и в качестве результата выдается «0».

В следующей таблице приведены результаты выполнения умножения для разных комбинаций умножаемого и множителя.

	Умножаемое				
Множи-тель	**0**	**Число**	**+∞**	**–∞**	**NaN**
0	0	0	См. примеч. 2	См. примеч. 2	
Число	0	См. примеч. 1	+/–∞	+/–∞	
+∞	См. примеч. 2	+/–∞	+∞	–∞	
–∞	См. примеч. 2	+/–∞	–∞	+∞	
NaN					См. примеч. 2

Примечание.

(1) Результатом может быть ноль (включая потерю значимости), числовое значение, +∞ или -∞.

(2) Устанавливается флаг ошибки, команда не выполняется.

Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если значение умножаемого или множителя не опознано как значение с плавающей запятой. Включен, если значение умножаемого или множителя не является числом (NaN). Включен, если +∞ умножается на 0. Включен, если -∞ умножается на 0. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если и показатель, и мантисса результата равны «0». Выключен во всех остальных случаях.
Флаг переполнения	OF	Включен, если абсолютное значение результата слишком велико и не может быть выражено числом с плавающей запятой двойной точности.
Флаг потери значимости	UF	Включен, если абсолютное значение результата слишком мало и не может быть выражено числом с плавающей запятой двойной точности.
Флаг отрицательного значения	N	Включен, если результат отрицателен. Выключен во всех остальных случаях.

Меры предосторожности

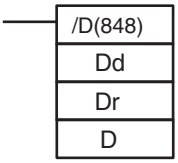
Значения умножаемого (Md...Md+3) и множителя (Mr...Mr+3) должны быть представлены в формате с плавающей запятой в соответствии с IEEE754.

3-15-8 ДЕЛЕНИЕ С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ: /D(848)

Назначение

Деление одного числа с плавающей запятой двойной точности (64 бита) на другое и запись результата в указанные слова.

Символ РКС



Dd: Первое слово делимого
Dr: Первое слово делителя
D: Первое слово назначения

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	/D(848)
	Однократное выполнение по положительному фронту	@/D(848)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

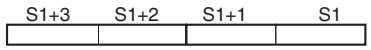
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	Dd	Dr	D
Область CIO	CIO 0...CIO 6140		
Рабочая область	W0...W508		
Область битов хранения	H0...H508		
Область вспомогательных битов	A0...A956		A448...A956
Область таймеров	T0000...T4092		
Область счетчиков	C0000...C4092		
Область DM	D0...D32764		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	---		
Регистры данных	---		
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

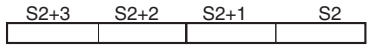
Описание

Команда /D(848) делит 64-битовое число с плавающей запятой двойной точности в словах Dd...Dd+3 на 64-битовое число с плавающей запятой двойной точности в словах Dr...Dr+3 и записывает результат в слова D...D+3. (Значения с плавающей запятой должны быть указаны в формате IEEE754.)

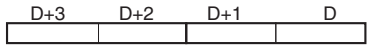


Делимое (с плавающей запятой, 64 бит)

,



Делитель (с плавающей запятой, 64 бит)



Результат (с плавающей запятой, 64 бит)

Если результат по модулю превосходит максимальное значение, которое может быть представлено в формате с плавающей запятой, включается флаг переполнения и в качестве результата выдается $\pm\infty$.

Если абсолютное значение результата меньше минимального значения, которое может быть выражено числом с плавающей запятой, включается флаг потери значимости и в качестве результата выдается «0».

В следующей таблице приведены результаты выполнения деления для разных комбинаций делимого и делителя.

	Делимое				
Делитель	**0**	**Число**	**+∞**	**–∞**	**NaN**
0	См. примеч. 3	+/–∞	+∞	–∞	
Число	0	См. примеч. 1	+/–∞	+/–∞	
+∞	0	См. примеч. 2	См. примеч. 3	См. примеч. 3	
–∞	0	См. примеч. 2	См. примеч. 3	См. примеч. 3	
NaN					См. примеч. 3

Примечание.

(1) Результатом может быть ноль (включая потерю значимости), числовое значение, +∞ или -∞.

(2) При потере значимости результатом будет «0».

(3) Устанавливается флаг ошибки, команда не выполняется.

Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если значение делимого или делителя не опознано как значение с плавающей запятой. Включен, если значение делимого или делителя не является числом (NaN). Включен, если и делимое и делитель оба равны «0». Включен, если и делимое и делитель оба равны +∞ или -∞. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если и показатель, и мантисса результата равны «0». Выключен во всех остальных случаях.
Флаг переполнения	OF	Включен, если абсолютное значение результата слишком велико и не может быть выражено числом с плавающей запятой двойной точности.
Флаг потери значимости	UF	Включен, если абсолютное значение результата слишком мало и не может быть выражено числом с плавающей запятой двойной точности.
Флаг отрицательного значения	N	Включен, если результат отрицателен. Выключен во всех остальных случаях.

Меры предосторожности

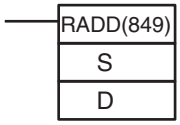
Значения делимого (Dd...Dd+3) и делителя (Dr...Dr+3) должны быть представлены в формате с плавающей запятой в соответствии с IEEE754.

3-15-9 ГРАДУСЫ В РАДИАНЫ ДВОЙНОЙ ТОЧНОСТИ: RADD(849)

Назначение

Преобразование указанного 64-битового числа с плавающей запятой двойной точности из градусов в радианы и запись результата в указанные слова.

Символ РКС



S: Первое исходное слово
D: Первое слово назначения

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	RADD(849)
	Однократное выполнение по положительному фронту	@RADD(849)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

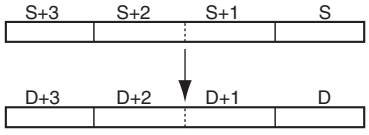
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	S	D
Область CIO	CIO 0...CIO 6140	
Рабочая область	W0...W508	
Область битов хранения	H0...H508	
Область вспомогательных битов	A0...A956	A448...A956
Область таймеров	T0000...T4092	
Область счетчиков	C0000...C4092	
Область DM	D0...D32764	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Постоянные	---	
Регистры данных	---	
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

Описание

Команда RADD(849) преобразует 64-битовое число с плавающей запятой двойной точности в словах S...S+3 из градусов в радианы и записывает результат в слова D...D+3. (Входное значение с плавающей запятой должно быть указано в формате IEEE754.)



Исходное значение
(градусы, с плавающей запятой, 64 бит)

Результат
(радианы, с плавающей запятой, 64 бит)

Преобразование из градусов в радианы производится по следующей формуле:

Значение в градусах $\times \pi/180$ = значение в радианах

Если результат по модулю превосходит максимальное значение, которое может быть представлено в формате с плавающей запятой, включается флаг переполнения и в качестве результата выдается $\pm\infty$.

Если абсолютное значение результата меньше минимального значения, которое может быть выражено числом с плавающей запятой, включается флаг потери значимости и в качестве результата выдается «0».

Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если входное значение не опознано как значение с плавающей запятой. Включен, если входное значение не является числом (NaN). Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если и показатель, и мантисса результата равны «0». Выключен во всех остальных случаях.
Флаг переполнения	OF	Включен, если абсолютное значение результата слишком велико и не может быть выражено числом с плавающей запятой двойной точности.
Флаг потери значимости	UF	Включен, если абсолютное значение результата слишком мало и не может быть выражено числом с плавающей запятой двойной точности.
Флаг отрицательного значения	N	Включен, если результат отрицателен. Выключен во всех остальных случаях.

Меры предосторожности

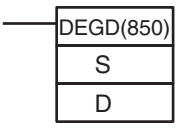
Входное значение в словах S...S+3 должно быть представлено в формате с плавающей запятой в соответствии с IEEE754.

3-15-10 РАДИАНЫ В ГРАДУСЫ ДВОЙНОЙ ТОЧНОСТИ: DEGD(850)

Назначение

Преобразование указанного 64-битового числа с плавающей запятой двойной точности из радианов в градусы и запись результата в указанные слова.

Символ РКС



S: Первое исходное слово
D: Первое слово назначения

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	DEGD(850)
	Однократное выполнение по положительному фронту	@DEGD(850)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

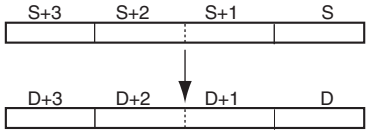
Характеристики операндов

Область	S	D
Область CIO	CIO 0...CIO 6140	
Рабочая область	W0...W508	
Область битов хранения	H0...H508	
Область вспомогательных битов	A0...A956	A448...A956
Область таймеров	T0000...T4092	

Область	S	D
Область счетчиков	C0000...C4092	
Область DM	D0...D32764	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Постоянные	---	
Регистры данных	---	
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

Описание

Команда DEGD(850) преобразует 64-битовое число с плавающей запятой двойной точности в словах S...S+3 из радианов в градусы и записывает результат в слова D...D+3. (Входное значение с плавающей запятой должно быть указано в формате IEEE754.)



Исходное значение
(радианы, с плавающей запятой, 64 бит)

Результат
(градусы, с плавающей запятой, 64 бит)

Преобразование из радиан в градусы производится по следующей формуле:

Значение в радианах $\times$ 180/π = значение в градусах

Если результат по модулю превосходит максимальное значение, которое может быть представлено в формате с плавающей запятой, включается флаг переполнения и в качестве результата выдается $\pm\infty$.

Если абсолютное значение результата меньше минимального значения, которое может быть выражено числом с плавающей запятой, включается флаг потери значимости и в качестве результата выдается «0».

Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если входное значение не опознано как значение с плавающей запятой. Включен, если входное значение не является числом (NaN). Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если и показатель, и мантисса результата равны «0». Выключен во всех остальных случаях.
Флаг переполнения	OF	Включен, если абсолютное значение результата слишком велико и не может быть выражено числом с плавающей запятой двойной точности.
Флаг потери значимости	UF	Включен, если абсолютное значение результата слишком мало и не может быть выражено числом с плавающей запятой двойной точности.
Флаг отрицательного значения	N	Включен, если результат отрицателен. Выключен во всех остальных случаях.

Меры предосторожности

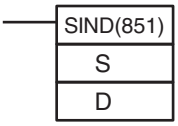
Входное значение в словах S...S+3 должно быть представлено в формате с плавающей запятой в соответствии с IEEE754.

3-15-11 СИНУС ДВОЙНОЙ ТОЧНОСТИ: SIND(851)

Назначение

Вычисление синуса угла (в радианах), заданного 64-битовым числом с плавающей запятой двойной точности, и запись результата в указанные слова.

Символ РКС



S: Первое исходное слово
D: Первое слово назначения

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	SIND(851)
	Однократное выполнение по положительному фронту	@SIND(851)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

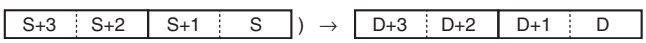
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	S	D
Область CIO	CIO 0...CIO 6140	
Рабочая область	W0...W508	
Область битов хранения	H0...H508	
Область вспомогательных битов	A0...A956	A448...A956
Область таймеров	T0000...T4092	
Область счетчиков	C0000...C4092	
Область DM	D0...D32764	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Постоянные	---	
Регистры данных	---	
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

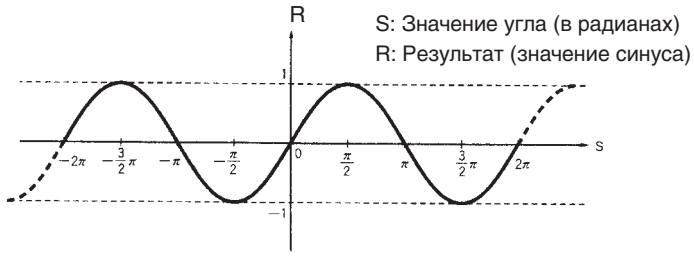
Описание

Команда SIND(851) рассчитывает синус угла (в радианах), представленного в виде 64-битового числа с плавающей запятой двойной точности в словах S...S+3, и записывает результат в слова D...D+3.
(Входное значение с плавающей запятой должно быть указано в формате IEEE754.)



В словах S...S+3 следует указать требуемое значение угла в радианах (от -65 535 до 65 535). Если значение угла будет выходить за диапазон -65 535...65 535, произойдет ошибка и команда выполнена не будет. Информацию о преобразовании из градусов в радианы или наоборот см. в разделе *3-15-9 ГРАДУСЫ В РАДИАНЫ ДВОЙНОЙ ТОЧНОСТИ: RADD(849)* или *3-15-10 РАДИАНЫ В ГРАДУСЫ ДВОЙНОЙ ТОЧНОСТИ: DEGD(850)*.

На следующем графике отображена взаимосвязь между величиной угла и результатом.



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если входное значение не является числом (NaN). Включен, если входное значение по модулю превосходит 65 535. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если и показатель, и мантисса результата равны «0». Выключен во всех остальных случаях.
Флаг переполнения	OF	Не изменяется
Флаг потери значимости	UF	Не изменяется
Флаг отрицательного значения	N	Включен, если результат отрицателен. Выключен во всех остальных случаях.

Меры предосторожности

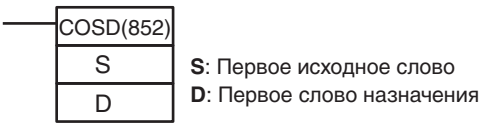
Входное значение в словах S...S+3 должно быть представлено в формате с плавающей запятой в соответствии с IEEE754.

3-15-12 КОСИНУС ДВОЙНОЙ ТОЧНОСТИ: COSD(852)

Назначение

Вычисление косинуса угла (в радианах), заданного 64-битовым числом с плавающей запятой двойной точности, и запись результата в указанные слова.

Символ РКС



Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	COSD(852)
	Однократное выполнение по положительному фронту	@COSD(852)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

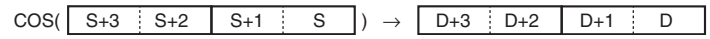
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	S	D
Область CIO	CIO 0...CIO 6140	
Рабочая область	W0...W508	
Область битов хранения	H0...H508	
Область вспомогательных битов	A0...A956	A448...A956
Область таймеров	T0000...T4092	
Область счетчиков	C0000...C4092	
Область DM	D0...D32764	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Постоянные	---	
Регистры данных	---	
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

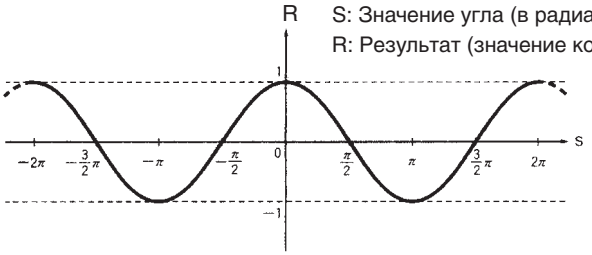
Описание

Команда COSD(852) рассчитывает косинус угла (в радианах), представленного в виде 64-битового числа с плавающей запятой двойной точности в словах S...S+3, и записывает результат в слова D...D+3.
(Входное значение с плавающей запятой должно быть указано в формате IEEE754.)



В словах S...S+3 следует указать требуемое значение угла в радианах (от -65 535 до 65 535). Если значение угла будет выходить за диапазон -65 535...65 535, произойдет ошибка и команда выполнена не будет. Информацию о преобразовании из градусов в радианы или наоборот см. в разделе *3-15-9 ГРАДУСЫ В РАДИАНЫ ДВОЙНОЙ ТОЧНОСТИ: RADD(849)* или *3-15-10 РАДИАНЫ В ГРАДУСЫ ДВОЙНОЙ ТОЧНОСТИ: DEGD(850)*.

На следующем графике отображена взаимосвязь между величиной угла и результатом.



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если входное значение не является числом (NaN). Включен, если входное значение по модулю превосходит 65 535. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если и показатель, и мантисса результата равны «0». Выключен во всех остальных случаях.
Флаг переполнения	OF	Не изменяется
Флаг потери значимости	UF	Не изменяется
Флаг отрицательного значения	N	Включен, если результат отрицателен. Выключен во всех остальных случаях.

Меры предосторожности

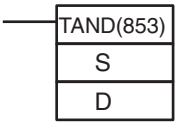
Входное значение в словах S...S+3 должно быть представлено в формате с плавающей запятой в соответствии с IEEE754.

3-15-13 ТАНГЕНС ДВОЙНОЙ ТОЧНОСТИ: TAND(853)

Назначение

Вычисление тангенса угла (в радианах), заданного 64-битовым числом с плавающей запятой двойной точности, и запись результата в указанные слова.

Символ РКС



S: Первое исходное слово
D: Первое слово назначения

Варианты выполнения

Варианты выполнения		
	Выполнение в каждом цикле при включенном условии выполнения	TAND(853)
	Однократное выполнение по положительному фронту	@TAND(853)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

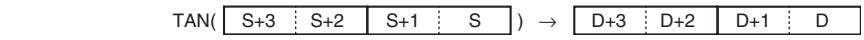
Характеристики операндов

Область	S	D
Область CIO	CIO 0...CIO 6140	
Рабочая область	W0...W508	
Область битов хранения	H0...H508	
Область вспомогательных битов	A0...A956	A448...A956
Область таймеров	T0000...T4092	
Область счетчиков	C0000...C4092	
Область DM	D0...D32764	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Постоянные	---	
Регистры данных	---	
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

Описание

Команда TAND(853) рассчитывает тангенс угла (в радианах), представленного в виде 64-битового числа с плавающей запятой двойной точности в словах S...S+3, и записывает результат в слова D...D+3.

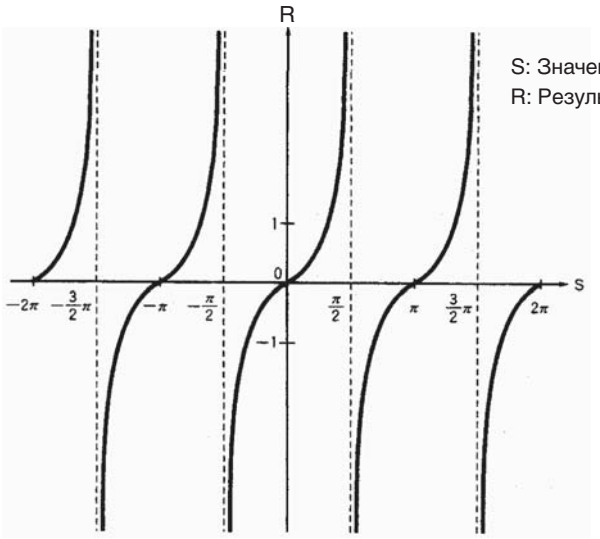
(Входное значение с плавающей запятой должно быть указано в формате IEEE754.)



В словах S...S+3 следует указать требуемое значение угла в радианах (от -65 535 до 65 535). Если значение угла будет выходить за диапазон -65 535...65 535, произойдет ошибка и команда выполнена не будет. Информацию о преобразовании из градусов в радианы или наоборот см. в разделе *3-15-9 ГРАДУСЫ В РАДИАНЫ ДВОЙНОЙ ТОЧНОСТИ: RADD(849)* или *3-15-10 РАДИАНЫ В ГРАДУСЫ ДВОЙНОЙ ТОЧНОСТИ: DEGD(850)*.

Если результат по модулю превосходит максимальное значение, которое может быть представлено в формате с плавающей запятой, включается флаг переполнения и в качестве результата выдается $\pm\infty$.

На следующем графике отображена взаимосвязь между величиной угла и результатом.



S: Значение угла (в радианах)
R: Результат (значение тангенса)

Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если входное значение не является числом (NaN). Включен, если входное значение по модулю превосходит 65 535. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если и показатель, и мантисса результата равны «0». Выключен во всех остальных случаях.
Флаг переполнения	OF	Включен, если абсолютное значение результата слишком велико и не может быть выражено числом с плавающей запятой двойной точности (64 бит).
Флаг потери значимости	UF	Не изменяется
Флаг отрицательного значения	N	Включен, если результат отрицателен. Выключен во всех остальных случаях.

Меры предосторожности

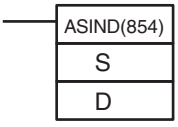
Входное значение в словах S...S+3 должно быть представлено в формате с плавающей запятой в соответствии с IEEE754.

3-15-14 АРКСИНУС ДВОЙНОЙ ТОЧНОСТИ: ASIND(854)

Назначение

Вычисление арксинуса угла (в радианах), заданного 64-битовым числом с плавающей запятой двойной точности, и запись результата в указанные слова. (Функция арксинуса является обратной по отношению к функции синуса. Она возвращает значение угла, которому соответствует указанное значение синуса от -1 до 1.)

Символ РКС



S: Первое исходное слово
D: Первое слово назначения

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	ASIND(854)
	Однократное выполнение по положительному фронту	@ASIND(854)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

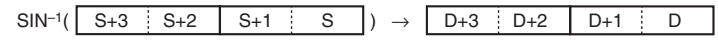
Характеристики операндов

Область	S	D
Область CIO	CIO 0...CIO 6140	
Рабочая область	W0...W508	
Область битов хранения	H0...H508	
Область вспомогательных битов	A0...A956	A448...A956
Область таймеров	T0000...T4092	
Область счетчиков	C0000...C4092	
Область DM	D0...D32764	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Постоянные	---	
Регистры данных	---	
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

Описание

Команда ASIND(854) рассчитывает арксинус угла (в радианах), представленного в виде 64-битового числа с плавающей запятой двойной точности в словах S...S+3, и записывает результат в слова D...D+3.

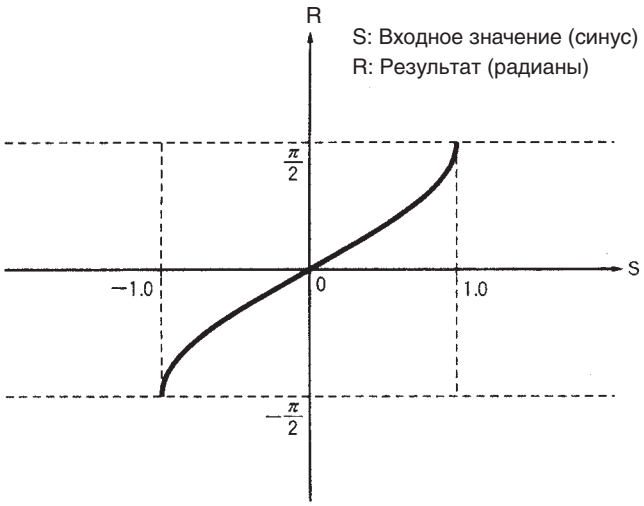
(Входное значение с плавающей запятой должно быть указано в формате IEEE754.)



Входное значение должно находиться в диапазоне от -1,0 до 1,0. Если входное значение по модулю превосходит 1,0, произойдет ошибка и команда выполнена не будет.

В качестве результата в слова D...D+3 выводится значение угла (в радианах) в диапазоне от $-\pi/2$ до $\pi/2$.

На следующем графике отображена взаимосвязь между входным значением и результатом.



Флаги

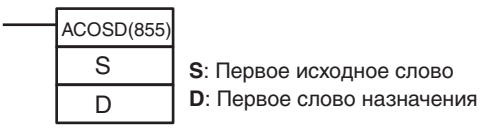
Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если входное значение не опознано как значение с плавающей запятой. Включен, если входное значение не является числом (NaN). Включен, если входное значение по модулю превосходит 1,0. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если и показатель, и мантисса результата равны «0». Выключен во всех остальных случаях.
Флаг переполнения	OF	Не изменяется
Флаг потери значимости	UF	Не изменяется
Флаг отрицательного значения	N	Включен, если результат отрицателен. Выключен во всех остальных случаях.

Меры предосторожности Входное значение в словах S...S+3 должно быть представлено в формате с плавающей запятой в соответствии с IEEE754.

3-15-15 АРККОСИНУС ДВОЙНОЙ ТОЧНОСТИ: ACOSD(855)

Назначение Вычисление арккосинуса угла (в радианах), заданного 64-битовым числом с плавающей запятой двойной точности, и запись результата в указанные слова. (Функция арккосинуса является обратной по отношению к функции косинуса. Она возвращает значение угла, которому соответствует указанное значение косинуса от -1 до 1.)

Символ РКС



S: Первое исходное слово
D: Первое слово назначения

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	ACOSD(855)
	Однократное выполнение по положительному фронту	@ACOSD(855)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

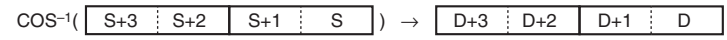
Характеристики операндов

Область	S	D
Область CIO	CIO 0...CIO 6140	
Рабочая область	W0...W508	
Область битов хранения	H0...H508	
Область вспомогательных битов	A0...A956	A448...A956
Область таймеров	T0000...T4092	
Область счетчиков	C0000...C4092	
Область DM	D0...D32764	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Постоянные	---	
Регистры данных	---	
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

Описание

ACOSD(855) рассчитывает арккосинус угла (в радианах), представленного в виде 64-битового числа с плавающей запятой двойной точности в словах S...S+3, и записывает результат в слова D...D+3.

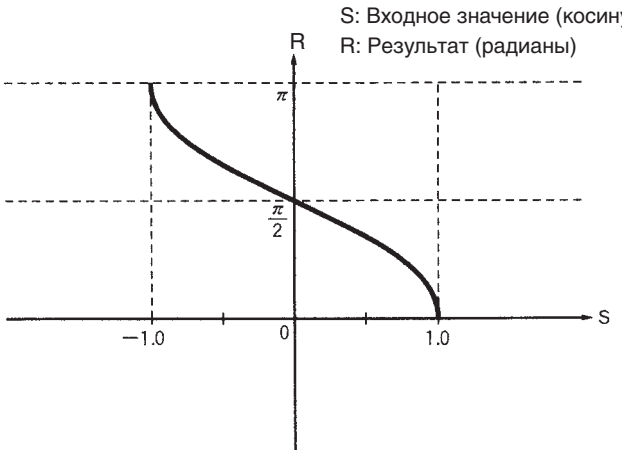
(Входное значение с плавающей запятой должно быть указано в формате IEEE754.)



Входное значение должно находиться в диапазоне от -1,0 до 1,0. Если входное значение по модулю превосходит 1,0, произойдет ошибка и команда выполнена не будет.

В качестве результата в слова D...D+3 выводится значение угла (в радианах) в диапазоне от 0 до π.

На следующем графике отображена взаимосвязь между входным значением и результатом.



Флаги

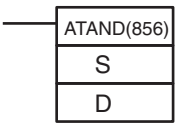
Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если входное значение не опознано как значение с плавающей запятой. Включен, если входное значение не является числом (NaN). Включен, если входное значение по модулю превосходит 1,0. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если и показатель, и мантисса результата равны «0». Выключен во всех остальных случаях.
Флаг переполнения	OF	Не изменяется
Флаг потери значимости	UF	Не изменяется
Флаг отрицательного значения	N	Не изменяется

Меры предосторожности Входное значение в словах S...S+3 должно быть представлено в формате с плавающей запятой в соответствии с IEEE754.

3-15-16 АРКТАНГЕНС ДВОЙНОЙ ТОЧНОСТИ: ATAND(856)

Назначение Вычисление арктангенса угла (в радианах), заданного 64-битовым числом с плавающей запятой двойной точности, и запись результата в указанные слова. (Функция арктангенса является обратной по отношению к функции тангенса. Она возвращает значение угла, которому соответствует указанное значение тангенса.)

Символ РКС



S: Первое исходное слово
D: Первое слово назначения

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	ATAND(856)
	Однократное выполнение по положительному фронту	@ATAND(856)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	S	D
Область CIO	CIO 0...CIO 6140	
Рабочая область	W0...W508	
Область битов хранения	H0...H508	
Область вспомогательных битов	A0...A956	A448...A956
Область таймеров	T0000...T4092	
Область счетчиков	C0000...C4092	
Область DM	D0...D32764	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Постоянные	---	
Регистры данных	---	
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

Описание

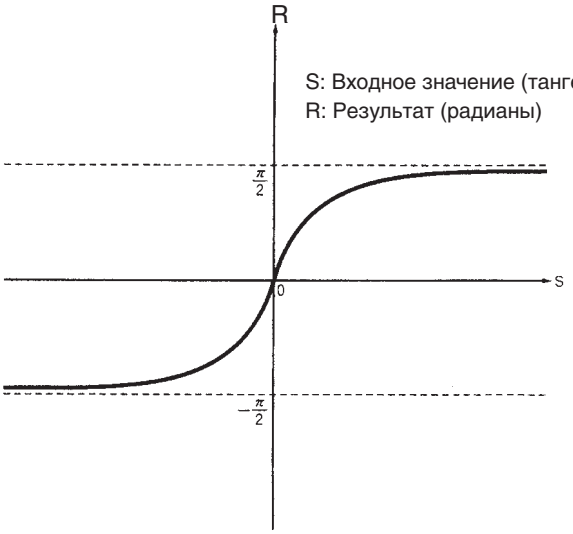
Команда ATAND(856) рассчитывает тангенс угла (в радианах), представленного в виде 64-битового числа с плавающей запятой двойной точности в словах S...S+3, и записывает результат в слова D...D+3.

(Входное значение с плавающей запятой должно быть указано в формате IEEE754.)



В качестве результата в слова D...D+3 выводится значение угла (в радианах) в диапазоне от $-\pi/2$ до $\pi/2$.

На следующем графике отображена взаимосвязь между входным значением и результатом.



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если входное значение не опознано как значение с плавающей запятой. Включен, если входное значение не является числом (NaN). Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если и показатель, и мантисса результата равны «0». Выключен во всех остальных случаях.
Флаг переполнения	OF	Не изменяется
Флаг потери значимости	UF	Не изменяется
Флаг отрицательного значения	N	Включен, если результат отрицателен. Выключен во всех остальных случаях.

Меры предосторожности

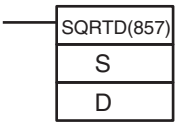
Входное значение в словах S...S+3 должно быть представлено в формате с плавающей запятой в соответствии с IEEE754.

3-15-17 КВАДРАТНЫЙ КОРЕНЬ ДВОЙНОЙ ТОЧНОСТИ: SQRTD(857)

Назначение

Вычисление квадратного корня 64-битового числа с плавающей запятой двойной точности и запись результата в указанные слова.

Символ РКС



S: Первое исходное слово
D: Первое слово назначения

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	SQRTD(857)
	Однократное выполнение по положительному фронту	@SQRTD(857)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

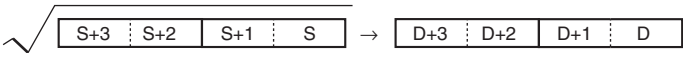
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	S	D
Область CIO	CIO 0...CIO 6140	
Рабочая область	W0...W508	
Область битов хранения	H0...H508	
Область вспомогательных битов	A0...A956	A448...A956
Область таймеров	T0000...T4092	
Область счетчиков	C0000...C4092	
Область DM	D0...D32764	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Постоянные	---	
Регистры данных	---	
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

Описание

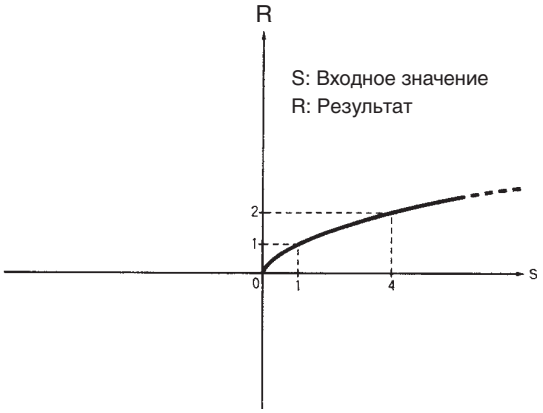
Команда SQRTD(857) рассчитывает квадратный корень 64-битового числа с плавающей запятой двойной точности в словах S...S+3 и записывает результат в слова D...D+3. (Входное значение с плавающей запятой должно быть указано в формате IEEE754.)



Входное значение должно быть положительным числом. Если оно будет отрицательным, произойдет ошибка и команда выполнена не будет.

Если результат по модулю превосходит максимальное значение, которое может быть представлено в формате с плавающей запятой, включается флаг переполнения и в качестве результата выдается $\pm\infty$.

На следующем графике отображена взаимосвязь между входным значением и результатом.



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если входное значение не опознано как значение с плавающей запятой. Включен, если входное значение отрицательно. Включен, если входное значение не является числом (NaN). Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если и показатель, и мантисса результата равны «0». Выключен во всех остальных случаях.
Флаг переполнения	OF	Включен, если абсолютное значение результата слишком велико и не может быть выражено числом с плавающей запятой двойной точности (64 бит).
Флаг потери значимости	UF	Не изменяется
Флаг отрицательного значения	N	Не изменяется

Меры предосторожности

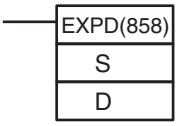
Входное значение в словах S...S+3 должно быть представлено в формате с плавающей запятой в соответствии с IEEE754.

3-15-18 ЭКСПОНЕНТА ДВОЙНОЙ ТОЧНОСТИ: EXPD(858)

Назначение

Расчет натуральной экспоненты (по основанию «e») от 64-битового числа с плавающей запятой двойной точности и запись результата в указанные слова.

Символ РКС



S: Первое исходное слово
D: Первое слово назначения

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	EXPD(858)
	Однократное выполнение по положительному фронту	@EXPD(858)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

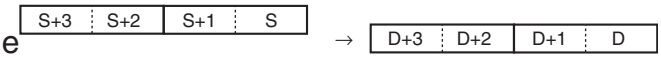
Область	S	D
Область CIO	CIO 0...CIO 6140	
Рабочая область	W0...W508	
Область битов хранения	H0...H508	
Область вспомогательных битов	A0...A956	A448...A956
Область таймеров	T0000...T4092	

Область	S	D
Область счетчиков	C0000...C4092	
Область DM	D0...D32764	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Постоянные	---	
Регистры данных	---	
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

Описание

Команда EXPD(858) рассчитывает натуральную экспоненту (по основанию «e») 64-битового числа с плавающей запятой двойной точности в словах S...S+3 и записывает результат в слова D...D+3.

Другими словами, команда EXP(467) рассчитывает e^x (x = входное значение) и записывает результат в слова D...D+3.



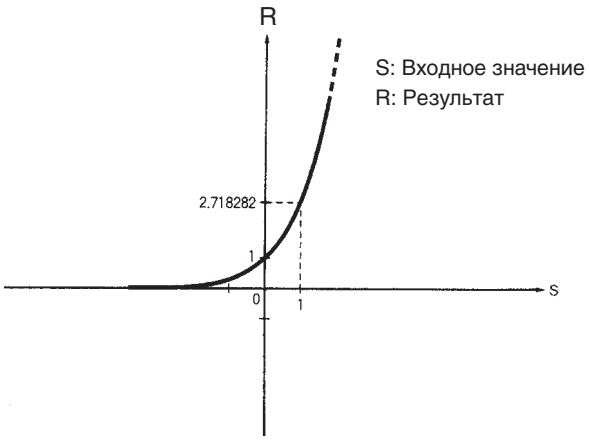
Если результат по модулю превосходит максимальное значение, которое может быть представлено в формате с плавающей запятой, включается флаг переполнения и в качестве результата выдается $\pm\infty$.

Если абсолютное значение результата меньше минимального значения, которое может быть выражено числом с плавающей запятой, включается флаг потери значимости и в качестве результата выдается «0».

Примечание.

Постоянная «e» = 2,718282.

На следующем графике отображена взаимосвязь между входным значением и результатом.



Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если входное значение не опознано как значение с плавающей запятой. Включен, если входное значение не является числом (NaN). Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если и показатель, и мантисса результата равны «0». Выключен во всех остальных случаях.
Флаг переполнения	OF	Включен, если абсолютное значение результата слишком велико и не может быть выражено числом с плавающей запятой двойной точности (64 бит).
Флаг потери значимости	UF	Включен, если абсолютное значение результата слишком мало и не может быть выражено 64-битовым числом с плавающей запятой двойной точности.
Флаг отрицательного значения	N	Не изменяется

Меры предосторожности

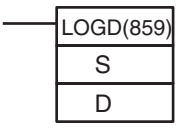
Входное значение в словах S...S+3 должно быть представлено в формате с плавающей запятой в соответствии с IEEE754.

3-15-19 ЛОГАРИФМ ДВОЙНОЙ ТОЧНОСТИ: LOGD(859)

Назначение

Вычисление натурального логарифма (по основанию «e») от 64-битового числа с плавающей запятой двойной точности и запись результата в указанные слова.

Символ РКС



S: Первое исходное слово
D: Первое слово назначения

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	LOGD(859)
	Однократное выполнение по положительному фронту	@LOGD(859)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	S	D
Область CIO	CIO 0...CIO 6140	
Рабочая область	W0...W508	
Область битов хранения	H0...H508	
Область вспомогательных битов	A0...A956	A448...A956
Область таймеров	T0000...T4092	

Область	S	D
Область счетчиков	C0000...C4092	
Область DM	D0...D32764	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Постоянные	---	
Регистры данных	---	
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

Описание

Команда LOGD(859) рассчитывает натуральный логарифм (по основанию «e») от 64-битового числа с плавающей запятой в S...S+3 и записывает результат в слова D...D+3.

$\log_e$ [S+3 | S+2 | S+1 | S] → [D+3 | D+2 | D+1 | D]

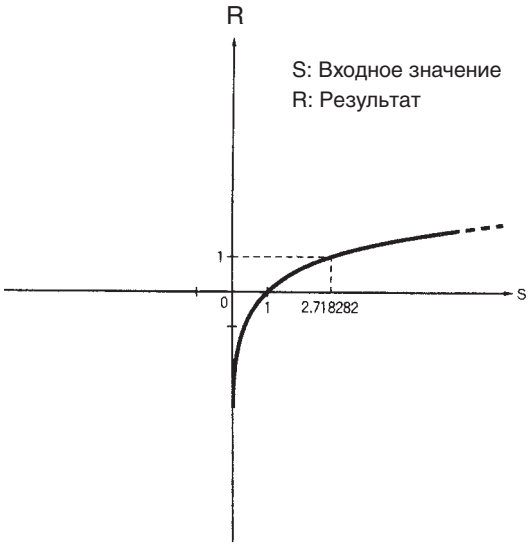
Входное значение должно быть положительным числом. Если оно будет отрицательным, произойдет ошибка и команда выполнена не будет.

Если результат по модулю превосходит максимальное значение, которое может быть представлено в формате с плавающей запятой, включается флаг переполнения и в качестве результата выдается $\pm\infty$.

Примечание.

Постоянная «e» = 2,718282.

На следующем графике отображена взаимосвязь между входным значением и результатом.



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если входное значение не опознано как значение с плавающей запятой. Включен, если входное значение отрицательно. Включен, если входное значение не является числом (NaN). Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если и показатель, и мантисса результата равны «0». Выключен во всех остальных случаях.
Флаг переполнения	OF	Включен, если абсолютное значение результата слишком велико и не может быть выражено числом с плавающей запятой двойной точности (64 бит).
Флаг потери значимости	UF	Не изменяется
Флаг отрицательного значения	N	Включен, если результат отрицателен. Выключен во всех остальных случаях.

Меры предосторожности

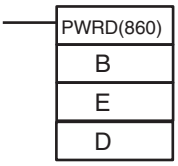
Входное значение в словах S...S+3 должно быть представлено в формате с плавающей запятой в соответствии с IEEE754.

3-15-20 ВОЗВЕДЕНИЕ В СТЕПЕНЬ ДВОЙНОЙ ТОЧНОСТИ: PWRD(860)

Назначение

Возведение 64-битового числа с плавающей запятой двойной точности в степень, определяемую другим 64-битовым числом с плавающей запятой двойной точности.

Символ РКС



B: Первое слово основания
E: Первое слово показателя
D: Первое слово назначения

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	PWRD(860)
	Однократное выполнение по положительному фронту	@PWRD(860)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

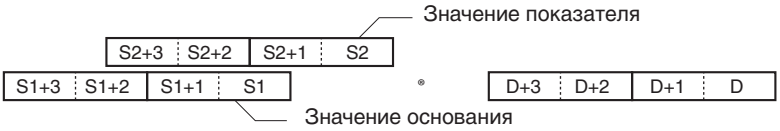
Характеристики операндов

Область	B	E	D
Область CIO	CIO 0...CIO 6140		
Рабочая область	W0...W508		
Область битов хранения	H0...H508		

Область	B	E	D
Область вспомогательных битов	A0...A956		A448...A956
Область таймеров	T0000...T4092		
Область счетчиков	C0000...C4092		
Область DM	D0...D32764		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	---		
Регистры данных	---		
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда PWRD(860) возводит 64-битовое число с плавающей запятой двойной точности в словах B...B+3 в степень 64-битового числа с плавающей запятой двойной точности в словах E...E+3. Другими словами, команда PWR(840) рассчитывает X^Y (X = содержимое слов B...B+3; Y = содержимое E...E+3).



Например, если основание степени (B...B+3) равно 3,1, а показатель степени (E...E+3) равен 3, результатом выполнения команды будет значение $3,1^3$ или 29,791.

Если результат по модулю превосходит максимальное значение, которое может быть представлено в формате с плавающей запятой, включается флаг переполнения.

Если абсолютное значение результата меньше минимального значения, которое может быть выражено числом с плавающей запятой, включается флаг потери значимости.

Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если значение основания (B...B+3) или показателя степени (E...E+3) не опознано как число с плавающей запятой. Включен, если значение основания (B...B+3) или показателя степени (E...E+3) не является числом (NaN). Включен, если основание степени (B...B+3) равно нулю, а показатель степени (E...E+3) меньше 0 (деление на 0). Включен, если основание степени (B...B+3) меньше нуля, а показатель степени (E...E+3) не является целым числом (корень из отрицательного числа). Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если и показатель, и мантисса результата равны «0». Выключен во всех остальных случаях.

Название	Обозначение	Описание работы
Флаг переполнения	OF	Включен, если абсолютное значение результата слишком велико и не может быть выражено числом с плавающей запятой двойной точности.
Флаг потери значимости	UF	Включен, если абсолютное значение результата слишком мало и не может быть выражено числом с плавающей запятой двойной точности.
Флаг отрицательного значения	N	Включен, если результат отрицателен. Выключен во всех остальных случаях.

Меры предосторожности

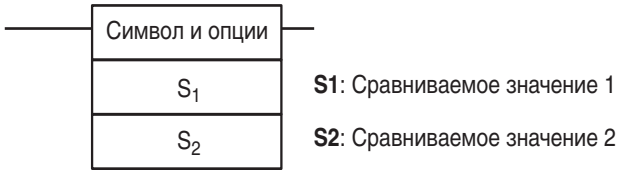
Основание (B...B+3) и показатель степени (E...E+3) должны быть представлены в формате с плавающей запятой в соответствии с IEEE754.

3-15-21 Входные команды для чисел с плавающей запятой двойной точности

Назначение

Эти входные команды сравнения служат для сравнения двух 64-битовых значений с плавающей запятой двойной точности (в формате IEEE754) и включают условие выполнения, если выполнено условие сравнения.

Символ РКС



Варианты выполнения

Варианты выполнения	Включение условия выполнения в каждом цикле, в котором результат сравнения = ИСТИНА.	Входная команда сравнения
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	S_1	S_2
Область CIO	CIO 0...CIO 6140	
Рабочая область	W0...W508	
Область битов хранения	H0...H508	
Область вспомогательных битов	A0...A956	
Область таймеров	T0000...T4092	
Область счетчиков	C0000...C4092	
Область DM	D0...D32764	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	

Область	S_1	S_2
Постоянные	---	
Регистры данных	---	
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

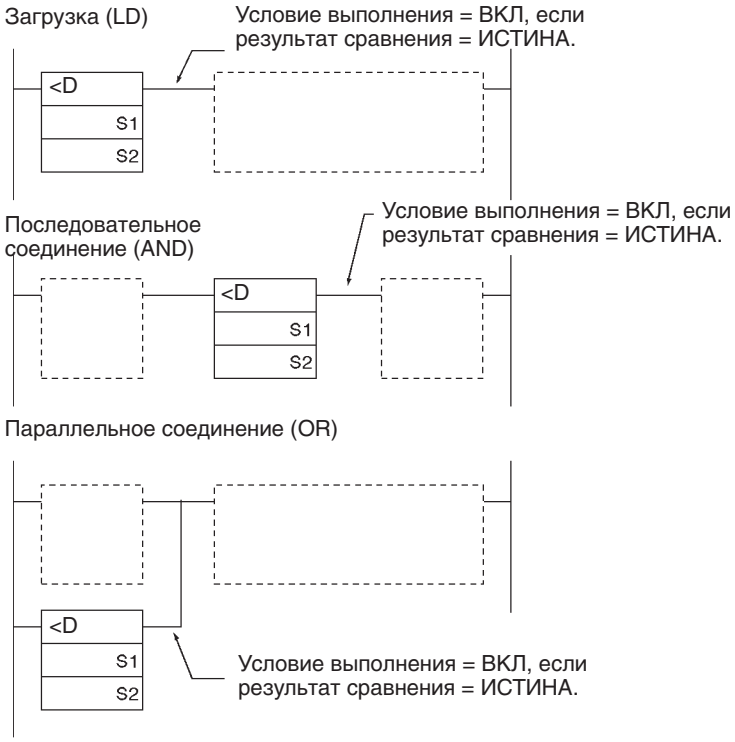
Описание

Входная команда сравнения сравнивает значения, указанные в S_1 и S_2, как значения с плавающей запятой двойной точности (64-битовые числа в формате IEEE754) и включает условие выполнения, если выполнено условие сравнения. После записи значений в слова S_1 и S_2 укажите первое из четырех слов, содержащих 64-битовое значение. 64-битовые значения с плавающей запятой невозможно ввести как константы.

Применение команд в программе

Входные команды сравнения используются в программе так же, как команды LD, AND и OR. Они управляют выполнением следующих за ними команд.

Тип ввода	Описание работы
LD	Команда может присоединяться непосредственно к левой шине.
AND	Команда не может присоединяться непосредственно к левой шине.
ИЛИ	Команда может присоединяться непосредственно к левой шине.



Варианты применения

Доступны шесть видов сравнения и три способа ввода команды: всего 18 комбинаций.

Символ		Вариант (формат данных)
=	(равно)	D: значение с плавающей запятой двойной точности
< >	(не равно)	
<	(меньше)	
<=	(меньше или равно)	
>	(больше)	
>=	(больше или равно)	

Краткий обзор входных команд сравнения

В приведенной ниже таблице перечислены коды функций, мнемонические коды, названия и назначение 18 входных команд сравнения чисел с плавающей запятой одинарной точности. (C1=S_1+3, S_1+2, S_1+1, S_1, а C2=S_2+3, S_2+2, S_2+1, S_2.)

Код	Мнемоническое обозначение	Название	Описание
335	LD=D	ЗАГРУЗКА С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ РАВНО	Истина, если C1 = C2
	AND=D	И С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ РАВНО	
	OR=D	ИЛИ С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ РАВНО	
336	LD<>D	ЗАГРУЗКА С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ НЕ РАВНО	Истина, если C1 ≠ C2
	AND<>D	И С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ НЕ РАВНО	
	OR<>D	ИЛИ С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ НЕ РАВНО	
337	LD<D	ЗАГРУЗКА С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ МЕНЬШЕ	Истина, если C1 < C2
	AND<D	И С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ МЕНЬШЕ	
	OR<D	ИЛИ С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ МЕНЬШЕ	
338	LD<=D	ЗАГРУЗКА С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ МЕНЬШЕ ИЛИ РАВНО	Истина, если C1 ≤ C2
	AND<=D	И С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ МЕНЬШЕ ИЛИ РАВНО	
	OR<=D	ИЛИ С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ МЕНЬШЕ ИЛИ РАВНО	
339	LD>D	ЗАГРУЗКА С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ БОЛЬШЕ	Истина, если C1 > C2
	AND>D	И С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ БОЛЬШЕ	
	OR>D	ИЛИ С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ БОЛЬШЕ	
340	LD>=D	ЗАГРУЗКА С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ БОЛЬШЕ ИЛИ РАВНО	Истина, если C1 ≥ C2
	AND>=D	И С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ БОЛЬШЕ ИЛИ РАВНО	
	OR>=D	ИЛИ С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ БОЛЬШЕ ИЛИ РАВНО	

Флаги

В данной таблице C1 = содержимое слов S1...S1+3, а C2 = содержимое слов S2...S2+3.

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если C1 или C2 не является допустимым числом с плавающей запятой (NaN). Включен, если C1 или C2 содержит значение +∞. Включен, если C1 или C2 содержит значение –∞. Выключен во всех остальных случаях.
Флаг «Больше»	>	Включен, если C1 > C2. Выключен во всех остальных случаях.
Флаг «Больше или равно»	> =	Включен, если C1 ≥ C2. Выключен во всех остальных случаях.
Флаг «Равно»	=	Включен, если C1 = C2. Выключен во всех остальных случаях.
Флаг «Не равно»	=	Включен, если C1 ≠ C2. Выключен во всех остальных случаях.
Флаг «Меньше»	<	Включен, если C1 < C2. Выключен во всех остальных случаях.
Флаг «Меньше или равно»	< =	Включен, если C1 ≤ C2. Выключен во всех остальных случаях.
Флаг отрицательного значения	N	Не изменяется

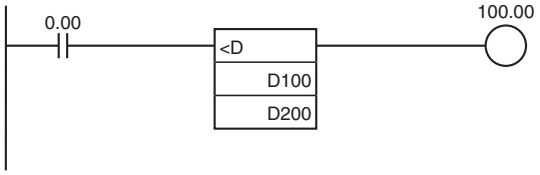
Меры предосторожности

Входная команда сравнения не может быть последней командой цепи. Между ней и правой шиной должна находиться как минимум еще одна команда.

Пример

И С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ МЕНЬШЕ: AND<D(331)

Если в следующем примере бит CIO 0.00 включен, значение с плавающей запятой в словах D100...D103 сравнивается со значением с плавающей запятой в словах D200...D203. Если содержимое слов D100...D103 меньше содержимого слов D200...D203, начинает выполняться следующая строка и включается бит CIO 100.00. Если содержимое слов D100...D103 больше или равно содержимому слов D200...D203, переход к следующей строке не происходит.





3-16 Команды обработки табличных данных

В данном разделе описаны команды, используемые для обработки табличных данных, стеков и других массивов значений.

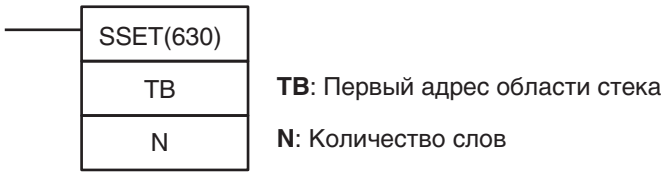
Команда	Мнемоническое обозначение	Код функции	Стр.
СОЗДАТЬ СТЕК	SSET	630	621
ЗАПИСАТЬ В СТЕК	PUSH	632	623
ПЕРВЫМ ПРИШЕЛ — ПЕРВЫМ ВЫШЕЛ	FIFO	633	626
ПОСЛЕДНИМ ПРИШЕЛ — ПЕРВЫМ ВЫШЕЛ	LIFO	634	629
СОЗДАТЬ ТАБЛИЦУ УКАЗАННЫХ РАЗМЕРОВ	DIM	631	632
ЗАПИСАТЬ РАСПОЛОЖЕНИЕ СТРОКИ	SETR	635	634
СЧИТАТЬ НОМЕР СТРОКИ	GETR	636	636
ПОИСК ЗНАЧЕНИЯ	SRCH	181	638
ПОМЕНЯТЬ МЕСТАМИ БАЙТЫ	SWAP	637	641
НАЙТИ МАКСИМУМ	MAX	182	643
НАЙТИ МИНИМУМ	MIN	183	646
СУММА	SUM	184	649
КОНТРОЛЬНАЯ СУММА КАДРА	FCS	180	653
СЧИТАТЬ РАЗМЕР СТЕКА	SNUM	638	655
ПРОЧИТАТЬ ДАННЫЕ ИЗ СТЕКА	SREAD	639	658
ЗАПИСАТЬ ДАННЫЕ В СТЕК	SWRIT	640	661
ВСТАВИТЬ ДАННЫЕ В СТЕК	SINS	641	664
УДАЛИТЬ ДАННЫЕ ИЗ СТЕКА	SDEL	642	667

3-16-1 СОЗДАТЬ СТЕК: SSET(630)

Назначение

Создание стека указанной длины, начинающегося с указанного слова.

Символ РКС



Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	SSET(630)
	Однократное выполнение по положительному фронту	@SSET(630)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

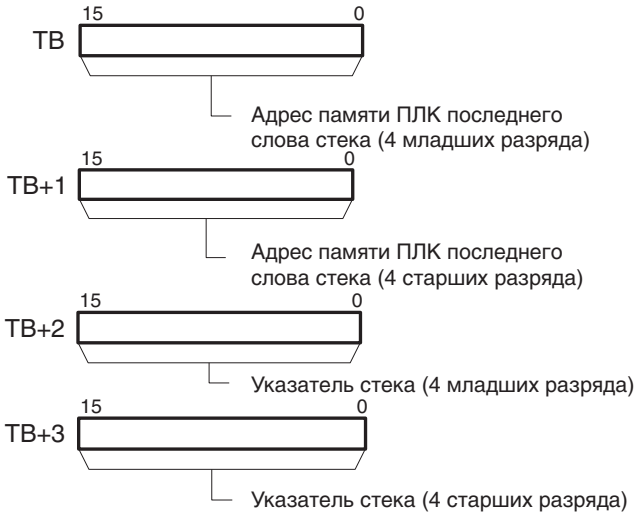
Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

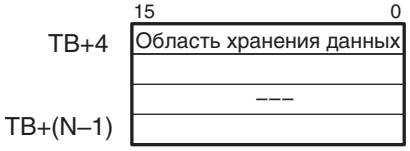
TB...TB+3: Слова управления стеком

Первые четыре слова стека содержат адрес памяти ПЛК последнего слова стека и указатель стека (адрес памяти ПЛК следующего слова, содержание которого должно быть перезаписано командой PUSH(632)).



TB+4...TB+(N-1): Область хранения данных

Остальное пространство стека служит для хранения данных.



Примечание.

(1) Начальным значением указателя стека всегда является адрес слова TB+4 в памяти ПЛК.

(2) Слова TB и TB+(N-1) должны принадлежать одной области.

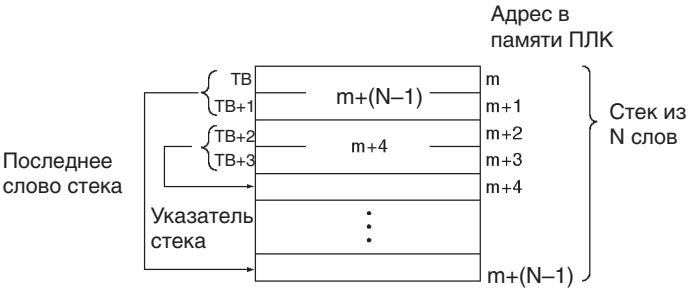
Характеристики операндов

Область	TB	N
Область CIO	CIO 0...CIO 6143	
Рабочая область	W0...W511	
Область битов хранения	H0...H511	
Область вспомогательных битов	A448...A959	A0...A959
Область таймеров	T0000...T4095	
Область счетчиков	C0000...C4095	
Область DM	D0...D32767	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Постоянные	---	#0005...#FFFF (двоичн.) или &5...&65535
Регистры данных	---	DR0...DR15
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

Описание

Команда SSET(630) формирует стек из N слов, начинающийся словом TB и заканчивающийся словом TB+(N-1). Первые два слова стека (TB+1 и TB) содержат 8-разрядный шестнадцатеричный адрес последнего слова стека в памяти ПЛК. Следующие два слова (TB+3 и TB+2) содержат указатель стека. Указатель стека — это адрес в памяти ПЛК следующего слова, значение которого будет перезаписано командой PUSH(632); первоначально указатель стека содержит адрес слова TB+4.

Команда SSET(630) автоматически инициализирует область данных стека (TB+4...TB+(N-1)) нулями. На следующей схеме показана структура стека.



Команда SSET(630) только формирует и инициализирует стек. Для записи данных в стек и чтения из стека используются следующие команды.

1,2,3...

1. Команда PUSH(632) при каждом выполнении записывает в стек одно слово данных.
2. Команды FIFO(633) и LIFO(634) считывают данные из стека. Команда FIFO(633) считывает слово, записанное первым. Команда LIFO(634) считывает слово, записанное последним.
3. Значение указателя стека в управляющем слове стека автоматически обновляется при выполнении команды PUSH(632), FIFO(633) или LIFO(634). В большинстве случаев у пользователей не возникает

потребности производить какие-либо операции с управляющим словом. В случае обращения к содержимому стека без использования указанных выше команды в целях косвенной адресации следует устанавливать значение указателя стека с помощью регистра указателя (IR).

Флаги

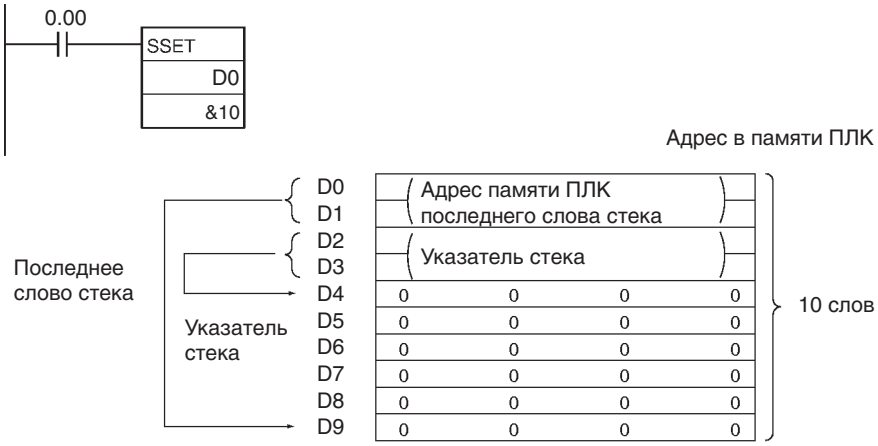
Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если N находится за пределами допустимого диапазона 0005...FFFF. Выключен во всех остальных случаях.

Меры предосторожности

Количество слов в стеке (N) должно быть не менее 5, так как в N входят четыре слова, содержащие указатель последнего слова стека и указатель стека. Если N не принадлежит диапазону 0005...FFFF, будет сгенерирована ошибка.

Примеры

Если в следующем примере включен бит CIO 0.00, команда SSET(630) создает стек из 10 слов в диапазоне D0...D9. Слова D0 и D1 содержат адрес последнего слова стека в памяти ПЛК. Слова D2 и D3 содержат указатель стека. Сам стек начинается со слова D4.

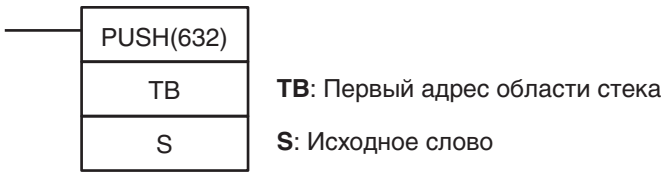


3-16-2 ЗАПИСАТЬ В СТЕК: PUSH(632)

Назначение

Запись одного слова данных в указанный стек.

Символ РКС



TB: Первый адрес области стека

S: Исходное слово

Варианты выполнения

Варианты выполнения		
	Выполнение в каждом цикле при включенном условии выполнения	PUSH(632)
	Однократное выполнение по положительному фронту	@PUSH(632)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

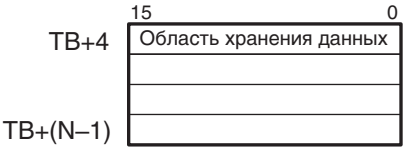
TB...TB+3: Слова управления стеком

Первые четыре слова стека содержат адрес памяти ПЛК последнего слова стека и указатель стека (адрес памяти ПЛК следующего слова, содержание которого должно быть перезаписано командой PUSH(632)).



TB+4...TB+(N-1): Область хранения данных

Остальное пространство стека служит для хранения данных.



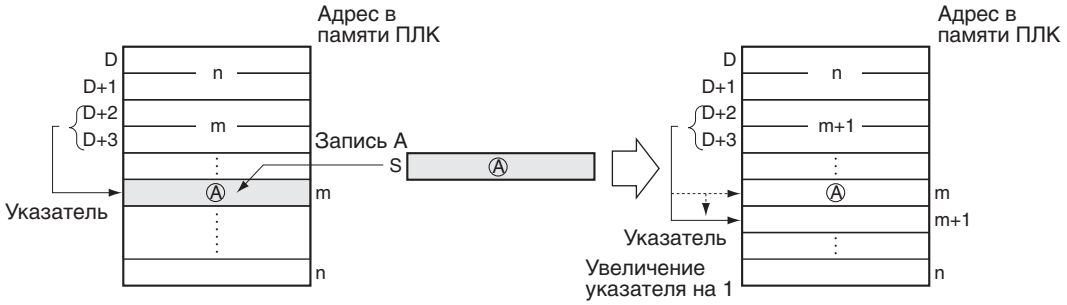
Характеристики операндов

Область	TB	S
Область CIO	CIO 0...CIO 6143	
Рабочая область	W0...W511	
Область битов хранения	H0...H511	
Область вспомогательных битов	A448...A959	A0...A959
Область таймеров	T0000...T4095	
Область счетчиков	C0000...C4095	
Область DM	D0...D32767	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Постоянные	---	#0000...#FFFF (двоичн.)
Регистры данных	---	DR0...DR15

Область	TB	S
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

Описание

Команда PUSH(632) записывает содержимое S по адресу, хранящемуся в указателе стека (TB+3 и TB+2), и увеличивает указатель стека на 1.



Для чтения данных, записанных в стек с помощью команды PUSH(632), можно использовать команды FIFO(633) и LIFO(634).

Флаги

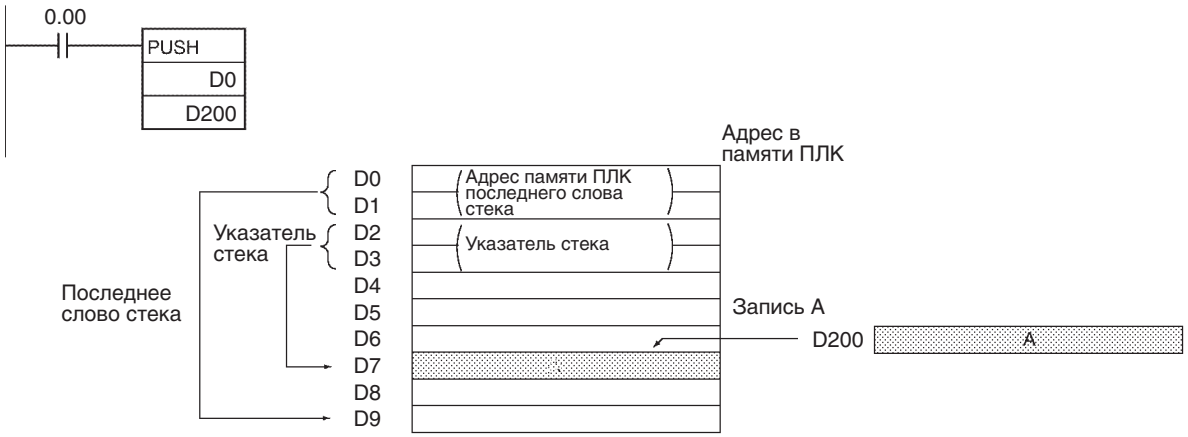
Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если адрес, хранящийся в указателе стека (TB+3 и TB+2), больше адреса последнего слова стека. (Это ошибка переполнения стека.) Выключен во всех остальных случаях.

Меры предосторожности

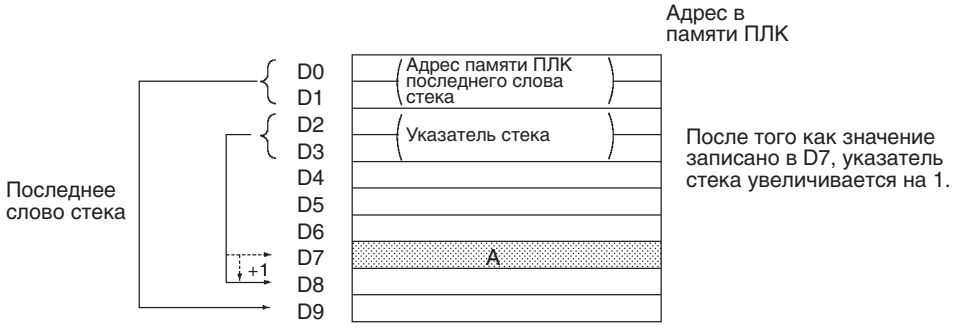
Стек должен быть определен заранее командой SSET(630).

Примеры

Если в следующем примере включен бит CIO 0.00, команда PUSH(632) копирует содержимое слова D200 в стек, начинающийся с D0. В этом случае указатель стека указывает на D7.



Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

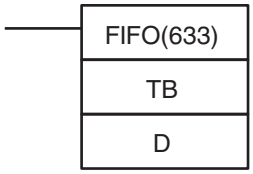


3-16-3 ПЕРВЫМ ПРИШЕЛ — ПЕРВЫМ ВЫШЕЛ: FIFO(633)

Назначение

Чтение слова данных, записанного в указанный стек самым первым (самого «старого» значения в стеке).

Символ РКС



TB: Первый адрес области стека

D: Слово назначения

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	FIFO(633)
	Однократное выполнение по положительному фронту	@FIFO(633)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

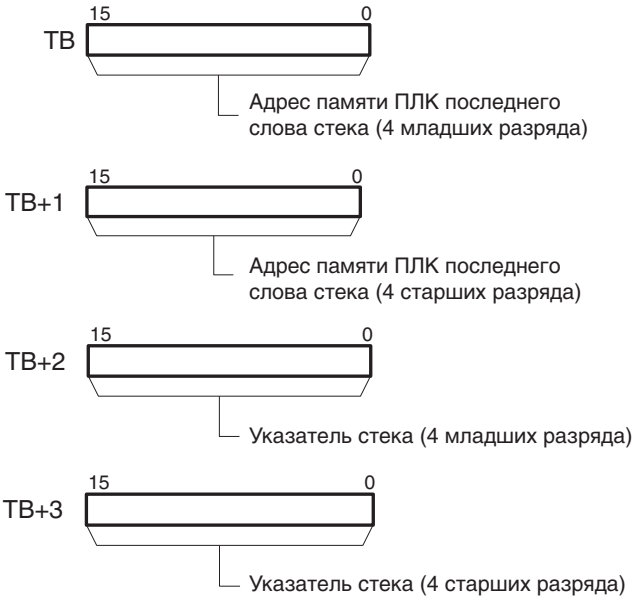
Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

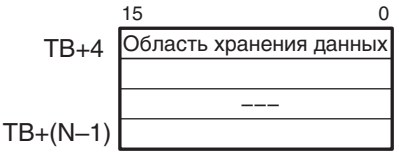
TB...TB+3: Слова управления стеком

Первые четыре слова стека содержат адрес памяти ПЛК последнего слова стека и указатель стека (адрес памяти ПЛК следующего слова, содержание которого должно быть перезаписано командой PUSH(632)).



TB+4...TB+(N-1): Область хранения данных

Остальное пространство стека служит для хранения данных.

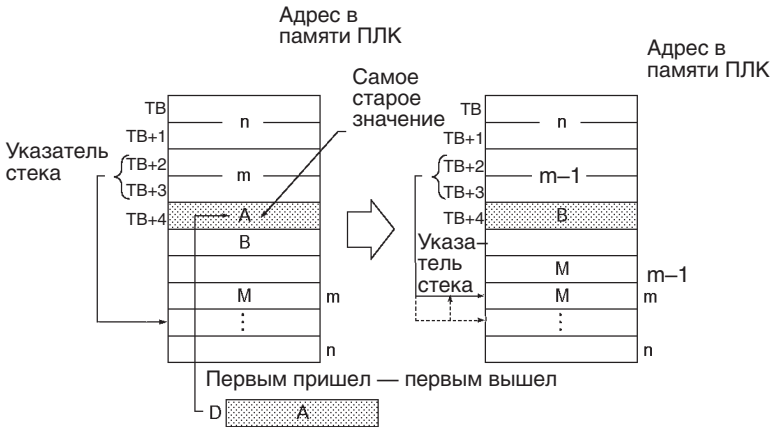


Характеристики операндов

Область	TB	D
Область CIO	CIO 0...CIO 6143	
Рабочая область	W0...W511	
Область битов хранения	H0...H511	
Область вспомогательных битов	A448...A959	
Область таймеров	T0000...T4095	
Область счетчиков	C0000...C4095	
Область DM	D0...D32767	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Постоянные	---	
Регистры данных	---	DR0...DR15
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

Описание

Команда FIFO(633) считывает содержимое самого «старого» слова данных из стека (TB+4) и выдает его в D. После этого указатель стека (TB+3 и TB+2) уменьшается на 1, считанное из TB+4 значение удаляется, а все остальные значения в стеке сдвигаются на одно слово вниз. Значение в самом конце стека (адрес которого содержался в указателе стека) остается неизменным.



Используйте команду FIFO(633) вместе с командой PUSH(632). Команду FIFO(633) можно использовать для чтения данных, записанных в стек командой PUSH(632), в порядке, совпадающем с порядком записи.

Команда FIFO(633) считывает начальное значение из стека и удаляет его для перемещения следующего значения вперед.

Флаги

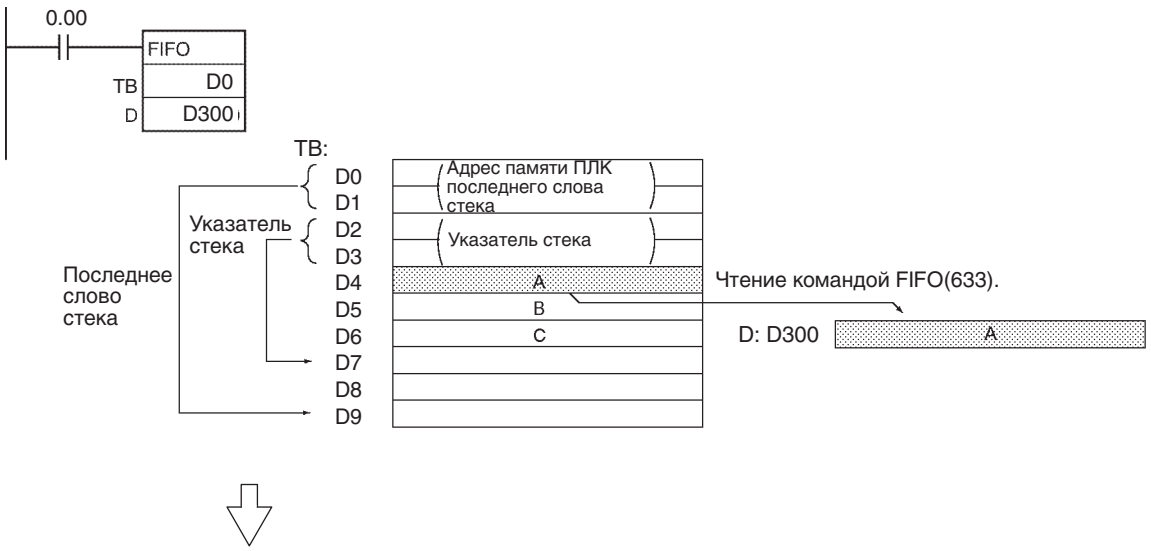
Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если содержание указателя стека (TB+3 и TB+2) меньше или равно адресу памяти ПЛК первого слова области данных стека (TB+4). (Это ошибка выхода за нижнюю границу стека.) Выключен во всех остальных случаях.

Меры предосторожности

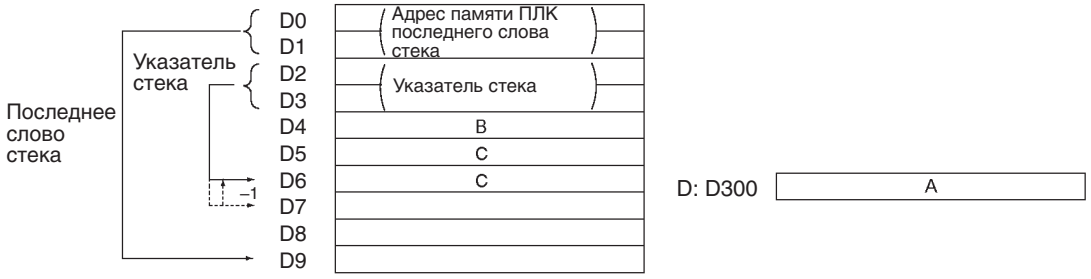
Стек должен быть определен заранее командой SSET(630).

Примеры

Если в следующем примере включен бит CIO 0.00, команда FIFO(633) считывает содержимое слова D4 (TB+4 для стека, начинающегося с D0) и записывает считанное значение в D300.



После записи значения в D300 указатель стека уменьшается на 1, а остальные данные сдвигаются вниз. (Содержимое слова D5 сдвигается в D4, а содержимое D6 — в D5.)

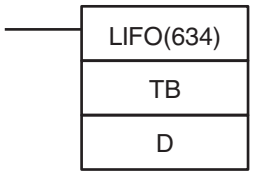


3-16-4 ПОСЛЕДНИМ ПРИШЕЛ — ПЕРВЫМ ВЫШЕЛ: LIFO(634)

Назначение

Чтение слова данных, записанного в указанный стек самым последним (самого нового значения в стеке).

Символ РКС



TB: Первый адрес области стека

D: Слово назначения

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	LIFO(634)
	Однократное выполнение по положительному фронту	@LIFO(634)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

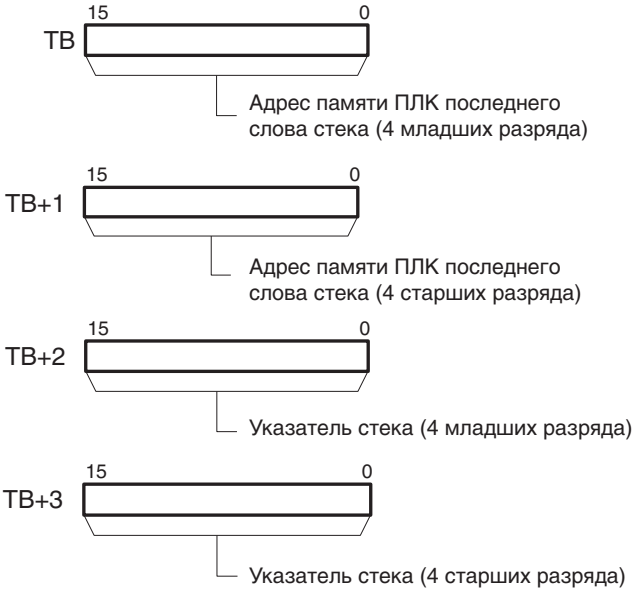
Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

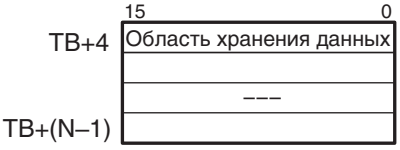
TB...TB+3: Слова управления стеком

Первые четыре слова стека содержат адрес памяти ПЛК последнего слова стека и указатель стека (адрес памяти ПЛК следующего слова, содержание которого должно быть перезаписано командой PUSH(632)).



TB+4...TB+(N-1): Область хранения данных

Остальное пространство стека служит для хранения данных.

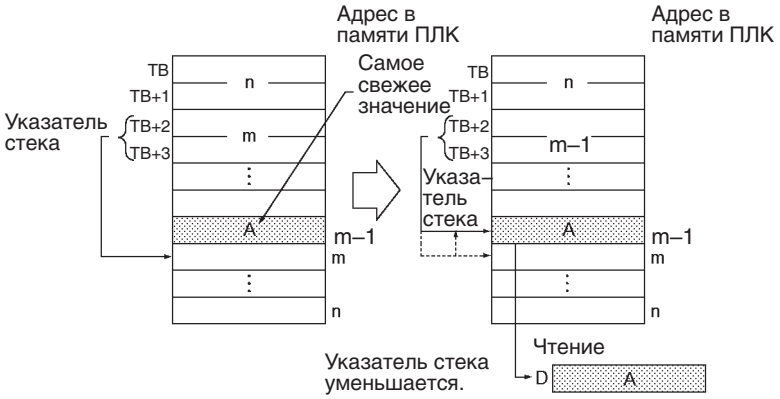


Характеристики операндов

Область	TB	D
Область CIO	CIO 0...CIO 6143	
Рабочая область	W0...W511	
Область битов хранения	H0...H511	
Область вспомогательных битов	A448...A959	
Область таймеров	T0000...T4095	
Область счетчиков	C0000...C4095	
Область DM	D0...D32767	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Постоянные	---	
Регистры данных	---	DR0...DR15
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

Описание

Команда LIFO(634) считывает значение по адресу, содержащемуся в указателе стека (самое новое значение в стеке), уменьшает указатель стека на 1 и выдает это значение в D. Считанное слово остается неизменным.



Используйте команду LIFO(634) вместе с командой PUSH(632). Команду LIFO(634) можно использовать для чтения данных, записанных в стек командой PUSH(632), в порядке, обратном порядку записи. После записи некоторого значения командой PUSH(632) указатель стека содержит адрес, следующий за адресом с самым последним записанным значением.

Флаги

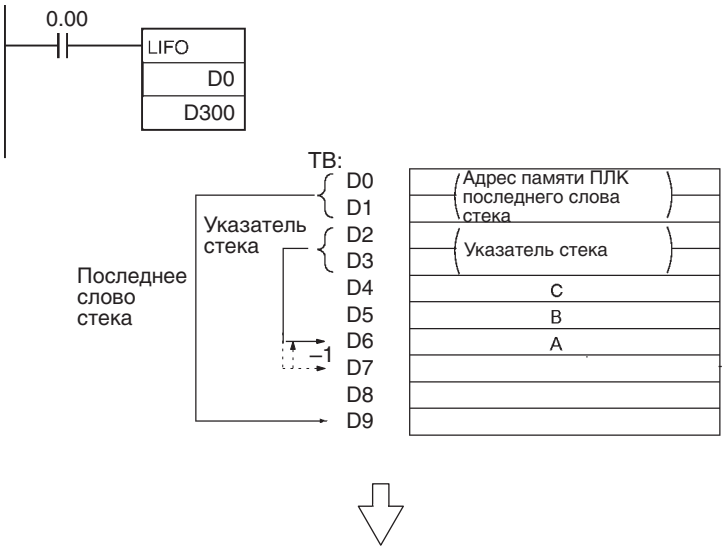
Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если содержание указателя стека (TB+3 и TB+2) меньше или равно адресу памяти ПЛК первого слова области данных стека (TB+4). (Это ошибка выхода за нижнюю границу стека.) Выключен во всех остальных случаях.

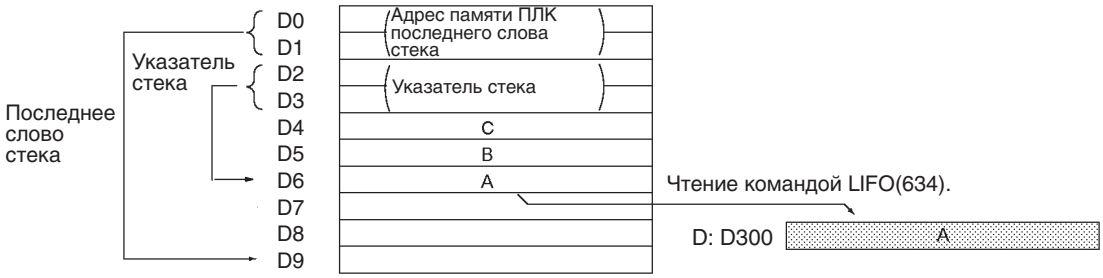
Меры предосторожности

Стек должен быть определен заранее командой SSET(630).

Примеры

Если в следующем примере включен бит CIO 0.00, команда LIFO(634) считывает содержимое слова, адрес которого содержится в указателе стека (D6), и выдает его в слово D300.





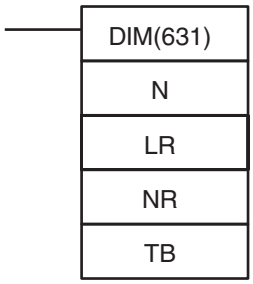
После записи значения в D300 указатель стека уменьшается на 1. Содержимое D6 остается неизменным.

3-16-5 СОЗДАТЬ ТАБЛИЦУ УКАЗАННЫХ РАЗМЕРОВ: DIM(631)

Назначение

Создание таблицы (из строк) в указанной области памяти ввода/вывода с объявлением длины каждой строки и количества строк. Можно создать таблицу размером до 16 строк.

Символ РКС



N: Номер таблицы

LR: Длина каждой строки

NK: Количество строк

TB: Первое слово таблицы

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	DIM(631)
	Однократное выполнение по положительному фронту	@DIM(631)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

N: Номер таблицы

Указывает номер таблицы. N должно находиться в диапазоне 0...15.

LR: Длина каждой строки

Указывает количество слов в каждой строке. Значение LR должно принадлежать диапазону 0001...FFFF hex (1...65535 слов).

NR: Количество строк

Указывает количество строк в таблице. Значение NR должно принадлежать диапазону 0001...FFFF hex (1...65535 строк).

TB: Первое слово таблицы

Указывает первое слово таблицы. Все слова таблицы должны принадлежать одной области. Другими словами, TB и TB+LR×NR-1 должны принадлежать одной области.

Характеристики операндов

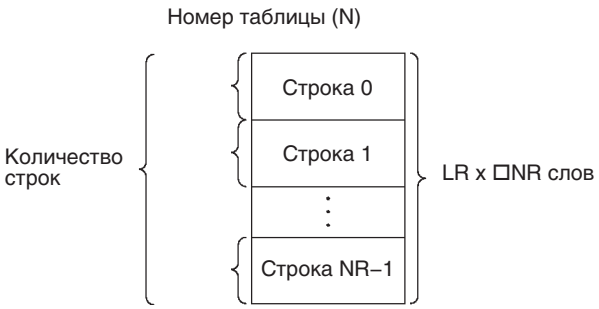
Область	N	LR	NR	TB
Область CIO	---	CIO 0...CIO 6143		
Рабочая область	---	W0...W511		
Область битов хранения	---	H0...H511		
Область вспомогательных битов	---	A0...A959		A448...A959
Область таймеров	---	T0000...T4095		
Область счетчиков	---	C0000...C4095		
Область DM	---	D0...D32767		
Косвенные адреса DM в двоичном формате	---	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	---	*D0...*D32767		
Постоянные	0...15	#0001...#FFFF (двоичный) или &1...&65 535		---
Регистры данных	---	DR0...DR15		---
Регистры указателей	---	---		
Косвенная адресация с помощью регистров указателей	---	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда DIM(631) регистрирует слова TB...TB+LR×NR-1 как таблицу под номером N. Таблица под номером N содержит NR строк по LR слов каждая. Данные, хранящиеся в области, объявленной таблицей, не могут быть изменены после такого объявления.

Используйте команду DIM(631) вместе с командой SETR(635) (ЗАПИСАТЬ РАСПОЛОЖЕНИЕ СТРОКИ) или GETR(636) (СЧИТАТЬ НОМЕР СТРОКИ) для упрощения вычисления адресов в таблицах данных. Используйте команду DIM(631), чтобы разбить данные на строки, а затем команду SETR(635) для записи адреса первого слова нужной строки в регистр указателя. Регистр указателя можно использовать в качестве указателя в других командах, например, в командах чтения, записи, поиска или сравнения.

К примеру, если заданные значения температуры, давления и других параметров хранятся в виде строк, и строки, относящиеся к разным моделям, объединены в таблицу, это облегчает считывание заданных значений для каждой модели при каких-либо конкретных условиях.



С командой DIM(631) используются две команды для работы с таблицами: SETR(635) и GETR(636). Команда SETR(635) записывает в указанный регистр указателя адрес в памяти ПЛК первого слова строки с указанным номером. Команда GETR(636) выводит номер строки, в которую входит значение указанного регистра указателя (адрес в памяти ПЛК).

Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если значение LR или NR равно «0000». Выключен во всех остальных случаях.

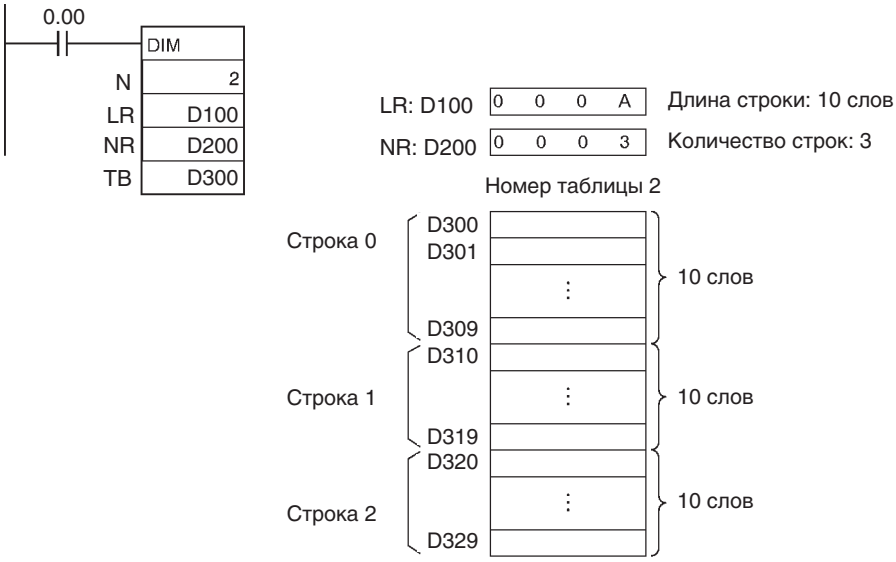
Меры предосторожности

Для идентификации строк в зарегистрированной таблице используются номера строк от 0 до NR-1.

В зависимости от настроек длины строки (LR) и количества строк (NR) одна таблица (TB...TB+LR×NR-1) может быть расположена сразу в двух областях данных. Прежде чем создавать таблицу, выходящую за пределы области данных, убедитесь, что это не вызовет никаких проблем.

Примеры

Если в следующем примере включен бит CIO 0.00, команда DIM(631) создает таблицу под номером 2 с тремя строками по десять слов каждая. Таблица начинается со слова D300.

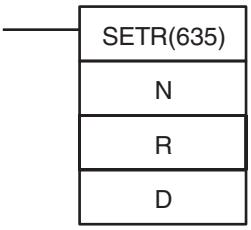


3-16-6 ЗАПИСАТЬ РАСПОЛОЖЕНИЕ СТРОКИ: SETR(635)

Назначение

Запись адреса расположения указанной строки (адреса начала строки в памяти ПЛК) в указанный регистр указателя.

Символ РКС



N: Номер таблицы

R: Номер строки

D: Адресуемый регистр указателя

Варианты выполнения

Варианты выполнения		
	Выполнение в каждом цикле при включенном условии выполнения	SETR(635)
	Однократное выполнение по положительному фронту	@SETR(635)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограм-мы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

N: Номер таблицы

Указывает номер таблицы. N должно находиться в диапазоне от 0 до 15.

R: Номер строки

Указывает номер нужной строки. R должно принадлежать диапазону 0000...FFFE hex (0...65534). Нумерация строк начинается с нуля, поэтому номер строки в таблице с NR строками должен принадлежать диапазону 0...NR-1.

D: Адресуемый регистр указателя

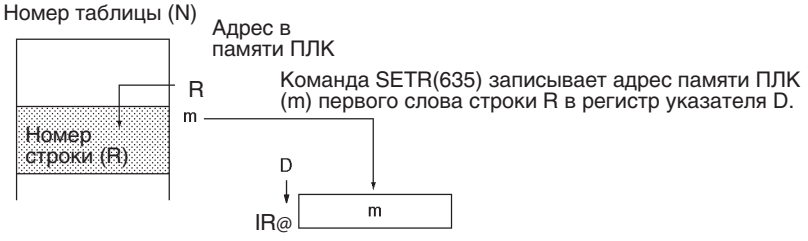
Указывает требуемый регистр указателя. D должно находиться в диапазоне от IR0 до IR15.

Характеристики операндов

Область	N	R	D
Область CIO	---	CIO 0...CIO 6143	---
Рабочая область	---	W0...W511	---
Область битов хранения	---	H0...H511	---
Область вспомогательных битов	---	A0...A959	---
Область таймеров	---	T0000...T4095	---
Область счетчиков	---	C0000...C4095	---
Область DM	---	D0...D32767	---
Косвенные адреса DM в двоичном формате	---	@ D0...@ D32767	---
Косвенные адреса DM в BCD-формате	---	*D0...*D32767	---
Постоянные	0...15	#0000...#FFFE (двоичн.) или &0...65534	---
Регистры данных	---	DR0...DR15	---
Регистры указателей	---		IR0...IR15
Косвенная адресация с помощью регистров указателей	---	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,– (– –)IR0..., – (– –)IR15	---

Описание

Команда SETR(635) записывает адрес первого слова указанной строки в памяти ПЛК в указанный регистр указателя. На следующей схеме показано выполнение команды SETR(635).



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если таблица с указанным номером (N) не была определена командой DIM(631). Включен, если указанный номер строки (R) больше номера последней строки в таблице (NR-1). Выключен во всех остальных случаях.

Меры предосторожности

Таблица должна быть создана заранее с помощью команды DIM(631).

Номера строк должны принадлежать диапазону 0...NR-1, где NR — число строк, указанное при создании таблицы командой DIM(631).

Примеры

Если в следующем примере включен бит CIO 0.00, команда SETR(635) находит адрес памяти ПЛК первого слова строки 3 в таблице 10 и записывает этот адрес в регистр указателя IR11.

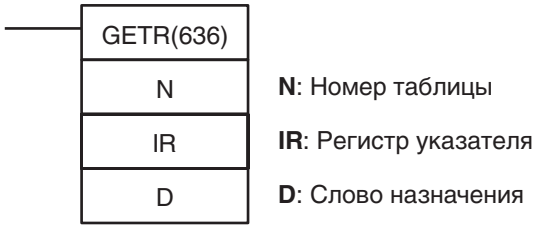


3-16-7 СЧИТАТЬ НОМЕР СТРОКИ: GETR(636)

Назначение

Чтение номера строки по адресу памяти ПЛК, содержащемуся в указанном регистре указателя.

Символ РКС



N: Номер таблицы

IR: Регистр указателя

D: Слово назначения

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	GETR(636)
	Однократное выполнение по положительному фронту	@GETR(636)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

N: Номер таблицы

Указывает номер таблицы. N должно находиться в диапазоне от 0 до 15.

IR: Регистр указателя

Указывает требуемый регистр указателя. D должно находиться в диапазоне от IR0 до IR15.

D: Слово назначения

Указывает слово, в которое будет записан номер строки.

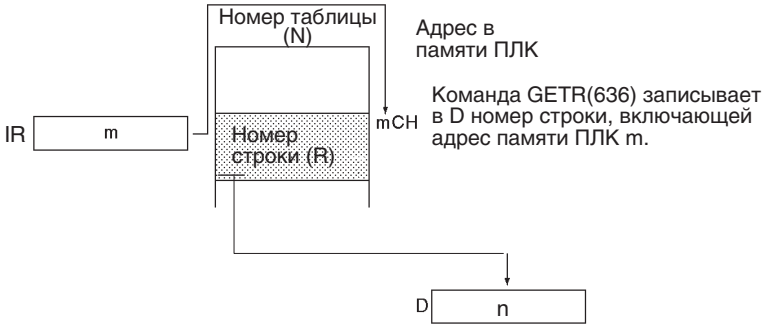
Характеристики операндов

Область	N	IR	D
Область CIO	---		CIO 0...CIO 6143
Рабочая область	---		W0...W511
Область битов хранения	---		H0...H511
Область вспомогательных битов	---		A448...A959
Область таймеров	---		T0000...T4095
Область счетчиков	---		C0000...C4095
Область DM	---		D0...D32767
Косвенные адреса DM в двоичном формате	---		@ D0...@ D32767
Косвенные адреса DM в BCD-формате	---		*D0...*D32767
Постоянные	0...15	---	---
Регистры данных	---		DR0...DR15
Регистры указателей	---	IR0...IR15	---
Косвенная адресация с помощью регистров указателей	---		,IR0...,IR15 -2048...+2047 ,IR0... -2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15

Описание

Команда GETR(636) находит строку, в которую входит адрес памяти ПЛК, содержащийся в указанном регистре указателя, и записывает номер этой строки в D. Адрес памяти ПЛК, содержащийся в регистре указателя, не обязательно должен быть адресом первого слова в строке. Это может быть адрес любого слова строки.

На следующей схеме показано выполнение команды GETR(636).



Флаги

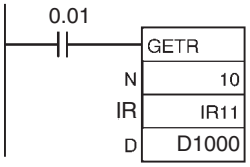
Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если адрес памяти ПЛК в указанном регистре указателя находится за пределами указанной таблицы (N). Включен, если таблица с указанным номером (N) не была определена командой DIM(631). Выключен во всех остальных случаях.

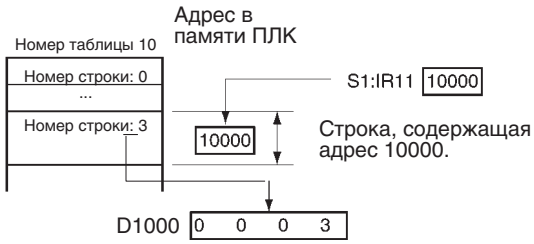
Меры предосторожности

Таблица должна быть создана заранее с помощью команды DIM(631), а адрес памяти ПЛК в указанном регистре указателя должен находиться в пределах указанной таблицы.

Примеры

Если в следующем примере бит CIO 0.01 включен, команда GETR(636) находит номер строки, содержащей адрес памяти ПЛК в регистре указателя IR11, и записывает этот номер строки в D1000.



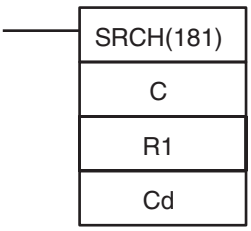


3-16-8 ПОИСК ЗНАЧЕНИЯ: SRCH(181)

Назначение

Поиск данных в пределах массива слов.

Символ РКС



C: Первое управляющее слово

R1: Первое слово диапазона

CD: Сравниваемое значение

Варианты выполнения

Варианты выполнения		
	Выполнение в каждом цикле при включенном условии выполнения	SRCH(181)
	Однократное выполнение по положительному фронту	@SRCH(181)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

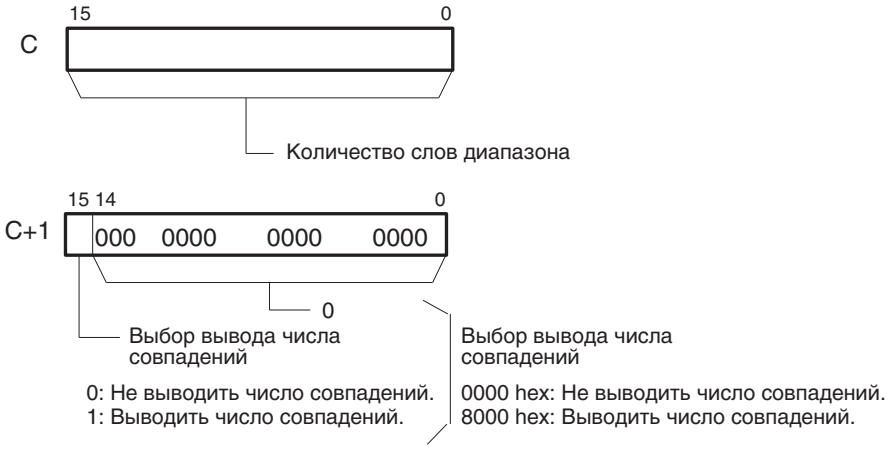
Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

C и C+1: Управляющие слова

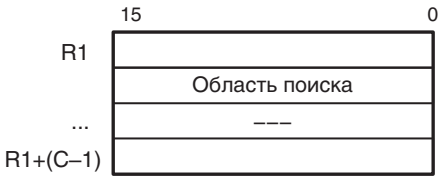
C указывает количество слов в массиве, а состояние бита 15 слова C+1 определяет, выводится ли число совпадений в слово DR0.



Примечание. C и C+1 должны принадлежать одной области.

R1: Первое слово диапазона

R1 указывает первое слово области поиска. Поиск указанных данных осуществляется в диапазоне R1...R1+(C-1). (C — количество слов, указанное в слове C.)



Примечание. R1 и R1+C-1 должны принадлежать одной области.

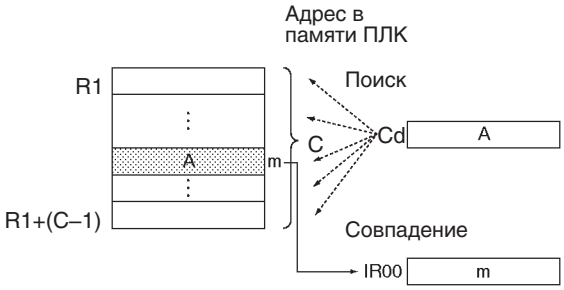
Характеристики операндов

Область	C	R1	Cd
Область CIO	CIO 0...CIO 6142	CIO 0...CIO 6143	
Рабочая область	W0...W510	W0...W511	
Область битов хранения	H0...H510	H0...H511	
Область вспомогательных битов	A0...A958	A0...A959	
Область таймеров	T0000...T4094	T0000...T4095	
Область счетчиков	C0000...C4094	C0000...C4095	
Область DM	D0...D32766	D0...D32767	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	Только указанные значения	---	#0000...#FFFF (двоичн.)
Регистры данных	---		DR0...DR15
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда SRCH(181) ищет слова, содержащие сравниваемое значение (Cd), в диапазоне R1...R1+C-1. При обнаружении совпадения команда SRCH(181) записывает адрес слова в памяти ПЛК в IR0 и устанавливает флаг равенства.
(При обнаружении двух совпадений и более в IR0 записывается только адрес первого слова, содержащего сравниваемое значение.)

Если установлен бит 15 слова C+1, команда SRCH(181) записывает число совпадений в DR0. Если бит 15 слова C+1 сброшен, DR0 остается неизменным.



Команда SRCH(181) осуществляет поиск в таблице, каждая строка которой содержит одно слово. Для поиска в таблице, строки которой содержат более одного слова, следует использовать команды DIM(631), SETR(635), GETR(636), FOR(512)–NEXT(513) или BREAK(514), а также регистр указателя (IR).

Чтобы узнать, были ли обнаружены совпадения, можно проверить состояние флага равенства сразу после выполнения поиска.

Флаги

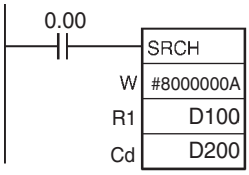
Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если содержимое C выходит за допустимый диапазон 0001...FFFF. Включен, если указано выполнение в фоновом режиме и при этом выключен флаг доступности порта связи, соответствующий номеру порта связи, указанному в *Com Port number (Номер COM-порта)* для *Background Execution (Выполнение в фоновом режиме)* (только для CP1H). Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если одно или несколько слов в области поиска содержат сравниваемое значение. Выключен во всех остальных случаях.

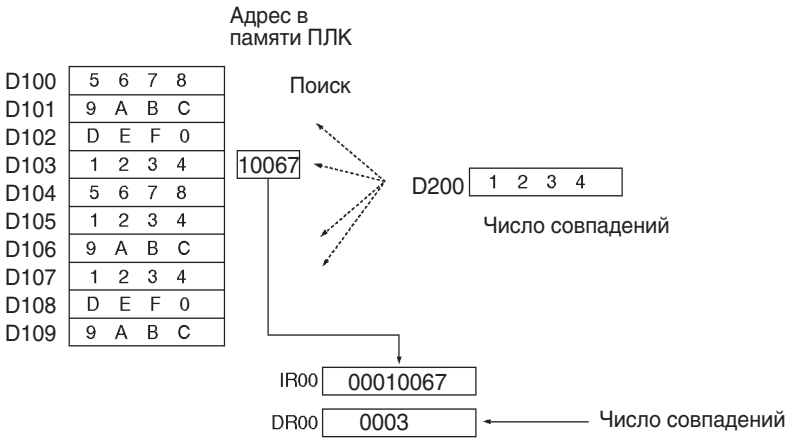
Меры предосторожности

Если совпадений не обнаружено, содержимое IR0 и DR0 остается неизменным.

Примеры

Если в следующем примере включен бит CIO 0.00, команда SRCH(181) осуществляет поиск слов, значение которых совпадает с содержимым D200, в массиве из 10 слов, начиная с D100. Адрес памяти ПЛК первого слова, значение которого совпадает со сравниваемым значением, записывается в IR0, а общее число совпадений записывается в DR0.





Если в качестве длины таблицы указано значение &10 (10 десятичн.) или A hex, количество совпадений не выводится в регистр данных DR0.

3-16-9 ПОМЕНЯТЬ МЕСТАМИ БАЙТЫ: SWAP(637)

Назначение

Обмен местами старшего и младшего байтов во всех словах массива.

Символ РКС



Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	SWAP(637)
	Однократное выполнение по положительному фронту	@SWAP(637)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	**Области пошаговых программ**	**Подпрограммы**	**Задачи обработки прерываний**
OK	OK	OK	OK

Операнды

N: Количество слов

N указывает число слов в массиве и должно принадлежать диапазону 0001...FFFF hex (или &1...&65535).

R1: Первое слово диапазона

R1 указывает первое слово диапазона.

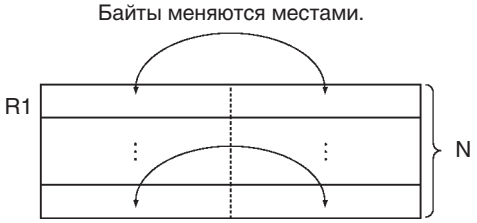


Характеристики операндов

Область	N	R1
Область CIO	CIO 0...CIO 6143	
Рабочая область	W0...W511	
Область битов хранения	H0...H511	
Область вспомогательных битов	A0...A959	A448...A959
Область таймеров	T0000...T4095	
Область счетчиков	C0000...C4095	
Область DM	D0...D32767	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Постоянные	#0001...#FFFF (двоичный) или &1...&65 535	---
Регистры данных	DR0...DR15	---
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

Описание

Команда SWAP(637) меняет местами два байта во всех словах массива памяти R1...R1+N-1. Эту команду можно использовать, чтобы заменить порядок расположения символов ASCII в каждом слове на обратный.

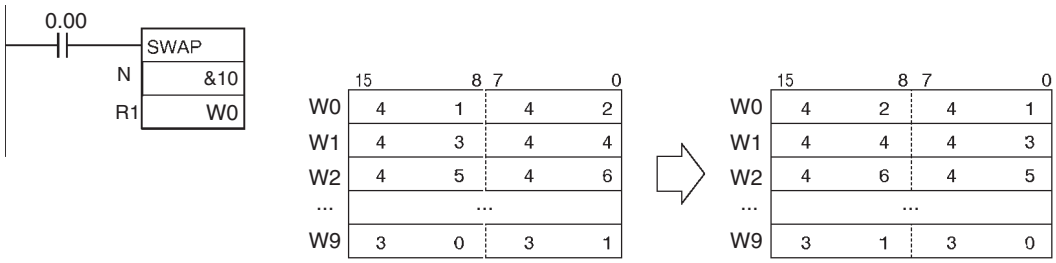


Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если N = 0000. Включен, если указано выполнение в фоновом режиме и при этом выключен флаг доступности порта связи, соответствующий номеру порта связи, указанному в *Com Port number (Номер COM-порта)* для *Background Execution (Выполнение в фоновом режиме)* (только для CP1H). Выключен во всех остальных случаях.

Примеры

Если в следующем примере включен бит CIO 0.00, команда SWAP(637) меняет местами значения старших и младших байтов в каждом слове массива из 10 слов от W0 до W9.

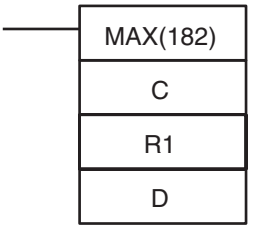


3-16-10 НАЙТИ МАКСИМУМ: MAX(182)

Назначение

Поиск максимального значения в массиве.

Символ РКС



C: Первое управляющее слово

R1: Первое слово диапазона

D: Слово назначения

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	MAX(182)
	Однократное выполнение по положительному фронту	@MAX(182)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

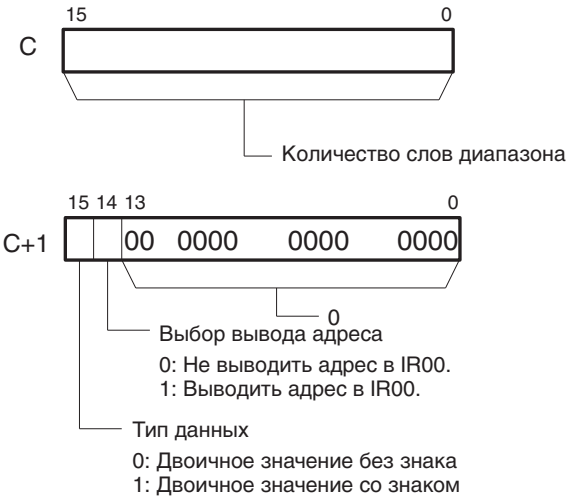
Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

C и C+1: Управляющие слова

C указывает число слов в массиве. В зависимости от состояния бита 15 слова C+1 значения воспринимаются как двоичные значения со знаком или двоичные значения без знака. В зависимости от состояния бита 14 слова C+1 адрес в памяти ПЛК слова, содержащего максимальное значение, выводится или не выводится в IR0.

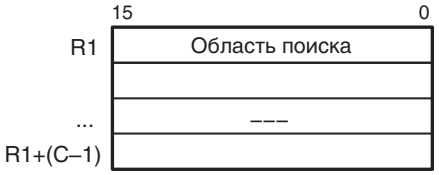


В следующей таблице приведены возможные значения C.

C+1	Тип данных	Вывод в регистр указателя
0000	Двоичное без знака	Нет
4000	Двоичное без знака	Да
8000	Двоичное со знаком	Нет
C000	Двоичное со знаком	Да

R1: Первое слово диапазона

R1 указывает первое слово области поиска. Поиск максимального значения осуществляется в массиве R1...R1+(C-1). (C — кол-во слов, указанное в C.)



Характеристики операндов

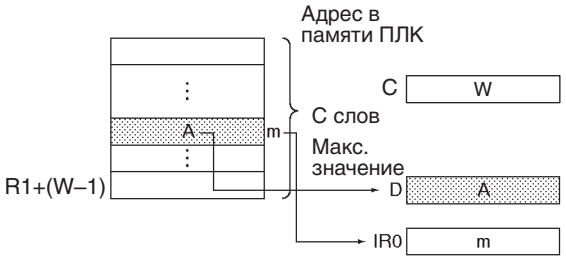
Область	C	R1	D
Область CIO	CIO 0...CIO 6142	CIO 0...CIO 6143	
Рабочая область	W0...W510	W0...W511	
Область битов хранения	H0...H510	H0...H511	
Область вспомогательных битов	A0...A958	A0...A959	A448...A959
Область таймеров	T0000...T4094	T0000...T4095	
Область счетчиков	C0000...C4094	C0000...C4095	
Область DM	D0...D32766	D0...D32767	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	Только указанные значения	---	
Регистры данных	---		DR0...DR15
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда MAX(182) ищет максимальное значение в массиве памяти R1...R1+C-1 и выводит его в D.

Если бит 14 слова C+1 установлен, команда MAX(182) записывает адрес памяти ПЛК слова, содержащего максимальное значение, в IR0. (Если максимальное значение содержат два слова или более, в IR0 записывается адрес первого слова с максимальным значением.)

Если бит 15 слова C+1 установлен, команда MAX(182) воспринимает значения массива как двоичные значения со знаком.



Флаги

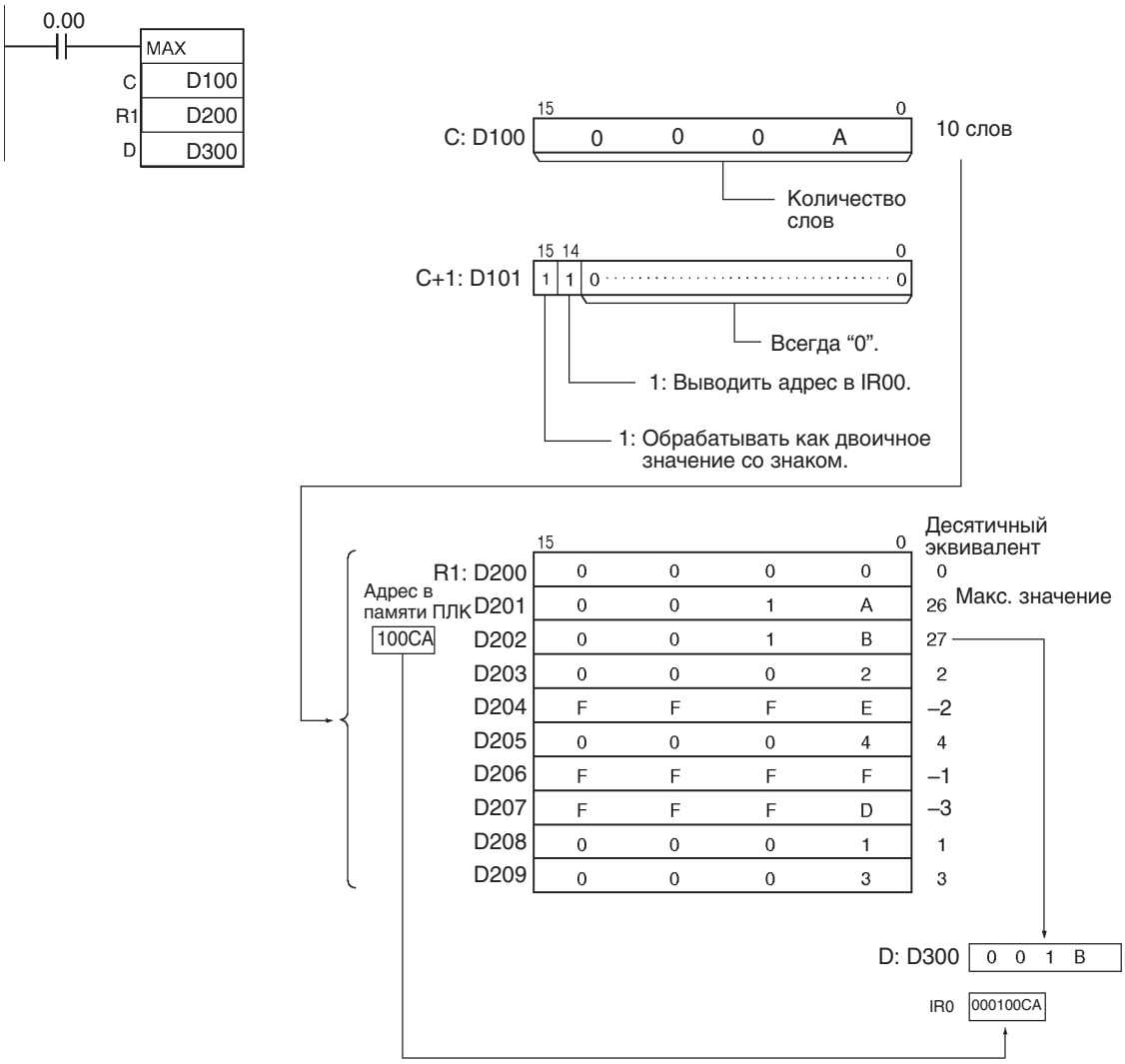
Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если содержимое C выходит за допустимый диапазон 0001...FFFF. Включен, если указано выполнение в фоновом режиме и при этом выключен флаг доступности порта связи, соответствующий номеру порта связи, указанному в *Com Port number (Номер COM-порта)* для *Background Execution (Выполнение в фоновом режиме)* (только для CP1H). Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если максимальное значение равно «0000». Выключен во всех остальных случаях.
Флаг отрицательного значения	N	Включен, если в слове, содержащем максимальное значение, установлен бит 15. Выключен во всех остальных случаях.

Меры предосторожности

Если бит 15 слова C+1 установлен в состояние «1», значение в пределах диапазона обрабатывается как двоичное значение со знаком, то есть шестнадцатеричные значения от 8000 до FFFF считаются отрицательными. Таким образом, результат поиска может варьироваться в зависимости от указанного типа данных.

Примеры

Если в следующем примере устанавливается бит CIO 0.00, команда MAX(182) выполняет поиск максимального значения в массиве из 10 слов (число слов указано в D100), начинающемся с D200. Максимальное значение записывается в D300, а адрес памяти ПЛК слова, содержащего максимальное значение, записывается в IR0.

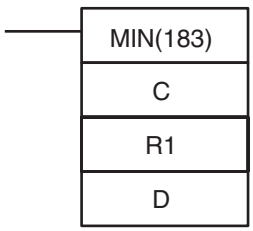


3-16-11 НАЙТИ МИНИМУМ: MIN(183)

Назначение

Поиск минимального значения в массиве.

Символ РКС



C: Первое управляющее слово

R1: Первое слово диапазона

D: Слово назначения

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	MIN(183)
	Однократное выполнение по положительному фронту	@MIN(183)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

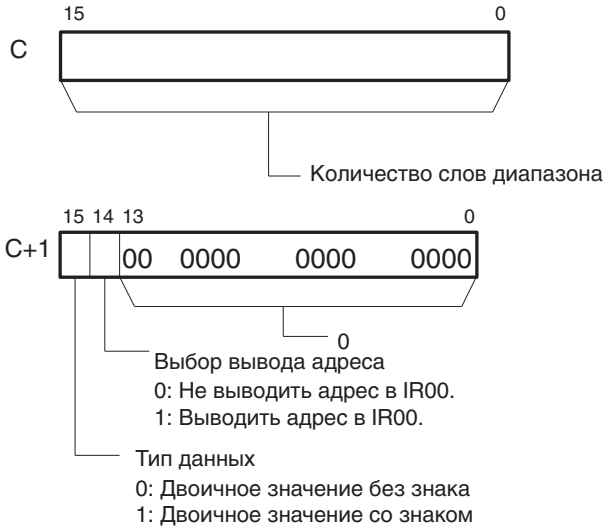
Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

C и C+1: Управляющие слова

C указывает число слов в массиве. В зависимости от состояния бита 15 слова C+1 значения воспринимаются как двоичные значения со знаком или двоичные значения без знака. В зависимости от состояния бита 14 слова C+1 адрес в памяти ПЛК слова, содержащего минимальное значение, выводится или не выводится в IR0.

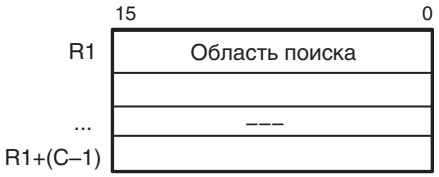


В следующей таблице приведены возможные значения C.

C+1	Тип данных	Вывод в регистр указателя
0000	Двоичное без знака	Нет
4000	Двоичное без знака	Да
8000	Двоичное со знаком	Нет
C000	Двоичное со знаком	Да

R1: Первое слово диапазона

R1 указывает первое слово области поиска. Поиск минимального значения осуществляется в массиве R1...R1+(C-1). (C — кол-во слов, указанное в C.)



Характеристики операндов

Область	C	R1	D
Область CIO	CIO 0...CIO 6142	CIO 0...CIO 6143	
Рабочая область	W0...W510	W0...W511	
Область битов хранения	H0...H510	H0...H511	
Область вспомогательных битов	A0...A958	A0...A959	A448...A959

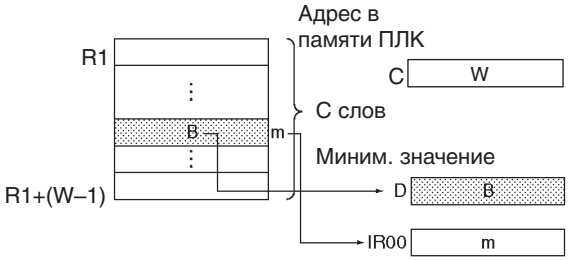
Область	C	R1	D
Область таймеров	T0000...T4094	T0000...T4095	
Область счетчиков	C0000...C4094	C0000...C4095	
Область DM	D0...D32766	D0...D32767	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	Только указанные значения	---	
Регистры данных	---		DR0...DR15
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда MIN(183) ищет минимальное значение в массиве памяти R1...R1+C-1 и выводит его в D.

Если установлен бит 14 слова C+1, команда MIN(183) записывает в IR0 адрес памяти ПЛК слова, содержащего минимальное значение. (Если минимальное значение содержат два слова или более, в IR0 записывается адрес первого слова с минимальным значением.)

Если бит 15 слова C+1 установлен, команда MIN(183) воспринимает значения массива как двоичные значения со знаком.



Флаги

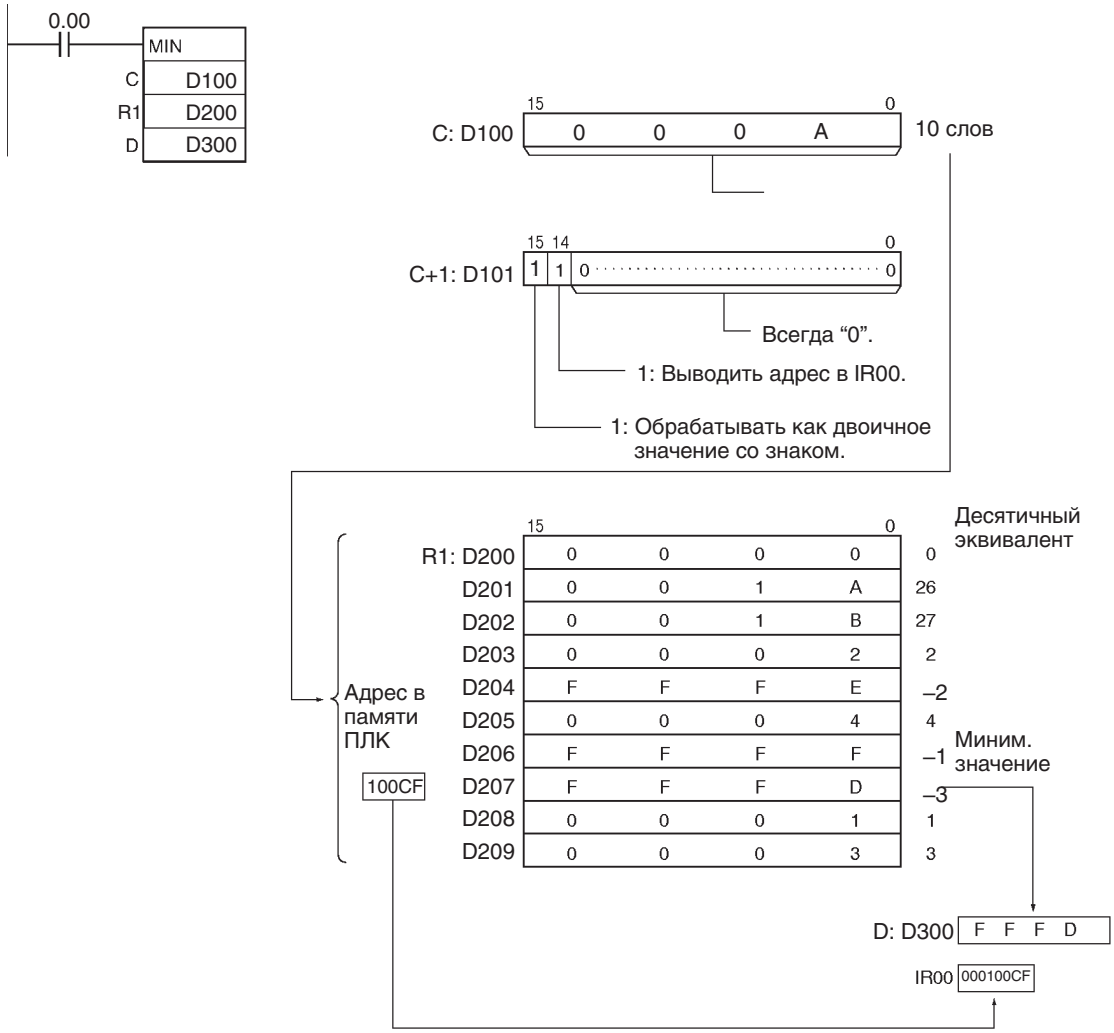
Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если содержимое C выходит за допустимый диапазон 0001...FFFF. Включен, если указано выполнение в фоновом режиме и при этом выключен флаг доступности порта связи, соответствующий номеру порта связи, указанному в *Com Port number (Номер COM-порта)* для *Background Execution (Выполнение в фоновом режиме)* (только для CP1H). Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если минимальное значение равно «0000». Выключен во всех остальных случаях.
Флаг отрицательного значения	N	Включен, если в слове, содержащем минимальное значение, установлен бит 15. Выключен во всех остальных случаях.

Меры предосторожности

Если бит 15 слова C+1 установлен в состояние «1», значение в пределах диапазона обрабатывается как двоичное значение со знаком, то есть шестнадцатеричные значения от 8000 до FFFF считаются отрицательными. Таким образом, результат поиска может варьироваться в зависимости от указанного типа данных.

Примеры

Если в следующем примере устанавливается бит CIO 0.00, команда MIN(183) выполняет поиск минимального значения в массиве из 10 слов (что указано в D100), начинающемся с D200. Минимальное значение записывается в D300, а адрес памяти ПЛК слова, содержащего минимальное значение, записывается в IR0.

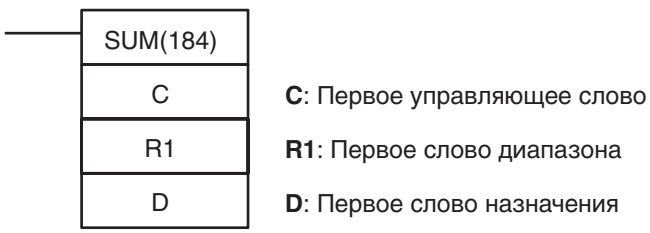


3-16-12 СЛОЖЕНИЕ: SUM(184)

Назначение

Сложение байтов или слов массива и запись результата в два слова.

Символ РКС



C: Первое управляющее слово

R1: Первое слово диапазона

D: Первое слово назначения

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	SUM(184)
	Однократное выполнение по положительному фронту	@SUM(184)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

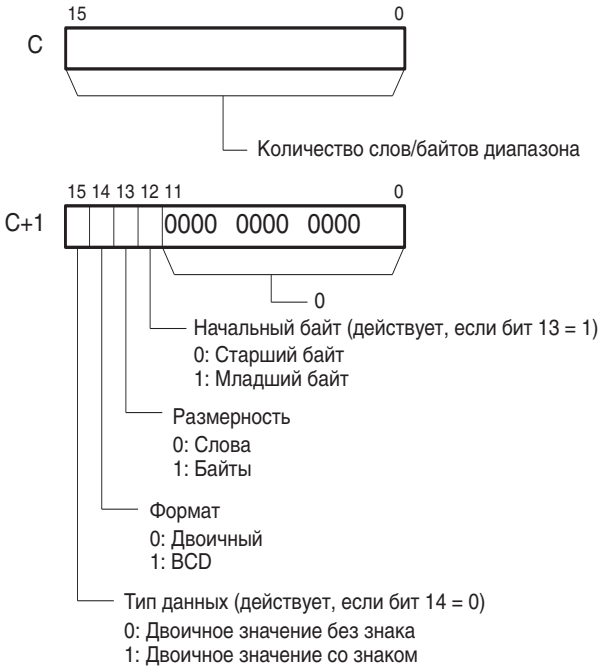
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

C и C+1: Управляющие слова

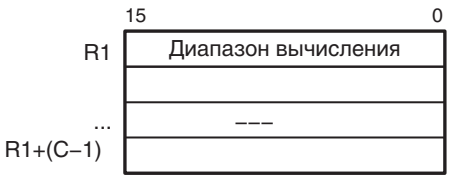
C указывает количество суммируемых единиц (байтов или слов). (В зависимости от состояния бита 13 слова C+1 суммируются байты или слова.)

Биты 12...15 слова C+1 указывают тип суммируемых значений, как показано на следующей схеме.



R1: Первое слово диапазона

R1 указывает первое слово диапазона. Длина массива зависит от количества единиц, а также от начального байта, если выполняется сложение байтов.



D: Первое слово назначения

Результат вычислений выводится в слова D+1 и D. Четыре старших разряда записываются в D+1, а четыре младших разряда записываются в D.

Характеристики операндов

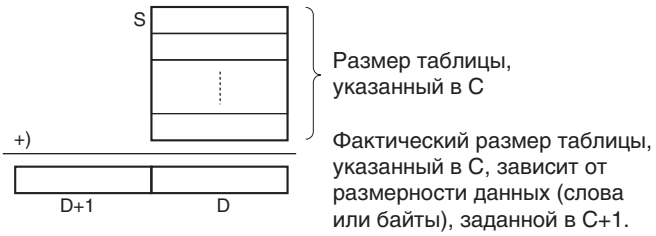
Область	C	R1	D
Область CIO	CIO 0...CIO 6142	CIO 0...CIO 6143	CIO 0...CIO 6142
Рабочая область	W0...W510	W0...W511	W0...W510
Область битов хранения	H0...H510	H0...H511	H0...H510
Область вспомогательных битов	A0...A958	A0...A959	A448...A958
Область таймеров	T0000...T4094	T0000...T4095	T0000...T4094
Область счетчиков	C0000...C4094	C0000...C4095	C0000...C4094
Область DM	D0...D32766	D0...D32767	D0...D32766
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	Только указанные значения	---	
Регистры данных	---		
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда SUM(184) складывает C слов или байтов, начиная с содержимого R1, и выводит результат в D+1 и D. В зависимости от значения параметров в слове C+1 осуществляется сложение слов или байтов, значения воспринимаются как двоичные (со знаком или без) или двоично-десятичные, сложение байтов начинается со старшего или младшего байта R1.

Если бит 14 слова C+1 сброшен, команда SUM(184) воспринимает значения как двоичные. В этом случае бит 15 определяет, как должны восприниматься данные: как значения со знаком (бит 15 = 1) или без знака (бит 15 = 0).

Если бит 13 слова C+1 установлен, команда SUM(184) складывает байты данных. В этом случае бит 12 определяет, с какого байта слова R1 должны начинаться вычисления: с младшего байта (бит 12 = 1) или старшего байта (бит 12 = 0).

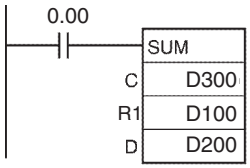


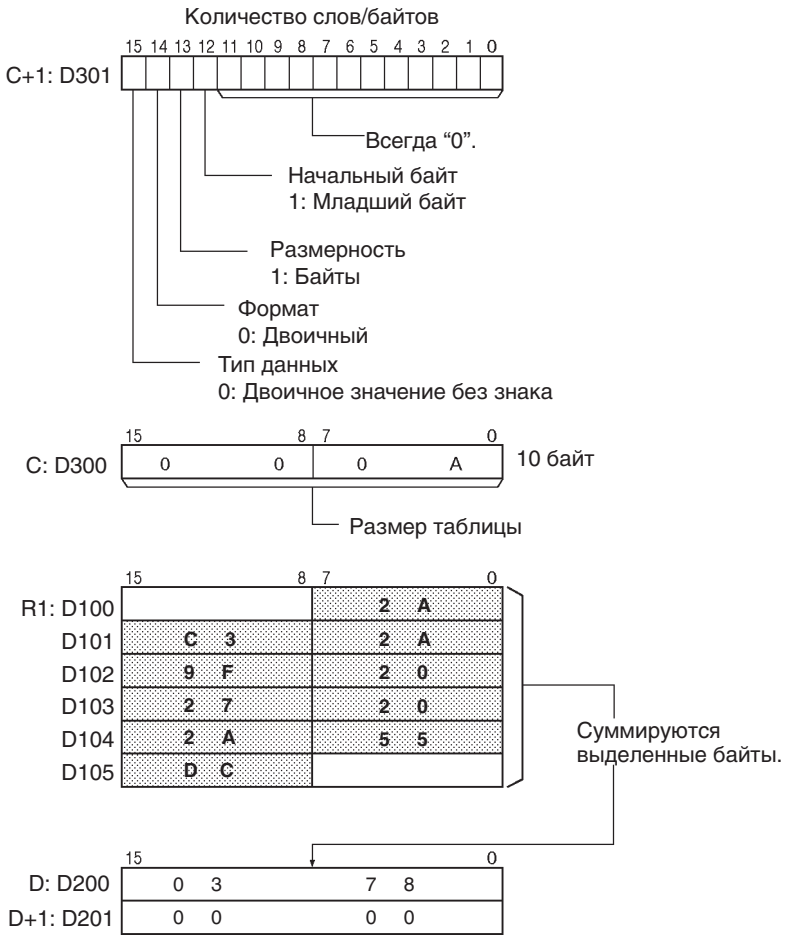
Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если содержимое C выходит за допустимый диапазон 0001...FFFF. Включен, если был выбран двоично-десятичный формат, а массив содержит двоичные значения. Включен, если указано выполнение в фоновом режиме и при этом выключен флаг доступности порта связи, соответствующий номеру порта связи, указанному в *Com Port number (Номер COM-порта)* для *Background Execution (Выполнение в фоновом режиме)* (только для CP1H). Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если результат = 0000. Выключен во всех остальных случаях.
Флаг отрицательного значения	N	Включен, если бит 15 результата = «1». Выключен во всех остальных случаях.

Примеры

Если в следующем примере включен бит CIO 0.00, команда SUM(184) складывает 10 байтов (число байтов указывается в D300) двоичных значений без знака, начиная с младшего байта слова D100, и записывает результат в D201 и D200.



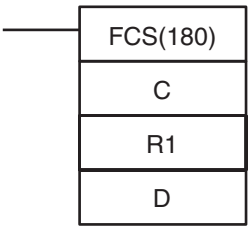


3-16-13 КОНТРОЛЬНАЯ СУММА КАДРА: FCS(180)

Назначение

Вычисление контрольной суммы кадра (FCS) в ASCII формате для указанного массива.

Символ РКС



C: Первое управляющее слово

R1: Первое слово диапазона

D: Первое слово назначения

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	FCS(180)
	Однократное выполнение по положительному фронту	@FCS(180)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

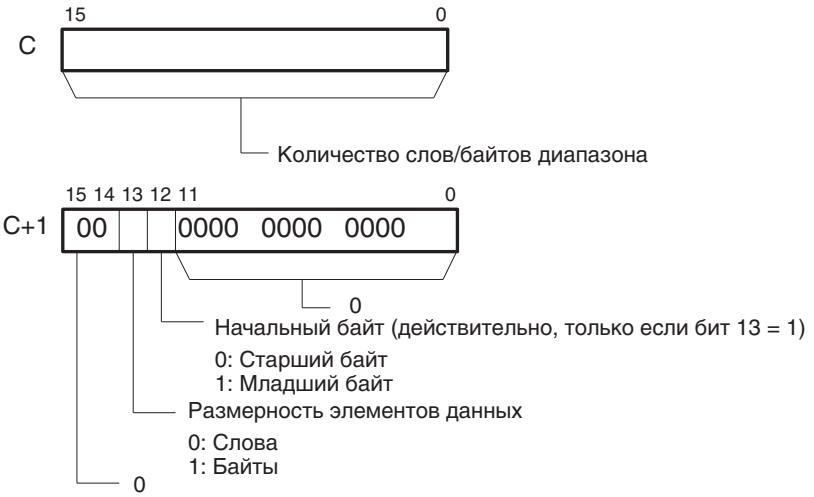
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

C и C+1: Управляющие слова

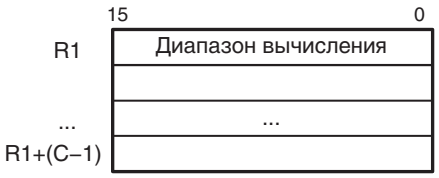
C указывает количество единиц (байтов или слов), используемых при расчете контрольной суммы кадра. (Бит 13 слова C+1 определяет, какие единицы будут использоваться: байты или слова.)

Если бит 13 слова C+1 установлен, команда FCS(180) рассчитывает значение контрольной суммы кадра для байтов данных. В этом случае бит 12 определяет, с какого байта слова R1 должны начинаться вычисления: с младшего байта (бит 12 = 1) или старшего байта (бит 12 = 0).



R1: Первое слово диапазона

R1 указывает первое слово диапазона. Длина массива зависит от количества единиц, а также начального байта, если в вычислениях используются байты.



D: Первое слово назначения

Если были выбраны байты, результат вычислений выводится в D.

Если были выбраны слова, результат выводится в D+1 и D. В этом случае четыре старших разряда записываются в D+1, а четыре младших разряда записываются в D.

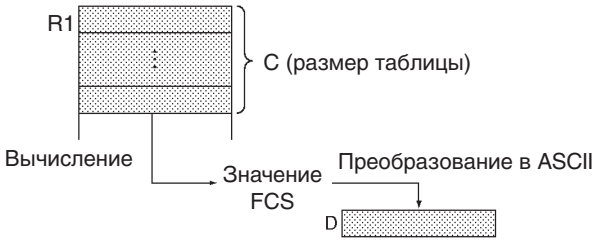
Характеристики операндов

Область	C	R1	D
Область CIO	CIO 0...CIO 6142	CIO 0...CIO 6143	
Рабочая область	W0...W510	W0...W511	
Область битов хранения	H0...H510	H0...H511	
Область вспомогательных битов	A0...A958	A0...A959	A448...A959
Область таймеров	T0000...T4094	T0000...T4095	
Область счетчиков	C0000...C4094	C0000...C4095	
Область DM	D0...D32766	D0...D32767	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	Только указанные значения	---	
Регистры данных	---		
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда FCS(180) рассчитывает значение контрольной суммы кадра для C единиц данных, начиная с содержимого слова R1, преобразует значение в ASCII формат и выводит результат в D (для байтов) или D+1 и D (для слов). В зависимости от значения параметров в слове C+1 при вычислении контрольной суммы используются слова или байты, значения воспринимаются как двоичные (со знаком или без) или двоично-десятичные, вычисление суммы по байтам начинается со старшего или младшего байта R1.

Если бит 13 слова C+1 установлен, команда FCS(180) вычисляет контрольную сумму кадра для байтов данных. В этом случае бит 12 определяет, с какого байта слова R1 должны начинаться вычисления: с младшего байта (бит 12 = 1) или старшего байта (бит 12 = 0).

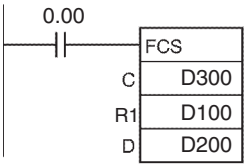


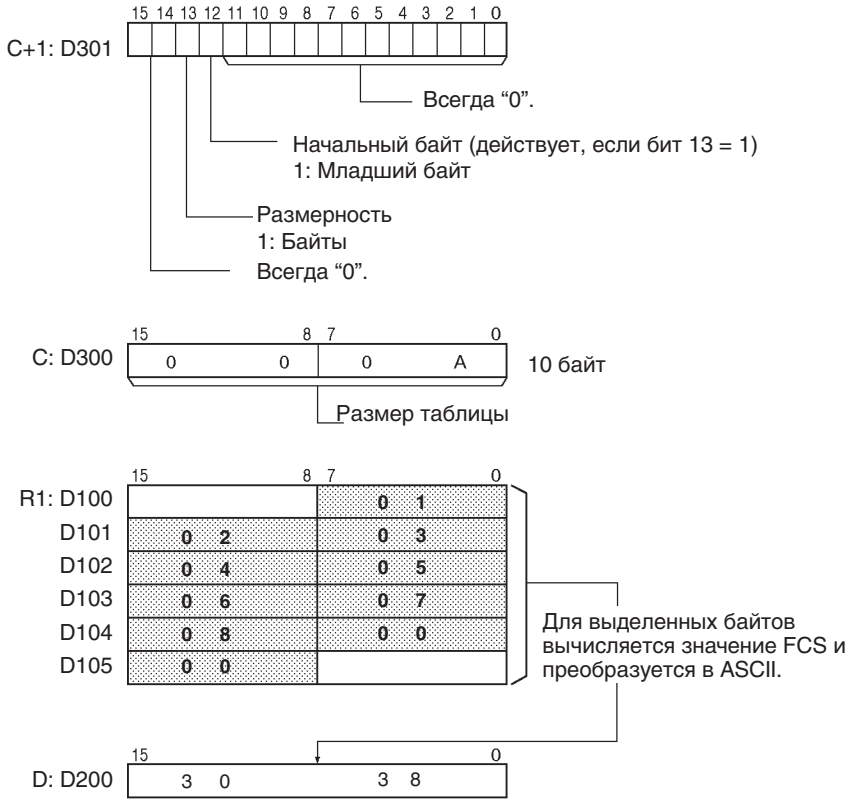
Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если содержимое C выходит за допустимый диапазон 0001...FFFF. Включен, если указано выполнение в фоновом режиме и при этом выключен флаг доступности порта связи, соответствующий номеру порта связи, указанному в *Com Port number (Номер COM-порта)* для *Background Execution (Выполнение в фоновом режиме)* (только для CP1H). Выключен во всех остальных случаях.

Примеры

Если в следующем примере включен бит CIO 0.00, команда FCS(180) рассчитывает значение контрольной суммы кадра для 10 байтов данных (число байтов указывается в D300), начиная с младшего байта слова D100, и записывает результат в D200.



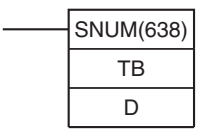


3-16-14 СЧИТАТЬ РАЗМЕР СТЕКА: SNUM(638)

Назначение

Расчет объема данных (количества слов) в указанном стеке.

Символ РКС



TB: Первый адрес области стека

D: Слово назначения

Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	SNUM(638)
	Однократное выполнение по положительному фронту	@SNUM(638)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

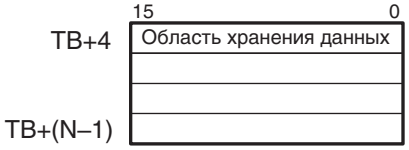
TB...TB+3: Слова управления стеком

Первые четыре слова стека содержат адрес памяти ПЛК последнего слова стека и указатель стека (адрес памяти ПЛК следующего имеющегося слова стека).



TB+4...TB+(N-1): Область хранения данных

Остальное пространство стека служит для хранения данных.



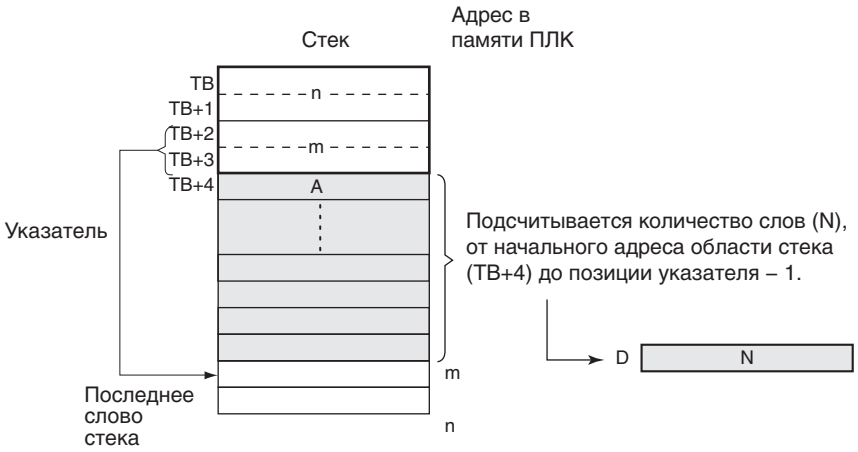
Характеристики операндов

Область	TB	D
Область CIO	CIO 0...CIO 6143	
Рабочая область	W0...W511	
Область битов хранения	H0...H511	
Область вспомогательных битов	A448...A959	
Область таймеров	T0000...T4095	

Область	TB	D
Область счетчиков	C0000...C4095	
Область DM	D0...D32767	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Постоянные	---	
Регистры данных	---	DR0...DR15
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

Описание

Команда SNUM(638) рассчитывает количество слов данных в указанном стеке от начала области данных (слово TB+4) до адреса, предшествующего адресу в указателе стека (TB+3 и TB+2). Команда SNUM(638) не изменяет данные в стеке и указатель стека.



Флаги

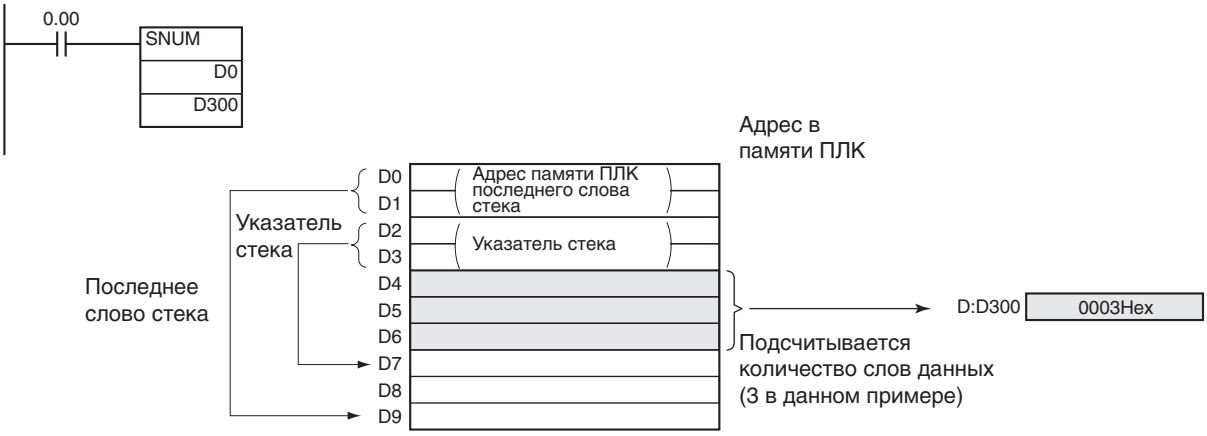
Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если количество слов в стеке (значение, которое выводится в D) равно «0». Выключен во всех остальных случаях.

Меры предосторожности

Стек должен быть определен заранее командой SSET(630).

Примеры

Если в следующем примере включен бит CIO 0.00, команда SNUM(638) рассчитывает количество слов с начала области данных (слово D4) до положения указателя стека - 1 (D6) и выводит результат в D300. (В данном случае указатель стека указывает на D7.) Область стека начинается с D0.

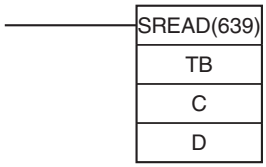


3-16-15 ПРОЧИТАТЬ ДАННЫЕ ИЗ СТЕКА: SREAD(639)

Назначение

Чтение данных из указанного элемента данных стека. Величина смещения указывает позицию требуемого элемента данных (сколько элементов данных находится до текущей позиции указателя).

Символ РКС



TB: Первый адрес области стека
C: Величина смещения
D: Слово назначения

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	SREAD(639)
	Однократное выполнение по положительному фронту	@SREAD(639)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

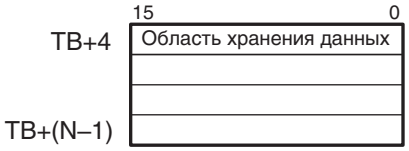
TB...TB+3: Слова управления стеком

Первые четыре слова стека содержат адрес памяти ПЛК последнего слова стека и указатель стека (адрес памяти ПЛК следующего имеющегося слова стека).



TB+4...TB+(N-1): Область хранения данных

Остальное пространство стека служит для хранения данных.

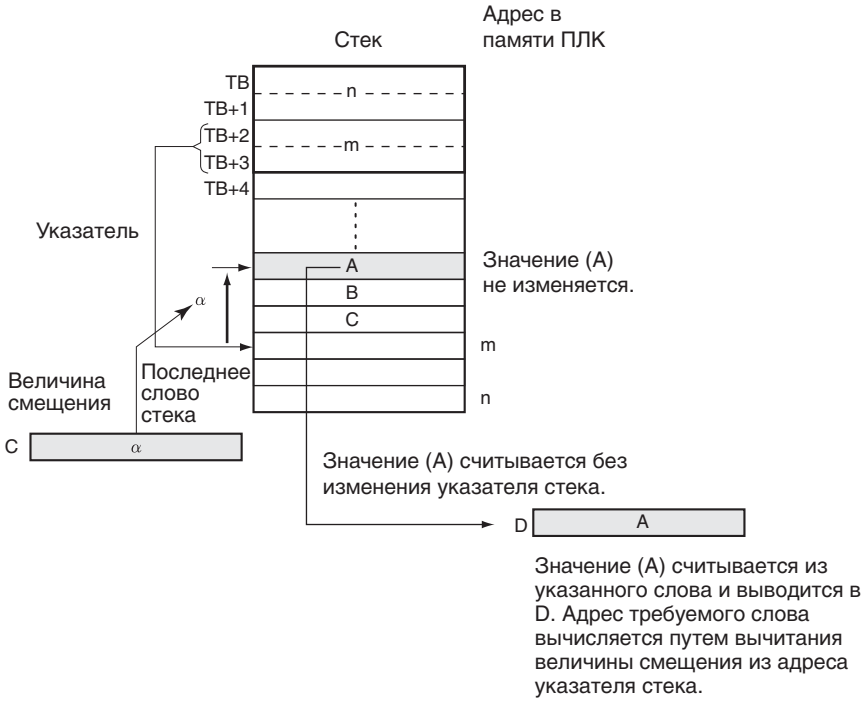


Характеристики операндов

Область	TB	C	D
Область CIO	CIO 0...CIO 6143		
Рабочая область	W0...W511		
Область битов хранения	H0...H511		
Область вспомогательных битов	A448...A959	A0...A959	A448...A959
Область таймеров	T0000...T4095		
Область счетчиков	C0000...C4095		
Область DM	D0...D32767		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	#0001...#FFFB		
Регистры данных	---	DR0...DR15	
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда SREAD(639) считывает данные по адресу, который определяется путем вычитания величины смещения, указанной в C, из адреса, содержащегося в указателе стека (TB+3 и TB+2). Команда SREAD(639) не изменяет данные в стеке и указатель стека.



Команду SREAD(639) можно использовать для считывания данных об объекте, находящемся на конвейере в данный момент. Для определения позиции требуемого объекта просто отсчитывается количество объектов (равное значению смещения) в обратном порядке от объекта, добавленного на конвейер самым последним.

Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если указанная область чтения находится за пределами области стека. Включен, если указанное в C значение смещения равно «0» или превышает максимальный размер области данных (FFFB hex). Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если выходное значение в D = 0000. Выключен во всех остальных случаях.

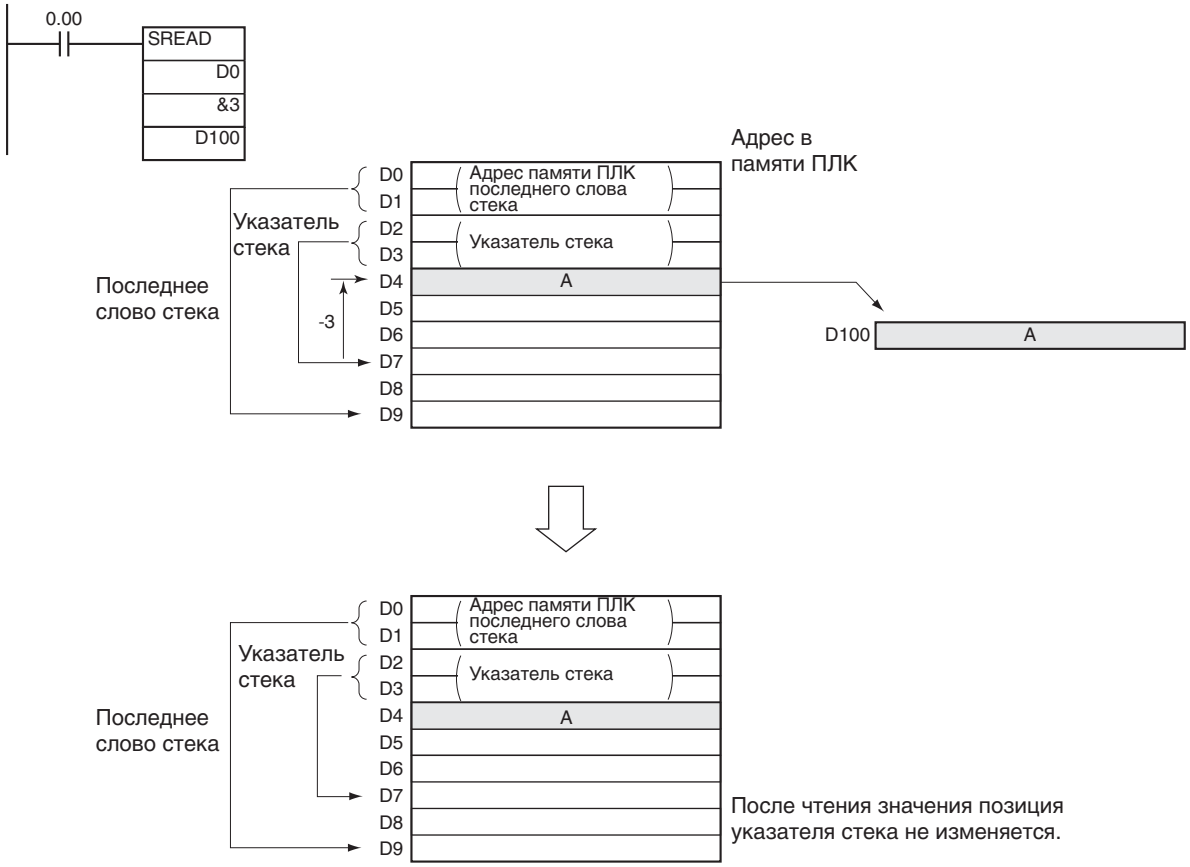
Меры предосторожности

Стек должен быть определен заранее командой SSET(630).

Адрес в указателе стека должен быть больше, чем адрес памяти ПЛК первого слова области данных (TB+4). Если значение указателя стека будет меньше, чем адрес памяти ПЛК слова TB+4, произойдет ошибка (ошибка выхода за нижнюю границу стека).

Примеры

Если в следующем примере включен бит CIO 0.00, команда SREAD(639) считывает данные из указанного слова стека, начинающегося с D0, и выводит их в D100. В этом случае указатель стека указывает на слово D7, а значение смещения равно 3, поэтому данные считываются из D4.

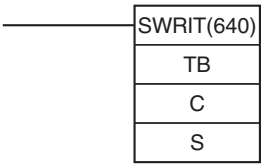


3-16-16 ЗАПИСАТЬ ДАННЫЕ В СТЕК: SWRIT(640)

Назначение

Запись исходных данных в указанный элемент данных стека (перезапись существующих данных). Величина смещения указывает позицию требуемого элемента данных (сколько элементов данных находится до текущей позиции указателя).

Символ РКС



TB: Первый адрес области стека

C: Величина смещения

S: Исходное слово

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	SWRIT(640)
	Однократное выполнение по положительному фронту	@SWRIT(640)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	**Области пошаговых программ**	**Подпрограммы**	**Задачи обработки прерываний**
OK	OK	OK	OK

Операнды

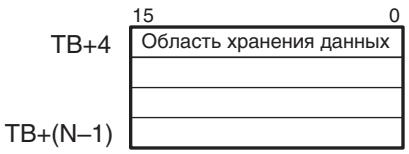
TB...TB+3: Слова управления стеком

Первые четыре слова стека содержат адрес памяти ПЛК последнего слова стека и указатель стека (адрес памяти ПЛК следующего имеющегося слова стека).



TB+4...TB+(N-1): Область хранения данных

Остальное пространство стека служит для хранения данных.

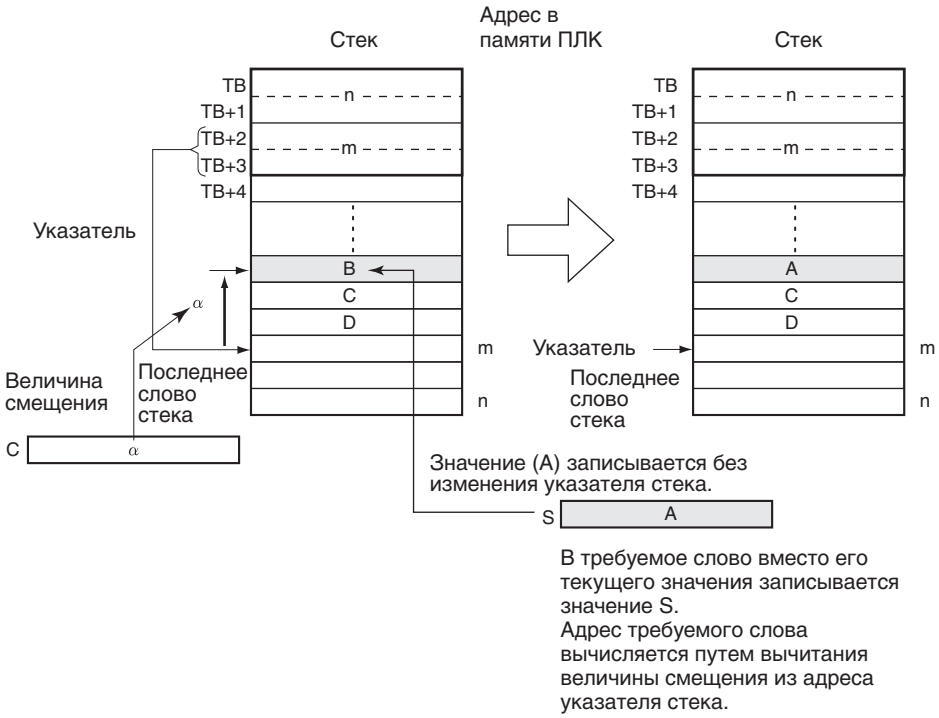


Характеристики операндов

Область	TB	C	S
Область CIO	CIO 0...CIO 6143		
Рабочая область	W0...W511		
Область битов хранения	H0...H511		
Область вспомогательных битов	A448...A959	A0...A959	
Область таймеров	T0000...T4095		
Область счетчиков	C0000...C4095		
Область DM	D0...D32767		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	---	#0001...#FFFB (Hex)	#0000...#FFFF (Hex)
Регистры данных	---	DR0...DR15	
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда SWRIT(640) записывает в требуемое слово данные, указанные в S. Адрес требуемого слова рассчитывается путем вычитания величины смещения, указанной в C, из указателя стека (TB+3 и TB+2). Команда SWRIT(640) не изменяет указатель стека.



Команду SWRIT(640) можно использовать для изменения данных об объекте, находящемся на конвейере в данный момент. Для определения позиции требуемого объекта просто отсчитывается количество объектов (равное значению смещения) в обратном порядке от объекта, добавленного на конвейер самым последним.

Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если указанная область записи находится за пределами области стека. Включен, если указанное в C значение смещения равно «0» или превышает максимальный размер области данных (FFFB hex). Выключен во всех остальных случаях.

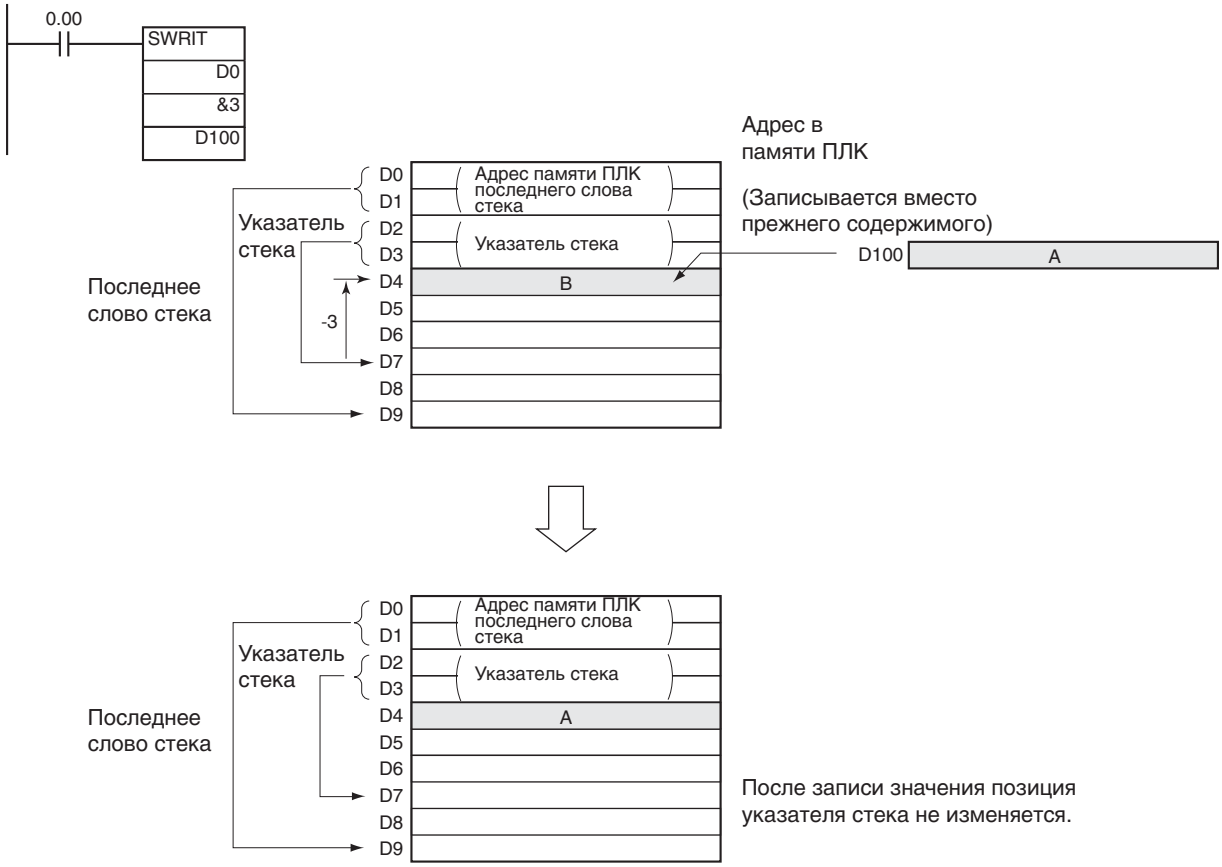
Меры предосторожности

Стек должен быть определен заранее командой SSET(630).

Адрес в указателе стека должен быть больше, чем адрес памяти ПЛК первого слова области данных (TB+4). Если значение указателя стека будет меньше, чем адрес памяти ПЛК слова TB+4, произойдет ошибка (ошибка выхода за нижнюю границу стека).

Примеры

Если в следующем примере включен бит CIO 0.00, команда SWRIT(640) записывает данные, содержащиеся в D100, в указанное слово стека, начинающегося с D0. В данном случае указатель стека указывает на D7, а значение смещения равно 3, поэтому перезаписывается значение слова D4.

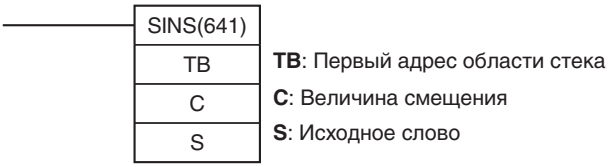


3-16-17 ВСТАВИТЬ ДАННЫЕ В СТЕК: SINS(641)

Назначение

Вставка входных данных в указанную позицию стека и смещение остальных данных стека вниз. Величина смещения указывает позицию требуемого элемента данных (сколько элементов данных находится до текущей позиции указателя).

Символ РКС



Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	SINS(641)
	Однократное выполнение по положительному фронту	@SINS(641)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

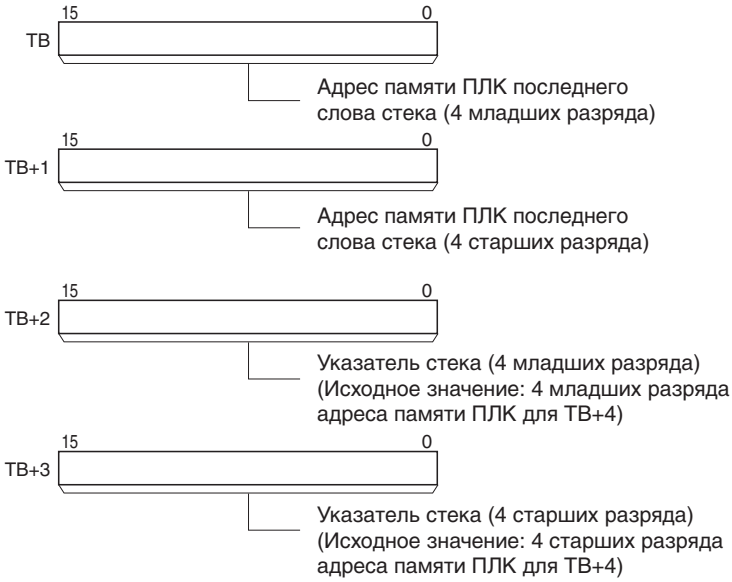
Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограм-мы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

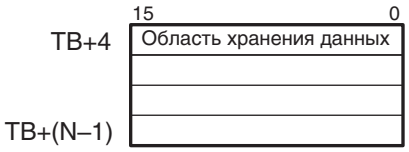
TB...TB+3: Слова управления стеком

Первые четыре слова стека содержат адрес памяти ПЛК последнего слова стека и указатель стека (адрес памяти ПЛК следующего имеющегося слова стека).



TB+4...TB+(N-1): Область хранения данных

Остальное пространство стека служит для хранения данных.

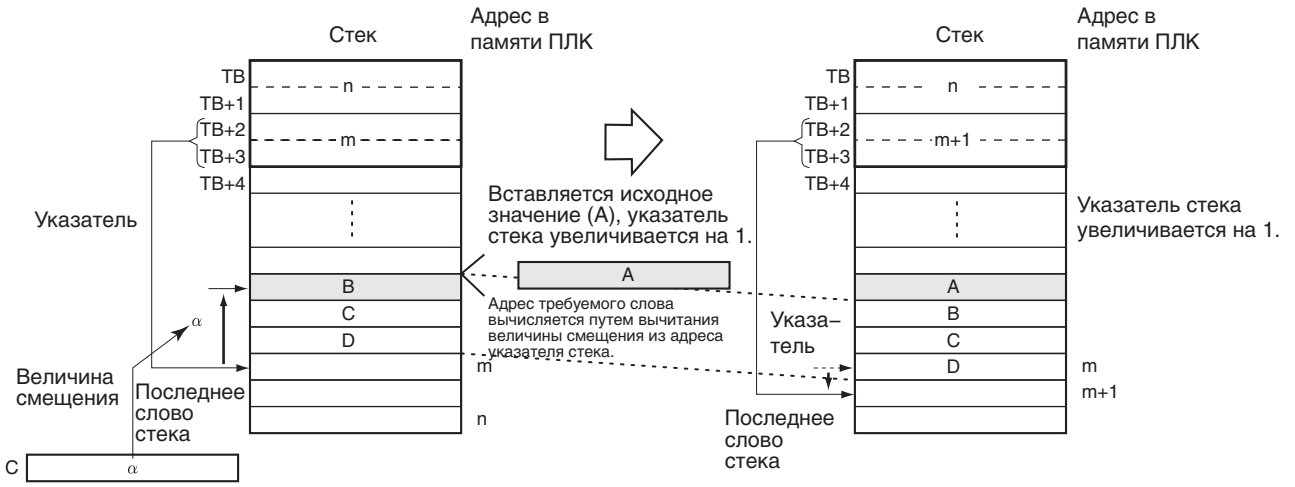


Характеристики операндов

Область	TB	C	S
Область CIO	CIO 0...CIO 6143		
Рабочая область	W0...W511		
Область битов хранения	H0...H511		
Область вспомогательных битов	A448...A959	A0...A959	
Область таймеров	T0000...T4095		
Область счетчиков	C0000...C4095		
Область DM	D0...D32767		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	---	#0001...#FFFB (Hex)	#0000...#FFFF (Hex)
Регистры данных	---	DR0...DR15	
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда SINS(641) вставляет исходные данные в указанный адрес и сдвигает существующие данные на одно слово вниз. При выполнении команды SINS(641) указатель стека (TB+3 и TB+2) увеличивается на 1. Требуемый адрес рассчитывается путем вычитания величины смещения, указанной в C, из указателя стека.



Команду SINS(641) можно использовать для вставки данных об объекте, помещенном между объектами, находящимися на конвейере. Для определения позиции вставки просто отсчитывается количество объектов (равное значению смещения) в обратном порядке от объекта, добавленного на конвейер самым последним.

Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если адрес, хранящийся в указателе стека (TB+3 и TB+2), больше, чем адрес памяти ПЛК последнего слова области данных стека. (Это ошибка переполнения стека.) Включен, если указанное значение смещения больше, чем максимальный размер области данных - 1 (FFFA hex). Выключен во всех остальных случаях.

Меры предосторожности

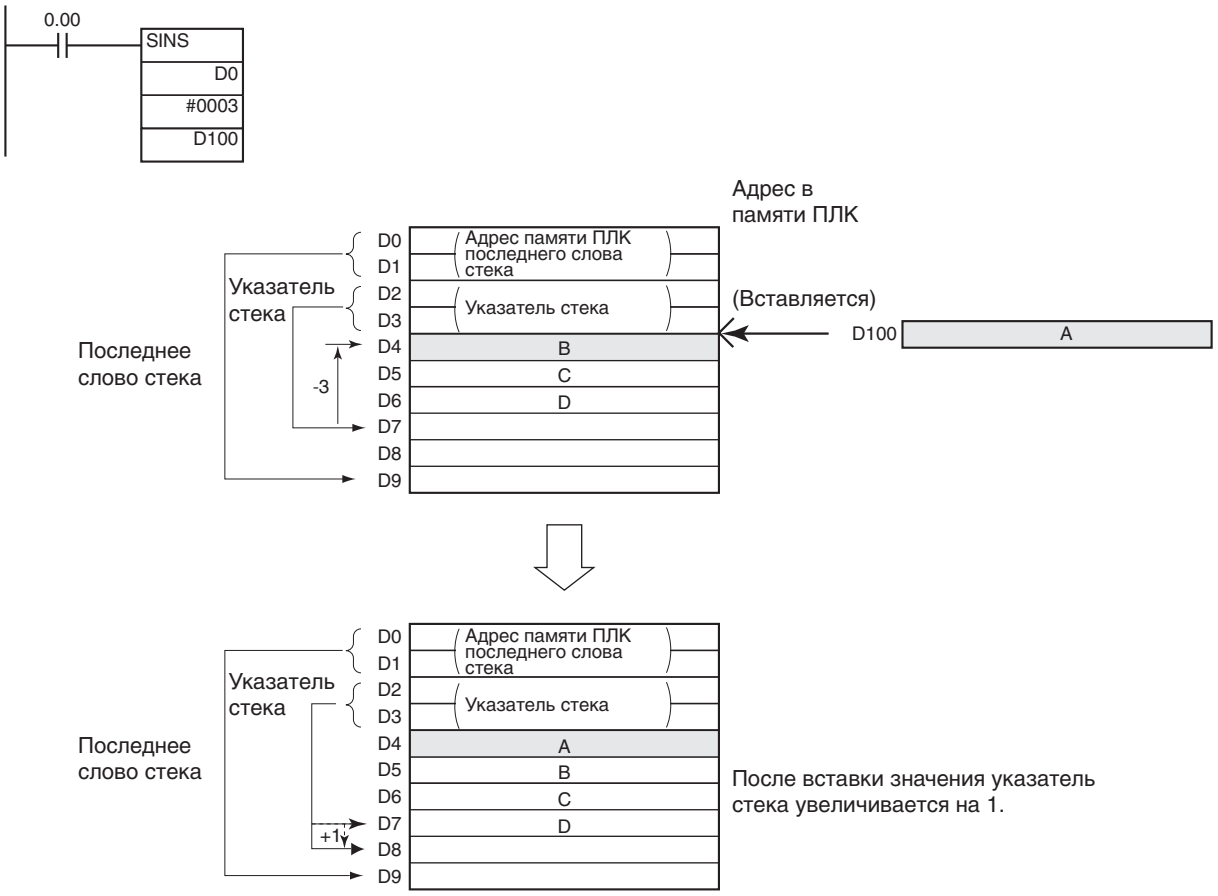
Стек должен быть определен заранее командой SSET(630).

Команда SINS(641) вставляет в стек одно слово данных, так что в конце стека должно быть хотя бы одно доступное слово. Если стек полон, будет сгенерирована ошибка и исходные данные не будут вставлены.

Если на момент выполнения команды SINS(641) адрес, хранящийся в указателе стека (TB+3 и TB+2), больше адреса последнего слова в стеке (TB+1 и TB), будет сгенерирована ошибка переполнения стека и исходные данные не будут вставлены.

Примеры

Если в следующем примере включен бит CIO 0.00, команда SINS(641) вставляет исходные данные, содержащиеся в D100, по указанному адресу в стек, начинающийся с D0. В данном случае указатель стека указывает на D7, а значение смещения равно 3, поэтому исходные данные вставляются в D4. Существующие данные сдвигаются на одно слово вниз, а данные в слове D7 перезаписываются. При этом указатель стека увеличивается с D7 до D8.

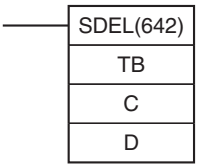


3-16-18 УДАЛИТЬ ДАННЫЕ ИЗ СТЕКА: SDEL(642)

Назначение

Удаление элемента данных, расположенного в указанной позиции стека, вывод этих данных в указанное слово назначения и сдвиг остальных данных стека вверх. Величина смещения указывает позицию требуемого элемента данных (сколько элементов данных находится до текущей позиции указателя).

Символ РКС



TB: Первый адрес области стека
C: Величина смещения
D: Слово назначения

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	SDEL(642)
	Однократное выполнение по положительному фронту	@SDEL(642)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

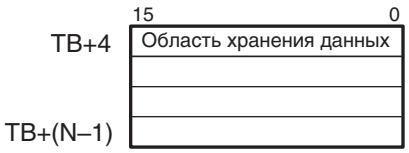
TB...TB+3: Слова управления стеком

Первые четыре слова стека содержат адрес памяти ПЛК последнего слова стека и указатель стека (адрес памяти ПЛК следующего имеющегося слова стека).



TB+4...TB+(N-1): Область хранения данных

Остальное пространство стека служит для хранения данных.

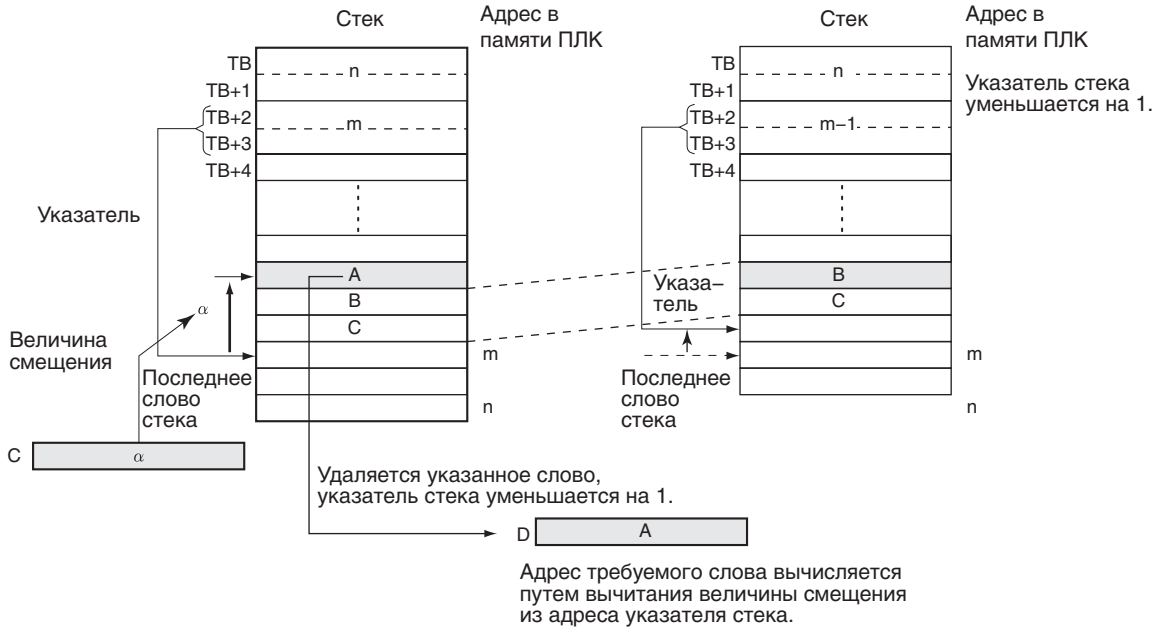


Характеристики операндов

Область	TB	C	D
Область CIO	CIO 0...CIO 6143		
Рабочая область	W0...W511		
Область битов хранения	H0...H511		
Область вспомогательных битов	A448...A959	A0...A959	A448...A959
Область таймеров	T0000...T4095		
Область счетчиков	C0000...C4095		
Область DM	D0...D32767		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	---	#0001...#FFFB (Hex)	---
Регистры данных	---	DR0...DR15	
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда SDEL(642) удаляет данные, расположенные в указанной позиции в стеке, выводит эти данные в указанное слово назначения и сдвигает остальные данные стека вверх. При выполнении команды SDEL(642) указатель стека (TB+3 и TB+2) уменьшается на 1. Требуемый адрес рассчитывается путем вычитания величины смещения, указанной в C, из указателя стека.



Команду SDEL(642) можно использовать для удаления данных об объекте, снятом с конвейера. Для определения позиции удаления просто отсчитывается количество объектов (равное значению смещения) в обратном порядке от объекта, добавленного на конвейер самым последним.

Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если содержимое указателя стека (TB+3 и TB+2) меньше или равно адресу памяти ПЛК первого слова области данных стека (TB+4). (Это ошибка выхода за нижнюю границу стека.) Включен, если указанное в C значение смещения равно «0» или превышает максимальный размер области данных (FFFB hex). Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если выходное значение в D = 0000. Выключен во всех остальных случаях.

Меры предосторожности

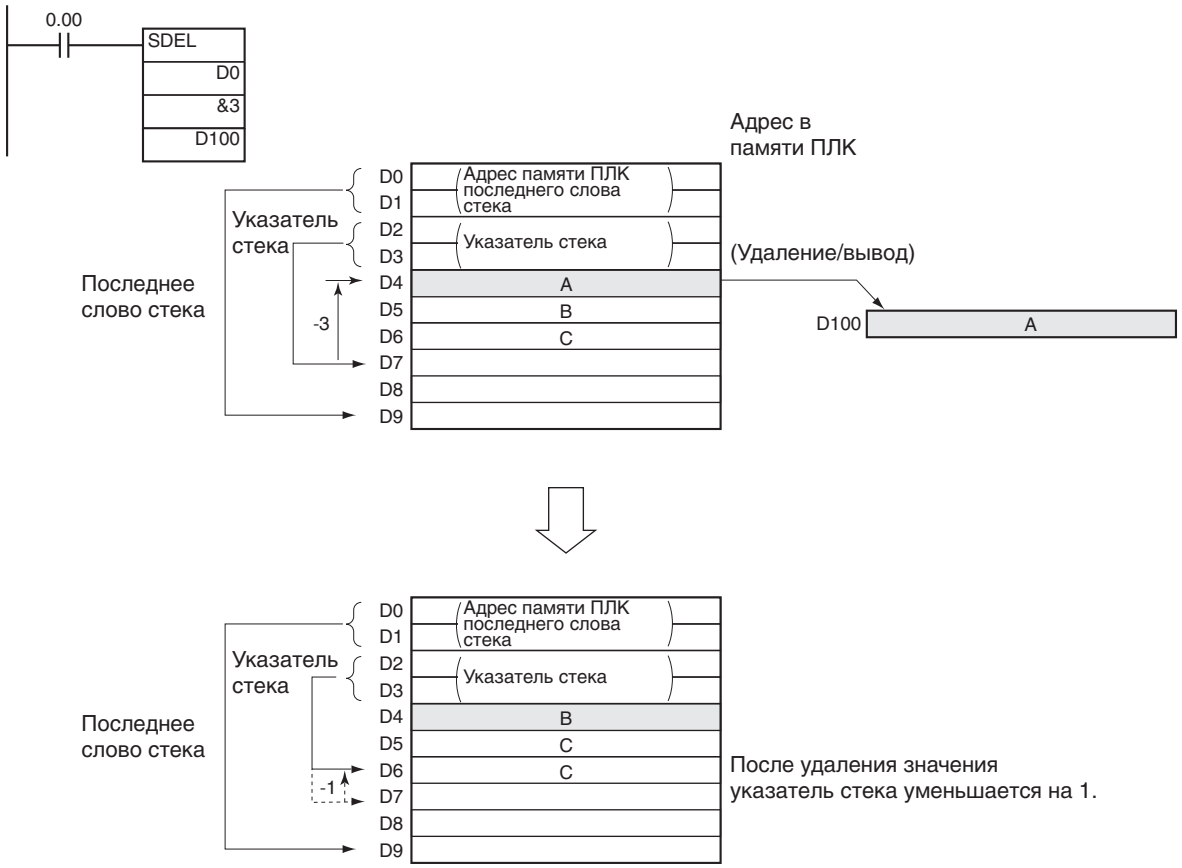
Стек должен быть определен заранее командой SSET(630).

Адрес в указателе стека должен быть больше, чем адрес памяти ПЛК первого слова области данных (TB+4). Если значение указателя стека будет меньше, чем адрес памяти ПЛК слова TB+4, произойдет ошибка (ошибка выхода за нижнюю границу стека).

Примеры

Если в следующем примере включен бит CIO 0.00, команда SDEL(642) удаляет слово по указанному адресу в стеке, начинающемся с D0, выводит удаленные данные в D100, сдвигает остальные данные вверх и уменьшает указатель стека на 1.

В этом случае указатель стека указывает на слово D7, а значение смещения равно 3, поэтому данные удаляются из D4. Остальные данные сдвигаются на одно слово вверх, а указатель стека уменьшается с D7 до D6.



3-17 Команды управления данными

В данном разделе описаны команды, предназначенные для управления отдельными специальными операциями.

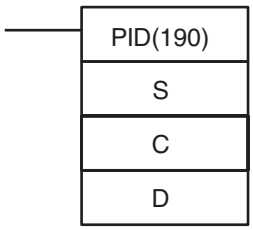
Команда	Мнемоническое обозначение	Код функции	Стр.
ПИД-РЕГУЛЯТОР	ПИД	190	671
ПИД-РЕГУЛЯТОР С АВТОНАСТРОЙКОЙ	PIDAT	191	684
ОГРАНИЧЕНИЕ	LMT	680	696
ЗОНА НЕЧУВСТВИТЕЛЬНОСТИ	BAND	681	698
МЕРТВАЯ ЗОНА	ZONE	682	701
ВЫХОД ШИМ	TPO	685	704
ИЗМЕНЕНИЕ МАСШТАБА	SCL	194	713
ИЗМЕНЕНИЕ МАСШТАБА 2	SCL2	486	718
ИЗМЕНЕНИЕ МАСШТАБА 3	SCL3	487	722
СРЕДНЕЕ ЗНАЧЕНИЕ	AVG	195	726

3-17-1 ПИД-РЕГУЛЯТОР: PID(190)

Назначение

Осуществление ПИД-регулирования в соответствии с заданными параметрами.

Символ РКС



S: Слово входных данных

C: Первое слово параметров

D: Слово выходных данных

Варианты выполнения

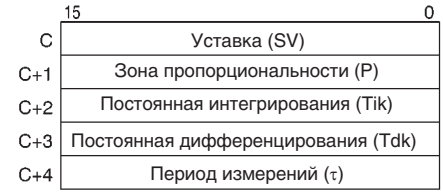
Варианты выполнения		
	Выполнение в каждом цикле при включенном условии выполнения	PID(190)
	Однократное выполнение по положительному фронту	Не предусмотрено
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

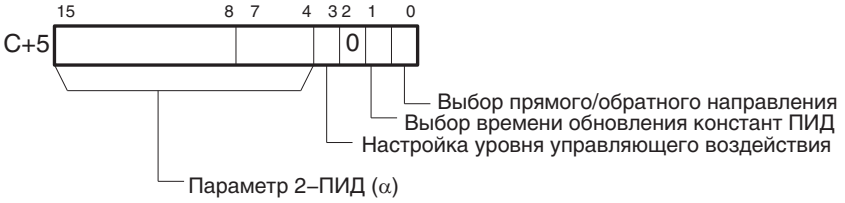
Доступные области программы

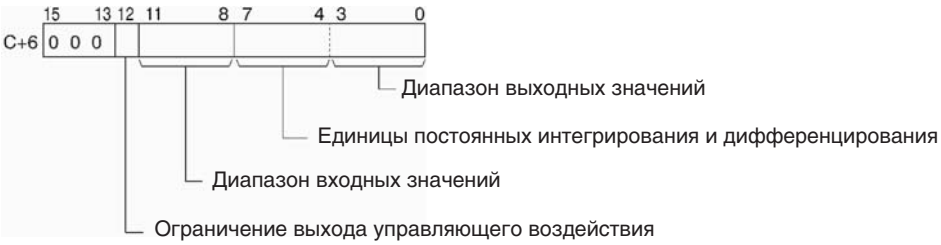
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
Не допускается	OK	OK	Не допускается

Параметры

Содержание слов параметров показано на следующем рисунке. Подробную информацию о параметрах см. в пункте *Описание и настройка параметров ПИД-регулирования* в данном разделе.







Характеристики операндов

Область	S	C	D
Область CIO	CIO 0...CIO 6143	CIO 0...CIO 6105	CIO 0...CIO 6143
Рабочая область	W0...W511	W0...W473	W0...W511
Область битов хранения	H0...H511	H0...H473	H0...H511
Область вспомогательных битов	A0...A959	A0...A921	A448...A959
Область таймеров	T0000...T4095	T0000...T4057	T0000...T4095
Область счетчиков	C0000...C4095	C0000...C4057	C0000...C4095
Область DM	D0...D32767	D0...D32729	D0...D32767
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767		
Постоянные	DR0...DR15	---	DR0...DR15
Регистры данных	---		
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15		

Описание

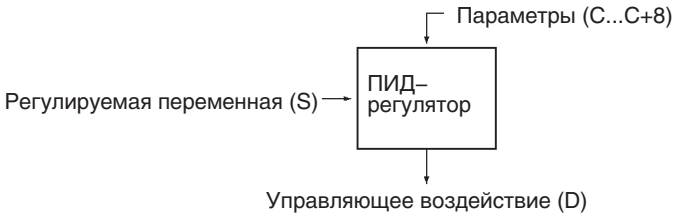
При включенном условии выполнения команда PID(190) осуществляет ПИД-регулирование (т. е. непрерывное приведение регулируемой переменной к заданному значению с применением пропорционального, интегрального и дифференциального звеньев) с фильтрацией задающего воздействия и двумя степенями свободы в соответствии с параметрами, заданными в C (уставка, ПИД-константы и др.). Она

принимает двоичное содержимое входного слова S, ограниченное указанным диапазоном (количеством действительных входных битов), и выполняет ПИД-управление в соответствии с заданными параметрами. Результат обработки (т. е. управляющее воздействие (MV)) выдается в слово D.

Заданные параметры считываются, когда условие выполнения переключается из состояния «0» в состояние «1». Если значение какого-либо параметра выходит за допустимый диапазон, устанавливается флаг ошибки.

Если значения параметров находятся в пределах допустимых диапазонов, в математических расчетах ПИД-регулятора используются исходные значения. Безударное управление в этот раз не производится. Оно будет применяться для вычисления величин управляющих воздействий в последующих циклах ПИД-регулирования. (Под безударным управлением понимается плавное и непрерывное изменение управляющего воздействия, исключающее нежелательные и потенциально опасные резкие изменения в состоянии процесса.)

После включения условия выполнения в ПИД-регулятор вводится значение регулируемой переменной (PV) для указанного периода измерения и производятся необходимые вычисления.



Количество действительных битов в 16-битовом слове входного значения S (регулируемая переменная) определяется битами 08...11 слова C+6 (настройка входного диапазона). Например, если входной диапазон ограничен 12 битами (4 hex), ПИД-регулятор работает со значениями регулируемой переменной в диапазоне от 0000 hex до 0FFF hex. (Любые значения свыше 0FFF hex воспринимаются как 0FFF hex.)

От выбранного входного диапазона также зависит диапазон значений уставки (SV) ПИД-регулятора.

Регулируемая переменная (измеряемый параметр технологического процесса, PV) и уставка (заданное значение параметра технологического процесса, SV) представляются в двоичном коде без знака и могут принимать значения в диапазоне от 0000 hex до максимального значения выбранного входного диапазона.

Количество действительных битов в 16-битовом слове выходного значения D (управляющее воздействие) определяется битами 00...03 слова C+6 (настройка выходного диапазона). Например, если выходной диапазон ограничен 12 битами (4 hex), ПИД-регулятор выдает значения управляющего воздействия в диапазоне от 0000 hex до 0FFF hex.

В случае только пропорционального управления выходное значение ПИД-регулятора (MV) в установившемся состоянии (PV = SV) определяется следующим образом:

0: выход 0%;
1: выход 50%.

Может быть указано прямое или обратное направление пропорционального воздействия.

Для выхода управляющего воздействия могут быть указаны верхнее и нижнее предельные значения.

Период измерения может быть задан с точностью до 10 мс (от 0,01 до 99,99 с), однако фактическое время работы ПИД-регулятора определяется не только периодом измерения, но и временем выполнения самой команды PID(190) (в каждом цикле).

В настройках можно выбрать, когда именно должны вступать в силу новые значения измененных ПИД-констант: 1) в начале выполнения команды ПИД-регулирования или 2) в начале выполнения команды ПИД-регулирования и в каждом периоде измерения. Изменение в каждом измерительном цикле (т. е. во время выполнения команды PID) возможно только для зоны пропорциональности (P), постоянной интегрирования (Tik) и постоянной дифференцирования (Tdk). Момент вступления изменения в силу задается битом 1 слова C+5.

Из всех параметров ПИД-регулятора (C...C+38) только уставка (SV) может изменяться при включенном условии выполнения. При изменении других параметров условие выполнения должно быть переключено из «0» в «1», чтобы новые значения вступили в силу.

Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	Включен, если значение C выходит за допустимый диапазон. Включен, если фактический период измерения более чем в два раза превосходит заданный период измерения. Выключен во всех остальных случаях.
Флаг большего значения («Больше»)	>	Включен, если управляющее воздействие на выходе ПИД-регулятора выходит за верхнее предельное значение. Выключен во всех остальных случаях.
Флаг меньшего значения («Меньше»)	<	Включен, если управляющее воздействие на выходе ПИД-регулятора находится ниже нижнего предельного значения. Выключен во всех остальных случаях.
Флаг переноса	CY	Включен во время выполнения ПИД-регулирования. Выключен во всех остальных случаях.

Меры предосторожности

Условие выполнения для команды PID(190) фактически выполняет роль сигнала остановки/запуска. ПИД-регулятор производит вычисления, если условие выполнения остается включенным в следующем цикле после инициализации операндов C+9...C+38. Если в качестве условия выполнения для команды PID(190) используется флаг «Всегда ВКЛ» (который всегда находится в состоянии «1»), в начале работы необходимо запускать отдельную операцию, инициализирующую параметры C+9...C+38.

Если значение какого-либо из операндов (C) выйдет за допустимый диапазон, произойдет ошибка и установится флаг ошибки.

Если фактический период измерения более чем в два раза превзойдет заданный период измерения, произойдет ошибка и установится флаг ошибки. ПИД-регулирование, однако, будет продолжаться.

Во время выполнения ПИД-регулирования флаг переноса находится во включенном состоянии.

Если управляющее воздействие на выходе ПИД-регулятора окажется больше верхнего предельного значения, будет установлен флаг большего значения (флаг «Больше»). В этот период в качестве выходного значения выдается верхнее предельное значение.

Предварительная версия
Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

Если управляющее воздействие на выходе ПИД-регулятора окажется меньше нижнего предельного значения, будет установлен флаг меньшего значения (флаг «Меньше»). В этот период в качестве выходного значения выдается нижнее предельное значение.

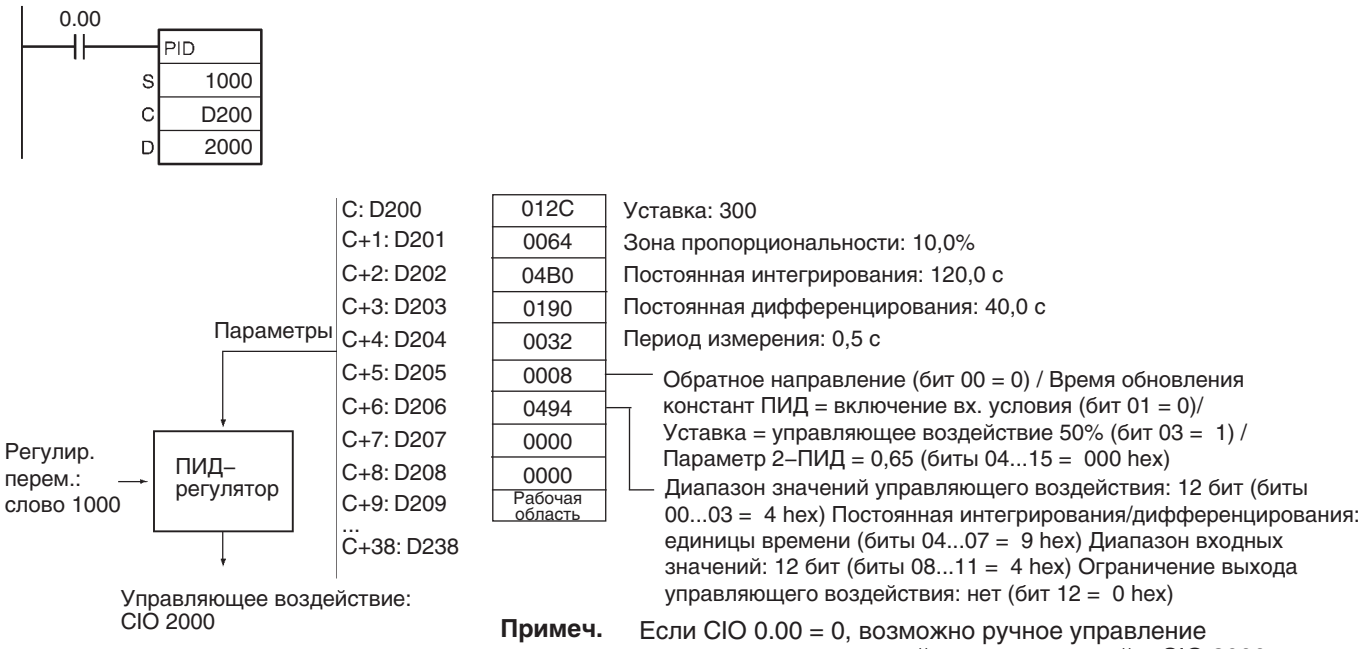
Из всех ПИД-параметров (C...C+38) единственный параметр, значение которого может быть изменено при включенном входном условии, это уставка в слове C. Для изменения любого другого значения входное условие необходимо выключить и вновь включить, чтобы новое значение вступило в силу.

Пример

По положительному фронту бита CIO 0.00 («0» -> «1») рабочая область (слова D209...D238) инициализируется в соответствии с параметрами (показанными ниже), заданными в словах D200...D208. После инициализации рабочей области производятся вычисления ПИД-регулирования и в слово CIO 2000 выдается полученное значение управляющего воздействия.

Далее ПИД-регулирование продолжает выполняться через заданные интервалы времени (равные периоду измерения) в соответствии с параметрами в словах D200...D208, пока остается включенным бит CIO 0.00. Значение управляющего воздействия выдается в CIO 2000.

Если какая-либо из ПИД-констант (зона пропорциональности (P), постоянная интегрирования (Tik) или постоянная дифференцирования (Tdk)) будет изменена после включения бита CIO 0.00, новое значение ПИД-регулятором использоваться не будет.



Примеч. Если CIO 0.00 = 0, возможно ручное управление путем непосредственной записи значений в CIO 2000.

Диапазоны входных и выходных значений

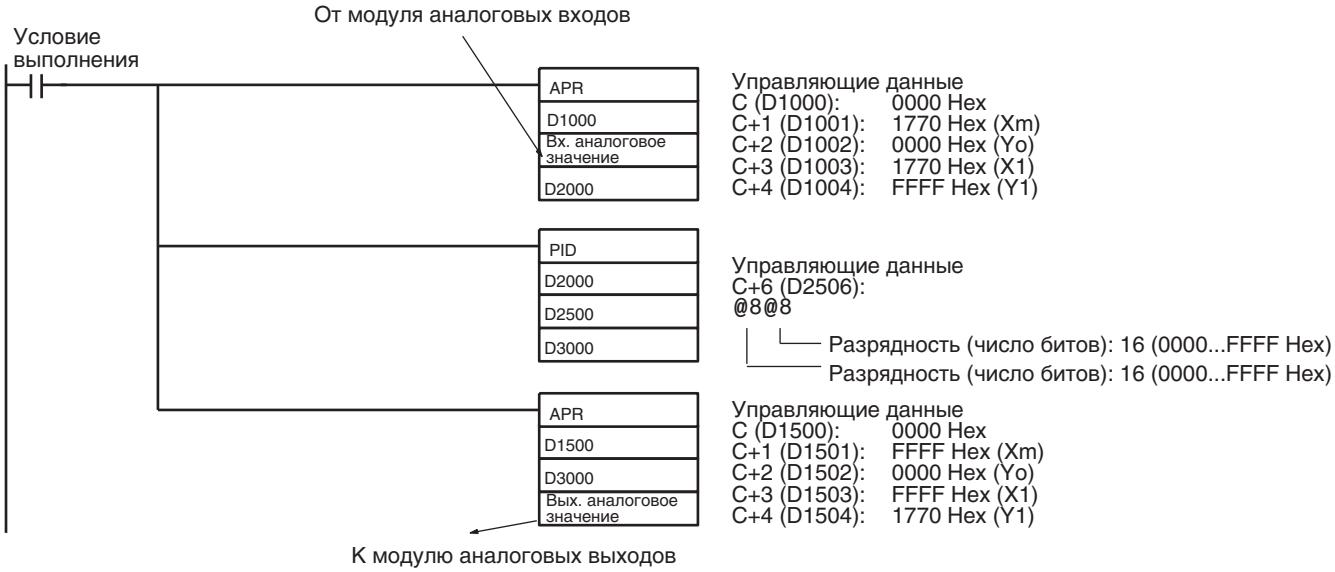
Количество действительных битов в слове входного значения (измеренная регулируемая переменная) определяется битами 08...11 слова C+6, а количество действительных битов в слове выходного значения (управляющее воздействие) определяется битами 00...03 слова C+6. Соответствующие диапазоны входных и выходных значений показаны в следующей таблице.

C+6, биты 08...11 или C+6, биты 00...03	Кол-во действительных битов	Диапазон
0	8	0000...00FF hex
1	9	0000...01FF hex

C+6, биты 08...11 или C+6, биты 00...03	Кол-во действительных битов	Диапазон
2	10	0000...03FF hex
3	11	0000...07FF hex
4	12	0000...0FFF hex
5	13	0000...1FFF hex
6	14	0000...3FFF hex
7	15	0000...7FFF hex
8	16	0000...FFFF hex

Если входной или выходной диапазон команды PID(190) не удается точно согласовать с рабочим диапазоном модуля аналоговых входов или модуля аналоговых выходов просто путем установки количества действительных битов, для более точного согласования диапазонов можно воспользоваться командой APR(069) (МАТЕМАТИЧЕСКАЯ ФУНКЦИЯ).

Ниже в качестве примера показан сегмент программы для модуля аналоговых входов DRT1-AD04 и модуля аналоговых выходов DRT1-DA02, функционирующих в качестве ведомых устройств в сети DeviceNet. Оба этих модуля работают со значениями в диапазоне от 0000 до 1770 hex, которые невозможно задать, просто указав требуемое количество разрядов. Поэтому на входе и выходе команды PID(190) используются команды APR(069). Первая приводит рабочий диапазон значений модуля аналоговых входов (0000...1770 hex) к входному диапазону команды PID(190) (0000...FFFF hex), а вторая — приводит выходной диапазон команды PID(190) к рабочему диапазону значений модуля аналоговых выходов (0000...1770 hex).



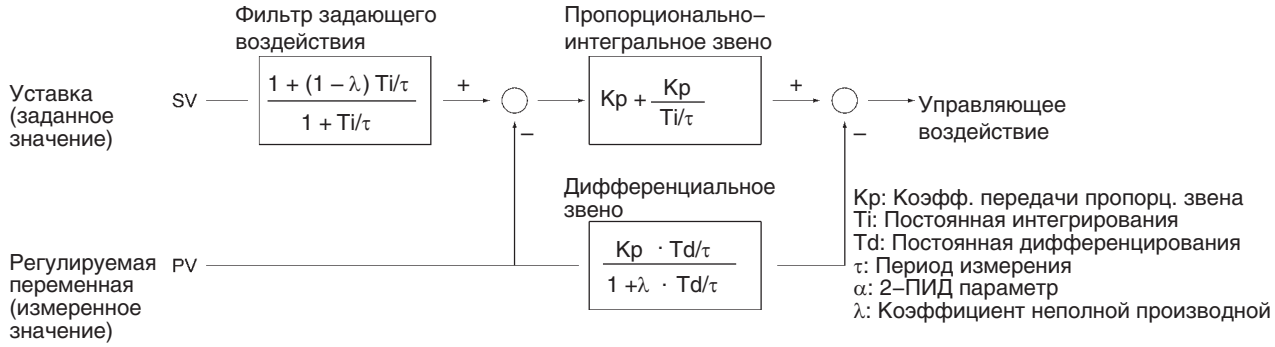
Эксплуатационные характеристики

Параметр		Характеристики
Метод ПИД-регулирования	---	ПИД-регулирование прямого или обратного действия с двумя степенями свободы, с фильтром задающего воздействия
Количество контуров ПИД-регулирования	---	Не ограничено (1 контур на команду)
Период измерения	τ	От 0,01 до 99,99 с

Параметр			Характеристики
ПИД-константы	Зона пропорциональности	P	0,1...999,9%
	Постоянная интегрирования	Tik	1...8191, 9999 (9999 = отключение интегрирования)
	Постоянная дифференцирования	Tdk	0...8191 (0 = отключение дифференцирования)
Уставка		SV	0...65535 (ограничено макс. значением входного диапазона)
Регулируемая переменная (измеренное значение)		PV	0...65535 (ограничено макс. значением входного диапазона)
Управляющее воздействие		MV	0...65535 (ограничено макс. значением выходного диапазона)

Метод вычисления

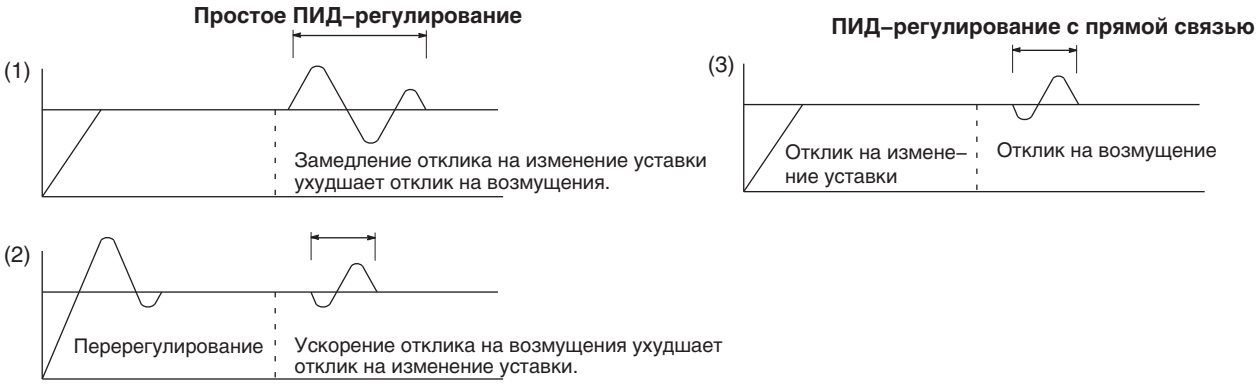
Функциональная схема ПИД-регулятора с фильтром для задающего воздействия (уставки) и двумя степенями свободы показана ниже.



Функциональная схема ПИД-регулятора с фильтром уставки и двумя степенями свободы

При простом ПИД-регулировании меры по предотвращению перерегулирования снижают скорость реакции на возмущения (1). С другой стороны, повышение скорости отработки возмущений приводит к перерегулированию и снижению скорости реакции на изменение уставки (2).

При использовании ПИД-регулятора с фильтром задающего воздействия и двумя степенями свободы отсутствует перерегулирование, возможно одновременное повышение скорости отклика на изменение уставки и скорости отклика на возмущения (3).



Описание и настройка параметров ПИД-регулирования

Управляющие данные	Параметр	Содержание	Диапазон значений	Изменение при включенном входном условии
C	Уставка (SV)	Заданное значение физической величины регулируемого техпроцесса.	Двоичное значение (с тем же числом битов, что указано для входного диапазона)	Разрешено
C+1	Зона пропорциональности	Параметр пропорционального воздействия, определяемый как отношение диапазона пропорционального регулирования к полному диапазону регулирования.	0001...270F hex (1...9999); (0,1%...999,9%, с шагом 0,1%)	Изменение при вкл. вх. условии возможно, если бит 1 слова C+5 = 1.
C+2	Tik Постоянная интегрирования	Константа, определяющая величину интегрального воздействия. С ростом этого параметра величина интегрального воздействия снижается.	0001...1FFF hex (1...8191); (9999 = интегральное регулирование не осуществляется) (см. примеч. 1.)	
C+3	Tdk Постоянная дифференцирования	Константа, определяющая величину воздействия по производной. С ростом этого параметра величина воздействия по производной снижается.	0001...1FFF hex (1...8191); (0000 = регулирование по производной не осуществляется) (см. примеч. 1.)	
C+4	Период измерения (τ)	Определяет периодичность, с которой выполняется команда ПИД-регулирования.	0001...270F hex (1...9999); (0,01...99,99 с, с шагом 10 мс)	Не допускается
Биты 04...15 слова C+5	Параметр 2-ПИД (α)	Коэффициент фильтрации входного значения. Обычно следует использовать значение 0,65 (т. е. значение параметра 000). По мере приближения коэффициента фильтрации к 0 эффективность фильтрации снижается.	000 hex: α = 0,65 Значениям от 100 до 163 hex соответствуют значения коэффициента от α= 0,00 до α= 0,99 (см. прим. 2).	
Бит 03 слова C+5	Настройка уровня управляющего воздействия	Определяет уровень управляющего воздействия при равенстве регулируемой переменной и уставки (PV = SV).	0: выход 0%; 1: выход 50%.	
Бит 01 слова C+5	Разрешение изменения ПИД-констант	Определяет, когда именно измененные значения констант (P, Tik и Tdk) вступают в силу и начинают использоваться для вычислений ПИД-регулятором.	0: в начале выполнения команды ПИД-регулирования; 1: в начале выполнения команды ПИД-регулирования и в каждом периоде измерения.	Разрешено

Управляющие данные	Параметр	Содержание	Диапазон значений	Изменение при включенном входном условии
Бит 00 слова C+5	Выбор прямого/ обратного направления для ПИД-регулятора	Определяет направление пропорционального воздействия.	0: обратное действие; 1: прямое действие.	Не допускается
Бит 12 слова C+6	Ограничение выхода управляющего воздействия	Определяет, должно ли применяться ограничение к выходу управляющего воздействия.	0: выключено (выход не ограничивается); 1: включено (выход ограничивается).	
Биты 08...11 слова C+6	Диапазон входных значений	Количество действительных битов в слове входного значения.	0: 8 битов; 5: 13 битов; 1: 9 битов; 6: 14 битов; 2: 10 битов; 7: 15 битов; 3: 11 битов; 8: 16 битов. 4: 12 битов;	
Биты 04...07 слова C+6	Единицы постоянных интегрирования и дифференцирования	Определяет единицы, в которых выражаются постоянные интегрирования и дифференцирования.	1: множитель к периоду измерения; 9: время (единица: 100 мс).	
Биты 00...03 слова C+6	Диапазон выходных значений	Количество действительных битов в слове выходного значения. (Количество битов выходного значения автоматически устанавливается равным количеству битов входного значения.)	0: 8 битов; 5: 13 битов; 1: 9 битов; 6: 14 битов; 2: 10 битов; 7: 15 битов; 3: 11 битов; 8: 16 битов. 4: 12 битов;	
C +7	Нижний предел выхода управл. воздействия	Нижнее предельное значение управляющего воздействия при включенном ограничении выхода управляющего воздействия.	0000...FFFF (двоичн.) (См. примеч. 3)	
C +8	Верхний предел выхода управл. воздействия	Верхнее предельное значение управляющего воздействия при включенном ограничении выхода управляющего воздействия.	0000...FFFF (двоичн.) (См. примеч. 3)	

Примечание.

(1) Если параметр выбора единиц измерения содержит «1», диапазон значений: от 1 до 8191 периодов измерения. Если параметр выбора единиц измерения содержит «9», диапазон значений: от 0,1 до 819,1 с. При значении «9» постоянные интегрирования и дифференцирования следует задавать так, чтобы они оставались в пределах от 1 до 8191 периодов измерения.

(2) При вводе значения «000» в параметр 2-ПИД (α) действует значение 0,65, являющееся нормой.

(3) Если включено ограничение выхода управляющего воздействия (т. е. выбрано «1»), значения следует задавать с соблюдением следующего правила:

0000 ≤ Нижн. предел выхода MV ≤ Верхн. предел выхода MV ≤ Макс. значение выходного диапазона

Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

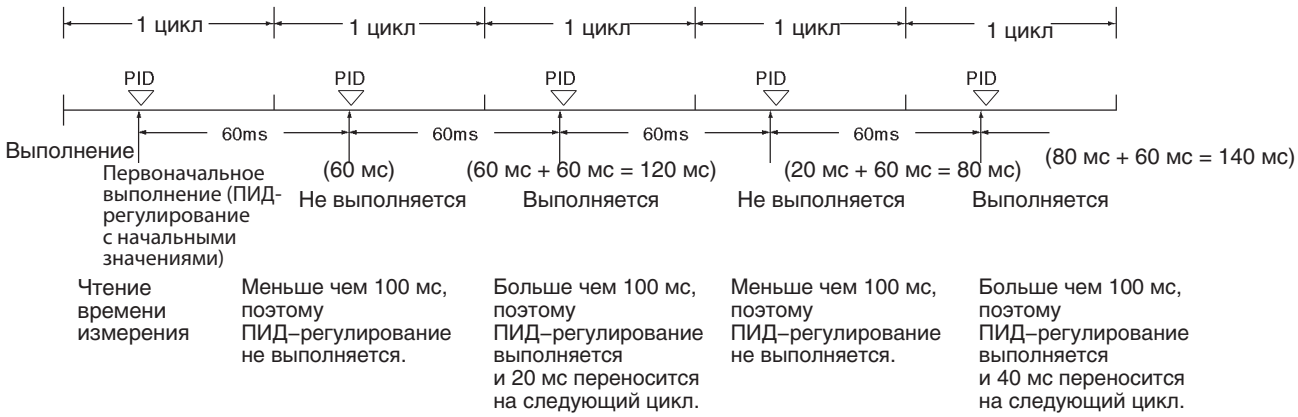
Период измерения и длительность цикла ПЛК

Период измерения может быть задан с точностью до 10 мс (от 0,01 до 99,99 с), однако фактическое время работы ПИД-регулятора определяется не только периодом измерения, но и временем выполнения самой команды PID(190) (в каждом цикле). Период измерения и длительность цикла ПЛК связаны между собой следующим образом.

- Если период измерения меньше длительности цикла, ПИД-регулирование выполняется в каждом цикле, но не в каждом периоде измерения.
- Если период измерения больше или равен длительности цикла, ПИД-регулирование не выполняется в каждом цикле. Команда PID(190) в этом случае выполняется тогда, когда с момента выполнения предыдущей команды PID(190) проходит время, равное периоду измерения или превышающее его. Избыток времени (т. е. разница между суммой длительностей цикла и периодом измерения) переносится на следующий период.

  Допустим, к примеру, что период измерения равен 100 мс, а длительность цикла постоянно имеет значение 60 мс. В первом цикле после начального выполнения команда PID(190) не будет выполнена, так как 60 мс меньше, чем 100 мс. Во втором цикле сумма двух длительностей цикла (60 мс + 60 мс) превысит 100 мс, поэтому команда PID(190) будет выполнена. Избыток в 20 мс (120 мс - 100 мс = 20 мс) будет учтен в следующем периоде.

  В третьем цикле избыток 20 мс добавляется к 60 мс. Поскольку суммарное значение 80 мс меньше, чем 100 мс, команда PID(190) не выполняется. В четвертом цикле 80 мс добавляется к 60 мс. Полученная сумма 140 мс превышает 100 мс, поэтому команда PID(190) выполняется, а избыток в 40 мс (140 мс - 100 мс = 40 мс) переносится на следующий период. Далее эта процедура многократно повторяется в последующих циклах.



Виды управления

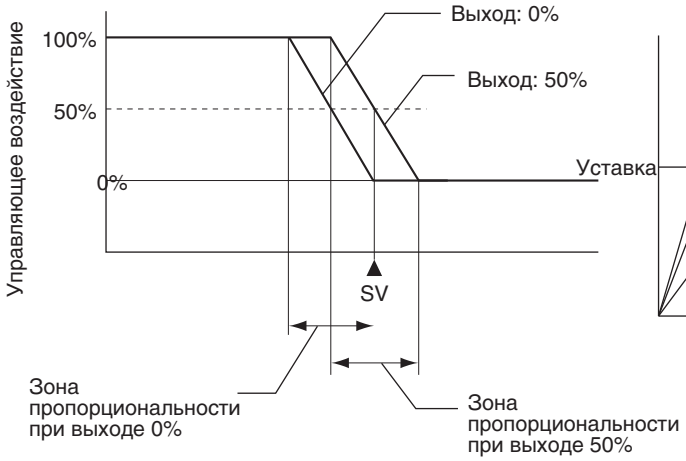
Пропорциональное управляющее воздействие (P)

Пропорциональное воздействие (или воздействие по отклонению) — вид управления, при котором величина управляющего воздействия (MV) изменяется пропорционально величине отклонения в некоторой области значений, определяемой относительно заданного значения (SV) и называемой зоною пропорциональности. На следующем рисунке показан пример обратного пропорционального воздействия.

Если при пропорциональном управлении текущее значение (PV) становится меньше левой границы зоны пропорциональности, величина управляющего воздействия (MV) составляет 100% (т. е. максимальна). В пределах зоны пропорциональности величина MV изменяется пропорционально отклонению (т. е. разнице между текущим значением и уставкой) и плавно снижается, пока текущее значение не становится равно уставке (т. е. отклонение не сводится к 0), после чего величина MV принимает минимальное значение 0% (или 50%, что определяется настройкой соответствующего параметра («уровень управляющего воздействия»)). Величина MV также составляет 0%, когда текущее значение превышает уставку.

Величина зоны пропорциональности выражается в процентах от полного диапазона изменения входных значений. Чем уже зона пропорциональности, тем больше постоянная пропорционального управления (коэффициент передачи пропорционального звена) и тем большее воздействие оказывается на регулируемую переменную. При чисто пропорциональном управлении всегда имеет место некоторое отклонение регулируемой переменной от уставки (статическая ошибка). Это отклонение можно уменьшить, сузив зону пропорциональности. Однако при слишком узкой зоне пропорциональности возникнут автоколебания.

Управляющее воздействие по отклонению (обратное)



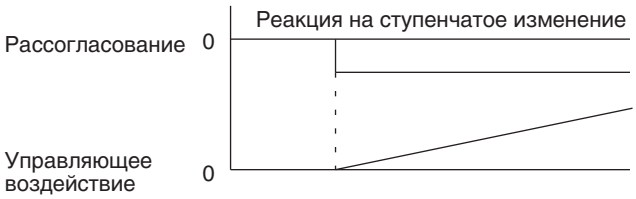
Регулировка зоны пропорциональности



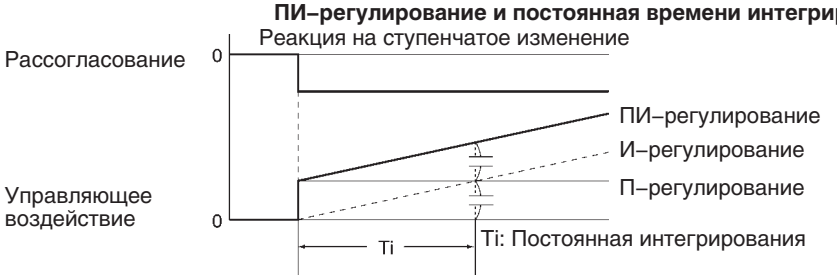
Управляющее воздействие по интегралу (I)

Интегральное воздействие в сочетании с пропорциональным управлением уменьшает величину смещения соразмерно времени, сводя, в конечном счете, отклонение регулируемой переменной от уставки (статическую ошибку) к нулю. Вклад интегрального звена в управление (сила воздействия) зависит от постоянной времени интегрирования, которая определяется как время, за которое интегральное звено способно привести управляющее воздействие к такому же уровню, что и пропорциональное звено при скачкообразном (ступенчатом) отклонении регулируемой переменной от уставки (см. рисунок ниже). Чем меньше время интегрирования, тем сильнее интегральное звено воздействует на регулируемую переменную. Если время интегрирования будет слишком мало, корректирующее воздействие будет избыточным и будет приводить к автоколебаниям.

Управляющее воздействие по интегралу



ПИ–регулирование и постоянная времени интегрирования



Управляющее воздействие по производной (D)

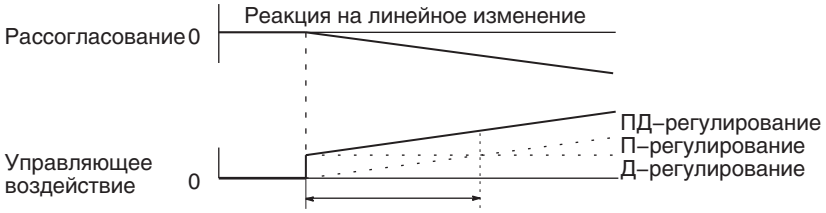
При пропорциональном и интегральном управлении корректирующее воздействие зависит от конечного результата управления, что неизбежно ведет к задержке отклика. Этот недостаток можно устранить, дополнив регулятор дифференцирующим звеном. Дифференцирующее звено формирует высокий уровень управляющего воздействия в ответ на резкие (скачкообразные) отклонения, чтобы быстро вернуть систему к исходному состоянию. При данном виде управления выходной сигнал регулятора (управляющее воздействие) изменяется пропорционально скорости отклонения регулируемой величины от заданного значения (или скорости изменения регулируемой величины).

Вклад дифференциального звена в управление (сила воздействия) зависит от постоянной времени дифференцирования, которая определяется как время, за которое дифференциальное звено способно привести управляющее воздействие к такому же уровню, что и пропорциональное звено при скачкообразном (ступенчатом) отклонении регулируемой переменной от уставки (см. рисунок ниже). Чем больше время дифференцирования, тем сильнее дифференциальное звено воздействует на регулируемую переменную.

Управляющее воздействие по производной



ПД–регулирование и постоянная времени дифференцирования

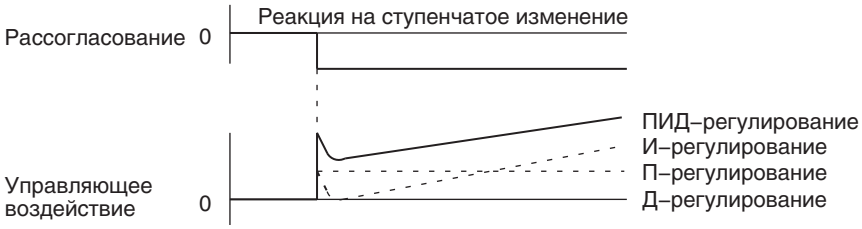


Td: Постоянная дифференцирования

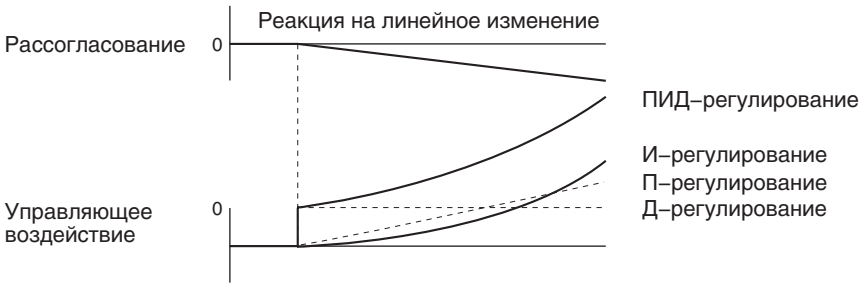
Пропорционально-интегрально-дифференциальное управление

ПИД-регулятор сочетает в себе все три вида управления: пропорциональное (P), интегральное (I), дифференциальное (D). Это обеспечивает высокое качество регулирования даже при управлении инерционными объектами. Пропорциональное воздействие обеспечивает плавное управление без автоколебаний, воздействие по интегралу автоматически устраняет любое отклонение, а воздействие по производной ускоряет отклик на возмущение.

Реакция на ступенчатое изменение



Реакция на линейное изменение

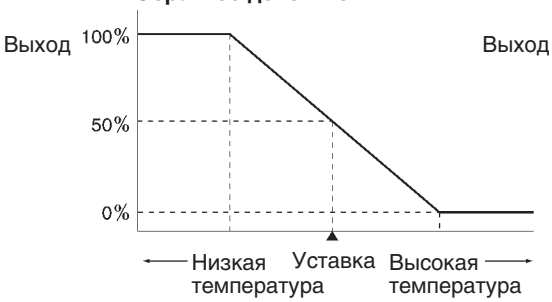


Направление действия

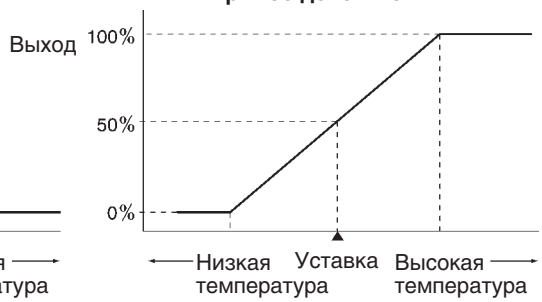
Для ПИД-регулятора можно выбрать одно из двух следующих направлений управляющего воздействия. В любом из этих направлений величина управляющего воздействия растет вместе с ростом отклонения регулируемой переменной от уставки.

- Прямое действие: управляющее воздействие растет, когда регулируемая переменная превышает уставку.
- Обратное действие: управляющее воздействие растет, когда регулируемая переменная меньше уставки.

Обратное действие



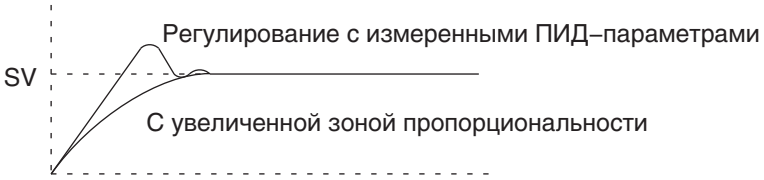
Прямое действие



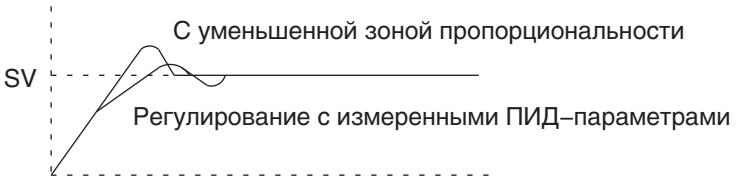
Настройка параметров ПИД-регулятора

Влияние параметров ПИД-регулятора на качество управления отражено на рисунках ниже.

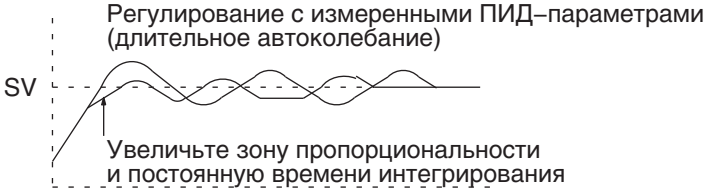
- Если наличие некоторой задержки в стабилизации процесса (времени установления) не является критичным, тогда как перерегулирование не допустимо, зону пропорциональности следует расширить.



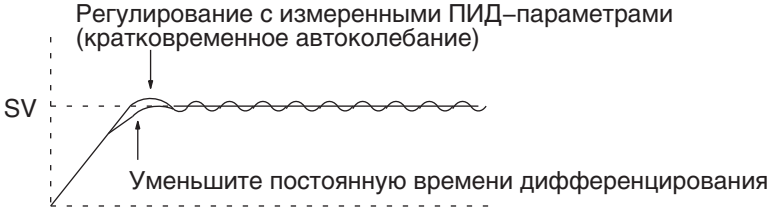
- Если требуется быстро стабилизировать регулируемую переменную, а перерегулирование допустимо, зону пропорциональности следует сузить. Следует, однако, помнить, что при очень узкой зоне пропорциональности могут возникать автоколебания.



- Если длительность и амплитуда автоколебаний чересчур высоки либо пере-/недорегулирование значительно ухудшает качество работы системы, причина, возможно, кроется в избыточном интегральном воздействии. Величину автоколебаний можно уменьшить путем повышения постоянной времени интегрирования или расширения зоны пропорциональности.



- Если период ПИД-регулирования мал и имеют место автоколебания, возможной причиной может быть слишком высокая скорость реакции системы управления и избыточное воздействие дифференциального звена. В этом случае следует снизить величину воздействия по производной.

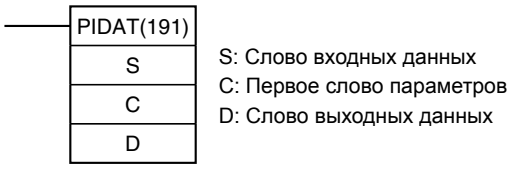


3-17-2 ПИД-РЕГУЛЯТОР С АВТОНАСТРОЙКОЙ: PIDAT(191)

Назначение

Осуществление ПИД-регулирования в соответствии с заданными параметрами. Константы ПИД-регулятора могут настраиваться автоматически.

Символ РКС



Варианты выполнения

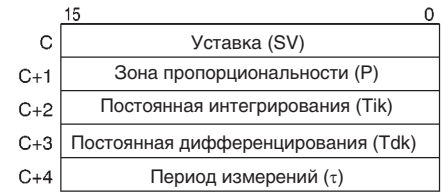
Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	PIDAT(191)
	Однократное выполнение по положительному фронту	Не предусмотрено
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

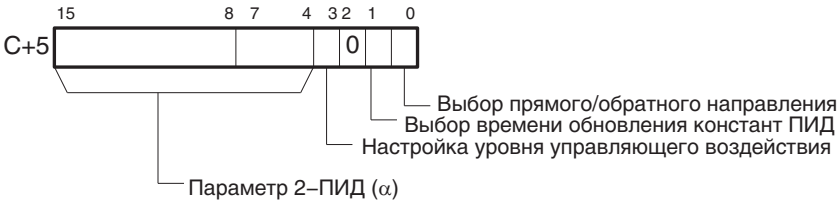
Доступные области программы

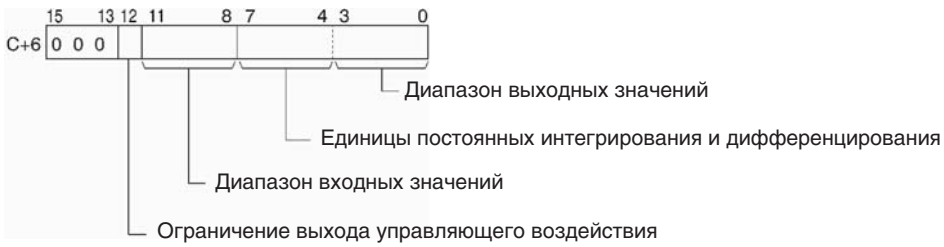
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
Не допускается	OK	OK	Не допускается

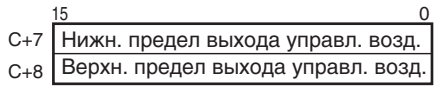
Параметры

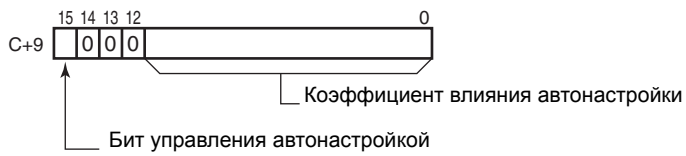
Содержание слов параметров показано на следующем рисунке. Подробную информацию о параметрах см. в пункте *Описание и настройка параметров ПИД-регулирования* в данном разделе.

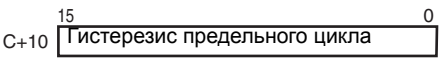


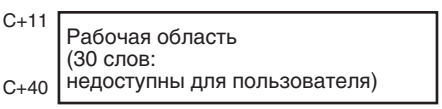












Характеристики операндов

Область	S	C	D
Область CIO	CIO 0...CIO 6143	CIO 0...CIO 6103	CIO 0...CIO 6143
Рабочая область	W0...W511	W0...W471	W0...W511
Область битов хранения	H0...H511	H0...H471	H0...H511
Область вспомогательных битов	A0...A959	A0...A919	A448...A959
Область таймеров	T0000...T4095	T0000...T4055	T0000...T4095
Область счетчиков	C0000...C4095	C0000...C4055	C0000...C4095
Область DM	D0...D32767	D0...D32727	D0...D32767
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767		
Постоянные	DR0...DR15	---	DR0...DR15
Регистры данных	---		
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15		

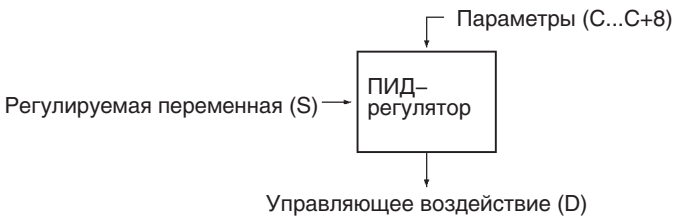
Описание

При включенном условии выполнения команда PIDAT(191) осуществляет ПИД-регулирование (т. е. непрерывное приведение регулируемой переменной к заданному значению с применением пропорционального, интегрального и дифференциального звеньев) с фильтрацией задающего воздействия и двумя степенями свободы в соответствии с параметрами, заданными в C (уставка, ПИД-константы и др.). Она принимает двоичное содержимое входного слова S, ограниченное указанным диапазоном (количеством действительных входных битов), и выполняет ПИД-управление в соответствии с заданными параметрами. Результат обработки (т. е. управляющее воздействие (MV)) выдается в слово D.

Заданные параметры считываются, когда условие выполнения переключается из состояния «0» в состояние «1». Если значение какого-либо параметра выходит за допустимый диапазон, устанавливается флаг ошибки.

Если значения параметров находятся в пределах допустимых диапазонов, в математических расчетах ПИД-регулятора используются исходные значения. Безударное управление в этот раз не производится. Оно будет применяться для вычисления величин управляющих воздействий в последующих циклах ПИД-регулирования. (Под безударным управлением понимается плавное и непрерывное изменение управляющего воздействия, исключающее нежелательные и потенциально опасные резкие изменения в состоянии процесса.)

После включения условия выполнения в ПИД-регулятор вводится значение регулируемой переменной (PV) для указанного периода измерения и производятся необходимые вычисления.



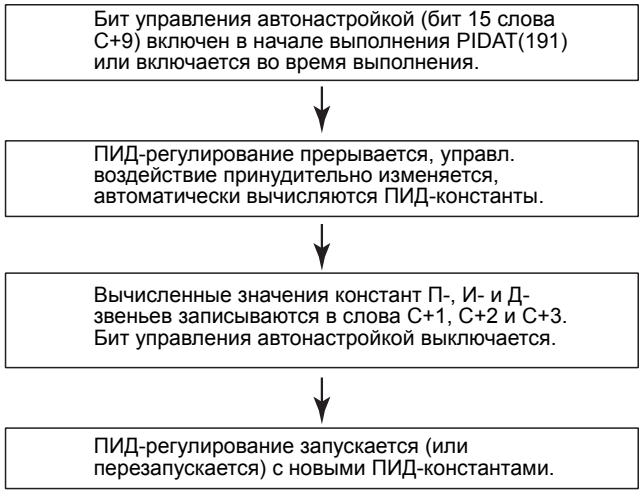
Автонастройка

В каждом цикле проверяется состояние бита управления автонастройкой (бит 15 слова C+9). Если в некотором цикле данный управляющий бит включен, команда PIDAT(191) начинает процедуру автонастройки констант ПИД-регулятора. (Во время выполнения автонастройки изменение уставки ПИД-регулятором не воспринимается.)

Для автонастройки используется метод предельного цикла. Команда PIDAT(191) принудительно изменяет уровень управляющего воздействия (макс. управл. воздействия↔ мин. управл. воздействия) и одновременно контролирует характеристики управляемой системы. По результатам наблюдения за характеристиками системы вычисляются новые значения констант ПИД-регулятора (П-, И- и Д-звеньев), которые автоматически сохраняются в слова C+1, C+2 и C+3. Сразу после этого выключается бит управления автонастройкой (бит 15 слова C+9), и ПИД-регулирование возобновляется с новыми значениями ПИД-констант, записанными в C+1, C+2 и C+3.

- Если бит управления автонастройкой включен при самом первом выполнении команды PIDAT(191), сначала выполняется автонастройка, и ПИД-регулирование запускается с новыми значениями.
- Если бит управления автонастройкой включается во время выполнения команды PIDAT(191), последняя прерывает ПИД-регулирование, выполняемое с использованием ПИД-констант пользователя, производит автонастройку, после чего возобновляет ПИД-регулирование с вычисленными значениями ПИД-констант.

Порядок выполнения автонастройки представлен на следующей блок-схеме.



Примечание.

(1) Если бит управления автонастройкой будет выключен во время процедуры автонастройки, автонастройка будет прервана и ПИД-регулирование будет запущено с использованием ПИД-констант, действовавших до запуска автонастройки.

(2) При возникновении ошибки автонастройки ПИД-регулирование также будет запущено с использованием ПИД-констант, действовавших до запуска автонастройки.

В обеих ситуациях, описанных в примечаниях 1 и 2, ПИД-константы могут использоваться, только если они были вычислены ранее (т. е. до прерывания автонастройки).

ПИД-регулирование

Количество действительных битов в 16-битовом слове входного значения S (регулируемая переменная) определяется битами 08...11 слова C+6 (настройка входного диапазона). Например, если входной диапазон ограничен 12 битами (4 hex), ПИД-регулятор работает со значениями регулируемой переменной в диапазоне от 0000 hex до 0FFF hex. (Любые значения свыше 0FFF hex воспринимаются как 0FFF hex.)

От выбранного входного диапазона также зависит диапазон значений уставки (SV) ПИД-регулятора.

Регулируемая переменная (измеряемый параметр технологического процесса, PV) и уставка (заданное значение параметра технологического процесса, SV) представляются в двоичном коде без знака и могут принимать значения в диапазоне от 0000 hex до максимального значения выбранного входного диапазона.

Количество действительных битов в 16-битовом слове выходного значения D (управляющее воздействие) определяется битами 00...03 слова C+6 (настройка выходного диапазона). Например, если выходной диапазон ограничен 12 битами (4 hex), ПИД-регулятор выдает значения управляющего воздействия в диапазоне от 0000 hex до 0FFF hex.

В случае только пропорционального управления выходное значение ПИД-регулятора (MV) в установившемся состоянии (PV = SV) определяется следующим образом:

0: выход 0%;
1: выход 50%.

Может быть указано прямое или обратное направление пропорционального воздействия.

Для выхода управляющего воздействия могут быть указаны верхнее и нижнее предельные значения.

Период измерения может быть задан с точностью до 10 мс (от 0,01 до 99,99 с), однако фактическое время работы ПИД-регулятора определяется не только периодом измерения, но и временем выполнения самой команды PIDAT(191) (в каждом цикле).

В настройках можно выбрать, когда именно должны вступать в силу новые значения измененных ПИД-констант: 1) в начале выполнения команды PIDAT(191) или 2) в начале выполнения команды PIDAT(191) и в каждом периоде измерения. Изменение в каждом измерительном цикле (т. е. во время выполнения команды PID) возможно только для зоны пропорциональности (P), постоянной интегрирования (Tik) и постоянной дифференцирования (Tdk). Момент вступления изменения в силу задается битом 1 слова C+5.

Если значения констант ПИД-регулятора задаются пользователем (вручную), параметр разрешения изменения ПИД-констант (бит 1 слова C+5) следует задать равным «1», чтобы значения слов C+1, C+2 и C+3 обновлялись в каждом периоде измерения. Такая настройка также позволяет корректировать вычисленные ПИД-константы вручную после автонастройки.

Из всех параметров ПИД-регулятора (C...C+40) только указанные ниже параметры могут быть изменены при включенном условии выполнения. При внесении изменений в любые другие параметры необходимо выключить и вновь включить условие выполнения команды, чтобы новые значения вступили в силу.

- Уставка (SV) в слове C.
  (Изменение возможно только во время ПИД-регулирования. Изменение уставки во время автонастройки не воспринимается.)

Предварительная версия
Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

- Параметр разрешения изменения ПИД-констант (бит 1 слова C+5).
- Константы П-, И- и Д- в словах C+1, C+2 и C+3.
  (Изменения в этих константах будут отражаться в каждом периоде измерения, только если параметр разрешения изменения ПИД-констант (бит 1 слова C+5) задан равным «1».)
- Бит управления автонастройкой (бит 15 слова C+9).
- Коэффициент влияния автонастройки (биты 0...14 слова C+9) и гистерезис предельного цикла (C+10) (эти значения считываются при запуске автонастройки).

Примечание. Команда PIDAT(191) отличается от команды PID(190) наличием функции автонастройки (AT). Во всем остальном эти команды полностью идентичны. Подробное описание операции ПИД-регулирования и примеры применения см. в разделе *3-17-1 ПИД-РЕГУЛЯТОР: PID(190)*.

Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	Включен, если значение C выходит за допустимый диапазон. Включен, если фактический период измерения более чем в два раза превосходит заданный период измерения. Включен, если во время автонастройки произошла ошибка. Выключен во всех остальных случаях.
Флаг большего значения («Больше»)	>	Включен, если управляющее воздействие на выходе ПИД-регулятора выходит за верхнее предельное значение. Выключен во всех остальных случаях.
Флаг меньшего значения («Меньше»)	<	Включен, если управляющее воздействие на выходе ПИД-регулятора находится ниже нижнего предельного значения. Выключен во всех остальных случаях.
Флаг переноса	CY	Включен во время выполнения ПИД-регулирования. Выключен во всех остальных случаях.

Меры предосторожности

Условие выполнения для команды PIDAT(191) фактически выполняет роль сигнала остановки/запуска. ПИД-регулятор производит вычисления, если условие выполнения остается включенным в следующем цикле после инициализации операндов C+11...C+40. Если в качестве условия выполнения для команды PIDAT(191) используется флаг «Всегда ВКЛ» (который всегда находится в состоянии «1»), в начале работы необходимо запускать отдельную операцию, инициализирующую параметры C+11...C+40.

Если значение какого-либо из операндов (C) выйдет за допустимый диапазон, произойдет ошибка и установится флаг ошибки.

Если во время автонастройки произойдет ошибка, будет установлен флаг ошибки.

Если фактический период измерения более чем в два раза превзойдет заданный период измерения, произойдет ошибка и установится флаг ошибки. ПИД-регулирование, однако, будет продолжаться.

Во время выполнения ПИД-регулирования флаг переноса находится во включенном состоянии.

Если управляющее воздействие на выходе ПИД-регулятора окажется больше верхнего предельного значения, будет установлен флаг большего значения (флаг «Больше»). В этот период в качестве выходного значения выдается верхнее предельное значение.

Если управляющее воздействие на выходе ПИД-регулятора окажется меньше нижнего предельного значения, будет установлен флаг меньшего значения (флаг «Меньше»). В этот период в качестве выходного значения выдается нижнее предельное значение.

Описание и настройка параметров ПИД-регулирования

Управляющие данные	Параметр	Содержание	Диапазон значений	Изменение при включенном входном условии
C	Уставка (SV)	Заданное значение физической величины регулируемого техпроцесса.	Двоичное значение (с тем же числом битов, что указано для входного диапазона)	Разрешено
C+1	Зона пропорциональности	Параметр пропорционального воздействия, определяемый как отношение диапазона пропорционального регулирования к полному диапазону регулирования.	0001...270F hex (1...9999); (0,1%...999,9%, с шагом 0,1%)	Изменение при вкл. вх. условии возможно, если бит 1 слова C+5 = 1.
C+2	Tik Постоянная интегрирования	Константа, определяющая величину интегрального воздействия. С ростом этого параметра величина интегрального воздействия снижается.	0001...1FFF hex (1...8191); (9999 = интегральное регулирование не осуществляется) (см. примеч. 1.)	
C+3	Tdk Постоянная дифференцирования	Константа, определяющая величину воздействия по производной. С ростом этого параметра величина воздействия по производной снижается.	0001...1FFF hex (1...8191); (0000 = регулирование по производной не осуществляется) (см. примеч. 1.)	
C+4	Период измерения (τ)	Определяет периодичность, с которой выполняется команда ПИД-регулирования.	0001...270F hex (1...9999); (0,01...99,99 с, с шагом 10 мс)	Не допускается
Биты 04...15 слова C+5	Параметр 2-ПИД (α)	Коэффициент фильтрации входного значения. Обычно следует использовать значение 0,65 (т. е. значение параметра 000). По мере приближения коэффициента фильтрации к 0 эффективность фильтрации снижается.	000 hex: α = 0,65 Значениям от 100 до 163 hex соответствуют значения коэффициента от α= 0,00 до α= 0,99 (см. прим. 2).	
Бит 03 слова C+5	Настройка уровня управляющего воздействия	Определяет уровень управляющего воздействия при равенстве регулируемой переменной и уставки (PV = SV).	0: выход 0%; 1: выход 50%.	
Бит 01 слова C+5	Разрешение изменения ПИД-констант	Определяет, когда именно измененные значения констант (P, Tik и Tdk) вступают в силу и начинают использоваться для вычислений ПИД-регулятором.	0: в начале выполнения команды ПИД-регулирования; 1: в начале выполнения команды ПИД-регулирования и в каждом цикле регулирования.	Разрешено

Управляющие данные	Параметр	Содержание	Диапазон значений	Изменение при включенном входном условии
Бит 00 слова C+5	Выбор прямого/ обратного направления для ПИД-регулятора	Определяет направление пропорционального воздействия.	0: обратное действие; 1: прямое действие.	Не допускается
Бит 12 слова C+6	Ограничение выхода управляющего воздействия	Определяет, должно ли применяться ограничение к выходу управляющего воздействия.	0: выключено (выход не ограничивается); 1: включено (выход ограничивается).	
Биты 08...11 слова C+6	Диапазон входных значений	Количество действительных битов в слове входного значения.	0: 8 битов; 5: 13 битов; 1: 9 битов; 6: 14 битов; 2: 10 битов; 7: 15 битов; 3: 11 битов; 8: 16 битов. 4: 12 битов;	
Биты 04...07 слова C+6	Единицы постоянных интегрирования и дифференцирования	Определяет единицы, в которых выражаются постоянные интегрирования и дифференцирования.	1: множитель к периоду измерения; 9: время (единица: 100 мс).	
Биты 00...03 слова C+6	Диапазон выходных значений	Количество действительных битов в слове выходного значения. (Количество битов выходного значения автоматически устанавливается равным количеству битов входного значения.)	0: 8 битов; 5: 13 битов; 1: 9 битов; 6: 14 битов; 2: 10 битов; 7: 15 битов; 3: 11 битов; 8: 16 битов. 4: 12 битов;	
C+7	Нижний предел выхода управл. воздействия	Нижнее предельное значение управляющего воздействия при включенном ограничении выхода управляющего воздействия.	0000...FFFF (двоичн.) (См. примеч. 3)	
C+8	Верхний предел выхода управл. воздействия	Верхнее предельное значение управляющего воздействия при включенном ограничении выхода управляющего воздействия.	0000...FFFF (двоичн.) (См. примеч. 3)	
Бит 15 слова C+9	Бит управления автонастройкой	Этот бит служит для запуска автонастройки. • Для запуска автонастройки бит управления автонастройкой следует переключить в состояние «1». (Автонастройку также можно запустить во время выполнения команды PIDAT(191).) • По завершении автонастройки этот бит автоматически выключается. Если бит управления автонастройкой будет выключен во время автонастройки (вручную), автонастройка будет прервана. В этом случае будут действовать ПИД-константы, которые были вычислены ранее (до запуска автонастройки).	В качестве бита управления: • 0 → 1: Выполняется автонастройка. • 1 → 0: Автонастройка прерывается. (Команда PID(191) выключает бит автоматически по завершении автонастройки.) Применение в качестве флага: 0: Автонастройка не выполняется. 1: В данный момент выполняется автонастройка.	Разрешено

Управляющие данные	Параметр	Содержание	Диапазон значений	Изменение при включенном входном условии
Биты 00...11 слова C+9	Коэффициент влияния автонастройки	Данный параметр позволяет отрегулировать соотношение между вычисленными значениями ПИД-констант и фактически сохраняемыми значениями. В большинстве случаев следует оставить принимаемые по умолчанию значения (0000). • Увеличьте это значение, чтобы повысить стабильность. • Уменьшите это значение, чтобы повысить скорость отклика.	0000 hex: 1,00 (по умолчанию); 0001...03E8 hex (1...1000); (0,01...10,00, с шагом 0,01).	Разрешено (Значения данных параметров считываются при запуске автонастройки.)
C+10	Гистерезис предельного цикла	Задает величину гистерезиса для предельного цикла. По умолчанию при обратном направлении управления управл. воздействие включается с гистерезисом SV−20%. Данный параметр рекомендуется повысить, если из-за нестабильности регулируемой переменной не удается сформировать надлежащий предельный цикл. В то же время, при избыточной величине гистерезиса предельного цикла повышается погрешность автонастройки.	0000 hex: 0,20% (по умолч); 0001...03E8 hex: 0,01...10,00% с шагом 0,01%; FFFF hex: 0,00%. **Примечание.** Значение в процентах от входного диапазона.	

Примечание.

(1) Если параметр выбора единиц измерения содержит «1», диапазон значений: от 1 до 8191 периодов измерения. Если параметр выбора единиц измерения содержит «9», диапазон значений: от 0,1 до 819,1 с. При значении «9» постоянные интегрирования и дифференцирования следует задавать так, чтобы они оставались в пределах от 1 до 8191 периодов измерения.

(2) При вводе значения «000» в параметр 2-ПИД (α) действует значение 0,65, являющееся нормой.

Если включено ограничение выхода управляющего воздействия (т. е. выбрано «1»), значения следует задавать с соблюдением следующего правила:

0000 ≤ Нижн. предел выхода MV ≤ Верхн. предел выхода MV ≤ Макс. значение выходного диапазона

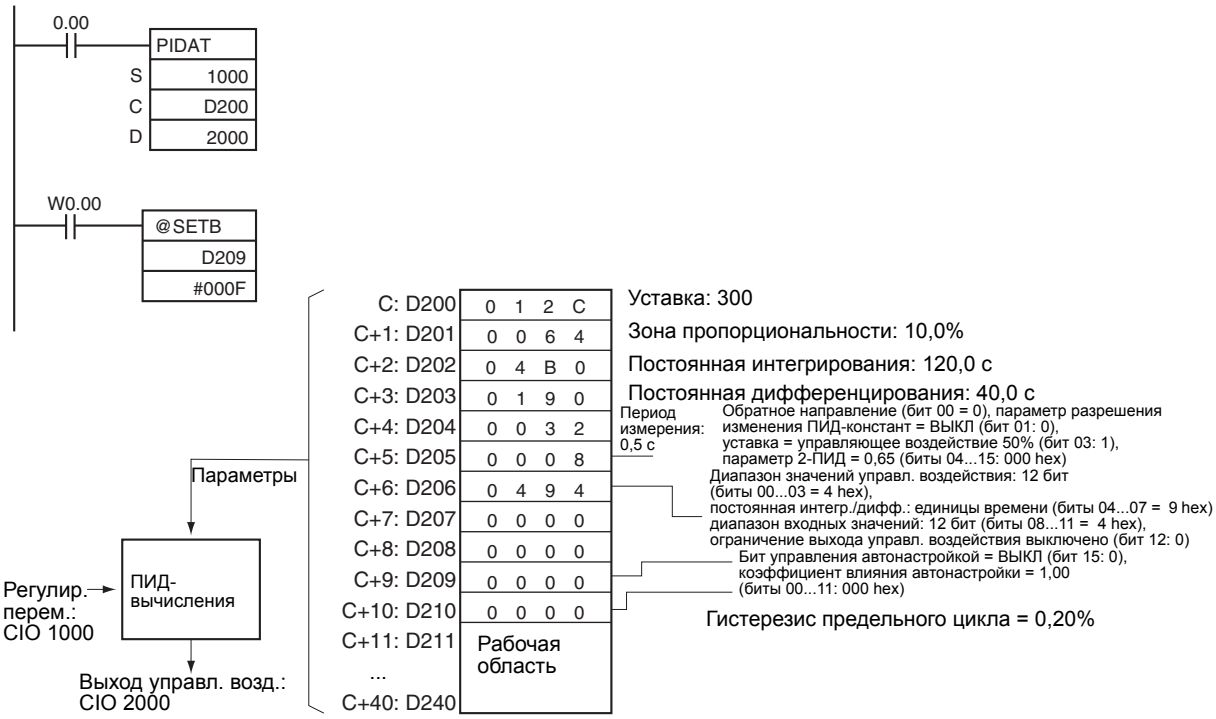
Пример 1. Прерывание ПИД-регулирования для выполнения автонастройки

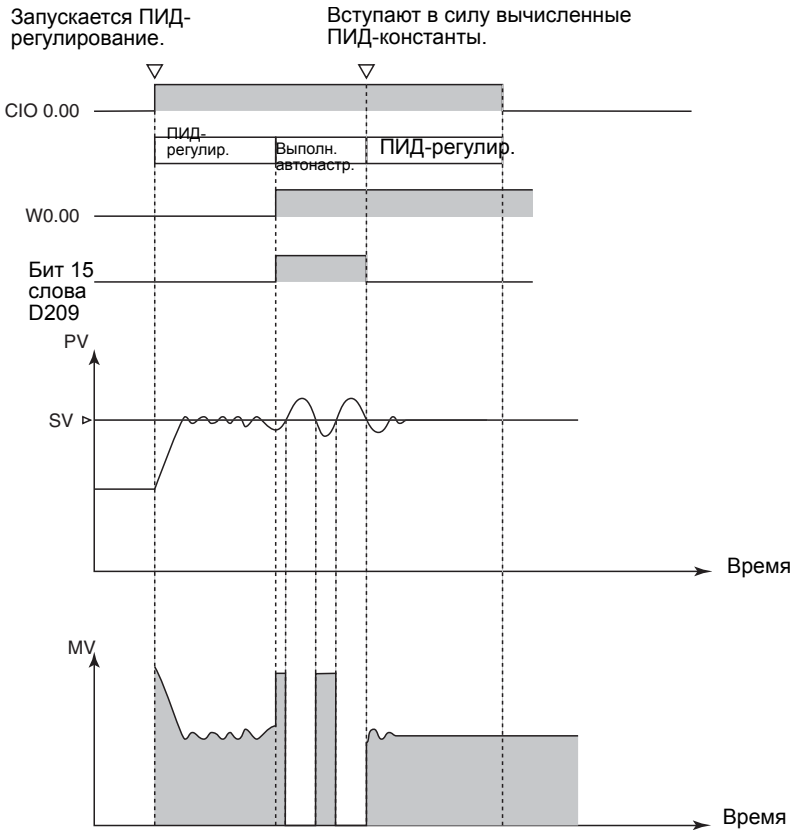
По положительному фронту бита CIO 0.00 («0» -> «1») рабочая область (слова D211...D240) инициализируется в соответствии с параметрами (показанными ниже), заданными в словах D200...D208. После инициализации рабочей области производятся вычисления ПИД-регулирования и в слово CIO 2000 выдается полученное значение управляющего воздействия.

Далее ПИД-регулирование продолжает выполняться через заданные интервалы времени (равные периоду измерения) в соответствии с параметрами в словах D200...D210, пока остается включенным бит CIO 0.00. Значение управляющего воздействия выдается в CIO 2000.

Если какая-либо из ПИД-констант (зона пропорциональности (P), постоянная интегрирования (Tik) или постоянная дифференцирования (Tdk)) будет изменена после включения бита CIO 0.00, новое значение ПИД-регулятором использоваться не будет.

По положительному фронту бита W 0.00 («0» -> «1») команда SETB(532) включает бит 15 слова D209 (C+9) и запускает автонастройку. По завершении автонастройки в слова C+1, C+2 и C+3 записываются вычисленные значения констант П-, И- и Д-звеньев. ПИД-регулирование возобновляется с использованием новых значений ПИД-констант.

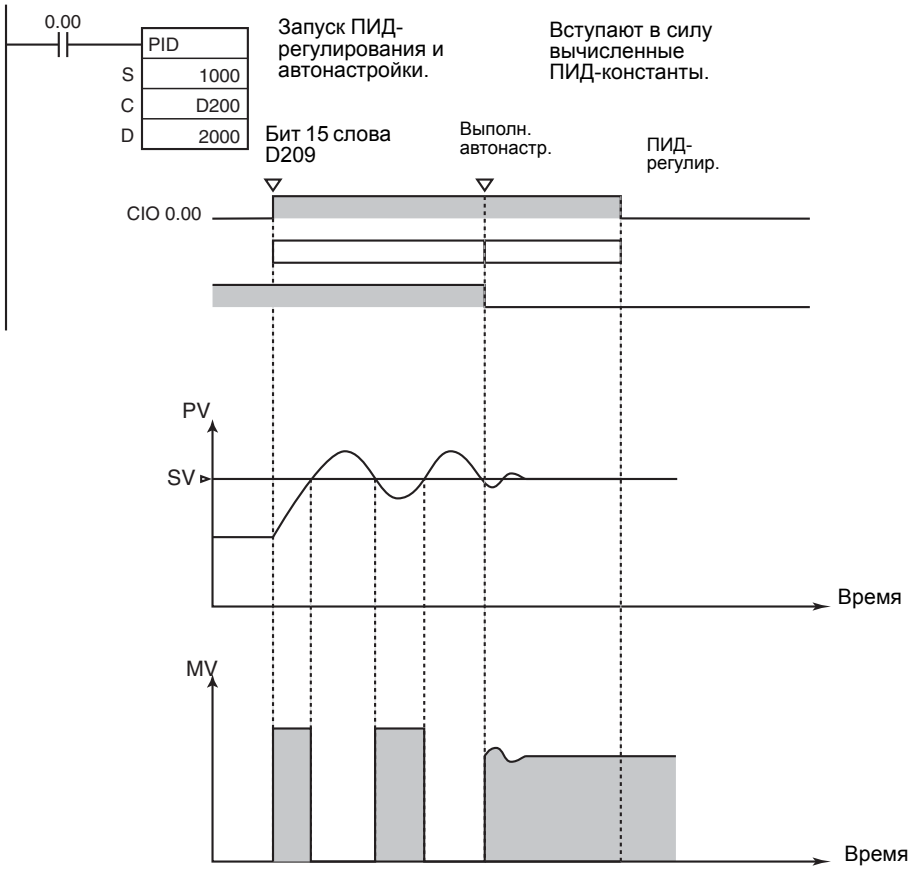




Предварительная версия
Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

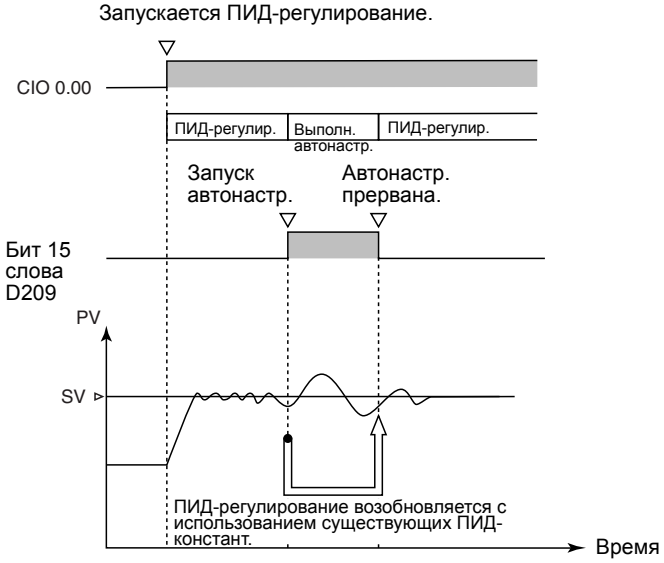
Пример 2.
Запуск PIDAT(191) с автонастройкой

По положительному фронту бита CIO 0.00 («0» -> «1») сначала выполняется автонастройка (если включен бит 15 слова D209 (C+9)). Когда автонастройка завершается, вычисленные значения П-, И- и Д-констант записываются в слова C+1, C+2 и C+3. ПИД-регулирование запускается с использованием вычисленных значений ПИД-констант.



Пример 3.
Досрочное прерывание автонастройки

Выполнение автонастройки можно прервать досрочно путем выключения бита 15 слова D209 (C+9). ПИД-регулирование будет возобновлено (или запущено) с использованием П-, И- и Д-констант, которые действовали до запуска автонастройки.

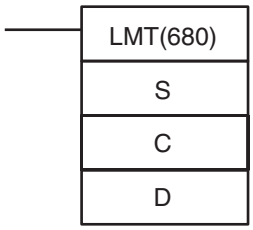


3-17-3 ОГРАНИЧЕНИЕ: LMT(680)

Назначение

Определение принадлежности входного значения диапазону, заданному верхним и нижним предельными значениями, и выдача соответствующего выходного значения.

Символ РКС



S: Слово входных данных

C: Первое слово предельных значений

D: Слово выходных данных

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	LMT(680)
	Однократное выполнение по положительному фронту	@LMT(680)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	S	C	D
Область CIO	CIO 0...CIO 6143	CIO 0...CIO 6142	CIO 0...CIO 6143
Рабочая область	W0...W511	W0...W510	W0...W511
Область битов хранения	H0...H511	H0...H510	H0...H511
Область вспомогательных битов	A0...959	A0...A958	A448...A959
Область таймеров	T0000...T4095	T0000...T4094	T0000...T4095
Область счетчиков	C0000...C4095	C0000...C4094	C0000...C4095
Область DM	D0...D32767	D0...D32766	D0...D32767
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767		
Постоянные	#0000...#FFFF (двоичн.)	---	
Регистры данных	DR0...DR15	---	DR0...DR15
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

При включенном условии выполнения команда LMT(680) ограничивает указанное входное значение (16-битовое двоичное значение со знаком) в пределах заданных верхней и нижней границ. Слова C и C+1 имеют следующее предназначение:

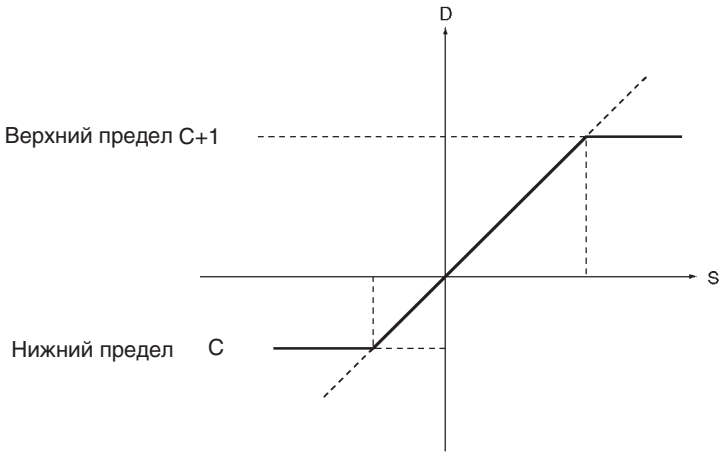
C	Нижнее предельное значение (минимальное выходное значение)
C+1	Верхнее предельное значение (максимальное выходное значение)

Слова C и C+1 должны принадлежать одной области памяти.

Если входное значение (S) меньше нижней границы диапазона (C), в слово D выдается нижнее предельное значение и устанавливается флаг меньшего значения.

Если входное значение (S) больше верхней границы диапазона (C+1), в слово D выдается верхнее предельное значение и устанавливается флаг большего значения.

Если входное значение (S) больше или равно нижней границе (C) или меньше или равно верхней границе (C+1), в слово D выдается входное значение (S).



Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	Включен, если верхнее предельное значение меньше нижнего предельного значения. Выключен во всех остальных случаях.
Флаг большего значения («Больше»)	>	Включен, если входное значение (S) больше верхнего предельного значения. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если результат = 0. Выключен во всех остальных случаях.
Флаг меньшего значения («Меньше»)	<	Включен, если входное значение (S) меньше нижнего предельного значения. Выключен во всех остальных случаях.
Флаг отрицательного значения	N	Включен, если старший бит результата = 1. Выключен во всех остальных случаях.

Меры предосторожности

Если верхнее предельное значение будет меньше нижнего предельного значения, произойдет ошибка и установится флаг ошибки.

Если входное значение (S) окажется больше верхнего предельного значения, будет установлен флаг большего значения (флаг «Больше»).

Если в D будет выдано значение 0000 hex, установится флаг равенства.

Если входное значение (S) окажется меньше нижнего предельного значения, будет установлен флаг меньшего значения (флаг «Меньше»).

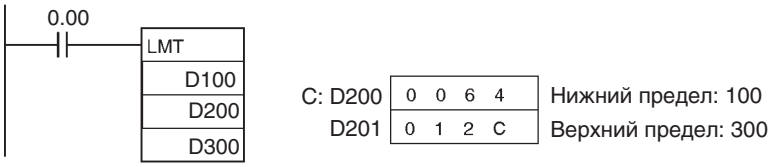
Если самый старший бит слова D будет содержать «1», будет установлен флаг отрицательного значения.

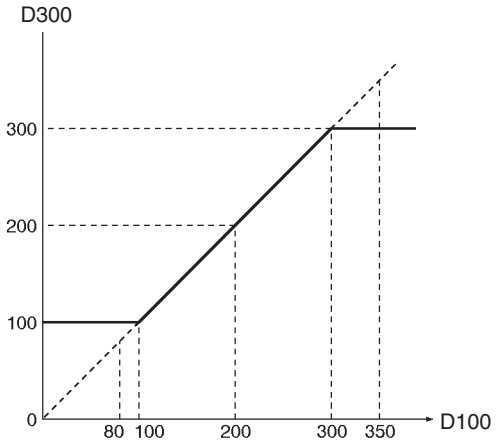
Пример

Если D100 = 0050 hex (80), в слово D300 будет выдано значение 0064 hex (100), так как 80 меньше, чем нижнее предельное значение 100.

Если D100 = 00C8 hex (200), в слово D300 будет выдано значение 00C8 hex (200), так как 200 находится в пределах верхней и нижней границ.

Если D100 = 015E hex (350), в слово D300 будет выдано значение 012C hex (300), так как 350 больше, чем верхнее предельное значение 300.



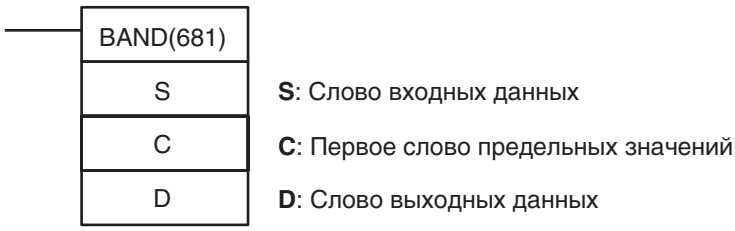


3-17-4 ЗОНА НЕЧУВСТВИТЕЛЬНОСТИ: BAND(681)

Назначение

Определение принадлежности входного значения диапазону (зоне нечувствительности), заданному верхним и нижним предельными значениями и выдача соответствующего выходного значения.

Символ РКС



S: Слово входных данных

C: Первое слово предельных значений

D: Слово выходных данных

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	BAND(681)
	Однократное выполнение по положительному фронту	@BAND(681)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Характеристики операндов

Область	S	C	D
Область CIO	CIO 0...CIO 6143	CIO 0...CIO 6142	CIO 0...CIO 6143
Рабочая область	W0...W511	W0...W510	W0...W511
Область битов хранения	H0...H511	H0...H510	H0...H511
Область вспомогательных битов	A0...A959	A0...A958	A448...A959
Область таймеров	T0000...T4095	T0000...T4094	T0000...T4095
Область счетчиков	C0000...C4095	C0000...C4094	C0000...C4095
Область DM	D0...D32767	D0...D32766	D0...D32767
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767		
Постоянные	#0000...#FFFF (двоичн.)	---	
Регистры данных	DR0...DR15	---	DR0...DR15
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

При включенном условии выполнения команда BAND(681) обнуляет указанное входное значение (16-битовое двоичное значение со знаком) в пределах заданных верхней и нижней границ зоны нечувствительности. Слова C и C+1 имеют следующее предназначение:

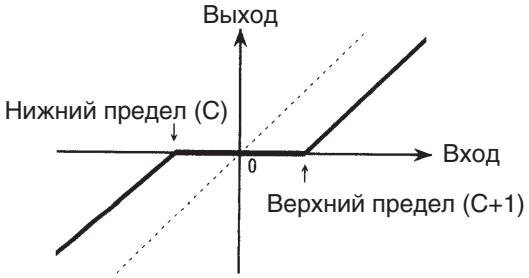
C	Нижнее предельное значение (нижняя граница зоны нечувствительности)
C+1	Верхнее предельное значение (верхняя граница зоны нечувствительности)

Слова C и C+1 должны принадлежать одной области памяти.

Если входное значение (S) больше или равно нижнему предельному значению (C) и меньше или равно верхнему предельному значению (C+1), в слово D выдается значение 0000 (hex) и устанавливается флаг равенства.

Если входное значение (S) меньше нижнего предельного значению (C), в слово D выдается разница между входным значением и нижним предельным значением, устанавливается флаг меньшего значения.

Если входное значение (S) больше верхнего предельного значения (C+1), в слово D выдается разница между входным значением и верхним предельных значением, устанавливается флаг большего значения.



Если выходное значение окажется меньше, чем 8000 (hex) или больше, чем 7FFF, знак будет изменен на противоположный. Например, при нижнем предельном значении 0100 (hex) и входном значении 8000 (hex) выходное значение будет следующим:
8000 (hex) [-32768] - 0100 (hex) [256] = 7F00 (hex) [32512].

Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	Включен, если верхнее предельное значение меньше нижнего предельного значения. Выключен во всех остальных случаях.
Флаг большего значения («Больше»)	>	Включен, если входное значение (S) больше верхнего предельного значения. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если результат = 0. Выключен во всех остальных случаях.
Флаг меньшего значения («Меньше»)	<	Включен, если входное значение (S) меньше нижнего предельного значения. Выключен во всех остальных случаях.
Флаг отрицательного значения	N	Включен, если старший бит результата = 1. Выключен во всех остальных случаях.

Меры предосторожности

Если верхнее предельное значение будет меньше нижнего предельного значения, произойдет ошибка и установится флаг ошибки.

Если входное значение (S) окажется больше верхнего предельного значения, будет установлен флаг большего значения (флаг «Больше»).

Если в D будет выдано значение 0000 hex, установится флаг равенства.

Если входное значение (S) окажется меньше нижнего предельного значения, будет установлен флаг меньшего значения (флаг «Меньше»).

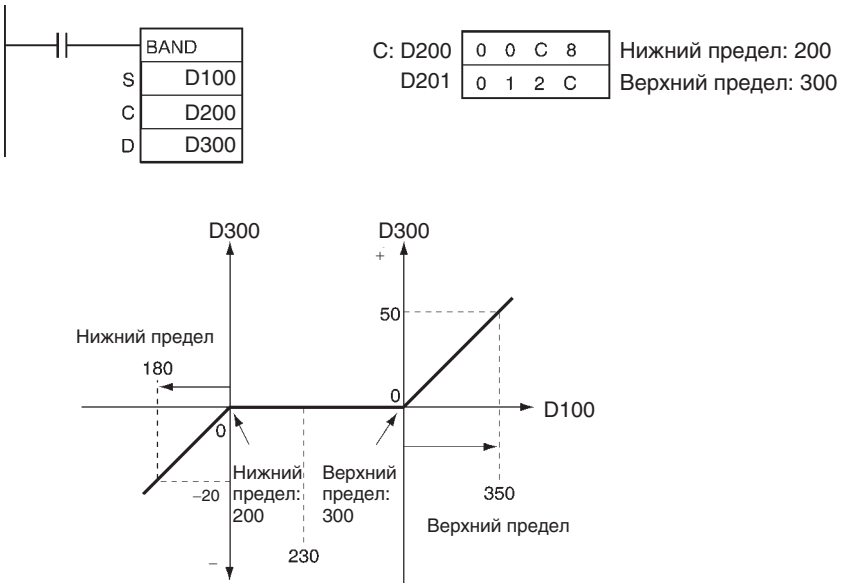
Если самый старший бит слова D будет содержать «1», будет установлен флаг отрицательного значения.

Пример

Если D100 = 00B4 hex (180), в слово D300 выдается значение 180-200=FFEC hex (-20), так как 180 меньше, чем нижнее предельное значение 200.

Если D100 = 00E6 hex (230), в слово D300 выдается 0, так как 230 находится в пределах верхней и нижней границ.

Если D100 = 015E hex (350), в слово D300 выдается значение 350-300=0032 hex (50), так как 350 больше, чем верхнее предельное значение 300.

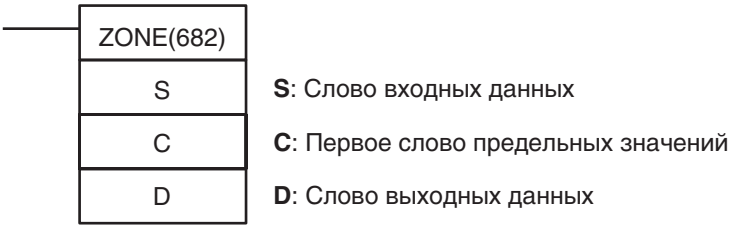


3-17-5 МЕРТВАЯ ЗОНА: ZONE(682)

Назначение

Добавление указанного смещения ко входному значению и выдача результата.

Символ РКС



S: Слово входных данных

C: Первое слово предельных значений

D: Слово выходных данных

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	ZONE(682)
	Однократное выполнение по положительному фронту	@ZONE(682)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	**Области пошаговых программ**	**Подпрограм-мы**	**Задачи обработки прерываний**
OK	OK	OK	OK

Характеристики операндов

Область	**S**	**C**	**D**
Область CIO	CIO 0...CIO 6143	CIO 0...CIO 6142	CIO 0...CIO 6143
Рабочая область	W0...W511	W0...W510	W0...W511
Область битов хранения	H0...H511	H0...H510	H0...H511
Область вспомогательных битов	A0...A959	A0...A958	A448...A959
Область таймеров	T0000...T4095	T0000...T4094	T0000...T4095

Область	S	C	D
Область счетчиков	C0000...C4095	C0000...C4094	C0000...C4095
Область DM	D0...D32767	D0...D32766	D0...D32767
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767		
Постоянные	#0000...#FFFF (двоичн.)	---	
Регистры данных	DR0...DR15	---	DR0...DR15
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

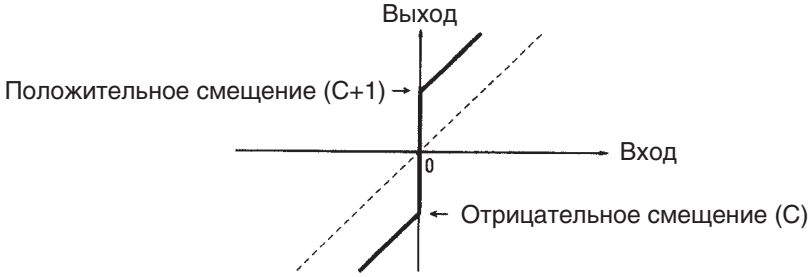
При включенном условии выполнения команда ZONE(682) добавляет указанное смещение к указанному входному значению (двоичному 16-битовому значению со знаком) и записывает результат в указанное слово. Слова C и C+1 имеют следующее предназначение:

C	Отрицательное смещение
C+1	Положительное смещение

При отрицательном входном значении (S < 0) в слово D выдается входное значение с добавленным к нему отрицательным смещением, устанавливается флаг меньшего значения.

При положительном входном значении (S > 0) в слово D выдается входное значение с добавленным к нему положительным смещением, устанавливается флаг большего значения.

При нулевом входном значении (S = 0) в слово D выдается значение 0000, устанавливается флаг равенства.



Если выходное значение окажется меньше, чем 8000 (hex) или больше, чем 7FFF, знак будет изменен на противоположный. Например, при величине отрицательного смещения FF00 (hex) и входном значении 8000 (hex) выходное значение будет следующим:
8000 (hex) [-32768] - FF00 (hex) [-256] = 7F00 (hex) [32512].

Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	Включен, если верхнее предельное значение меньше нижнего предельного значения. Выключен во всех остальных случаях.
Флаг большего значения («Больше»)	>	Включен, если входное значение (S) больше верхнего предельного значения. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если результат = 0. Выключен во всех остальных случаях.
Флаг меньшего значения («Меньше»)	<	Включен, если входное значение (S) меньше нижнего предельного значения. Выключен во всех остальных случаях.
Флаг отрицательного значения	N	Включен, если старший бит результата = 1. Выключен во всех остальных случаях.

Меры предосторожности

Если верхнее предельное значение будет меньше нижнего предельного значения, произойдет ошибка и установится флаг ошибки.

Если входное значение (S) окажется больше верхнего предельного значения, будет установлен флаг большего значения (флаг «Больше»).

Если в D будет выдано значение 0000 hex, установится флаг равенства.

Если входное значение (S) окажется меньше нижнего предельного значения, будет установлен флаг меньшего значения (флаг «Меньше»).

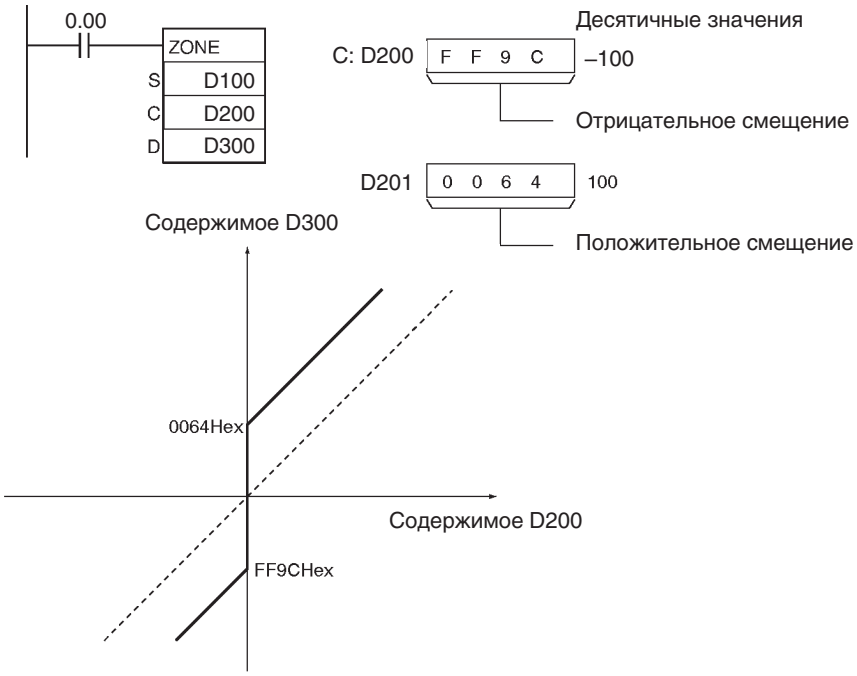
Если самый старший бит слова D будет содержать «1», будет установлен флаг отрицательного значения.

Пример

Если бит CIO 0.00 включен и значение слова D100 < 0, к значению D100 добавляется смещение -100 и полученное значение записывается в слово D300.

Если значение слова D100 = 0, в слово D300 записывается значение 0000 hex.

Если значение слова D100 > 0, к слову D100 добавляется смещение +100 и полученное значение записывается в слово D300.



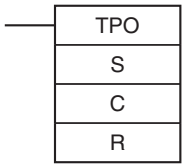
Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

3-17-6 ВЫХОД ШИМ: TPO(685)

Назначение

Чтение величины коэффициента заполнения или управляющего воздействия из указанного слова, преобразование полученного коэффициента заполнения в выход ШИМ в соответствии с указанными параметрами и выдача результата на указанный выход.

Символ РКС



S: Слово входных данных
C: Первое слово параметров
R: Бит импульсного выхода

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	TPO(685)
	Однократное выполнение по положительному фронту	Не предусмотрено
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
Не допускается	OK	OK	OK

Операнды

S: Слово входных данных

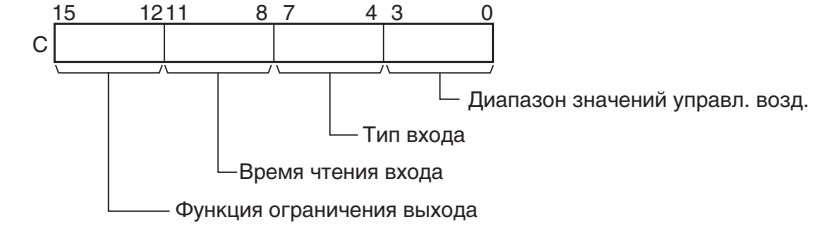
Указывает слово входных данных, содержащее величину коэффициента заполнения или управляющего воздействия. Тип входного значения (коэффициент заполнения или управляющее воздействие) указывается битами 04...07 слова C (введите в эти биты значение 0 hex — для коэффициента заполнения или 1 hex — для управляющего воздействия).

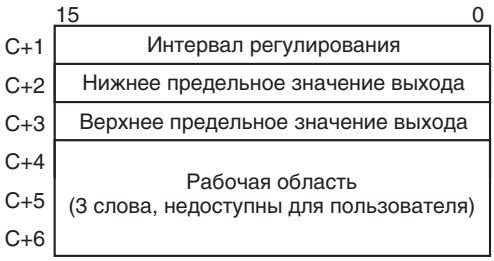
- Ввод коэффициента заполнения: 0000...2710 hex (0,00%...100,00%)
- Ввод управляющего воздействия (см. примеч.): 0000...FFFF hex (0...65 535 макс.). (Биты 00...03 слова C указывают диапазон значений управляющего воздействия, т. е. количество допустимых битов в слове управляющего воздействия. Следует указать такое же количество битов, какое указано для выходного диапазона в параметрах команды PID(190).)

Примечание. Для ввода управляющего воздействия в S следует указать адрес выходного слова команды PID(190) или PIDAT(191).

C...C+6: Параметры

Содержание слов параметров показано на следующем рисунке. Подробное описание параметров см. в пункте *Настройка параметров* в данном разделе.





Прим.: каждый из этих параметров будет подробно описан ниже.

R: Бит импульсного выхода

Указывает адрес выходного бита для импульсного выхода.

Обычно в качестве выхода используется транзисторный выход модуля транзисторных выходов, к которому подключено твердотельное реле.

Характеристики операндов

Область	S	C	R
Область CIO	CIO 0...CIO 6143	CIO 0...CIO 6137	CIO 0.00...CIO 6143.15
Рабочая область	W0...W511	W0...W505	W0.00...W511.15
Область битов хранения	H0...H511	H0...H505	H0.00...H511.15
Область вспомогательных битов	A0...959	A0...A953	A448.00...A959.15
Область таймеров	T0000...T4095	T0000...T4089	---
Область счетчиков	C0000...C4095	C0000...C4089	---
Область DM	D0...D32767	D0...D32761	---
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		---
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767		---
Постоянные	#0000...#FFFF (двоичн.)	---	---
Регистры данных	DR0...DR15	---	---
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Данная команда считывает из слова по адресу S величину коэффициента заполнения или величину управляющего воздействия, полученного на выходе ПИД-регулятора, в соответствии с параметрами в словах C...C+3 преобразует полученный коэффициент заполнения в импульсный сигнал для широтно-импульсного управления (см. примеч.) и выдает этот сигнал в бит по адресу R.

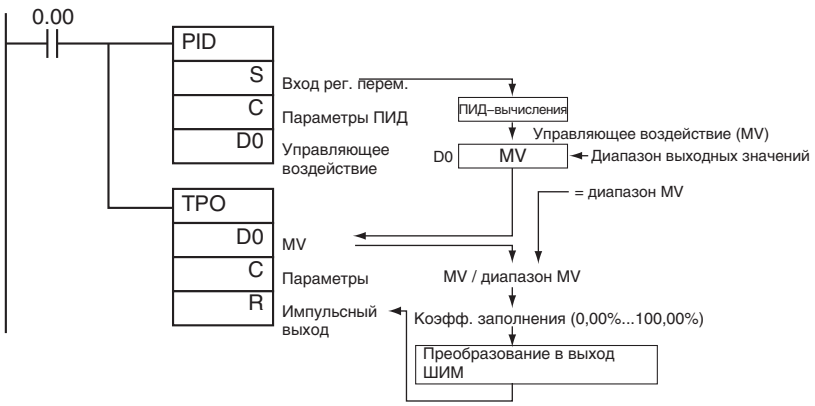
Примечание. Широтно-импульсное управление — это такое управление, при котором некоторая конечная физическая величина (например, ток или напряжение) изменяется пропорционально коэффициенту заполнения импульсного сигнала (т. е. отношению длительности импульса к периоду следования импульсов). Коэффициент заполнения указывается входным операндом S. Период следования импульсов (время, за которое выход находится один раз во включенном состоянии и один раз в выключенном состоянии, также называемое периодом регулирования) задается с помощью параметра в слове C+1.

Пример. При периоде регулирования 1 с и входном значении 50% выходной бит включается на 0,5 с и выключается на 0,5 с. При периоде регулирования 1 с и входном значении 80% выходной бит включается на 0,8 с и выключается на 0,2 с.

Как правило, команда TPO(685) используется вместе с командой PID(190) или PIDAT(191), и во входном слове (S) для команды TPO(685) указывается выходное слово команды ПИД-регулирования (т. е. управляющее воздействие). В операнде R обычно указывается бит транзисторного выхода модуля транзисторных выходов. Нагрузкой транзисторного выхода обычно является полупроводниковое реле, управляющее работой нагревательного прибора. Мощность нагрева изменяется пропорционально устанавливаемому коэффициенту заполнения.

Применение TPO(685) с командой ПИД-регулирования

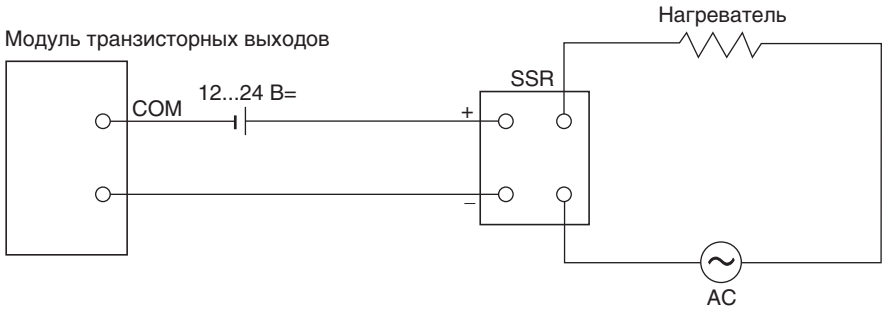
В случае объединения команды TPO(685) с командой ПИД-регулирования команда TPO(685) вычисляет коэффициент заполнения путем деления управляющего воздействия, получаемого с выхода ПИД-регулятора, на полный диапазон значений управляющего воздействия. Полученный коэффициент заполнения преобразуется в соответствующий импульсный сигнал для широтно-импульсного управления.



Параметр диапазона изменения значений управляющего воздействия команды TPO(685) должен иметь такое же значение, что и параметр выходного диапазона команды ПИД-регулирования. Например, если выходной диапазон команды ПИД-регулирования и диапазон управляющих воздействий команды TPO(685) ограничены 12 битам (0000...0FFF hex), коэффициент заполнения, на основании которого команда TPO(685) формирует импульсный сигнал, вычисляется путем деления текущего управляющего воздействия на выходе ПИД-регулятора на значение 0FFF hex.

Пример подключения к внешним устройствам

Подключите твердотельное реле (TTP) к модулю транзисторных выходов по схеме, которая представлена на рисунке ниже.



Настройка параметров

Управляющие данные		Параметр	Содержание	Диапазон значений	Изменение при включенном входном условии
Слово	Биты				
C	00...03	Диапазон значений управл. возд.	Указывает количество битов во входном значении.	0 hex: 8 битов;5 hex: 13 битов; 1 hex: 9 битов;6 hex: 14 битов; 2 hex: 10 битов;7 hex: 15 битов; 3 hex: 11 битов;8 hex: 16 битов. 4 hex: 12 битов;	Разрешено
	04...07	Тип входа	Указывает, что содержит S: коэффициент заполнения или управляющее воздействие.	0 hex: коэффициент заполнения. Диапазон значений S: 0000...2710 hex (0,00...100,00%) 1 hex: управляющее воздействие. Диапазон значений S: 0000...FFFF hex (0...65 535) (Максимальное значение зависит от диапазона значений MV, установленного битами 00...03 слова C.)	Разрешено
	08...11	Время чтения входа	Указывает, когда производится чтение входного значения.	0 hex: использовать начальное значение периода регулирования; 1 hex: использовать наименьшее значение; 2 hex: использовать наибольшее значение; 3 hex: непрерывно обновлять.	Разрешено
	12...15	Ограничение выхода	Служит для включения или выключения функции ограничения выходного значения.	0 hex: выключено; 1 hex: включено (см. примеч.).	Разрешено

Управляющие данные		Параметр	Содержание	Диапазон значений	Изменение при включенном входном условии
Слово	Биты				
C+1	00...15	Период регулирования	Период регулирования (фактически, период следования управляющих импульсов).	0064...270F hex (1,00...99,99 с) **Примечание.** Например, значению 1,00 соответствует 0064 hex, а не 0001 hex.	Разрешено
C+2	00...15	Нижнее предельное значение выхода	Указывает нижнее предельное значение при включенном ограничении выхода.	0000...2710 hex (0...100,00%)	Разрешено
C+3	00...15	Верхнее предельное значение выхода	Указывает верхнее предельное значение при включенном ограничении выхода.	0000...2710 hex (0...100,00%)	Разрешено
C+4	00...15	Рабочая область	Эта рабочая область используется системой. Для пользователя она недоступна.	Использовать невозможно.	---
C+5	00...15				
C+6	00...15				

Примечание. Если включена функция ограничения выхода, необходимо также задать нижние и верхние предельные значения с соблюдением следующего правила: 0000 hex ≤ нижний предел ≤ верхний предел ≤ 2710 hex.

Выполнение

- Команда выполняется, пока включено входное условие.
- В процессе работы команда включает и выключает выходной бит (R) в соответствии с заданным коэффициентом заполнения.
- При каждом выполнении команды считываются текущие значения параметров (в словах C...C+3). Если некоторые параметры требуется изменить, их требуется изменять вместе в одно и то же время, чтобы не смешать разные наборы параметров.
- В процессе выполнения команды выход (R) последовательно включается и выключается, при этом максимальная погрешность периода включения/выключения выхода составляет 10 мс.
- Когда входное условие выключается, выполнение команды прекращается. При этом значение истекшего времени сбрасывается и при следующем выполнении команды период регулирования отсчитывается с самого начала.
- Содержимое входного слова (S) (коэффициент заполнения или управляющее воздействие) определяется параметром выбора типа входного значения (биты 04...07 слова C). Если S содержит величину управляющего воздействия, команда вычисляет соответствующий ей коэффициент заполнения путем деления введенного управляющего воздействия на диапазон изменения значений управляющего воздействия (биты 00...03 слова C).

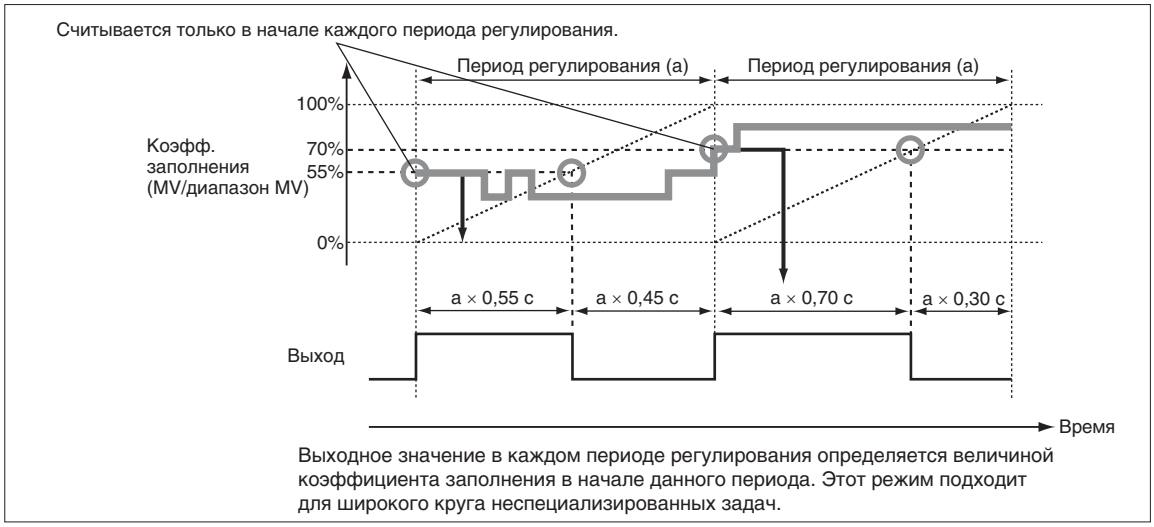
Время чтения входного значения (биты 08...11 слова C)

Данный параметр указывает, когда именно должно производиться чтение содержимого входного слова (S). Значения параметра поясняются в следующей таблице.

Время чтения входа	Описание
0: использовать начальные значения периода регулирования	Введенный коэффициент заполнения считывается в самом начале периода регулирования, в течение периода регулирования изменить коэффициент заполнения невозможно.
1: использовать наименьшее значение	Если входной коэффициент заполнения в какой-то момент времени становится меньше коэффициента заполнения, действовавшего в начале периода регулирования, то вступает в силу это меньшее значение и время включенного состояния выхода (длительность импульса) уменьшается соответствующим образом.
2: использовать наибольшее значение	Если входной коэффициент заполнения в некоторый момент времени становится больше коэффициента заполнения, действовавшего в начале периода регулирования, вступает в силу это большее значение и время включенного состояния выхода (длительность импульса) увеличивается соответствующим образом.
3: непрерывно обновлять.	Текущее значение коэффициента заполнения считывается при каждом выполнении команды и сразу же вступает в силу.

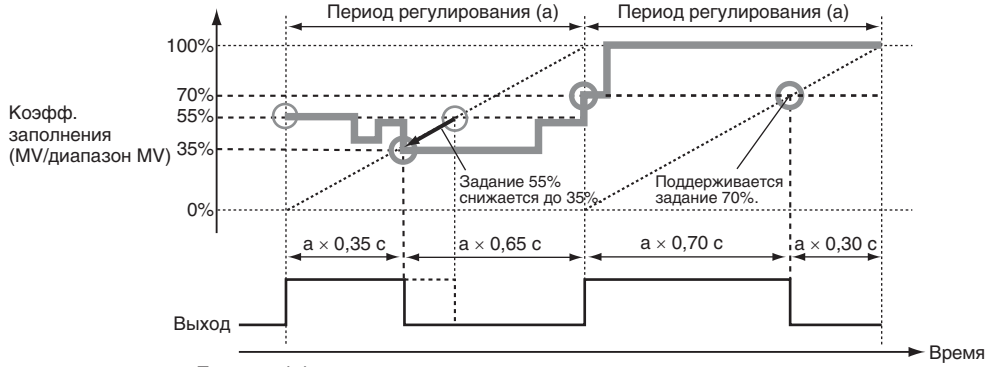
Работа выхода широтно-импульсного управления для каждого из трех режимов чтения входного значения показана на следующих рисунках.

- Время чтения входа = 0 (использовать начальное значение периода регулирования)



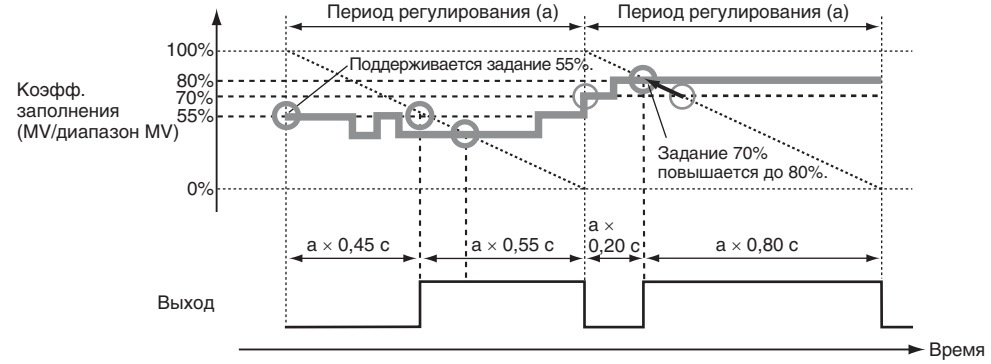
Выходное значение в каждом периоде регулирования определяется величиной коэффициента заполнения в начале данного периода. Этот режим подходит для широкого круга неспециализированных задач.

• Время чтения входа = 1 (использовать наименьшее значение)



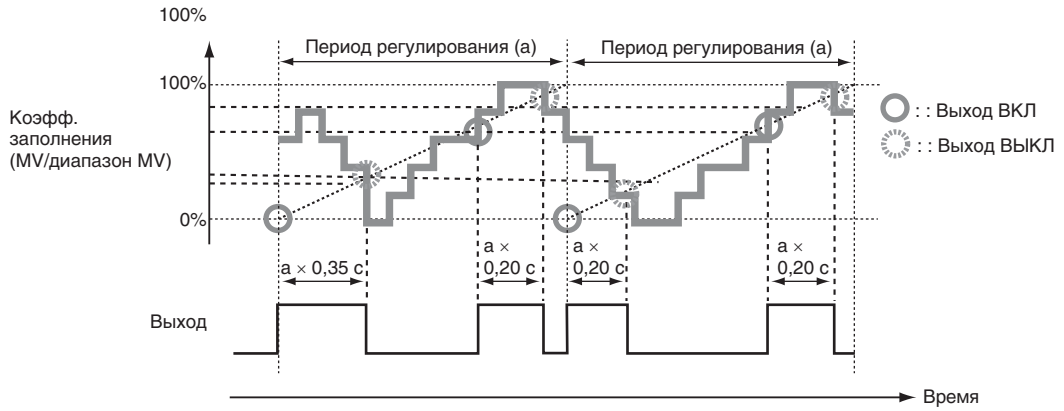
Если коэфф. заполнения достаточно рано опускается ниже первоначального значения, вступает в силу новое значение коэфф. заполнения и выход отключается раньше.
Данный режим следует использовать, например, для устранения эффекта перерегулирования при управлении нагревом методом ШИМ с относительно долгим периодом регулирования.

• Время чтения входа = 2 (использовать наибольшее значение)



Если коэфф. заполнения достаточно рано становится больше первоначального значения, вступает в силу новое значение коэфф. заполнения и выход включается раньше. (В этом режиме порядок чередования состояний выхода (ВКЛ и ВЫКЛ) меняется на противоположный, и выход переходит из выключенного состояния во включенное.)
Данный режим следует использовать, например, для устранения эффекта недорегулирования при управлении охлаждением методом ШИМ с относительно долгим периодом регулирования.

• Время чтения входа = 3 (непрерывное обновление)



Изменение коэффициента заполнения контролируется в реальном времени. Если коэфф. заполнения достаточно рано опускается ниже первоначального значения, вступает в силу новое значение коэфф. заполнения и выход отключается раньше. Если коэфф. заполнения после этого снова возрастает, он вновь корректируется и выход включается. Этот процесс непрерывно повторяется.
Использование данного режима способствует улучшению отклика системы на быстрые изменения коэфф. заполнения при относительно долгом периоде регулирования. Это режим также рекомендуется для высокоточных систем регулирования освещения или мощности.

Флаги

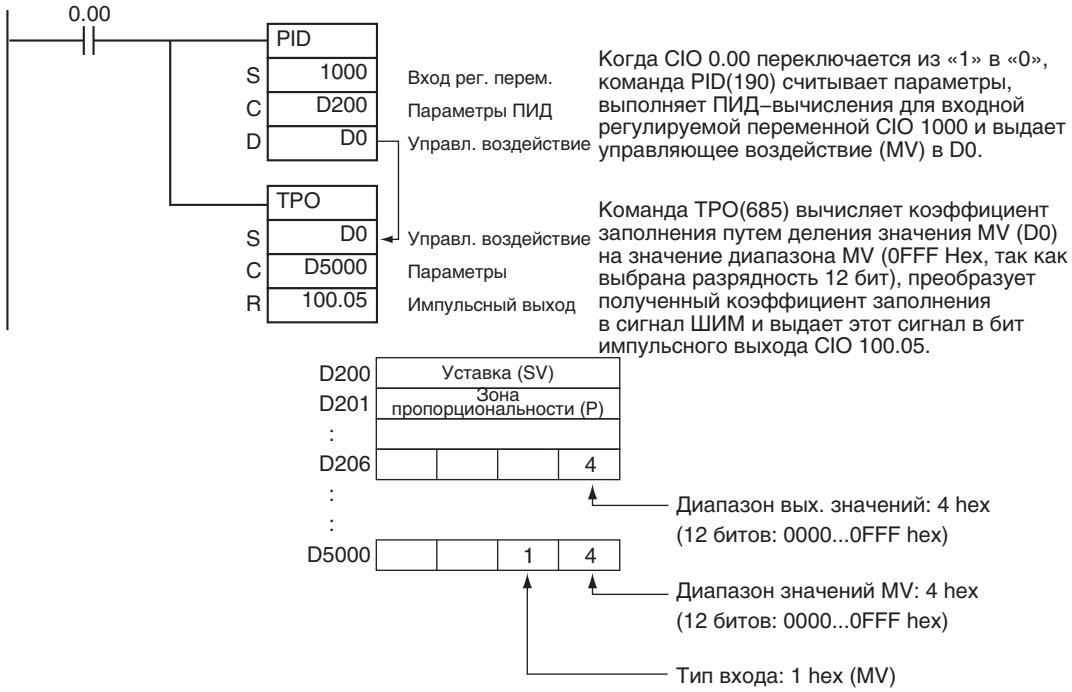
Название	Обозначение	Действие
Флаг ошибки	ER	Включен, если входное значение в S выходит за допустимый диапазон. (Диапазон входных значений зависит от выбранного типа входных значений.) Включен, если значение C выходит за допустимый диапазон. (Неверный диапазон управляющих воздействий вызовет ошибку, только если выбран ввод управляющих воздействий.) Включен, если период регулирования в C+1 выходит за допустимый диапазон. Включен, если при включенной функции ограничения выхода задано недопустимое значение нижней (C+2) или верхней (C+3) границы диапазона. Включен, если при включенной функции ограничения выхода нижняя граница диапазона (C+2) меньше или равна верхней границе диапазона (C+3). Выключен во всех остальных случаях.

Пример

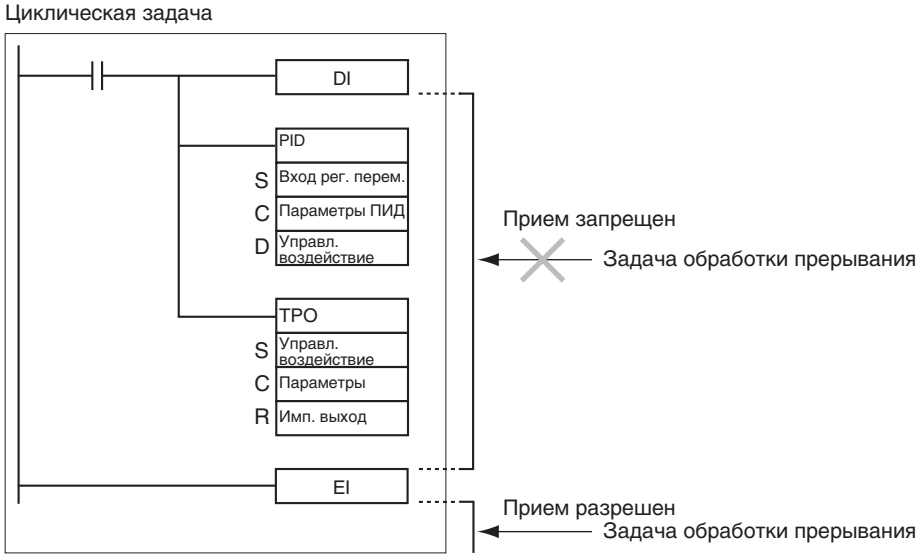
Пример 1. Применение команды TPO(685) с командой PID(190).

Если включен бит CIO 0.00, команда TPO(685) получает величину управляющего воздействия с выхода команды PID(190) (слово D0), по этому значению вычисляет соответствующий коэффициент заполнения (коэфф. заполн. = MV ÷ диапазон MV), в соответствии с полученным коэффициентом заполнения формирует импульсный сигнал и выдает его в бит CIO 100.05.

В данном случае слово CIO 100 назначено модулю транзисторных выходов, и к выходу, которому соответствует бит CIO 100.05, подключено полупроводниковое реле, управляющее питанием нагревательного прибора.



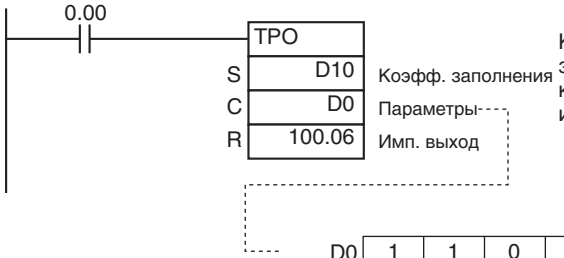
Примечание. Если команда TPO(685) используется в комбинации с командой PID(190) в циклической задаче и в программе также используется прерывание, непосредственно перед командой PID(190) и командой TPO(685) необходимо разместить команду DI(693) (ЗАПРЕТИТЬ ПРЕРЫВАНИЯ), чтобы временно запретить обработку прерываний. Если прерывания не будут отключены и в период между командой PID(190) и командой TPO(685) поступит прерывание, период регулирования может быть сдвинут.



Пример 2. Автономное использование команды TPO(685).

Если включен бит CIO 0.00, команда TPO(685) считывает коэффициент заполнения из слова D10, преобразует его в импульсный сигнал и выдает этот сигнал в бит CIO 100.06.

В данном примере используется период регулирования 1 с, включена функция ограничения выхода с нижним граничным значением 20,00% и верхним граничным значением 80,00%.



Команда TPO(685) считывает коэфф. заполнения в D10, преобразует полученный коэффициент заполнения в сигнал ШИМ и выдает этот сигнал в бит CIO 100.06.

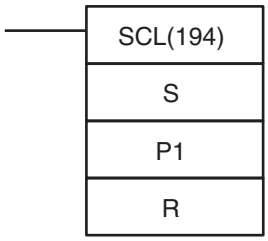
D0	1	1	0	0	Ввод коэфф. заполнения, время чтения входа и включение ограничения выхода
D1	0	0	6	4	Период регулирования = 1,00 с
D2	0	7	D	0	Нижний предел выхода = 20,00%
D3	1	F	4	0	Верхний предел выхода = 80,00%
D4	Не изменять.				
D5	Не изменять.				
D6	Не изменять.				
:					
:					
D10	0...2710 hex				0...100,00%

3-17-7 ИЗМЕНЕНИЕ МАСШТАБА: SCL(194)

Назначение

Преобразование двоичного значения без знака в двоично-десятичное значение без знака в соответствии с заданной линейной функцией.

Символ РКС



S: Исходное слово

P1: Первое слово параметров

R: Слово результата

Варианты выполнения

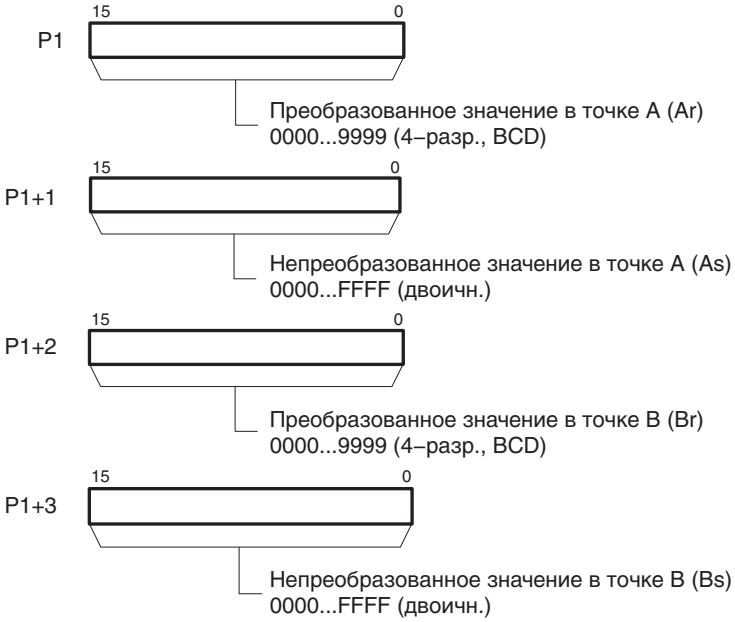
Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	SCL(194)
	Однократное выполнение по положительному фронту	@SCL(194)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	**Области пошаговых программ**	**Подпрограммы**	**Задачи обработки прерываний**
OK	OK	OK	OK

Операнды

Содержание четырех слов параметров (P1...P1+3) показано на следующем рисунке.



Характеристики операндов

Область	S	P1	R
Область CIO	CIO 0...CIO 6143	CIO 0...CIO 6140	CIO 0...CIO 6143
Рабочая область	W0...W511	W0...W508	W0...W511
Область битов хранения	H0...H511	H0...H508	H0...H511
Область вспомогательных битов	A0...A959	A0...A956	A448...A959
Область таймеров	T0000...T4095	T0000...T4092	T0000...T4095
Область счетчиков	C0000...C4095	C0000...C4092	C0000...C4095
Область DM	D0...D32767	D0...D32764	D0...D32767
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767		
Постоянные	---		
Регистры данных	DR0...DR15	---	DR0...DR15
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда SCL(194) преобразует двоичное значение без знака, содержащееся в исходном слове S, в двоично-десятичное значение без знака, используя линейную функцию, заданную точками (As, Ad) и (Bs, Bd), и выдает результат в выходное слово R. Координаты точек (As, Ar) и (Bs, Br) указываются четырьмя словами (P1...P1+3), адрес первого из которых указывается в операнде P1. Координаты этих точек по оси абсцисс (As и Bs) — это значения до преобразования, а координаты по оси ординат (Ar и Br) — это значение после преобразования.

Для преобразования используются следующие формулы.

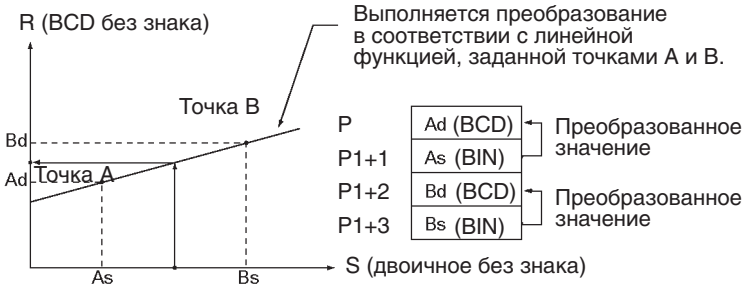
$$R = Bd - \frac{(Bd - Ad)}{(Bs - As)\ (BCD)} \times (Bs - S)\ (BCD)$$

Угловой коэффициент определяется отношением:

$$к = \frac{(Bd - Ad)}{(Bs - As)\ (BCD)}$$

Точками A и B можно задать линию как с положительным, так и с отрицательным угловым коэффициентом (наклоном). При отрицательном угловом коэффициенте шкала значений становится обратной.

Результат округляется до ближайшего целого значения. При получении результата меньше 0000 в качестве результата выдается 0000. Если полученный результат превышает 9999, выдается значение 9999.



С помощью команды SCL(194) некоторый диапазон значений (например, диапазон цифровых значений аналогового сигнала на аналоговом входе) может быть приведен к требуемому диапазону (шкале) значений. Например, если сигналу напряжения 1...5 В после аналого-цифрового преобразования в памяти модуля ЦПУ соответствуют значения 0000...0FA0 hex, эти значения с помощью команды SCL(194) можно привести к шкале 50...200°C.

Команда SCL(194) преобразует значение без знака в двоичном коде в значение без знака в двоично-десятичном коде. Для преобразования отрицательного значения к нему сначала следует добавить максимальное по модулю отрицательное значение (с помощью отдельной команды в программе) и лишь после этого использовать команду SCL(194) (см. пример).

Команда SCL(194) не может выдать отрицательное значение в слово результата R. При получении отрицательного результата в R выдается значение 0000.

Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	Включен, если содержимое P1 (Ar) или P1+2 (Br) не в формате BCD. Включен, если содержимое P1+1 (As) равно содержимому P1+3 (Bs). Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если результат = 0. Выключен во всех остальных случаях.

Меры предосторожности

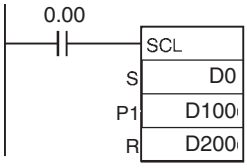
Если значения Ar (P1) и Br (P1+2) не будут представлены в двоично-десятичном коде или если значения As (P1+1) и Bs (P1+3) будут равны друг другу, произойдет ошибка и будет установлен флаг ошибки.

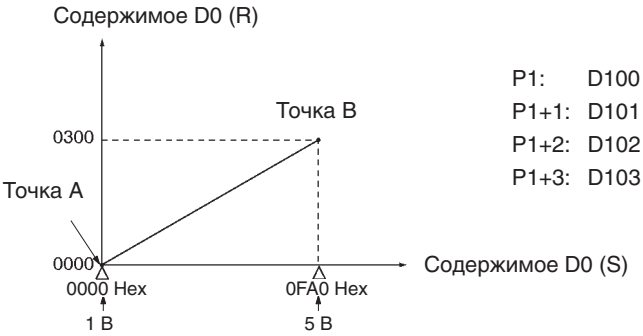
Если в результате выполнения команды слово D будет содержать «0000», будет установлен флаг равенства.

Примеры

В представленном ниже примере предполагается, что аналоговому сигналу напряжения 1...5 В после АЦ-преобразования соответствуют значения 0000...0FA0 hex, которые записываются по адресу D0. Команда SCL(194) преобразует (изменяет масштаб) содержимого слова CIO 200, приводя его к диапазону 0000...0300 BCD.

Если бит CIO 0.00 включен, содержимое D0 приводится к новой шкале с использованием линейной функции, которая задана точкой A (0000, 0000) и точкой B (0FA0, 0300). Координаты этих точек заданы словами D100...D103, результат выдается в слово D200.

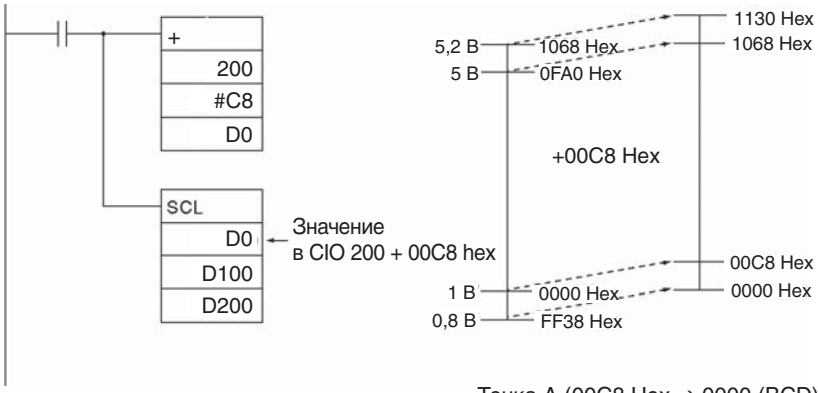


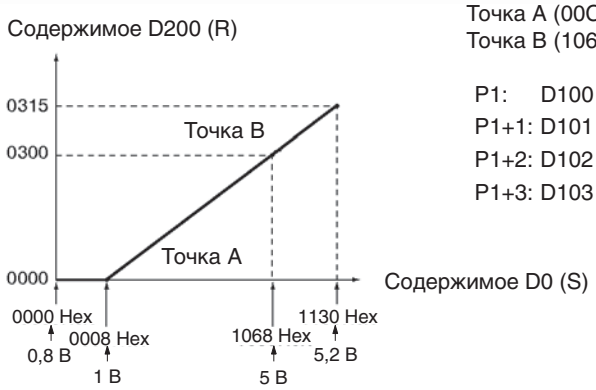


P1:	D100	0	0	0	0	Ad (BCD)
P1+1:	D101	0	0	0	0	As (BIN)
P1+2:	D102	0	3	0	0	Bd (BCD)
P1+3:	D103	0	F	A	0	Bs (BIN)

Отрицательные значения

Модуль аналоговых входов в действительности выдает значения от FF38 до 1068 hex, соответствующие диапазону от 0,8 до 5,2 В. Команда SCL(194), однако, может работать только с двоичными значениями без знака в диапазоне от 0000 до FFFF hex, поэтому ее невозможно использовать непосредственно для двоичных значений со знаком, которые выдаются модулем аналоговых входов при напряжении ниже 1 В (0000 hex), то есть для значений в диапазоне от FF38 до FFFF hex. По этой причине в прикладной программе перед использованием команды SCL(194) ко всем значениям требуется добавлять 00C8 hex (максимальное по модулю отрицательное значение), чтобы FF38 hex представлялось как 0000 hex (см. пример ниже).





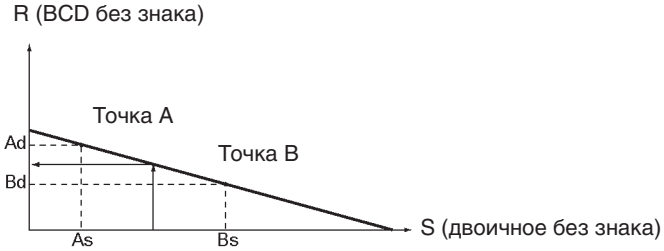
Точка A (00C8 Hex → 0000 (BCD))
Точка B (1068 Hex → 0300 (BCD))

P1:	D100	0 0 0 0	Ad	(BCD)
P1+1:	D101	0 0 C 8	As	(BIN)
P1+2:	D102	0 3 0 0	Bd	(BCD)
P1+3:	D103	1 0 6 8	Bs	(BIN)

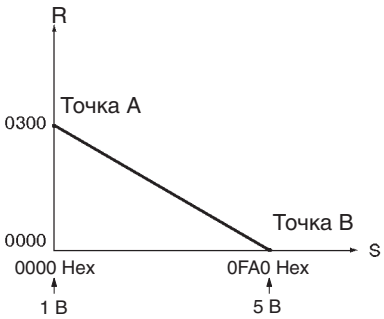
В данном примере преобразование значений в диапазоне от 0000 до 00C8 hex дает отрицательные значения. Однако команда SCL(194) может выдавать только двоично-десятичные значения без знака в диапазоне от 0000 до 9999, поэтому для всех значений D0 в диапазоне от 0000 до 00C8 hex будет выдаваться значение 0000 BCD.

Изменение направления шкалы значений

Задав As < Bs и Ar > Br, шкалу значений можно сделать обратной. График линейной функции будет выглядеть следующим образом.



Пример обратной шкалы показан на рисунке ниже. Здесь значениям напряжения 1...5 В (0000...0FA0 hex) в новом масштабе соответствуют значения 0300...0000.

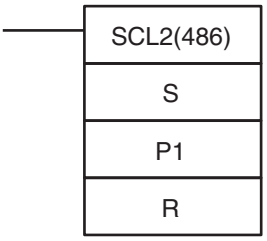


3-17-8 ИЗМЕНЕНИЕ МАСШТАБА 2: SCL2(486)

Назначение

Преобразование двоичного значения со знаком в двоично-десятичное значение со знаком в соответствии с заданной линейной функцией. Для линейной функции также можно указать величину смещения.

Символ РКС



S: Исходное слово

P1: Первое слово параметров

R: Слово результата

Варианты выполнения

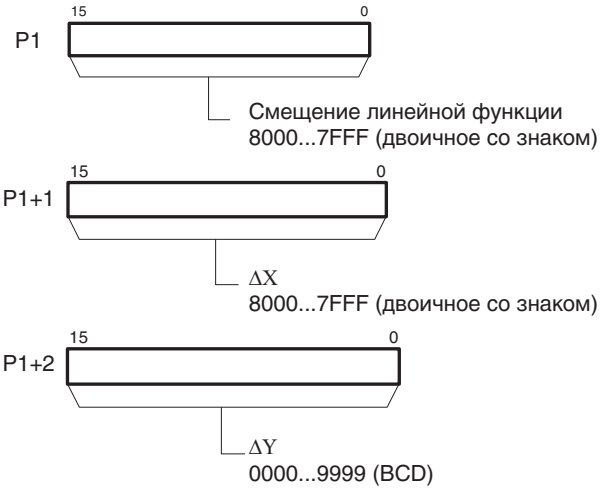
Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	SCL2(486)
	Однократное выполнение по положительному фронту	@SCL2(486)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

Содержание трех слов параметров (P1...P1+2) показано на следующем рисунке.



Характеристики операндов

Область	S	P1	R
Область CIO	CIO 0...CIO 6143	CIO 0...CIO 6141	CIO 0...CIO 6143
Рабочая область	W0...W511	W0...W509	W0...W511
Область битов хранения	H0...H511	H0...H509	H0...H511
Область вспомогательных битов	A0...A959	A0...A957	A448...A959
Область таймеров	T0000...T4095	T0000...T4093	T0000...T4095

Область	S	P1	R
Область счетчиков	C0000...C4095	C0000...C4093	C0000...C4095
Область DM	D0...D32767	D0...D32765	D0...D32767
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767		
Постоянные	---		
Регистры данных	DR0...DR15	---	DR0...DR15
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда SCL2(486) преобразует двоичное значение со знаком, содержащееся во входном слове S, в двоично-десятичное значение со знаком (фактически выдает абсолютное значение, а знак указывается флагом переноса), используя линейную функцию, которая задана угловым коэффициентом (отрезками ΔX, ΔY) и смещением, и записывает результат в выходное слово R. Значения углового коэффициента (ΔX, ΔY) и смещения задаются в трех словах, адрес первого из которых указывается операндом команды P1. Знак результата сигнализируется состоянием флага переноса (ВКЛ: отрицательный, ВЫКЛ: положительный).

Для преобразования используются следующие формулы.

$$R = \frac{\Delta Y}{\Delta X\ (BCD)} \times S\ (BCD) - \text{смещение (BCD)}$$

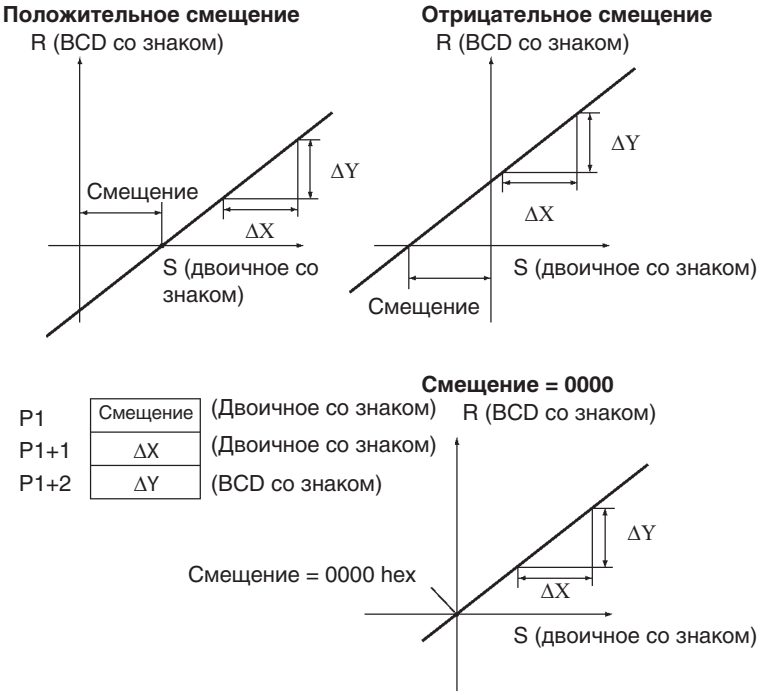
Угловой коэффициент: $\Delta Y/\Delta X$.

Смещение и угловой коэффициент могут иметь положительное, нулевое или отрицательное значение. При отрицательном угловом коэффициенте шкала значений становится обратной.

Результат округляется до ближайшего целого значения.

В слово R записывается модуль полученного двоично-десятичного значения, а знак указывается флагом переноса. Таким образом, результат может варьироваться в диапазоне от -9999 до 9999.

При получении результата меньше -9999 в качестве результата выдается значение -9999. Если полученный результат превышает 9999, выдается значение 9999.



С помощью команды SCL2(486) некоторый диапазон значений (например, диапазон цифровых значений аналогового сигнала на аналоговом входе) может быть приведен к требуемому диапазону (шкале) значений. Например, если сигналу напряжения 1...5 В после аналого-цифрового преобразования в памяти модуля ЦПУ соответствуют значения 0000...0FA0 hex, эти значения с помощью команды SCL2(486) можно привести к шкале -100...200°C.

Команда SCL2(486) преобразует двоичное значение со знаком в двоично-десятичное значение со знаком. В операнд S, таким образом, можно непосредственно вводить отрицательные значения. Команда SCL2(486) способна выдавать отрицательные значения, используя для этих целей флаг переноса.

Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	Включен. Если содержимое C+1 (ΔX) = 0000. Включен, если содержимое C+2 (ΔY) не в формате BCD. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если результат = 0. Выключен во всех остальных случаях.
Флаг переноса	CY	Включен, если результат отрицателен. Выключен при нулевом или положительном результате.

Меры предосторожности

Если для ΔX (C+1) задано значение «0000» или если значение ΔY (C+2) не представлено в двоично-десятичном коде, произойдет ошибка и будет установлен флаг ошибки.

Если в результате выполнения команды слово D будет содержать «0000», будет установлен флаг равенства.

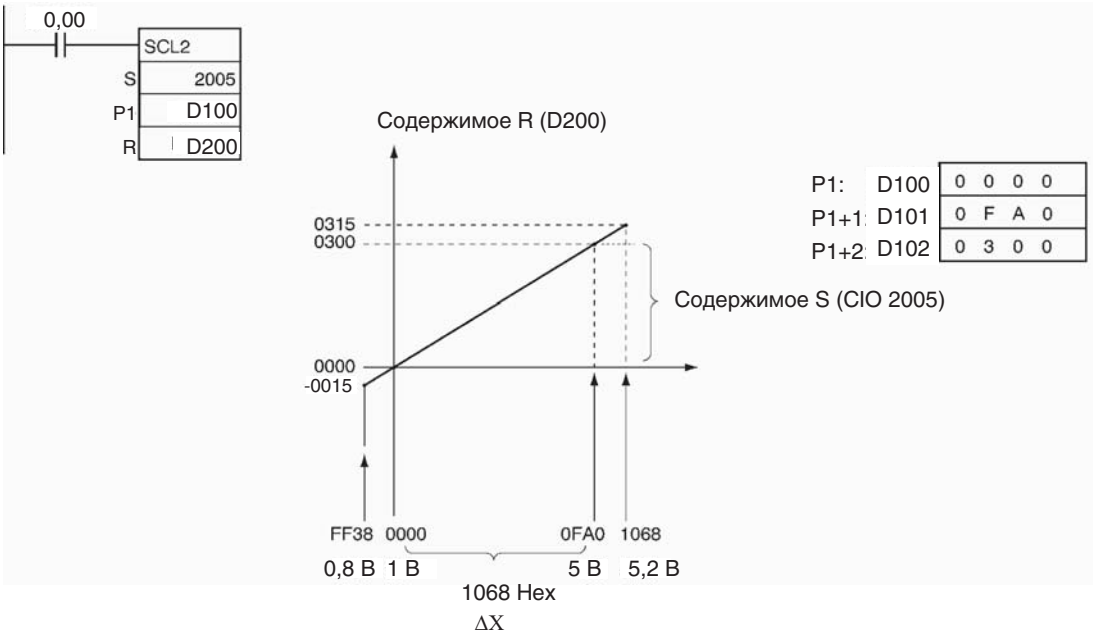
Если в результате выполнения команды будет получено отрицательное значение, будет установлен флаг переноса.

Примеры

Приведение шкалы аналогового входа 1...5 В к масштабу 0...300

В приведенном ниже примере предполагается, что аналоговому сигналу напряжения 1...5 В после АЦ-преобразования соответствуют значения 0000...0FA0 hex в слове CIO 2005. С помощью команды SCL2(486) шкала значений аналогового сигнала (содержимое слова CIO 2005) преобразуется в шкалу значений 0000...0300 BCD.

Если включен бит CIO 0.00, содержимое слова CIO 2005 приводится к новому масштабу (диапазону) с использованием линейной функции, которая задана угловым коэффициентом (величинами ΔX (0FA0) и ΔY (0300)) и смещением (0). Эти значения содержатся в словах D100...D102, а результат выдается в слово D200.

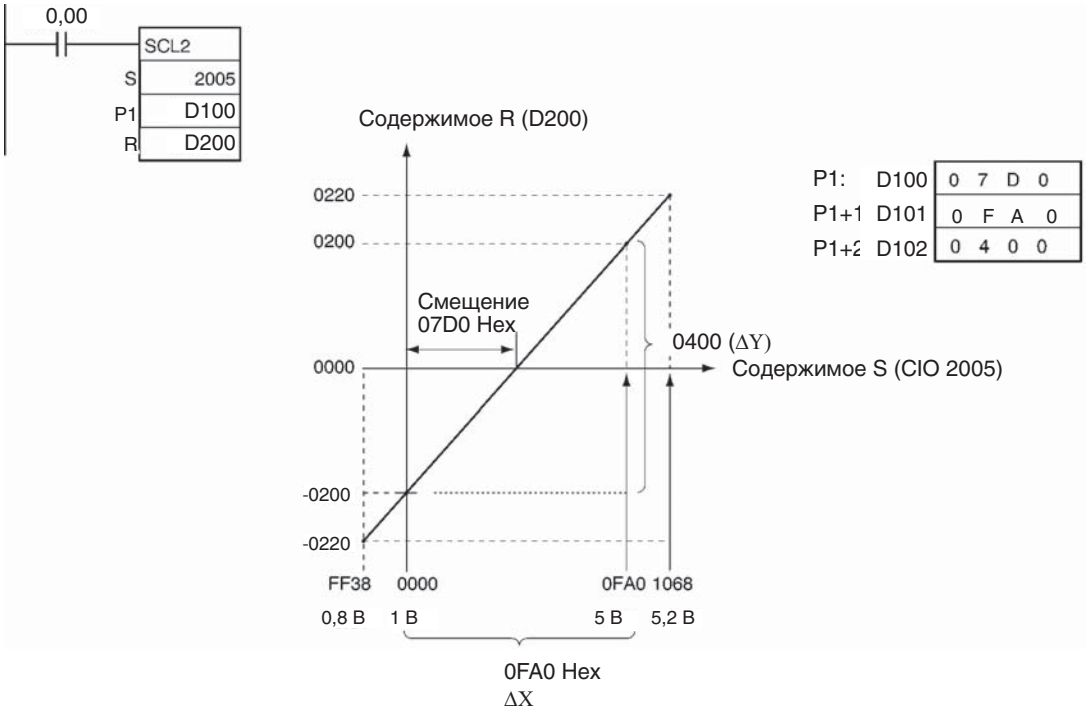


Приведение шкалы аналогового входа 1...5 В к масштабу -200...200

В приведенном ниже примере предполагается, что аналоговому сигналу напряжения 1...5 В после АЦ-преобразования соответствуют значения в диапазоне 0000...0FA0 hex в слове CIO 2005. С помощью команды SCL2(486) шкала аналогового сигнала (содержимое слова CIO 2005) преобразуется в шкалу значений -0200...0200 BCD.

Если бит CIO 0.00 включен, содержимое слова CIO 2005 приводится к новому масштабу (диапазону) с использованием линейной функции, которая задана угловым коэффициентом (величинами ΔX (0FA0), ΔY

(0400)) и смещением (07D0). Эти значения содержатся в словах D100...D102, а результат выдается в слово D200.

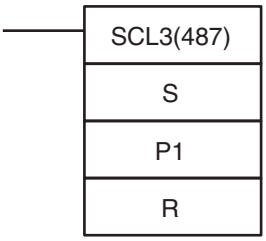


3-17-9 ИЗМЕНЕНИЕ МАСШТАБА 3: SCL3(487)

Назначение

Преобразование двоично-десятичного значения со знаком в двоичное значение со знаком в соответствии с заданной линейной функцией. Для линейной функции также можно указать величину смещения.

Символ РКС



S: Исходное слово

P1: Первое слово параметров

R: Слово результата

Варианты выполнения

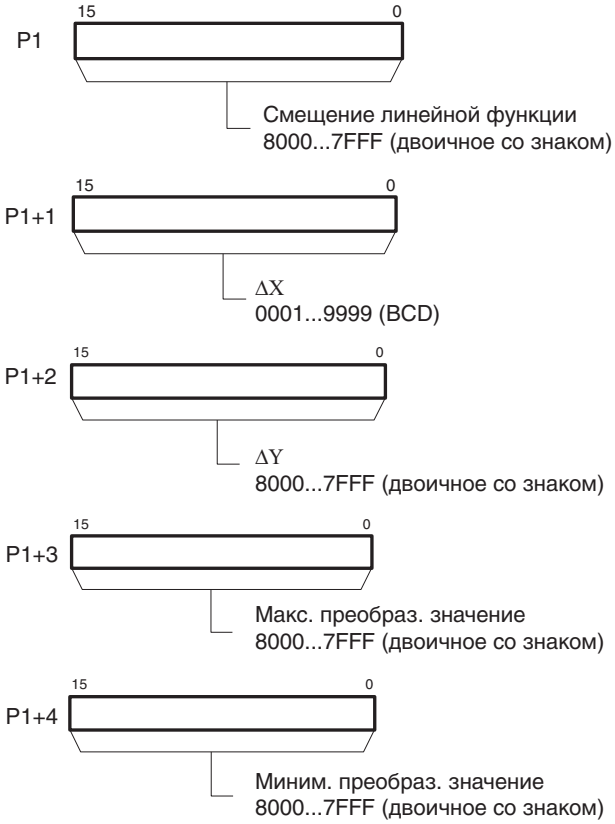
Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	SCL3(487)
	Однократное выполнение по положительному фронту	@SCL3(487)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

Содержание пяти слов параметров (P1...P1+4) показано на следующем рисунке.



Характеристики операндов

Область	S	P1	R
Область CIO	CIO 0...CIO 6143	CIO 0...CIO 6139	CIO 0...CIO 6143
Рабочая область	W0...W511	W0...W507	W0...W511
Область битов хранения	H0...H511	H0...H507	H0...H511
Область вспомогательных битов	A0...A447 A448...A959	A0...A443 A448...A955	A448...A959
Область таймеров	T0000...T4095	T0000...T4091	T0000...T4095
Область счетчиков	C0000...C4095	C0000...C4091	C0000...C4095
Область DM	D0...D32767	D0...D32763	D0...D32767
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767		
Постоянные	---		
Регистры данных	DR0...DR15	---	DR0...DR15
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда SCL3(487) преобразует двоично-десятичное значение со знаком (фактически вводится абсолютное значение, а знак указывается флагом переноса), содержащееся во входном слове S, в двоичное значение со знаком, используя линейную функцию, которая задана угловым коэффициентом (отрезками ΔX, ΔY) и смещением, и выдает результат в выходное слово R. Также указываются максимальное и минимальное значения для результата преобразования. Угловой коэффициент (ΔX, ΔY), смещение, максимальный и минимальный результаты преобразования задаются в пяти словах параметров, адрес первого из которых указывается операндом P1.

Знак результата сигнализируется состоянием флага переноса (ВКЛ: отрицательный, ВЫКЛ: положительный). Для включения и выключения флага переноса следует использовать, соответственно, команды STC(040) и CLC(041).

Для преобразования используются следующие формулы.

$$R = \frac{\Delta Y}{\Delta X \text{ (двоичн.)}} \times S \text{ (двоичн.)} + \text{Смещение}$$

Угловой коэффициент: $\Delta Y/\Delta X$.

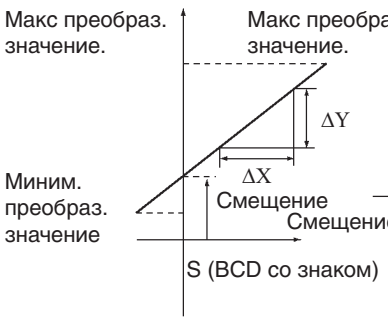
Смещение и угловой коэффициент могут иметь положительное, нулевое или отрицательное значение. При отрицательном угловом коэффициенте шкала значений становится обратной.

Результат округляется до ближайшего целого значения.

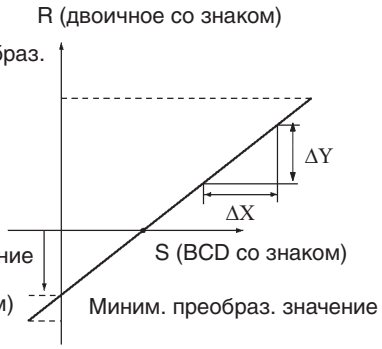
В слово S записывается модуль входного двоично-десятичного значения, а знак указывается флагом переноса. Входное значение, таким образом, может варьироваться в диапазоне -9999...9999.

Если полученный результат окажется меньше указанного минимального значения, будет выдано минимальное значение. Если полученный результат окажется больше указанного максимального значения, будет выдано максимальное значение.

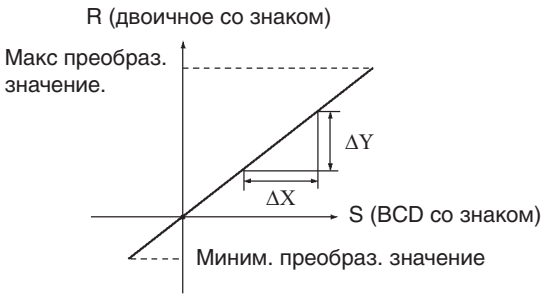
Положительное смещение



Отрицательное смещение



Смещение = 0000



Команда SCL3(487) предназначена для преобразования некоторой шкалы двоично-десятичных значений в шкалу двоичных значений со знаком для модулей аналоговых выходов. Например, с помощью команды SCL3(487) шкалу температур 0...200 °C можно привести к диапазону значений 0000...0FA0 (hex) и выдать эти значения в виде аналогового сигнала 1...5 В с помощью модуля аналоговых выходов.

Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	Включен, если содержимое S не в формате BCD. Включен, если содержимое C+1 (ΔX) выходит за диапазон 0001...9999 BCD. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если результат = «0». Выключен во всех остальных случаях.
Флаг отрицательного значения	N	Включен, если старший бит результата (R) = «1». Выключен во всех остальных случаях.

Меры предосторожности

Если содержимое слова S не будет представлено в двоично-десятичном коде или значение ΔX (C+1) не будет лежать в диапазоне 0001...9999 BCD, произойдет ошибка и будет установлен флаг ошибки.

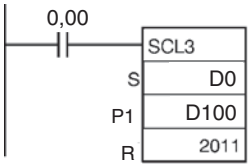
Если в результате выполнения команды слово D будет содержать «0000», будет установлен флаг равенства.

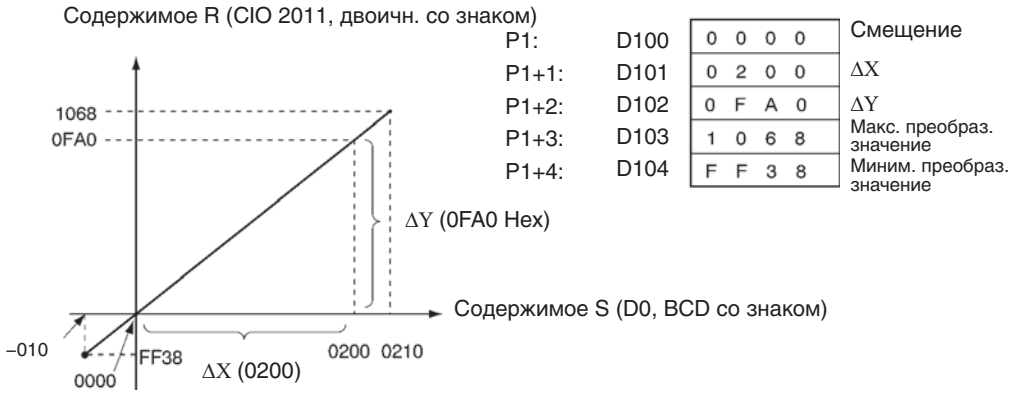
Если в результате выполнения команды старший бит слова результата (R) будет находиться в состоянии «1» (т. е. будет получен отрицательный результат), будет установлен флаг отрицательного значения.

Примеры

Значения в диапазоне от 0 до 200 требуется привести к шкале аналогового сигнала (1...5 В). Двоично-десятичные значения со знаком 0000...0200 преобразуются (масштабируются) в двоичные значения со знаком 0000...0FA0 для вывода в форме аналогового сигнала модулем аналоговых выходов.

При включении бита CIO 0.00 содержимое по адресу D0 преобразуется с использованием линейной функции, которая определена угловым коэффициентом (отрезками ΔX (0200), ΔY (0FA0)) и смещением (0). Эти параметры заданы словами D100...D102. Знак двоично-десятичного значения по адресу D0 указывается флагом переноса. Результат выдается в слово CIO 2011.



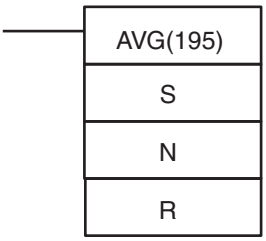


3-17-10 СРЕДНЕЕ ЗНАЧЕНИЕ: AVG(195)

Назначение

Вычисление среднего значения содержимого входного слова за указанное количество циклов.

Символ РКС



S: Исходное слово

N: Количество циклов усреднения

R: Слово результата

R+1: Первое слово рабочей области

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	AVG(195)
	Однократное выполнение по положительному фронту	Не предусмотрено
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
Не допускается	OK	OK	OK

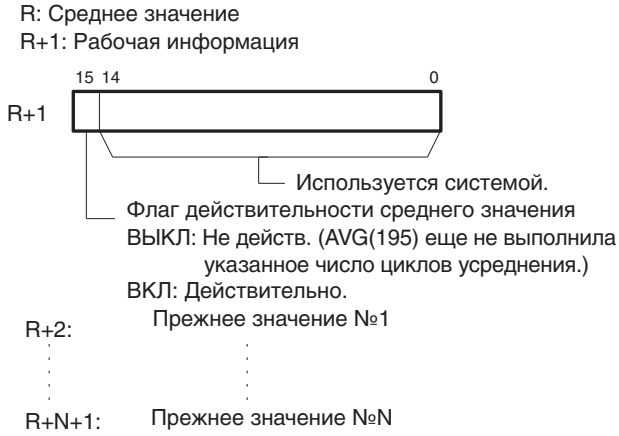
Операнды

N: Количество циклов

Значение количества циклов должно находиться в пределах от 0001 до 0040 hex (от 0 до 64 циклов).

R: Слово результата и R+1: Первое слово рабочей области

По истечении указанного количества циклов слово R будет содержать среднее значение. Слово R+1 содержит параметры операции усреднения, а слова R+2...R+N+1 содержат предыдущие значения S, как показано на рисунке ниже.



Характеристики операндов

Область	S	N	R
Область CIO	CIO 0...CIO 6143		
Рабочая область	W0...W511		
Область битов хранения	H0...H511		
Область вспомогательных битов	A0...A959		A448...A959
Область таймеров	T0000...T4095		
Область счетчиков	C0000...C4095		
Область DM	D0...D32767		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767		
Постоянные	#0000...#FFFF (двоичн.)	#0001...#0040 (двоичн.)	---
Регистры данных	DR0...DR15		---
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15		

Описание

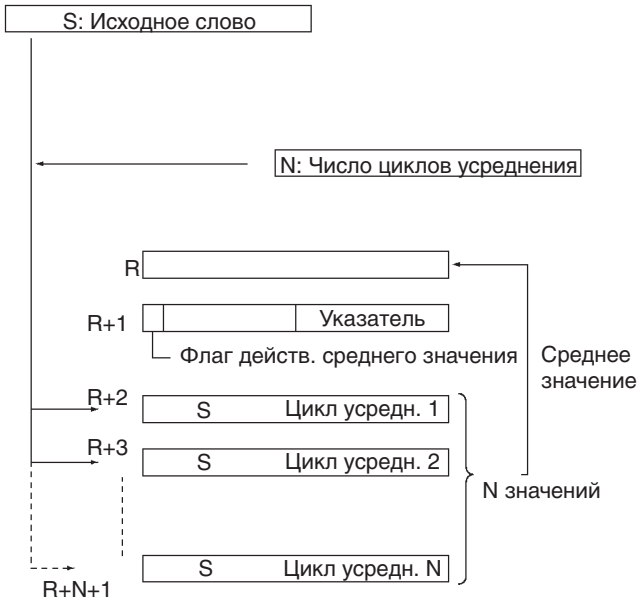
В течение первых N-1 циклов при включенном условии выполнения команда AVG(195) записывает по порядку значения слова S в слова рабочей области, начиная со слова R+2. Каждый раз, когда в рабочую область записывается значение, указатель предыдущего значения (биты 00...07 слова R+1) увеличивается на 1. До тех пор пока в рабочую область не будет записано N значений, в R будет выдаваться без изменений содержимое слова S, а флаг среднего значения (бит 15 слова R+1) будет оставаться выключенным.

Когда в слово R+N+1 оказывается записано значение с порядковым номером N, вычисляется среднее арифметическое всей совокупности сохраненных значений, которое выдается в слово R в формате двоичного значения без знака, и устанавливается флаг среднего значения (бит 15 слова R+1). В каждом последующем цикле в рабочую область сохраняется новое значение S (а наиболее старое удаляется), и значение R соответствующим образом обновляется.

Количество циклов усреднения (N) не может превышать 64. Даже если будет указано больше 64 циклов усреднения, среднее значение все равно будет определяться за 64 цикла.

После того как в рабочую область оказывается записано N-1 значений, указатель предыдущего значения обнуляется.

Среднее значение, выдаваемое в слово R, округляется до ближайшего целого значения.



Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	Включен, если содержимое N = 0. Выключен во всех остальных случаях.

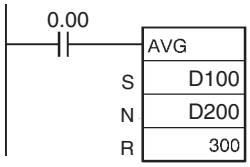
Меры предосторожности

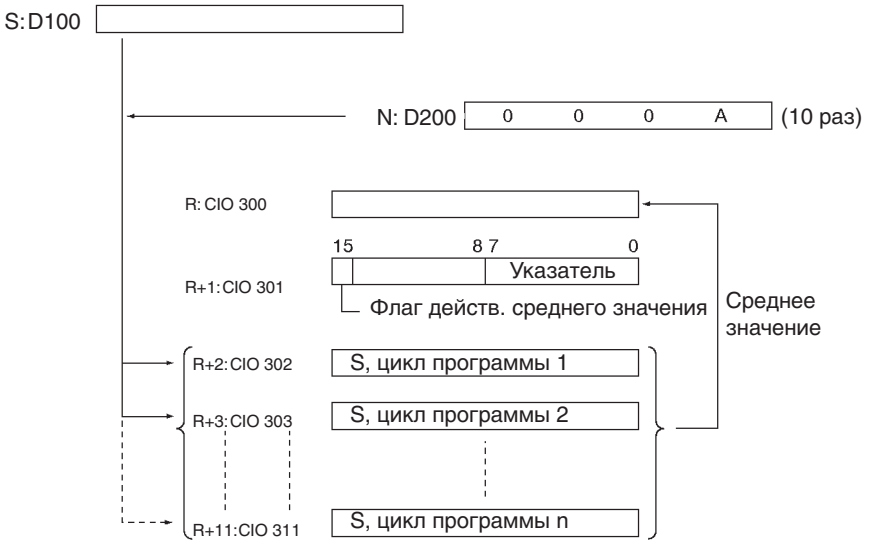
Всякий раз, когда условие выполнения переключается из «0» в «1», содержимое первого слова рабочей области (D+1) обнуляется (т. е. в него записывается 0000).

При первом выполнении программы в начале работы ПЛК содержимое первого слова рабочей области (D+1) не обнуляется. Если команда AVG(195) должна выполняться в самом первом цикле ПЛК, в программе следует предусмотреть обнуление первого слова рабочей области.

Если операнд N (количество циклов усреднения) будет содержать значение «0000», произойдет ошибка и будет установлен флаг ошибки

При включенном бите CIO 0.00 приведенная ниже программа в течение некоторого количества циклов, указанного в D200, сохраняет в рабочую область содержимое слова D100 (один раз в каждом цикле). Эти значения сохраняются по порядку в десять слов в диапазоне от CIO 302 до CIO 311. Среднее значение содержимого этих десяти слов записывается в слово CIO 300, после чего включается бит 15 слова CIO 301.





Примеры

В приведенном ниже примере в слово CIO 200 сначала записывается значение #0000, а затем его содержимое увеличивается на 1 в каждом цикле. В двух первых циклах команда AVG(195) передает содержимое слова CIO 200 в слова D1002 и D1003. Содержимое слова D1001 также изменится (что можно использовать в качестве признака изменения результата выполнения команды AVG(195)). В третьем и всех последующих циклах команда AVG(195) вычисляет среднее значение содержимого слов D1002...D1004 и выдает это значение в слово D1000.

	1–й цикл	2–й цикл	3–й цикл	4–й цикл
CIO 200	0000	0001	0002	0003

D1000	0000	0001	0001	0002	Среднее значение
D1001	0001	0002	8000	8001	Указатель
D1002	0000	0000	0000	0003	
D1003	–––	0001	0001	0001	3 прежних значения IR 40
D1004	–––	–––	0002	0002	

3-18 Команды подпрограмм

В данном разделе описаны команды, предназначенные для создания подпрограмм и управления их выполнением.

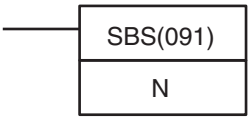
Команда	Мнемоническое обозначение	Код функции	Стр.
ВЫЗОВ ПОДПРОГРАММЫ	SBS	091	730
МАКРОС	MCRO	099	736
ВХОД В ПОДПРОГРАММУ	SBN	092	740
ВЫХОД ИЗ ПОДПРОГРАММЫ	RET	093	743
ВЫЗОВ ГЛОБАЛЬНОЙ ПОДПРОГРАММЫ	GSBS	750	743
ВХОД В ГЛОБАЛЬНУЮ ПОДПРОГРАММУ	GSBN	751	751
ВЫХОД ИЗ ГЛОБАЛЬНОЙ ПОДПРОГРАММЫ	GRET	752	754

3-18-1 ВЫЗОВ ПОДПРОГРАММЫ: SBS(091)

Назначение

Вызов подпрограммы с указанным номером и выполнение ее программы.

Символ РКС



N: Номер подпрограммы

Варианты выполнения

Варианты выполнения		
Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	SBS(091)
	Однократное выполнение по положительному фронту	@SBS(091)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

N: Номер подпрограммы

Указывает номер подпрограммы в диапазоне от 0 до 255 (десятичн).

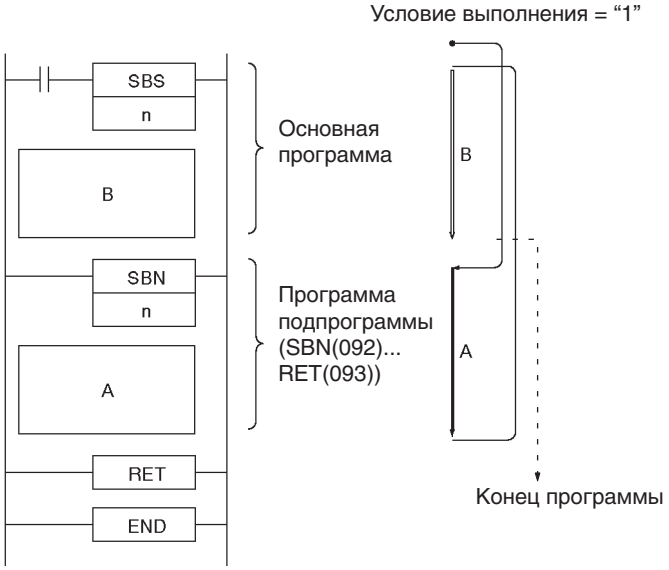
Характеристики операндов

Область	N
Область CIO	---
Рабочая область	---
Область битов хранения	---
Область вспомогательных битов	---
Область таймеров	---
Область счетчиков	---
Область DM	---
Косвенные адреса DM в двоичном формате	---

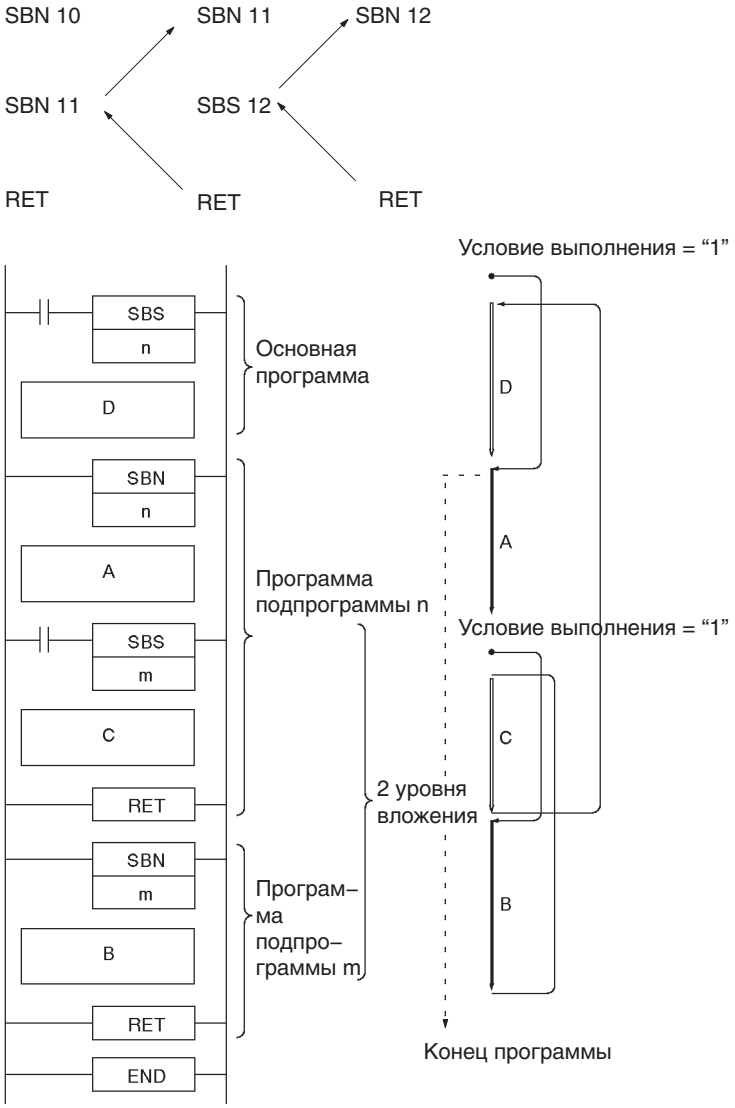
Область	N
Косвенные адреса DM в двоично-десятичном формате (BCD)	---
Постоянные	0...255 (десятичн.)
Регистры данных	---
Регистры указателей	---
Косвенная адресация с помощью регистров указателей	---

Описание

Команда SBS(091) вызывает подпрограмму с указанным номером. Под подпрограммой понимается сегмент программы, заключенный между командами SBN(092) и RET(093). Когда выполнение подпрограммы завершается, основная программа продолжает выполняться с команды, расположенной сразу после команды SBS(091).



Допускается до 16 уровней вложения подпрограмм. Слово «вложение» означает, что из программы некоторой подпрограммы вызывается другая подпрограмма. Пример программы с тремя уровнями вложения подпрограмм показан на рисунке ниже.

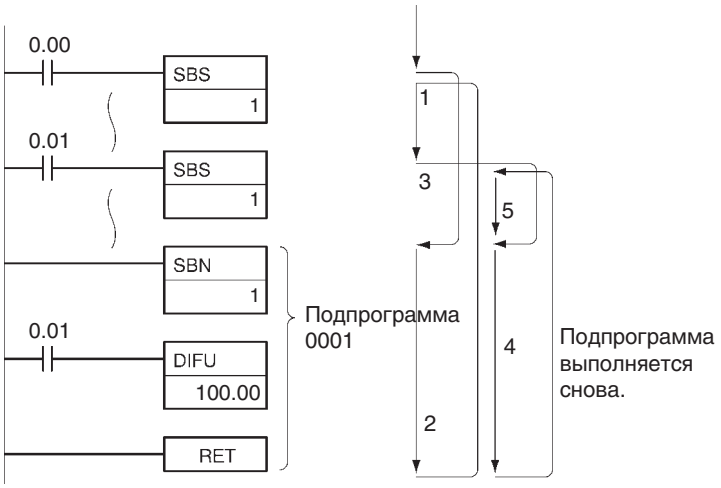


Примечание. Одна и та же подпрограмма может вызываться в программе более одного раза.

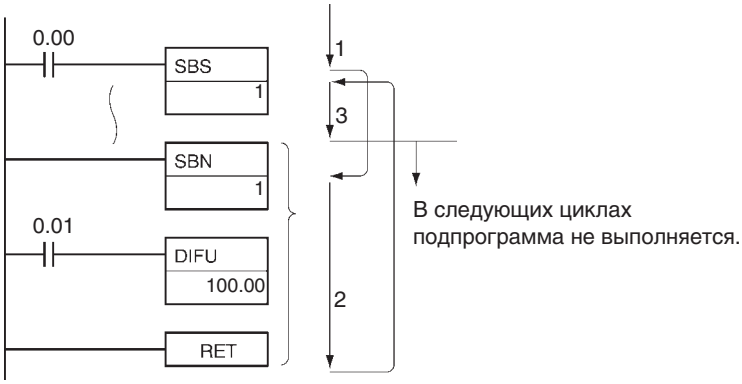
Подпрограммы и различение фронта

В случае использования в подпрограммах команд с различением фронта (DIFU(013), DIFU(014) или команд с префиксом положительного/ отрицательного фронта) необходимо учитывать описанные ниже особенности.

Если некоторая подпрограмма вызывается более одного раза в пределах одного цикла, команды с различением фронта условия выполнения в этой подпрограмме работают непредсказуемым образом. В приведенном ниже примере подпрограмма 0001 выполняется, когда бит CIO 0.00 включен. Бит CIO 100.00 в подпрограмме включается командой DIFU(013), когда бит CIO 0.01 переключается из «0» в «1». Если бит CIO 0.01 включен в том же цикле, подпрограмма 0001 будет выполнена еще раз, но в этот раз команда DIFU(013) выключит бит CIO 100.00, не проверяя состояние бита CIO 0.01.



С другой стороны, выход команды с различением фронта (DIFU(013) или DIFD(014)) останется включенным, если команда будет выполнена и выход будет включен, но во второй раз та же подпрограмма вызвана не будет.



В приведенном выше примере подпрограмма 0001 выполняется, если бит CIO 0.00 включен. Выход CIO 100.00 включается командой DIFU(013) после того, как CIO 0.01 переключается из «0» в «1». Если в следующем цикле бит CIO 0.00 будет выключен, подпрограмма 0001 больше выполнена не будет и выход CIO 100.00 останется включенным.

Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	Включен, если число уровней вложения превышает 16. Включен, если подпрограмма с указанным номером не существует. Включен, если подпрограмма вызывает сама себя. Включен, если вызванная подпрограмма уже выполняется. Включен, если указанная подпрограмма не определена в текущей задаче. Выключен во всех остальных случаях.

Меры предосторожности

Соответствующие друг другу команды SBS(091) и SBN(092) должны находиться в пределах одной задачи. В противном случае произойдет ошибка.

Если команда SBS(091) расположена в сегменте программы, который заблокирован с помощью команд IL(002) и ILC(003), она выполняется как команда NOP(000).

При выполнении команды SBS(091) в перечисленных ниже случаях подпрограмма не будет вызвана и будет установлен флаг ошибки.

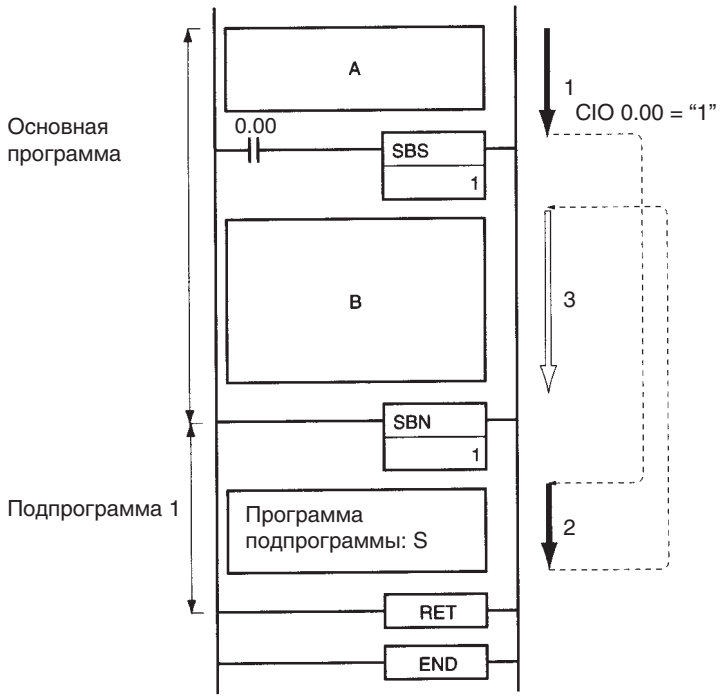
1,2,3...

1. Указанная подпрограмма не определена в текущей задаче.
2. Программа вызывает сама себя.
3. Число уровней вложения подпрограмм превышает 16.
4. Указанная подпрограмма уже выполняется.

Примеры

Пример 1. Последовательность подпрограмм (без вложения).

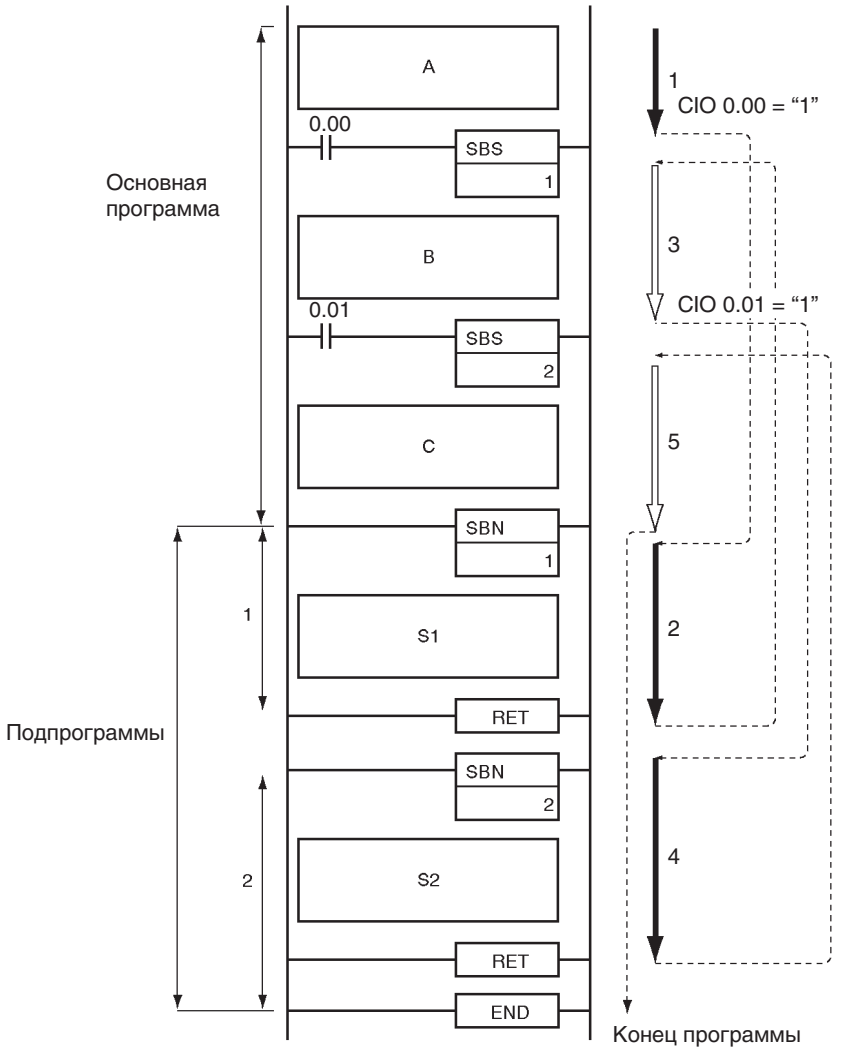
При включенном бите CIO 0.00 выполняется подпрограмма 1, после чего выполняется оставшаяся часть основной программы, начиная с команды, расположенной после SBS(091), и вплоть до команды, расположенной перед SBN(092).



0.00	Порядок выполнения
ВКЛ	A →S →B
ВЫКЛ	A →B

Пример 2. Последовательность подпрограмм (без вложения).

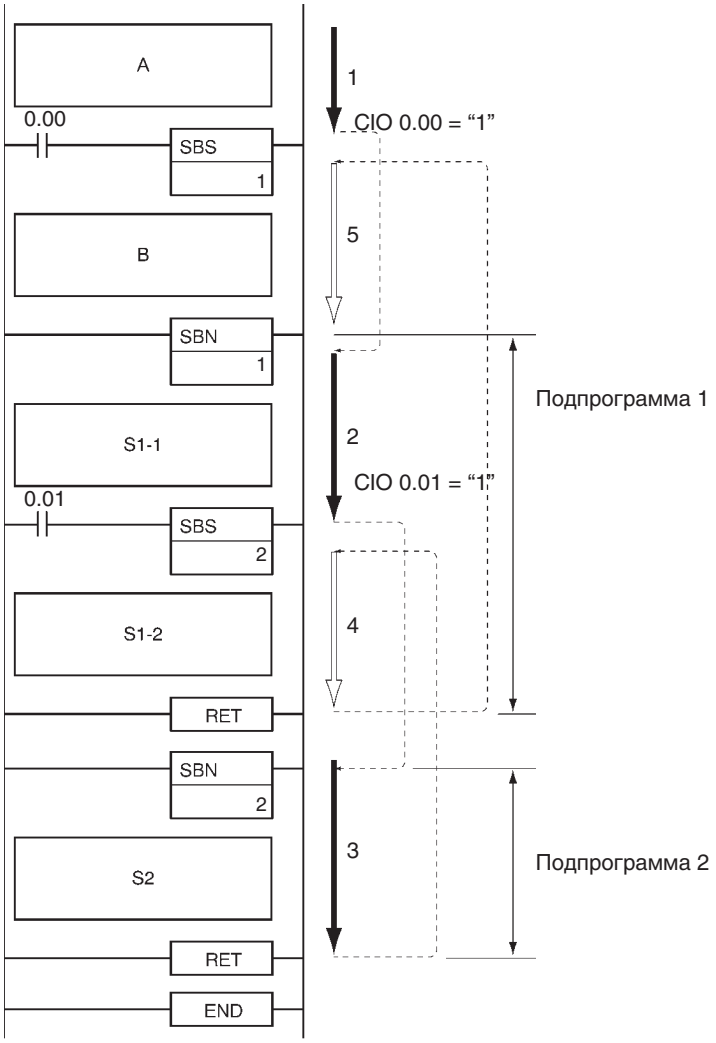
При включенном бите CIO 0.00 выполняется подпрограмма 1, после чего программа возвращается к команде, расположенной после SBS(091) 1. Если включен бит CIO 0.01, выполняется подпрограмма 2, после чего программа возвращается к команде, расположенной после команды SBS(091) 2.



0.00	0.01	Порядок выполнения
ВКЛ	ВКЛ	A→S1→B→S2→C
ВКЛ	ВЫКЛ	A→S1→B→C
ВЫКЛ	ВКЛ	A→B→S2→C
ВЫКЛ	ВЫКЛ	A→B→C

Пример 3. Вложенные подпрограммы.

При включенном бите CIO 0.00 выполняется подпрограмма 1. Если включен бит CIO 0.01, выполняется подпрограмма 2 внутри подпрограммы 1, после чего программа возвращается к команде, расположенной после команды SBS(091) 2. Возобновляется выполнение подпрограммы 1, после чего программа возвращается к команде, расположенной после команды SBS(091) 1.



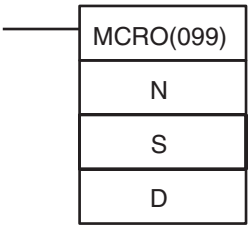
0.00	0.01	Порядок выполнения
ВКЛ	ВКЛ	A→S1–1→S2→S1–2→B
ВКЛ	ВЫКЛ	A→S1–1→S1–2→B
ВЫКЛ	ВКЛ	A→B
ВЫКЛ	ВЫКЛ	A→B

3-18-2 МАКРОС: MCRO(099)

Назначение

Вызов подпрограммы с указанным номером и выполнение ее программы с использованием входных параметров в словах S...S+3 и выходных параметров в словах D...D+3.

Символ РКС



N: Номер подпрограммы

S: Первое слово входных параметров

D: Первое слово выходных параметров

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	MCRO(099)
	Однократное выполнение по положительному фронту	@MCRO(099)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограм-мы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

N: Номер подпрограммы

Указывает номер подпрограммы в диапазоне от 0 до 255 (десятичн).

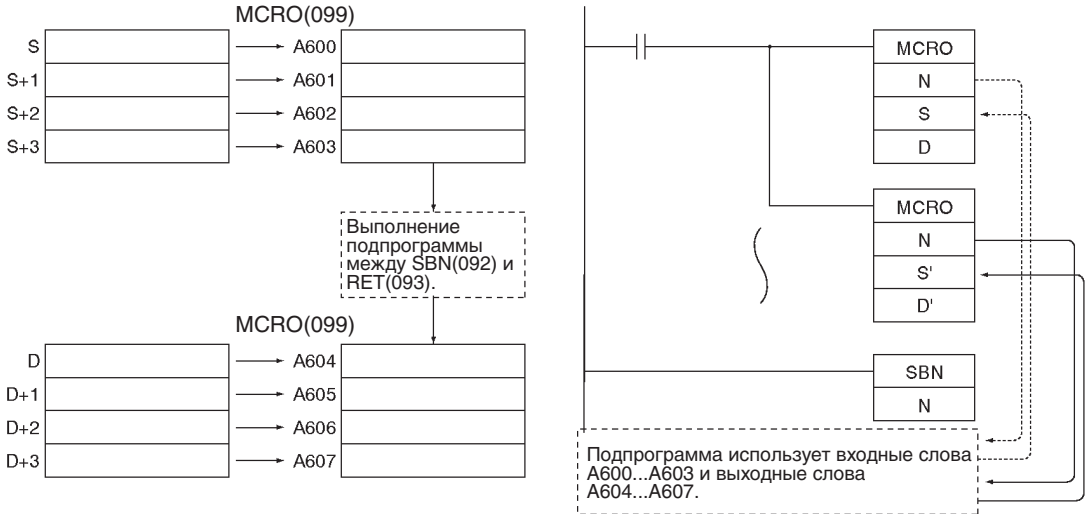
Характеристики операндов

Область	N	S	D
Область CIO	---	CIO 0...CIO 6140	
Рабочая область	---	W0...W508	
Область битов хранения	---	H0...H508	
Область вспомогательных битов	---	A0...A444 A448...A956	A448...A956
Область таймеров	---	T0000...T4092	
Область счетчиков	---	C0000...C4092	
Область DM	---	D0...D32764	
Косвенные адреса DM в двоичном формате	---	@ D0...@ D32767	
Косвенные адреса DM в двоично-десятичном формате (BCD)	---	*D0...*D32767	
Постоянные	0...255 (десятичн.)	---	
Регистры данных	---		
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	---	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

Описание

Команда MCRO(099) вызывает подпрограмму с указанным номером, так же как команда SBS(091). Но в отличие от SBS(091), в команде MCRO(099) предусмотрены операнды S и D, которые позволяют изменять адреса битов и слов в подпрограмме, хотя структура подпрограммы при этом остается неизменной.

При выполнении команды MCRO(099) содержимое операндов S...S+3 копируется в слова A600...A603 (входные слова области макросов) и выполняется указанная подпрограмма. Когда подпрограмма завершается, содержимое слов A604...A607 (выходные слова области макросов) копируется в операнды D...D+3, после чего продолжается выполнение основной программы, начиная с команды, следующей за командой MCRO(099).



Благодаря команде MCRO(099) вместо нескольких (двух и более) подпрограмм с одинаковой структурой, но с разными адресами входных и выходных данных можно создать и использовать всего одну подпрограмму. При выполнении команды MCRO(099) в указанную подпрограмму передаются указанные входные и выходные данные.

Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	Включен, если число уровней вложения превышает 16. Включен, если подпрограмма с указанным номером не существует. Включен, если подпрограмма вызывает сама себя. Включен, если вызванная подпрограмма уже выполняется. Включен, если указанная подпрограмма не определена в текущей задаче. Выключен во всех остальных случаях.

В следующей таблице перечислены сопутствующие слова вспомогательной области.

Название	Адрес	Действие
Входные слова области макросов	A600... A603	При выполнении MCRO(099) содержимое четырех операндов S...S+3 копируется в слова A600...A603. Эти входные слова передаются в подпрограмму.
Выходные слова области макросов	A604... A607	После того как подпрограмма, указанная в MCRO(099), выполнена, содержимое этих выходных слов копируется в операнды D...D+3.

Меры предосторожности

В тексте подпрограммы, вызываемой командой MCRO(099), должны использоваться четыре слова входных данных (слова или биты) A600...A603 и четыре слова выходных данных (слова или биты) A604...A607. Может быть передано не более четырех слов данных.

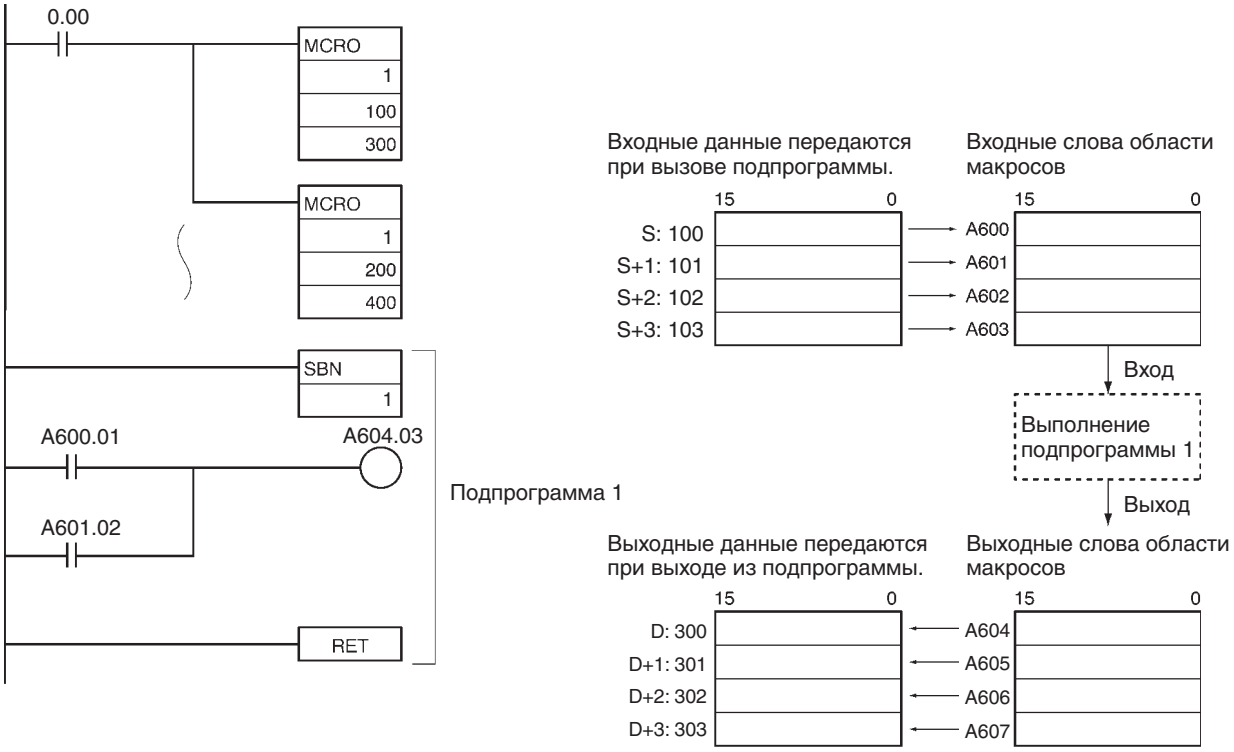
Команда MCRO(099) допускает вложение, но перед вызовом другой подпрограммы требуется сохранять содержимое входных и выходных слов области макросов (A600...A607), так как все команды MCRO(099) пользуются только этими 8 словами.

Пример

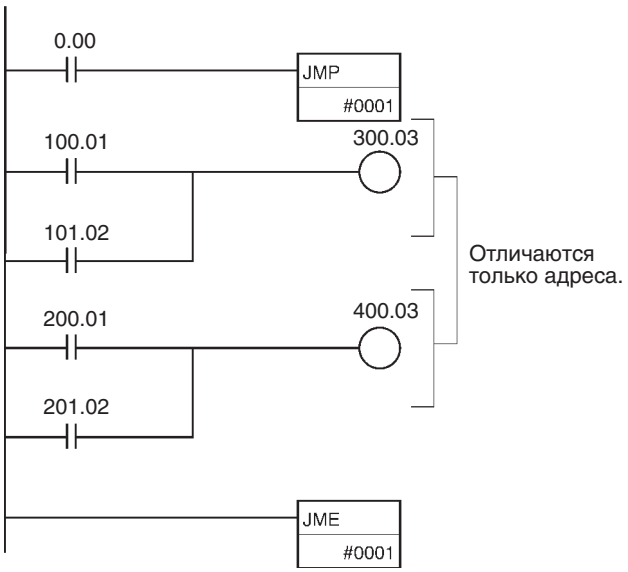
Если в приведенном ниже примере бит CIO 0.00 включен, две команды MCRO(099) передают разные входные и выходные данные в подпрограмму 1.

1,2,3...

1. Первая команда MCRO(099) передает входные данные, содержащиеся в словах CIO 100...CIO 103, и выполняет подпрограмму. По завершении подпрограммы в слова CIO 300...CIO 303 записываются выходные данные.
2. Вторая команда MCRO(099) передает входные данные, содержащиеся в словах CIO 200...CIO 203, и выполняет подпрограмму. По завершении подпрограммы в слова CIO 400...CIO 403 записываются выходные данные.



Вторая команда MCRO(099) выполняет абсолютно те же действия, что и первая команда MCRO(099), но передает в слова A600...A603 содержимое слов CIO 200...CIO 203, а содержимое слов выходных данных A604...A607 передает в слова CIO 400...CIO 403.

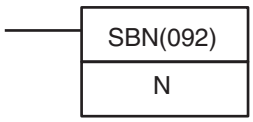


3-18-3 ВХОД В ПОДПРОГРАММУ: SBN(092)

Назначение

Указывает на начало программы подпрограммы с указанным номером подпрограммы. Используется в сочетании с командой RET(093) для указания границ области подпрограммы.

Символ РКС



N: Номер подпрограммы

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	SBN(092)
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	**Области пошаговых программ**	**Подпрограм-мы**	**Задачи обработки прерываний**
Не допускается	Не допускается	OK	OK

Операнды

N: Номер подпрограммы

Указывает номер подпрограммы в диапазоне от 0 до 255 (десятичн).

Характеристики операндов

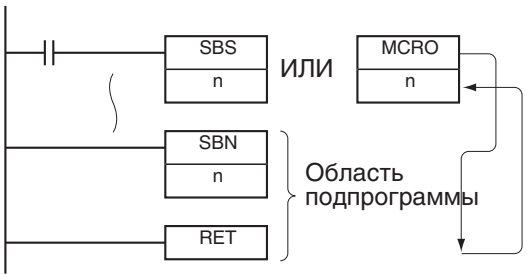
Область	**N**
Область CIO	---
Рабочая область	---
Область битов хранения	---
Область вспомогательных битов	---
Область таймеров	---
Область счетчиков	---
Область DM	---
Косвенные адреса DM в двоичном формате	---

Область	N
Косвенные адреса DM в двоично-десятичном формате (BCD)	---
Постоянные	0...255 (десятичн.)
Регистры данных	---
Регистры указателей	---
Косвенная адресация с помощью регистров указателей	---

Описание

Команда SBN(092) указывает, что после нее начинается подпрограмма с указанным номером подпрограммы. Конец подпрограммы обозначается командой RET(093).

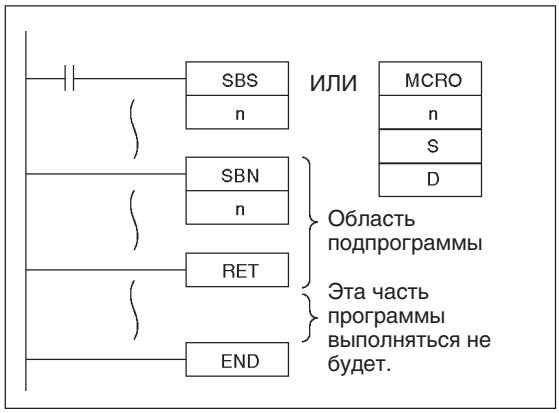
Область программы пользователя, расположенная после самой первой команды SBN(092), является областью подпрограмм. Подпрограммы выполняются, только когда они вызываются командами SBS(091) или MCRO(099).



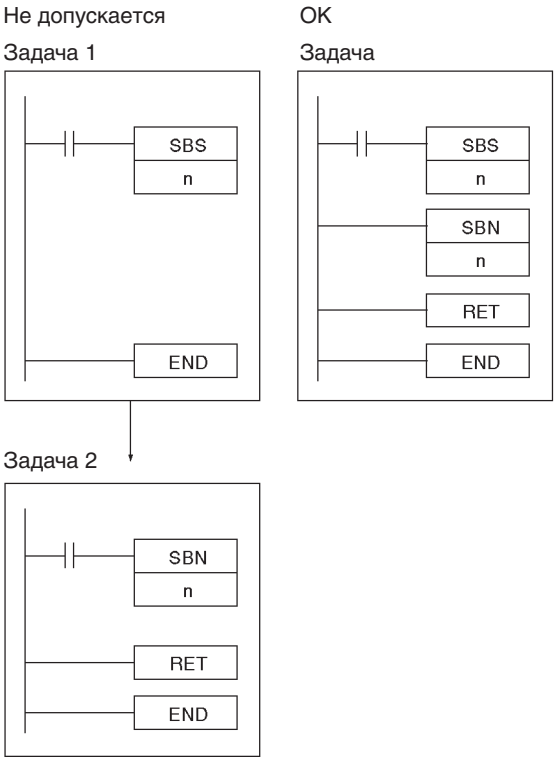
Меры предосторожности

Когда подпрограмма не выполняется, команды обрабатываются как NOP(000).

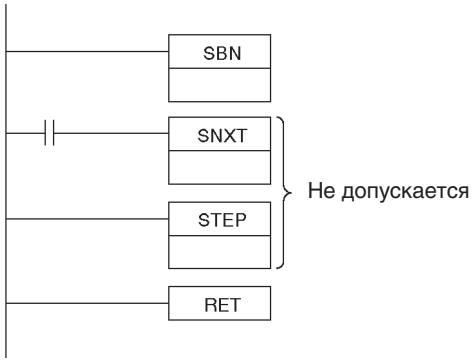
Подпрограммы во всех задачах должны располагаться после текста основной программы, непосредственно перед командой END(001). Части основной программы, расположенные после области подпрограмм перед командой END(001), игнорируются.



Текст подпрограммы должен находиться в той программе (задаче), в которой он вызывается соответствующей командой SBS(091) или MCRO(099). Подпрограмма, созданная в одной задаче, не может быть вызвана из другой задачи. Создание подпрограмм также возможно в задачах обработки прерываний.

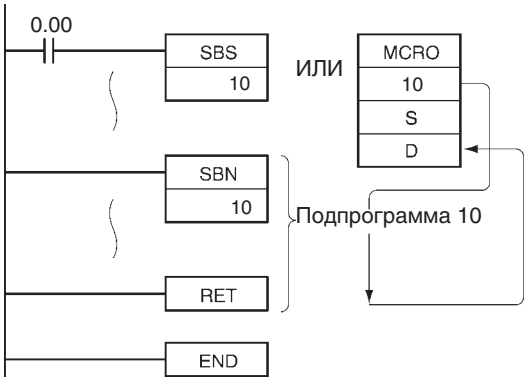


В подпрограммах невозможно использовать команды пошагового выполнения STEP(008) и SNXT(009).



Пример

При включенном бите CIO 0.00 выполняется подпрограмма 10, после чего программа возвращается к команде, расположенной после команды SBS(091) или MCRO(099), вызвавшей подпрограмму.



3-18-4 ВЫХОД ИЗ ПОДПРОГРАММЫ: RET(093)

Назначение

Указывает на конец подпрограммы. Используется в комбинации с командой SBN(092) для указания границ области подпрограммы.

Символ РКС



Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	RET(093)
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограм-мы	Задачи обработки прерываний
Не допускается	Не допускается	OK	OK

Описание

Команда RET(093) указывает на конец подпрограммы, а команда SBN(092) указывает на начало подпрограммы. Более подробно работа подпрограмм описана в разделе *3-18-3 ВХОД В ПОДПРОГРАММУ: SBN(092)* .

Когда выполняемая программа доходит до команды RET(093), она автоматически возвращается к команде, расположенной после команды SBS(091) или MCRO(099), вызвавшей подпрограмму. Если подпрограмма была вызвана командой MCRO(099), перед выходом из подпрограммы в операнды D...D+3 записываются выходные данные, содержащиеся в словах A604...A607.

Меры предосторожности

Когда подпрограмма не выполняется, команды обрабатываются как NOP(000).

Пример

Примеры применения команды RET(093) приведены в разделе *3-18-3 ВХОД В ПОДПРОГРАММУ: SBN(092)*.

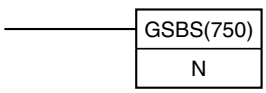
3-18-5 ВЫЗОВ ГЛОБАЛЬНОЙ ПОДПРОГРАММЫ: GSBS(750)

Назначение

Вызов глобальной подпрограммы с указанным номером и выполнение ее программы. Одну и ту же глобальную подпрограмму можно вызывать из разных задач.

Команда GSBS(750) используется в сочетании с командами GSBN(751) и GRET(752) (ВХОД В ГЛОБАЛЬНУЮ ПОДПРОГРАММУ и ВЫХОД ИЗ ГЛОБАЛЬНОЙ ПОДПРОГРАММЫ).

Символ РКС



N: Номер глобальной подпрограммы

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	GSBS(750)
	Однократное выполнение по положительному фронту	@GSBS(750)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограм-мы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

N: Номер глобальной подпрограммы

Указывает номер глобальной подпрограммы в диапазоне от 0 до 255 (десятичн).

Характеристики операндов

Область	N
Область CIO	---
Рабочая область	---
Область битов хранения	---
Область вспомогательных битов	---
Область таймеров	---
Область счетчиков	---
Область DM	---
Косвенные адреса DM в двоичном формате	---
Косвенные адреса DM в двоично-десятичном формате (BCD)	---
Постоянные	0...255 (десятичн.)
Регистры данных	---
Регистры указателей	---
Косвенная адресация с помощью регистров указателей	---

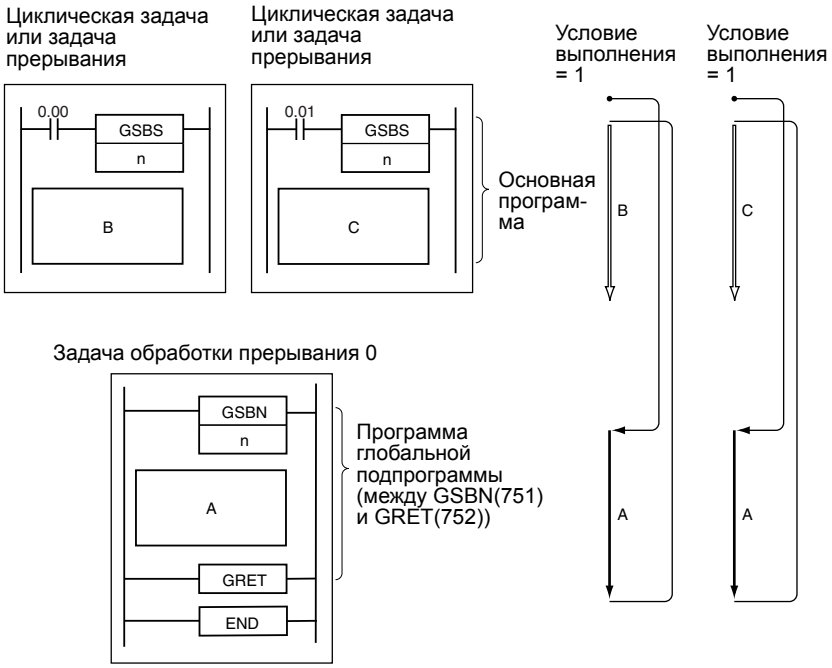
Описание

Команда GSBS(750) вызывает глобальную подпрограмму с указанным номером глобальной подпрограммы. Глобальная подпрограмма — это сегмент программы, заключенный между командами GSBN(751) и GRET(752). По завершении выполнения глобальной подпрограммы возобновляется выполнение основной программы, начиная с команды, расположенной после команды GSBS(750).

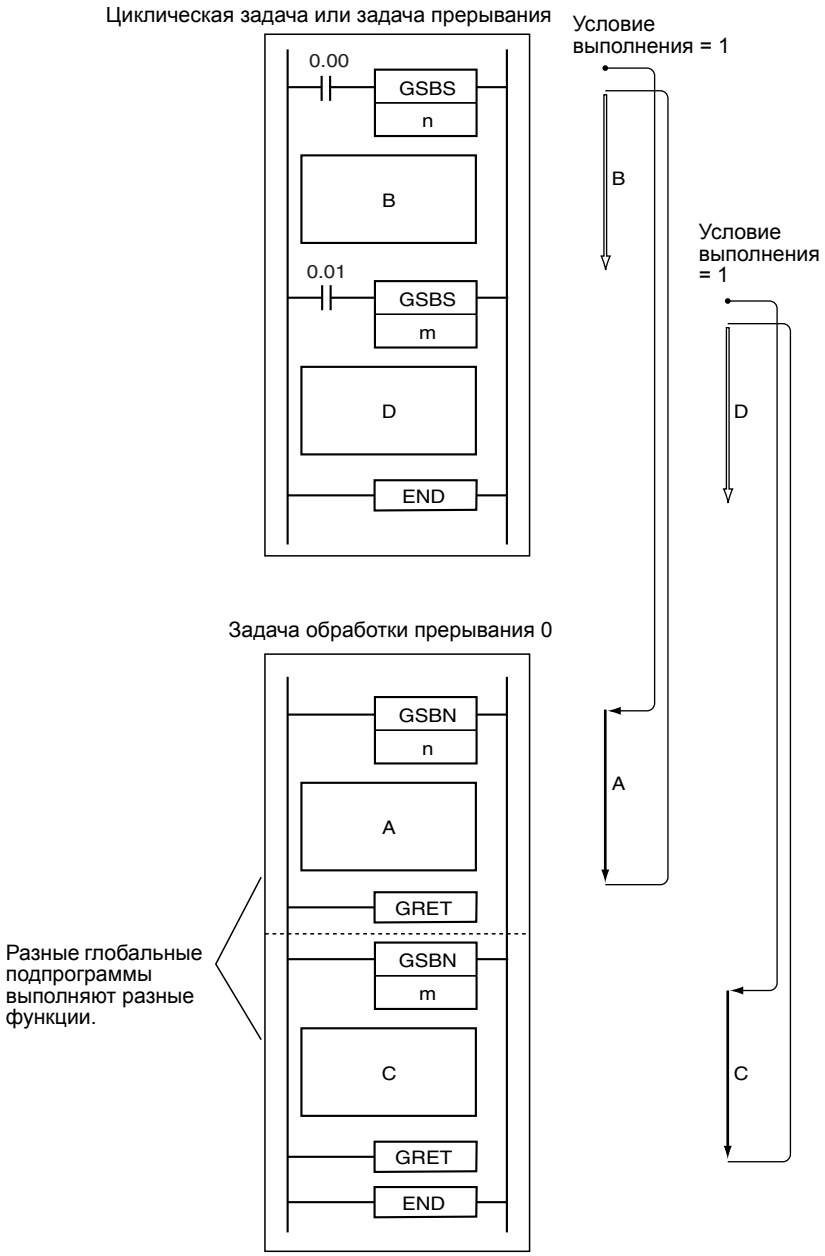
Эту команду можно использовать во множестве разных задач, указывая один и тот же номер глобальной подпрограммы (для вызова этой подпрограммы из разных задач). Таким образом, некоторые стандартные операции, общие для многих задач прикладной программы, можно запрограммировать в виде одной глобальной подпрограммы.

Содержимое глобальной подпрограммы (область между GSBN(751) и GRET(752)) должно находиться в задаче обработки прерывания 0. Если глобальная подпрограмма будет определена в любой другой задаче, при выполнении команды GSBS(750) произойдет ошибка и будет установлен флаг ошибки.

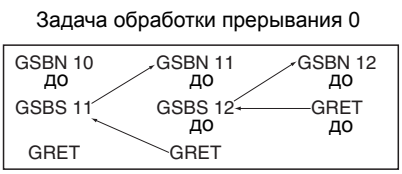
Команда GSBS(750) может использоваться как в циклических задачах (в том числе дополнительных циклических задачах), так и в задачах, выполняемых по прерыванию.



В задаче обработки прерывания 0 можно создать множество подпрограмм (областей GSBN(751) – GRET(752)).



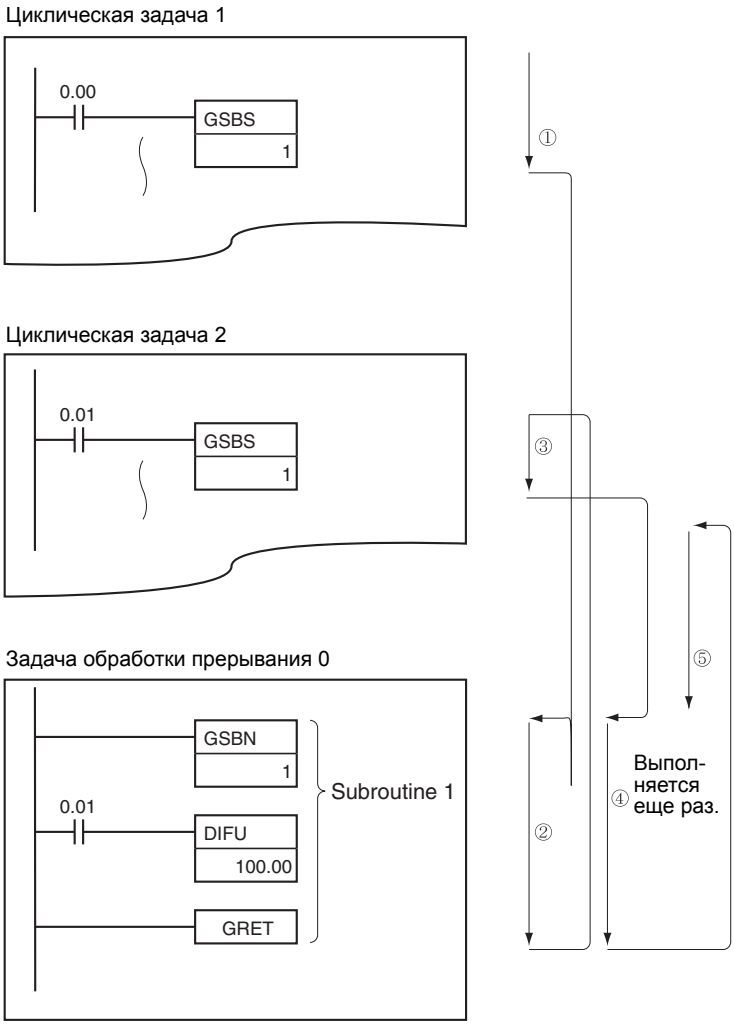
Команда SBS(091) или GSBS(750) может быть применена внутри обычной подпрограммы (область между SBN(092) и RET(093)) или внутри другой глобальной подпрограммы (область между GSBN(751) и GRET(752)), то есть допускается вложение подпрограмм (возможно до 16 уровней вложения).



Глобальные подпрограммы и различение фронтов

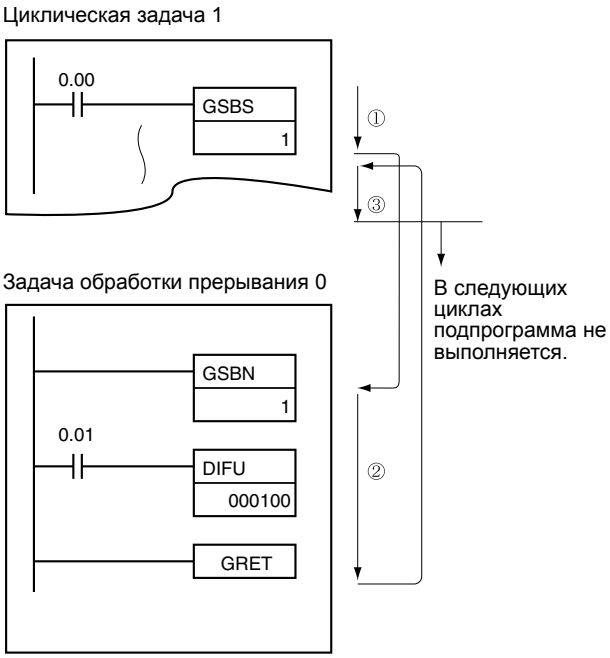
В случае использования в подпрограммах команд с различением фронта (DIFU(013), DIFU(014) или команд с префиксом положительного/ отрицательного фронта) необходимо учитывать описанные ниже особенности.

Если некоторая глобальная подпрограмма вызывается более одного раза в пределах одного цикла, команды с различением фронта условия выполнения в этой подпрограмме работают непредсказуемым образом. В приведенном ниже примере глобальная подпрограмма 0001 выполняется, когда бит CIO 0.00 включен. Бит CIO 100.00 в подпрограмме включается командой DIFU(013), когда бит CIO 0.01 переключается из «0» в «1». Если бит CIO 0.01 включен в том же цикле, глобальная подпрограмма 0001 будет выполнена еще раз, но в этот раз команда DIFU(013) не обнаружит положительный фронт бита CIO 0.01 и бит CIO 100.00 будет выключен.



С другой стороны, выход команды с различением фронта (DIFU(013) или DIFD(014)) останется включенным, если команда будет выполнена и выход будет включен, но во второй раз та же глобальная подпрограмма вызвана не будет.

В приведенном ниже примере глобальная подпрограмма 0001 выполняется, если бит CIO 0.00 включен. Выход CIO 100.00 включается командой DIFU(013) после того, как CIO 0.01 переключается из «0» в «1». Если в следующем цикле бит CIO 0.00 будет выключен, подпрограмма 0001 больше выполнена не будет и выход CIO 100.00 останется включенным.



Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	Включен, если число уровней вложения превышает 16 (в сумме для обычных и глобальных подпрограмм). Включен, если глобальная подпрограмма с указанным номером не существует. Включен, если глобальная подпрограмма вызывает сама себя. Включен, если вызванная глобальная подпрограмма уже выполняется. Включен, если указанная глобальная подпрограмма не определена в задаче обработки прерывания 0. Выключен во всех остальных случаях.

Меры предосторожности

Обе команды, GSBN(751) (ВХОД В ГЛОБАЛЬНУЮ ПОДПРОГРАММУ) и GRET(752) (ВЫХОД ИЗ ГЛОБАЛЬНОЙ ПОДПРОГРАММЫ), должны находиться в задаче обработки прерывания 0. Если глобальная подпрограмма будет определена в любой другой задаче, при выполнении команды GSBS(750) произойдет ошибка и будет установлен флаг ошибки.

С помощью команды SBS(091), предназначенной для вызова обычной подпрограммы, невозможно вызвать глобальную подпрограмму (заключенную между командами GSBN(751) и GRET(752)).

Если команда GSBS(750) находится в сегменте программы, который заблокирован командами IL(002) и ILC(003), команда GSBS(750) не выполняется. По этой причине не допускается использовать блокировки в пределах области глобальной подпрограммы.

Одна и та же глобальная подпрограмма (область между GSBN(751) и GRET(752)) может быть вызвана более одного раза.

При выполнении команды GSBS(750) в перечисленных ниже случаях глобальная подпрограмма не будет вызвана и будет установлен флаг ошибки.

1,2,3...

1. Указанная глобальная подпрограмма не определена.
2. Число уровней вложения подпрограмм (включая обычные и глобальные подпрограммы) превышает 16.
3. Глобальная подпрограмма вызывает сама себя.
4. Указанная глобальная подпрограмма уже выполняется.
5. Указанная глобальная подпрограмма не определена в задаче обработки прерывания 0.

Примеры

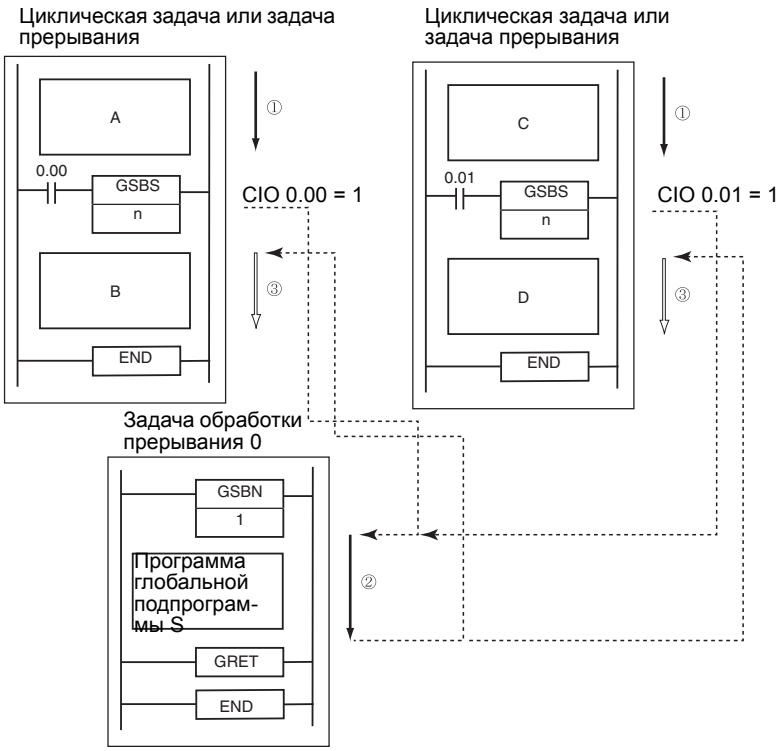
Пример 1

Если в приведенном ниже примере включен бит CIO 0.00, выполняется глобальная подпрограмма под номером 1, после чего программа продолжает выполняться со следующей команды после GSBS(750).

Состояние CIO 0.00	Порядок выполнения программы
ВКЛ	A → S → B
ВЫКЛ	A → B

Если в приведенном ниже примере включен бит CIO 0.01, выполняется глобальная подпрограмма под номером 1, после чего программа продолжает выполняться со следующей команды после GSBS(750).

Состояние CIO 0.01	Порядок выполнения программы
ВКЛ	C → S → D
ВЫКЛ	C → D

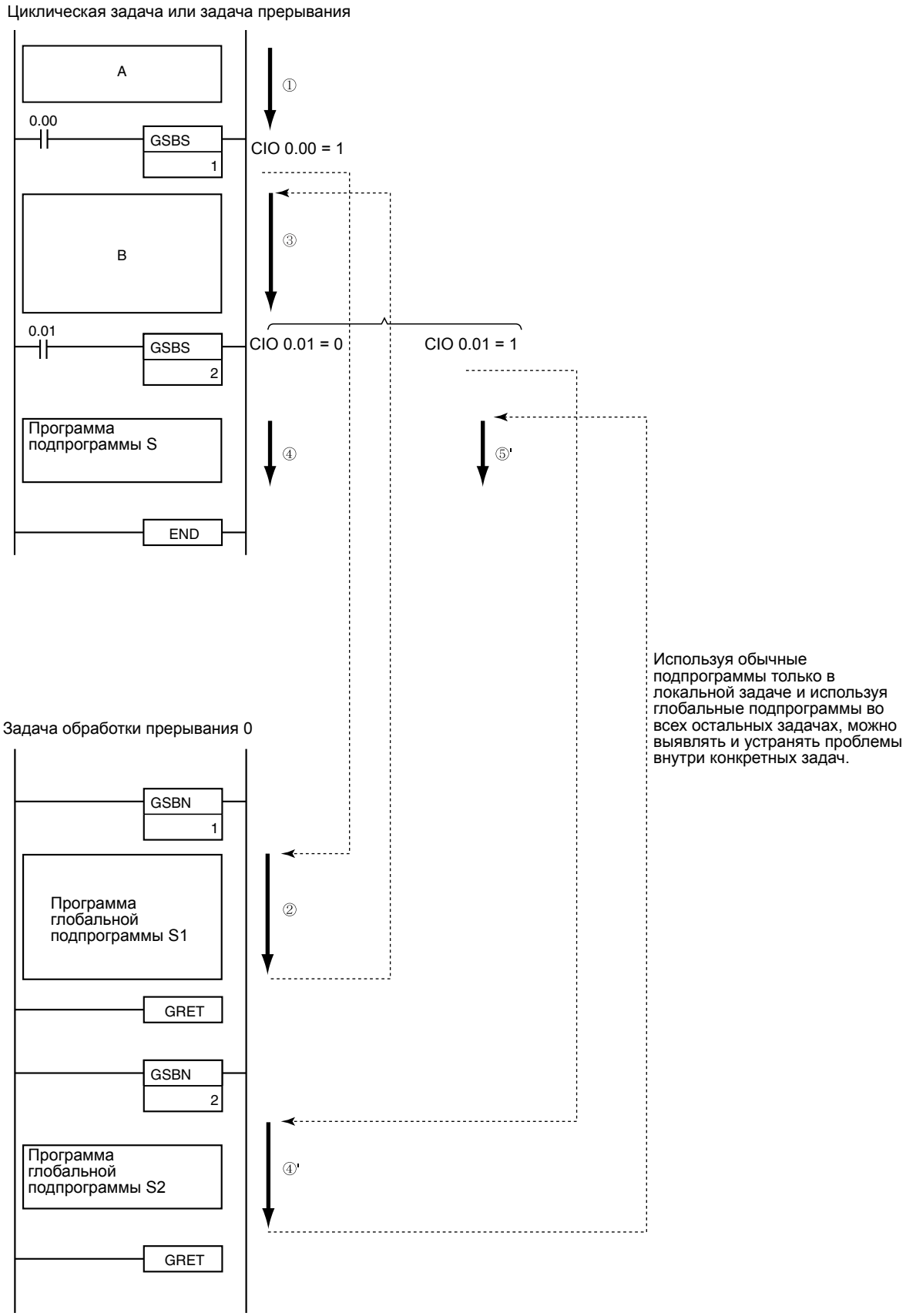


Пример 2

В задаче обработки прерывания 0 может быть запрограммировано две и более глобальных подпрограмм. В этом случае задача обработки прерывания 0 будет фактически состоять из нескольких подзадач и будет использоваться как задача для создания подпрограмм.

Если включен бит CIO 0.00, выполняется программа глобальной подпрограммы 1.

Если включен бит CIO 0.01, выполняется программа глобальной подпрограммы 2.


Циклическая задача или задача прерывания
A
0.00
GSBS
1
CIO 0.00 = 1
①
B
③
0.01
GSBS
2
CIO 0.01 = 0
CIO 0.01 = 1
Программа подпрограммы S
④
⑤'
END
Используя обычные подпрограммы только в локальной задаче и используя глобальные подпрограммы во всех остальных задачах, можно выявлять и устранять проблемы внутри конкретных задач.
Задача обработки прерывания 0
GSBN
1
Программа глобальной подпрограммы S1
②
GRET
GSBN
2
Программа глобальной подпрограммы S2
④'
GRET

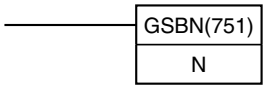
3-18-6 ВХОД В ГЛОБАЛЬНУЮ ПОДПРОГРАММУ: GSBN(751)

Назначение

Указывает на начало программы глобальной подпрограммы с указанным номером подпрограммы. Используется в сочетании с командой GRET(752) для указания границ области глобальной подпрограммы.

Команда GSBN(751) используется в сочетании с командами GSBS(750) и GRET(752) (ВЫЗОВ ГЛОБАЛЬНОЙ ПОДПРОГРАММЫ и ВЫХОД ИЗ ГЛОБАЛЬНОЙ ПОДПРОГРАММЫ).

Символ РКС



N: Номер глобальной подпрограммы

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	GSBN(751)
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	**Области пошаговых программ**	**Подпрограммы**	**Задачи обработки прерываний**
Не допускается	Не допускается	---	OK

Операнды

N: Номер глобальной подпрограммы

Указывает номер глобальной подпрограммы в диапазоне от 0 до 255 (десятичн).

Характеристики операндов

Область	**N**
Область CIO	---
Рабочая область	---
Область битов хранения	---
Область вспомогательных битов	---
Область таймеров	---
Область счетчиков	---
Область DM	---
Косвенные адреса DM в двоичном формате	---
Косвенные адреса DM в двоично-десятичном формате (BCD)	---
Постоянные	0...255 (десятичн.)
Регистры данных	---
Регистры указателей	---
Косвенная адресация с помощью регистров указателей	---

Описание

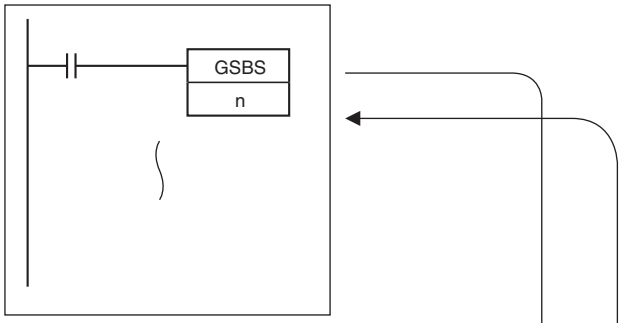
Команда GSBN(751) указывает, что после нее начинается глобальная подпрограмма с указанным номером подпрограммы. Конец подпрограммы обозначается командой GRET(752).

Область программы пользователя, расположенная после самой первой команды GSBN(751), является областью подпрограмм. Глобальная подпрограмма выполняется, только когда она вызывается командой GSBS(750).

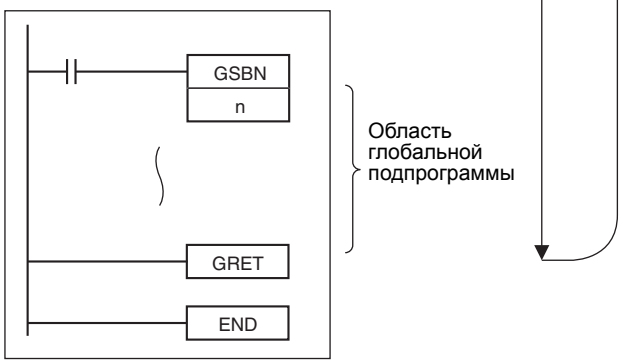
Содержимое глобальной подпрограммы (область между GSBN(751) и GRET(752)) должно находиться в задаче обработки прерывания 0. Если глобальная подпрограмма будет определена в любой другой задаче, при выполнении команды GSBS(750) произойдет ошибка и будет установлен флаг ошибки.

Команда GSBS(750) может быть применена как в циклической задаче (в том числе в дополнительной циклической задаче), так и в задаче обработки прерывания.

Циклическая задача или задача прерывания



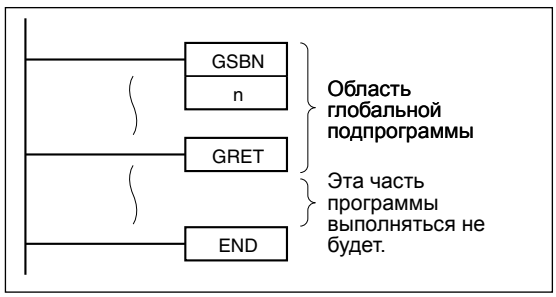
Задача обработки прерывания 0



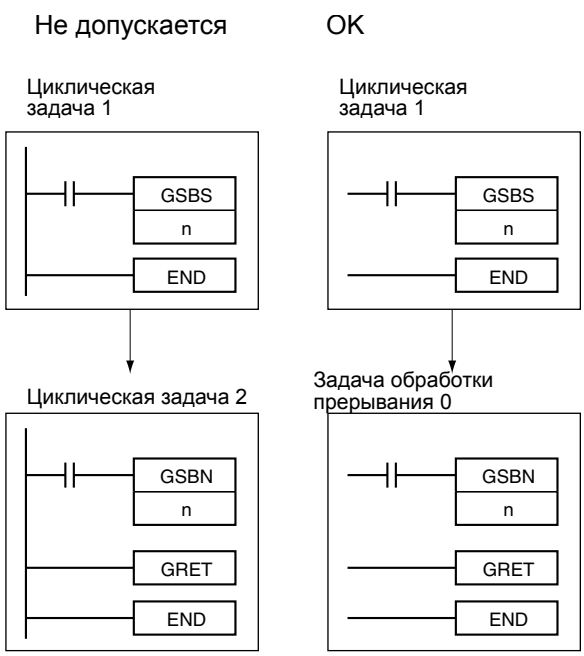
Меры предосторожности

- Когда подпрограмма не выполняется, команды обрабатываются как NOP(000).
- Программу глобальной подпрограммы (область между GSBN(751) и GRET(752)) следует вводить в задаче обработки прерывания 0 непосредственно перед командой END(001). Для использования двух и более глобальных подпрограмм их следует расположить друг за другом в задаче обработки прерывания 0 сразу после текста основной программы. Части основной программы, расположенные после области глобальных подпрограмм перед командой END(001), игнорируются.

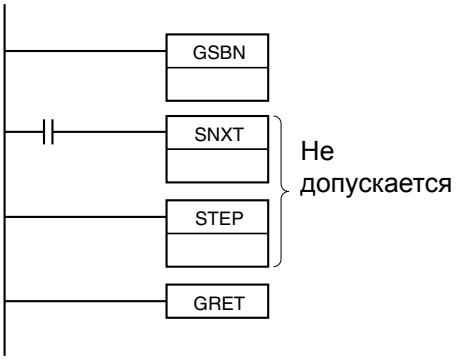
Задача обработки прерывания 1



- Глобальные подпрограммы должны создаваться только в задаче обработки прерывания 0. Если вызванная глобальная подпрограмма находится не в задаче обработки прерывания 0, произойдет ошибка.

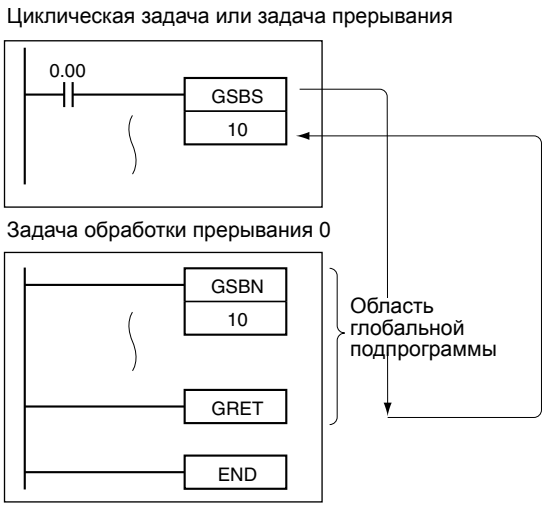


- В глобальных подпрограммах невозможно использовать команды пошагового выполнения STEP(008) и SNXT(009).



Пример

Если в приведенном ниже примере включен бит CIO 0.00, выполняется глобальная подпрограмма под номером 10, после чего программа продолжает выполняться со следующей команды после GSBS(750), вызвавшей эту глобальную подпрограмму.



3-18-7 ВЫХОД ИЗ ГЛОБАЛЬНОЙ ПОДПРОГРАММЫ: GRET(752)

Назначение

Указывает на конец подпрограммы. Используется в комбинации с командой GSBN(751) для указания границ области подпрограммы.

Команда GRET(752) используется в сочетании с командами GSBS(750) и GSBN(751) (ВЫЗОВ ГЛОБАЛЬНОЙ ПОДПРОГРАММЫ и ВХОД В ГЛОБАЛЬНУЮ ПОДПРОГРАММУ).

Символ РКС



Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	GRET(752)
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
Не допускается	Не допускается	Не допускается	OK

Описание

Команда GRET(752) указывает на конец глобальной подпрограммы, а команда GSBN(751) указывает на начало глобальной подпрограммы. Более подробно работа глобальных подпрограмм описана в разделе *3-18-6 ВХОД В ГЛОБАЛЬНУЮ ПОДПРОГРАММУ: GSBN(751)* .

Когда выполняемая программа доходит до команды GRET(752), она автоматически возвращается к команде, расположенной после команды GSBS(750), вызвавшей глобальную подпрограмму.

Меры предосторожности

Когда подпрограмма не выполняется, команды обрабатываются как NOP(000).

Пример

Примеры применения команды GRET(752) приведены в разделе *3-18-6 ВХОД В ГЛОБАЛЬНУЮ ПОДПРОГРАММУ: GSBN(751)*.

3-19 Команды управления прерываниями

В данном разделе описаны команды, предназначенные для управления прерываниями.

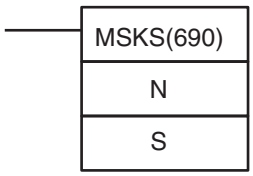
Команда	Мнемоническое обозначение	Код функции	Стр.
УСТАНОВИТЬ МАСКУ ПРЕРЫВАНИЙ	MSKS	690	755
СЧИТАТЬ МАСКУ ПРЕРЫВАНИЙ	MSKR	692	759
ОЧИСТИТЬ ПРЕРЫВАНИЕ	CLI	691	763
ЗАПРЕТИТЬ ПРЕРЫВАНИЯ	DI	693	766
РАЗРЕШИТЬ ПРЕРЫВАНИЯ	EI	694	768

3-19-1 УСТАНОВИТЬ МАСКУ ПРЕРЫВАНИЙ: MSKS(690)

Назначение

Когда ПЛК переходит в режим выполнения (RUN), задачи обработки прерываний от входов и запланированных прерываний изначально отключены (маскированы), а внутренний таймер запланированных прерываний остановлен. С помощью команды MSKS(690) можно снять или установить маску для требуемых прерываний от входов, а также задать интервалы формирования запланированных прерываний.

Символ РКС



N: Идентификатор прерывания

S: Параметры прерывания

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	MSKS(690)
	Однократное выполнение по положительному фронту	@MSKS(690)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

Прерывания от входов

Операнд N указывает номер прерывания по входу и функцию команды MSKS(690), а операнд S указывает выполняемые действия.

Операнд		Содержание	
		Выбор формирования прерывания по включению или выключению	**Маскирование или размаскирование прерывания**
N	Прерывание от входа 0 (задача обр. прерывания 140)	110 (или 10)	100 (или 6)
	Прерывание от входа 1 (задача обр. прерывания 141)	111 (или 11)	101 (или 7)
	Прерывание от входа 2 (задача обр. прерывания 142)	112 (или 12)	102 (или 8)
	Прерывание от входа 3 (задача обр. прерывания 143)	113 (или 13)	103 (или 9)
	Прерывание от входа 4 (задача обр. прерывания 144)	114	104
	Прерывание от входа 5 (задача обр. прерывания 145)	115	105
	Прерывание от входа 6 (задача обр. прерывания 146)	116	106
	Прерывание от входа 7 (задача обр. прерывания 147)	117	107
S		0000 hex: распознавать включение; 0001 hex: распознавать выключение.	0000 hex: размаскировать, прямой режим; 0001 hex: маскировать; 0002 hex: размаскировать, режим счетчика, начать обратный счет; 0003 hex: размаскировать, режим счетчика, начать прямой счет.

Примечание. Прерывания от входов 6 и 7 невозможно использовать в модулях ЦПУ CP1H Y и CP1L. Прерывания от входов 4, 5, 6 и 7 невозможно использовать в модулях ЦПУ CP1L с 14 входами/выходами.

Запланированные прерывания

Операнд N указывает номер запланированного прерывания и способ запуска, а операнд S содержит значение интервала формирования прерываний.

Операнд		Содержание
N	Запланированное прерывание 0 (задача обработки прерывания 2)	14 hex: запуск со сбросом (сброс внутреннего таймера и запуск отсчета времени); 4 hex: запуск без сброса (время до первого прерывания задается отдельно с помощью CLI(691)).
S		0000 hex: запретить запланированные прерывания и остановить внутренний таймер.

Операнд		Содержание
Параметр в настройках ПЛК (интервал формирования прерываний)	10 мс	Интервал запланированных прерываний: 0001...270F hex (от 10 до 99 990 мс)
	1 мс	Интервал запланированных прерываний: 0001...270F hex (1...9 999 мс)
	0,1 мс	Интервал запланированных прерываний: 0005...270F hex (5...999,9 мс) **Примечание.** Если будет задано значение 0001...0004 hex, произойдет ошибка.

Характеристики операндов

Область	N	S
Область CIO	---	CIO 0...CIO 6143
Рабочая область	---	W0...W511
Область битов хранения	---	H0...H511
Область вспомогательных битов	---	A0...A447 A448...A959
Область таймеров	---	T0000...T4095
Область счетчиков	---	C0000...C4095
Область DM	---	D0...D32767
Косвенные адреса DM в двоичном формате	---	@ D0...@ 32767
Косвенные адреса DM в двоично-десятичном формате (BCD)	---	*D0...*D32767
Постоянные	Только указанные значения	
Регистры данных	---	DR0...DR15
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	---	,IR0...,IR15 -2048...+2047 ,IR0... -2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15

Описание

Команда MSKS(690) управляет работой прерываний от входов и запланированных прерываний. Значение операнда N идентифицирует прерывание.

Прерывания от входов: N = 100...107, 110...117 или 6...13

- В зависимости от значений операндов, команда MSKS(690) маскирует или размаскирует указанное прерывание и устанавливает направление переключения (включение или выключение) входного сигнала, при котором формируется прерывание. Если направление в команде не будет указано, прерывание будет формироваться по включению входа прерывания.
- При снятии маски с прерывания указывается либо прямой режим, либо режим счетчика (с прямым или обратных счетом). Дополнительную информацию см. в разделе *Функции обработки прерываний* в *Руководстве по эксплуатации ЦПУ CP1H или CP1L*.
- При размаскировании прерывания или при изменении направления переключения сигнала (ВКЛ/ВЫКЛ), приводящего к формированию прерывания, информация о любом прежнем состоянии этого прерывания сбрасывается.

Запланированные прерывания: N = 4 или 14

- Команда MSKS(690) указывает интервал формирования прерываний и запускает внутренний таймер. Интервал формирования прерываний также зависит от настройки параметра *Интервал формирования прерываний* в настройках ПЛК.
- В операндах команды MSKS(690) также указывается, должен ли сбрасываться внутренний таймер при запуске.
- Если внутренний таймер сбрасывается, отсчет времени начинается с момента сброса таймера и первое запланированное прерывание формируется, когда с момента выполнения команды MSKS(690) проходит время, указанное в S.
- Если внутренний таймер не сбрасывается, он продолжает работать с текущим временем, а время до формирования первого прерывания указывается отдельно с помощью команды CLI(691). Если время до первого прерывания не будет указано с помощью CLI(691), оно останется неопределенным, но формирование запланированных прерываний начнется не позже, чем по истечении двух интервалов формирования запланированных прерываний.

Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	Включен, если N выходит за указанный диапазон. Для прерываний от входов: включен, если значение S выходит за допустимый диапазон 0000...0003. Для запланированных прерываний: включен, если значение S выходит за допустимый диапазон 0000...270F hex (когда интервал формирования задается с шагом в 10 или 1 мс) или 0005...270F hex (когда интервал задается с шагом 0,1 мс). Выключен во всех остальных случаях.
Флаг равенства	=	ВЫКЛ
Флаг отрицательного значения	N	ВЫКЛ

В следующей таблице перечислены сопутствующие флаги вспомогательной области.

Название	Адрес	Действие
Флаг ошибки задачи прерывания	A402.13	Включен в следующем случае: В задаче обработки прерывания была выполнена команда IORF(097), при этом не было отключено циклическое обновление для спец. модулей ввода/вывода.
Флаг причины ошибки задачи прерывания	A426.15	Указывает, какая именно ошибка задачи обработки прерывания произошла: 1 или 2.
Номер модуля, вызвавшего ошибку задачи обработки прерывания	A426.00... A426.11	Указывает номер специального модуля ввода/вывода, вызвавшего ошибку конфликта обновления входов/выходов.

Меры предосторожности

Позаботьтесь, чтобы период формирования прерываний был больше времени, необходимого для выполнения задачи обработки запланированного прерывания.

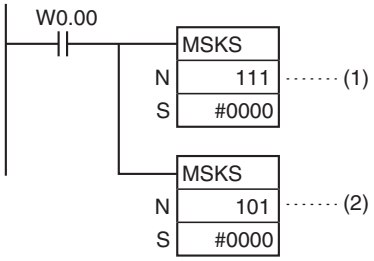
Если в задаче обработки прерывания выполняется команда IORF(097) с целью обновления входов/выходов специального модуля ввода/вывода, для этого специального модуля ввода/вывода в настройках ПЛК должно быть отключено циклическое обновление. Если циклическое обновление для специального модуля ввода/вывода не будет отключено, команда IORF(097) может оказаться выполненной во время циклического обновления, что вызовет некритическую ошибку дублированного обновления и установку флага ошибки задачи прерывания (A402.13).

Слово A440 содержит максимальное время выполнения задачи обработки прерывания, а младший байт слова A441 содержит номер задачи обработки прерывания, на выполнение которой потребовалось больше всего времени.

Примеры

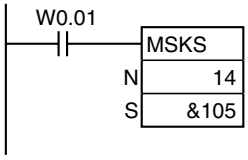
Разрешение обработки прерываний от входов в прямом режиме

При включении бита W0.00 первая команда MSKS(690) (1) устанавливает, что прерывание по входу 1 (CIO 0.01 для CP1H и CIO 0.05 для CP1L) должно формироваться при включении входа прерывания, а вторая команда MSKS(690) (2) снимает маску с данного прерывания.



Запуск запланированных прерываний со сбросом таймера

При включении бита W0.01 команда MSKS(690) устанавливает интервал формирования прерываний для запланированного прерывания 0 равным 10,5 мс (предполагается, что в настройках ПЛК выбран шаг установки 0,1 мс), сбрасывает внутренний таймер и запускает внутренний таймер.

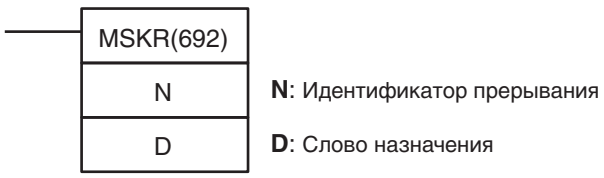


3-19-2 СЧИТАТЬ МАСКУ ПРЕРЫВАНИЙ: MSKR(692)

Назначение

Чтение текущих значений параметров обработки прерываний, заданных командой MSKS(690).

Символ РКС



Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	MSKR(692)
	Однократное выполнение по положительному фронту	@MSKR(692)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

Прерывания от входов

Операнд N указывает номер входа прерывания (то же самое, что «номер прерывания от входа») и данные, которые должны быть прочитаны. Операнд D указывает адрес области памяти, по которому должны быть записаны прочитанные данные.

Операнд		Содержание	
		Чтение направления переключения, вызывающего прерывание (ВКЛ или ВЫКЛ)	**Чтение состояния маски**
N	Прерывание от входа 0 (задача обр. прерывания 140)	110 (или 10)	100 (или 6)
	Прерывание от входа 1 (задача обр. прерывания 141)	111 (или 11)	101 (или 7)
	Прерывание от входа 2 (задача обр. прерывания 142)	112 (или 12)	102 (или 8)
	Прерывание от входа 3 (задача обр. прерывания 143)	113 (или 13)	103 (или 9)
	Прерывание от входа 4 (задача обр. прерывания 144)	114	104
	Прерывание от входа 5 (задача обр. прерывания 145)	115	105
	Прерывание от входа 6 (задача обр. прерывания 146)	116	106
	Прерывание от входа 7 (задача обр. прерывания 147)	117	107
D		0000 hex: распознавать включение; 0001 hex: распознавать выключение.	0000 hex: размаскировать, прямой режим; 0001 hex: маскировать; 0002 hex: размаскировать, режим счетчика, начать обратный счет; 0003 hex: размаскировать, режим счетчика, начать прямой счет.

Примечание. Прерывания от входов 6 и 7 невозможно использовать в модулях ЦПУ CP1H Y и CP1L. Прерывания от входов 4, 5, 6 и 7 невозможно использовать в модулях ЦПУ CP1L с 14 входами/выходами.

Запланированные прерывания

Операнд N указывает номер запланированного прерывания и данные, которые должны быть прочитаны. Операнд D указывает адрес области памяти, по которому должны быть записаны прочитанные данные.

Операнд			Содержание	
			Чтение интервала формирования прерываний	**Чтение текущего значения внутреннего таймера (т. е. времени, прошедшего с момента обработки первого прерывания или предыдущего прерывания)**
N			4	14
D			0000 hex: запланированные прерывания запрещены.	0000...270F hex (1...9999): текущее значение внутреннего таймера
			0001...270F hex (1...9999): интервал запланированных прерываний	
	Параметр в настройках ПЛК (интервал формирования прерываний)	10 мс	10...99990 мс	0...99990 мс
		1 мс	1...9999 мс	0...9999 мс
		0,1 мс	0,1...999,9 мс	0...999,9 мс

Примечание. Если обработка запланированных прерываний была запрещена, может быть прочитано значение истекшего времени, достигнутое к моменту остановки внутреннего таймера запланированных прерываний. Если запуск запланированных прерываний не производился вообще, текущее значение будет содержать 0.

Характеристики операндов

Область	N	D
Область CIO	---	CIO 0...CIO 6143
Рабочая область	---	W0...W511
Область битов хранения	---	H0...H511
Область вспомогательных битов	---	A448...A959
Область таймеров	---	T0000...T4095
Область счетчиков	---	C0000...C4095
Область DM	---	D0...D32767
Косвенные адреса DM в двоичном формате	---	@ D0...@ D32767
Косвенные адреса DM в двоично-десятичном формате (BCD)	---	*D0...*D32767
Постоянные	Только указанные значения	---
Регистры данных	---	DR0...DR15

Область	N	D
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	---	,IR0...,IR15 -2048...+2047 ,IR0... -2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15

Описание

Значение операнда N идентифицирует прерывание.

Прерывания от входов: N = 100...107, 110...117 или 6...13

В операнд D записывается состояние маски или выбранное направление переключения (ВКЛ или ВЫКЛ) для прерывания, указанного в N.

Запланированные прерывания: N = 4 или 14

В операнд D записывается шестнадцатеричное значение интервала формирования прерываний (заданное значение) или текущее значение внутреннего таймера для запланированного прерывания, указанного в N. Единицы, в которых отсчитывается интервал формирования прерываний, указываются в настройках ПЛК параметром *Интервал формирования прерываний*.

Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	Включен, если N выходит за допустимых диапазон 0...5 (0...15 для CJ1M). Выключен во всех остальных случаях.

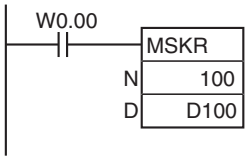
Меры предосторожности

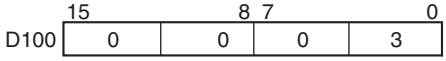
Команда MSKR(692) может быть выполнена как в основной программе, так и в любой задаче обработки прерываний.

Примеры

Прерывания от входов

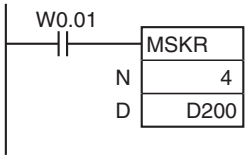
При включении бита W0.00 в слово D100 записывается состояние маски прерывания от входа 0 (CIO 0.00). Полученное в данном примере значение 0003 говорит о том, что прерывание размаскировано и для него выбран режим счетчика прямого счета.

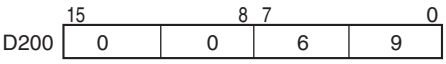




Запланированные прерывания

При включении бита W0.01 в слово D200 записывается интервал формирования прерываний. Полученное в данном примере значение 0069 говорит о том, что установлен интервал 10,5 мс (0069 hex = 105 десятичн.) при условии, что в настройках ПЛК выбран шаг установки интервала запланированных прерываний 0,1 мс.



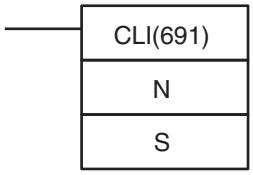


3-19-3 ОЧИСТИТЬ ПРЕРЫВАНИЕ: CLI(691)

Назначение

Сброс или сохранение зарегистрированных прерываний (для прерываний от входов и прерываний от скоростных счетчиков) или установка времени до первого запланированного прерывания (для запланированных прерываний).

Символ РКС



N: Идентификатор прерывания

S: Параметры прерывания

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	CLI(691)
	Однократное выполнение по положительному фронту	@CLI(691)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

Прерывания от входов

Операнд N указывает номер входа прерывания (то же, что и «номер прерывания от входа»), а операнд S указывает выполняемое действие.

Операнд	Содержание
N	Номер входа прерывания 100 или 6: прерывание от входа 0 (задача обр. прерывания 140); 101 или 7: прерывание от входа 1 (задача обр. прерывания 141); 102 или 8: прерывание от входа 2 (задача обр. прерывания 142); 103 или 9: прерывание от входа 3 (задача обр. прерывания 143); 104: прерывание от входа 4 (задача обр. прерывания 144); 105: прерывание от входа 5 (задача обр. прерывания 145); 106: прерывание от входа 6 (задача обр. прерывания 146); 107: прерывание от входа 7 (задача обр. прерывания 147).
S	Выбор сброса зарегистрированного прерывания 0000 hex: зарегистрированное прерывание сохраняется; 0001 hex: зарегистрированное прерывание сбрасывается.

Примечание. Прерывания от входов 6 и 7 невозможно использовать в модулях ЦПУ CP1H Y и CP1L. Прерывания от входов 4, 5, 6 и 7 невозможно использовать в модулях ЦПУ CP1L с 14 входами/выходами.

Запланированные прерывания

Операнд N указывает номер запланированного прерывания, а операнд S указывает время до поступления первого запланированного прерывания.

Операнд	Содержание
N	Указывает номер запланированного прерывания. 4: запланированное прерывание 0 (задача обработки прерывания 2)
S	0000...270F hex: Время до первого запланированного прерывания (0...9999) **Примечание.** Шаг установки интервала запланированных прерываний можно задать в настройках ПЛК: 10 мс / 1,0 мс / 0,1 мс.

Прерывания от скоростных счетчиков

Операнд N указывает номер прерывания от скоростного счетчика, а операнд S указывает выполняемое действие.

Операнд	Содержание
N	Номер прерывания от скоростного счетчика 10: вход скоростного счетчика 0; 11: вход скоростного счетчика 1; 12: вход скоростного счетчика 2; 13: вход скоростного счетчика 3.
S	Выбор сброса зарегистрированного прерывания 0000 hex: зарегистрированное прерывание сохраняется; 0001 hex: зарегистрированное прерывание сбрасывается.

Характеристики операндов

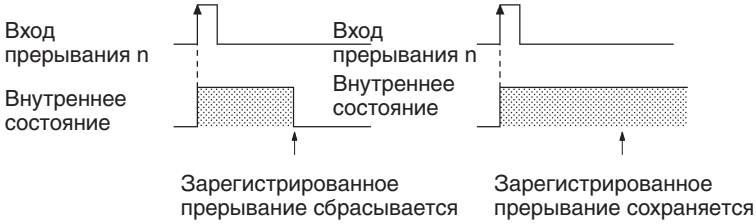
Область	N	S
Область CIO	---	CIO 0...CIO 6143
Рабочая область	---	W0...W511
Область битов хранения	---	H0...H511
Область вспомогательных битов	---	A0...A959
Область таймеров	---	T0000...T4095
Область счетчиков	---	C0000...C4095
Область DM	---	D0...D32767
Косвенные адреса DM в двоичном формате	---	@ D0...@ D32767
Косвенные адреса DM в двоично-десятичном формате (BCD)	---	*D0...*D32767
Постоянные	---	DR0...DR15
Регистры данных	Только указанные значения	
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	---	,IR0...,IR15 -2048...+2047 ,IR0... -2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15

Описание

В зависимости от значения операнда N, команда CLI(691) либо сбрасывает указанные зарегистрированные прерывания от входов или прерывания от скоростных счетчиков, либо задает время до выполнения первого запланированного прерывания.

Прерывания от входов: N = 100...107 или 6...9

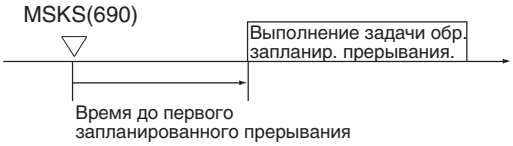
В зависимости от значения операнда S, зарегистрированные прерывания, поступившие от входа прерывания, указанного операндом N, либо сбрасываются, либо сохраняются.



Если во время обработки прерывания от входа с одним номером поступает прерывание от входа с другим номером, номер этого прерывания запоминается во внутренней памяти. Зарегистрированные прерывания от входов обрабатываются позже в порядке очередности (фактически, в порядке возрастания номеров). С помощью команды CLI(691) эти зарегистрированные прерывания можно сбросить, прежде чем они будут выполнены.

Запланированные прерывания: N = 4

Если операнд N = 4, операнд S указывает время, по истечении которого после выполнения MSKS(690) будет сформировано первое запланированное прерывание.



Прерывания от скоростных счетчиков: N = 10...13

В зависимости от значения операнда S, зарегистрированное прерывание от скоростного счетчика, номер которого указан в N, либо сбрасывается, либо сохраняется.

Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	Включен, если N выходит за указанный диапазон. Включен, если S не содержит 0000 или 0001 hex (только для прерываний от скоростных счетчиков и прерываний от входов). Включен, если S выходит за допустимый диапазон 0000...270F hex (для запланированных прерываний). Выключен во всех остальных случаях.

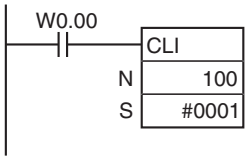
Меры предосторожности

Если прерывание от некоторого входа уже зарегистрировано (и еще не обработано), любые другие прерывания от этого входа игнорируются. Также прерывание от входа игнорируется, если во время его поступления уже выполняется задача обработки прерывания от этого же входа.

Примеры

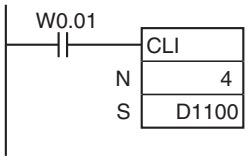
Прерывания от входов

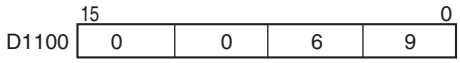
При включении бита W0.00 команда CLI(691) сбрасывает все зарегистрированные прерывания, поступившие от входа прерывания 0.



Запланированные прерывания

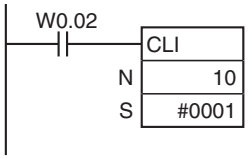
При включении бита W0.01 команда CLI(691) задает время до формирования первого запланированного прерывания равным 10,5 мс (0069 hex = 105 десятичн.) при условии, что в настройках ПЛК выбран шаг установки интервала запланированных прерываний 0,1 мс.





Прерывания от скоростных счетчиков

При включении бита W0.02 команда CLI(691) сбрасывает все зарегистрированные прерывания, поступившие от скоростного счетчика прерываний 0.

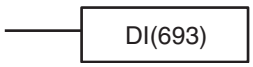


3-19-4 ЗАПРЕТИТЬ ПРЕРЫВАНИЯ: DI(693)

Назначение

Запрет выполнения любых (т. е. абсолютно всех) задач обработки прерываний.

Символ РКС



Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	DI(693)
	Однократное выполнение по положительному фронту	@DI(693)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	Не допускается

Описание

Команда DI(693) выполняется в основной программе с целью временного запрета выполнения любых задач обработки прерываний (прерываний от входов, запланированных прерываний, прерываний от скоростных счетчиков и внешних прерываний).

Все задачи обработки прерываний будут запрещены, пока их выполнение снова не будет разрешено путем выполнения команды EI(694).

Флаги

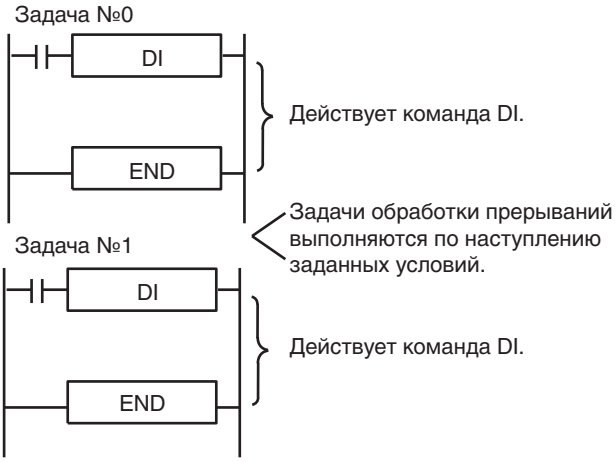
Название	Обозначение	Действие
Флаг ошибки	ER	Включен, если команда DI(693) выполняется в задаче обработки прерывания. Выключен во всех остальных случаях.

Меры предосторожности

Выполнение всех задач обработки прерываний будет запрещено до тех пор, пока не будет выполнена команда EI(694).

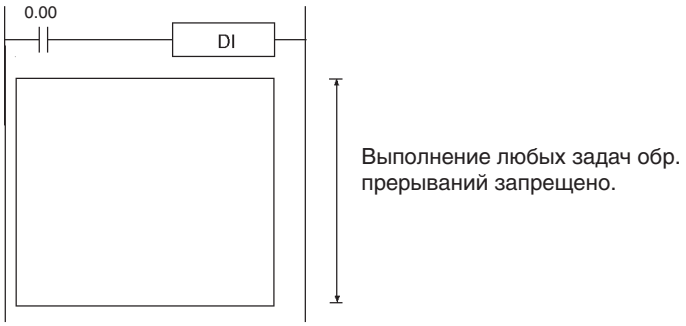
Команду DI(693) невозможно выполнить из задачи обработки прерывания.

Действие каждой команды DI(693) распространяется только на одну циклическую задачу. Чтобы запретить обработку прерываний в нескольких циклических задачах, команду DI(693) следует предусмотреть в каждой такой циклической задаче. Любые прерывания, поступившие во время выполнения циклической задачи, в которой обработка прерываний была запрещена, будут обработаны после завершения этой циклической задачи, если только зарегистрированные прерывания не будут сброшены командой CLI(691). Сказанное демонстрирует следующий рисунок.



Примеры

При включенном бите CIO 0.00 команда DI(693) запрещает выполнение всех задач обработки прерываний.

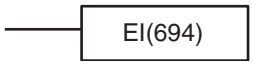


3-19-5 РАЗРЕШИТЬ ПРЕРЫВАНИЯ: EI(694)

Назначение

Разрешение выполнения любых (т. е. абсолютно всех) задач обработки прерываний, ранее запрещенных командой DI(693).

Символ РКС



Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле с нормально включенным условием выполнения**	EI(694)
	Однократное выполнение по положительному фронту	Не предусмотрено
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	Не допускается

Описание

Команда EI(694) выполняется в основной программе с целью разрешения выполнения всех задач обработки прерываний, которые были запрещены командой DI(693). Команда DI(693) запрещает обработку любых прерываний (прерываний от входов, запланированных прерываний, прерываний от скоростных счетчиков и внешних прерываний).

Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	Включен, если команда EI(694) выполняется из задачи обработки прерывания. Выключен во всех остальных случаях.

Меры предосторожности

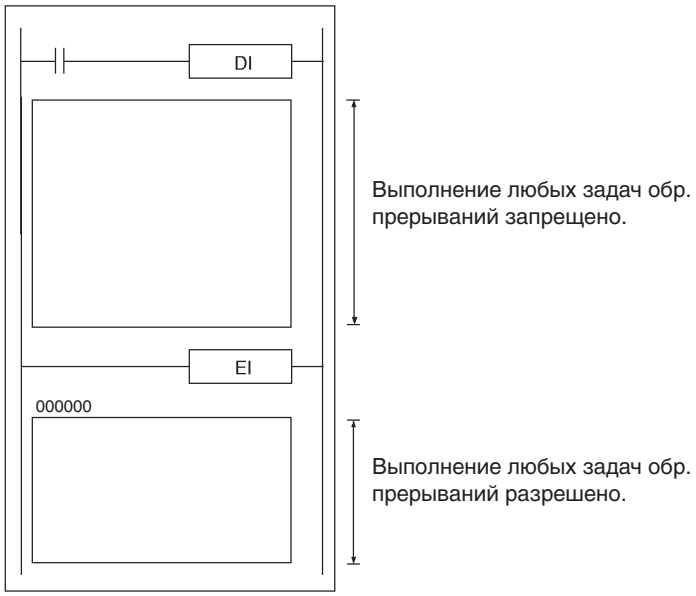
Для команды EI(694) не требуется условие выполнения. Она всегда выполняется с включенным условием выполнения.

Команда EI(694) не способна размаскировать прерывания от входов, которые не были размаскированы командой MSKS(690). Она также не позволяет задать интервал формирования запланированных прерываний, если он не был задан командой MSKS(690).

Команду EI(694) невозможно выполнить в задаче обработки прерывания.

Примеры

В приведенном ниже примере команда EI(694) разрешает обработку любых прерываний, ранее запрещенную командой DI(693).



3-20 Команды управления скоростными счетчиками и импульсными выходами

В данном разделе описаны команды, предназначенные для управления работой скоростных счетчиков и импульсных выходов.

Команда	Мнемоническое обозначение	Код функции	Стр.
УПРАВЛЕНИЕ РЕЖИМОМ	INI	880	770
ЧТЕНИЕ ТЕКУЩ. ЗНАЧ. СКОРОСТНОГО СЧЕТЧИКА	PRV	881	775
ПРЕОБРАЗОВАНИЕ ЧАСТОТЫ НА ВХОДЕ СЧЕТЧИКА	PRV2	881	781
ЗАРЕГИСТРИРОВАТЬ ТАБЛИЦУ СРАВНЕНИЯ	CTBL	882	785
ВЫДАЧА ИМПУЛЬСОВ СКОРОСТИ	SPED	885	790
ЗАДАТЬ КОЛ-ВО ИМПУЛЬСОВ	PULS	886	796
ВЫДАЧА ИМПУЛЬСОВ	PLS2	887	799
УПРАВЛЕНИЕ РАЗГОНОМ	ACC	888	808
ПОИСК ИСХОДНОГО ПОЛОЖЕНИЯ	ORG	889	817
ИМПУЛЬСЫ С ПЕРЕМЕННОЙ СКВАЖНОСТЬЮ	PWM	891	822

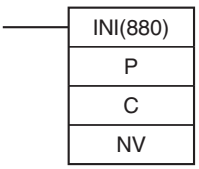
3-20-1 УПРАВЛЕНИЕ РЕЖИМОМ: INI(880)

Назначение

Команда INI(880) предназначена для управления работой встроенных входов/выходов. Она позволяет выполнять следующие операции:

- запуск сравнения текущего значения скоростного счетчика с таблицей сравнения;
- остановка сравнения текущего значения скоростного счетчика с таблицей сравнения;
- изменение текущего значения скоростного счетчика;
- изменение текущих значений входов прерывания в режиме счетчика;
- изменение текущего значения импульсного выхода (при фиксированном нулевом исходном положении);
- прекращение выдачи импульсов.

Символ РКС



P: Указатель порта
C: Управляющие данные
NV: Первое слово нового текущего значения

Варианты выполнения

Варианты выполнения		
	Выполнение в каждом цикле при включенном условии выполнения	INI(880)
	Однократное выполнение по положительному фронту	@INI(880)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

P: Указатель порта

Операнд P указывает порт, для которого выполняется операция.

P	Порт
0000 hex	Импульсный выход 0
0001 hex	Импульсный выход 1
0002 hex	Импульсный выход 2 (только для CP1H)
0003 hex	Импульсный выход 3 (только для CP1H)
0010 hex	Скоростной счетчик 0
0011 hex	Скоростной счетчик 1
0012 hex	Скоростной счетчик 2
0013 hex	Скоростной счетчик 3
0020 hex	Позиционирование на базе ПЧ 0 (только для CP1L)
0021 hex	Позиционирование на базе ПЧ 1 (только для CP1L)
0100 hex	Вход прерывания 0 в режиме счетчика
0101 hex	Вход прерывания 1 в режиме счетчика
0102 hex	Вход прерывания 2 в режиме счетчика
0103 hex	Вход прерывания 3 в режиме счетчика
0104 hex	Вход прерывания 4 в режиме счетчика (см. примеч. 1)
0105 hex	Вход прерывания 5 в режиме счетчика (см. примеч. 1)
0106 hex	Вход прерывания 6 в режиме счетчика (см. примеч. 2)
0107 hex	Вход прерывания 7 в режиме счетчика (см. примеч. 2)
1000 hex	Выход ШИМ 0
1001 hex	Выход ШИМ 1

Примечание.

(1) Входы прерывания 4 и 5 невозможно использовать в модулях ЦПУ CP1L L с 14 входами/выходами.

(2) Входы прерывания 6 и 7 невозможно использовать в модулях ЦПУ CP1H Y и в модулях CP1L L с 14 входами/выходами.

C: Управляющие данные

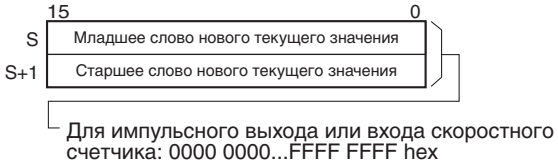
Функция команды INI(880) определяется содержанием управляющих слов (C).

C	Функция команды INI(880)
0000 hex	Запуск сравнения.
0001 hex	Остановка сравнения.
0002 hex	Изменение текущего значения.
0003 hex	Прекращение выдачи импульсов.

NV: Первое слово нового текущего значения

В словах NV и NV+1 содержится новое текущее значение в случае изменения текущего значения.

Если C = 0002 hex (что означает изменение текущ. значения), слова NV и NV+1 содержат новое текущее значение. При любом другом значении операнда C содержимое слов NV и NV+1 не играет роли.



Для импульсного выхода или входа скоростного счетчика: 0000 0000...FFFF FFFF hex

Для входа прерывания в режиме счетчика: 0000 0000...0000 FFFF hex

Характеристики операндов

Область	P	C	NV
Область CIO	---	---	CIO 0...CIO 6142
Рабочая область	---	---	W0...W510
Область битов хранения	---	---	H0...H510
Область вспомогательных битов	---	---	A448...A958
Область таймеров	---	---	T0000...T4094
Область счетчиков	---	---	C0000...C4094
Область DM	---	---	D0...D32766
Косвенные адреса DM в двоичном формате	---	---	@ D0...@ D32767
Косвенные адреса DM в двоично-десятичном формате (BCD)	---	---	*D0...*D32767
Постоянные	См. описание операнда.	См. описание операнда.	---
Регистры данных	---	---	---
Регистры указателей	---	---	---
Косвенная адресация с помощью регистров указателей	---	---	,IR0...,IR15 -2048...+2047 ,IR0... -2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15

Описание

Команда INI(880) выполняет действие, указанное операндом C, для порта, указанного операндом P. В следующей таблице приведены возможные комбинации действий и портов.

P: Указатель порта	C: Управляющие данные			
	0000 hex: Начать сравнение	0001 hex: Остановить сравнение	0002 hex: Изменить текущ. знач.	0003 hex: Прекратить выдачу импульсов
0000...0003 hex: Импульсный выход	Не допустимо.	Не допустимо.	OK	OK
0010...0013 hex: Вход скоростного счетчика	OK	OK	OK	Не допустимо.
0100...0107 hex: Вход прерывания в режиме счетчика	Не допустимо.	Не допустимо.	OK	Не допустимо.
1000 или 1001 hex: Выход ШИМ	Не допустимо.	Не допустимо.	Не допустимо.	OK

■ Запуск операции сравнения (C = 0000 hex)

Если операнд C = 0000 hex, команда INI(880) запускает операцию сравнения текущего значения скоростного счетчика со значениями таблицы сравнения, зарегистрированной с помощью команды CTBL(882).

Примечание. Таблица с заданными значениями (таблица сравнения) должна быть зарегистрирована заранее с помощью команды CTBL(882). В противном случае при выполнении команды INI(880) будет установлен флаг ошибки.

■ Остановка операции сравнения (C = 0001 hex)

Если операнд C = 0001 hex, команда INI(880) останавливает операцию сравнения текущего значения скоростного счетчика со значениями таблицы сравнения, зарегистрированной с помощью команды CTBL(882).

■ Изменение текущего значения (C = 0002 hex)

Если операнд C = 0002 hex, команда INI(880) изменяет текущее значение в соответствии со следующей таблицей.

<table>
<tr><th colspan="3">Порт и режим</th><th>Действие</th><th>Диапазон значений</th></tr>
<tr><td colspan="3">Импульсный выход
(P = 0000...0003 hex)</td><td>Изменяется текущее значение для импульсного выхода. Новое значение указывается в словах NV и NV+1.
Примечание.
Данная команда может быть выполнена, только когда выдача импульсов остановлена. Если она будет выполнена во время выдачи импульсов, произойдет ошибка.</td><td>8000 0000...
7FFF FFFF hex
(-2 147 483 648...
2 147 483 647)</td></tr>
<tr><td rowspan="3">Вход скоростного счетчика
(P = 0010...
0013 hex)</td><td rowspan="2">Линейный режим</td><td>Квадратурные входы (x4), импульсные входы прямого/обратного счета или импульсный вход + вход направления</td><td rowspan="3">Изменяется текущее значение скоростного счетчика. Новое значение указывается в словах NV и NV+1.
Примечание.
Если указанный порт не назначен для работы в режиме скоростного счетчика, при выполнении команды произойдет ошибка.</td><td>8000 0000...
7FFF FFFF hex
(-2 147 483 648...
2 147 483 647)</td></tr>
<tr><td>Импульсный вход прямого счета</td><td>0000 0000...
FFFF FFFF hex
(0...4 294 967 295)</td></tr>
<tr><td colspan="2">Кольцевой режим</td><td>0000 0000...
FFFF FFFF hex
(0...4 294 967 295)</td></tr>
<tr><td colspan="3">Входы прерывания в режиме счетчика
(P = 0100...0107 hex)</td><td>Изменяется текущее значение счетчика входа прерывания. Новое значение указывается в словах NV и NV+1.</td><td>0000 0000...0000 FFFF hex
(0...65 535)
Примечание.
Указание значения за пределами данного диапазона вызовет ошибку.</td></tr>
</table>

■ Прекращение выдачи импульсов (P = 0000...0003, 1000 или 1001 hex и C = 0003 hex)

Если операнд C = 0003 hex, команда INI(880) немедленно прекращает выдачу импульсов с указанного порта. Выполнение этой команды при уже остановленной выдаче импульсов приведет к обнулению установленного количества импульсов.

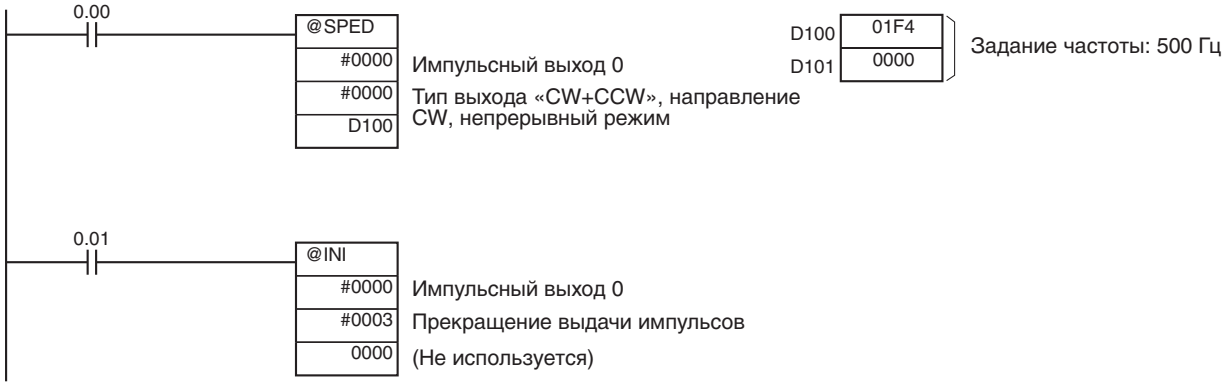
Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	Включен, если значение P, C или NV выходит за допустимый диапазон. Включен при недопустимой комбинации P и C. Включен, если был указан запуск сравнения, но не была зарегистрирована таблица сравнения. Включен, если для порта, который в данный момент выдает импульсы, указано новое текущее значение. Включен, если операция изменения текущего значения скоростного счетчика выбрана для порта, который не назначен в качестве скоростного счетчика. Включен, если текущее значение, указанное для входа прерывания в режиме счетчика, выходит за допустимый диапазон. Включен, если команда INI(880) выполняется для скоростного счетчика задачей обработки прерывания и это прерывание возникает во время выполнения команды CTBL(882). Включен, если команда выполнена для порта, не выбранного в качестве входа прерывания в режиме счетчика.

Пример

При включении бита CIO 0.00 в приведенном ниже примере команда SPED(885) запускает выдачу импульсов с импульсного выхода 0 в непрерывном режиме с частотой 500 Гц. Когда включается бит CIO 0.01, выдача импульсов прекращается командой INI(880).



3-20-2 ЧТЕНИЕ ТЕКУЩ. ЗНАЧ. СКОРОСТНОГО СЧЕТЧИКА: PRV(881)

Назначение

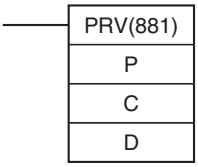
Команда PRV(881) считывает и возвращает указанную ниже информацию о встроенном входе или выходе.

- Текущие значения (PV): текущ. значение скоростного счетчика, текущ. значение импульсного выхода, текущ. значение входа прерывания в режиме счетчика.
- Данные о состоянии, указанные в следующей таблице.

Состояние	Содержание
Состояние импульсного выхода	Флаг состояния выдачи импульсов Флаг переполнения/потери значимости текущего значения Флаг установки числа выходных импульсов Флаг завершения выдачи импульсов Флаг выдачи импульсов Флаг неопределения исходного положения Флаг достижения исходного положения Флаг остановки выдачи импульсов из-за ошибки
Состояние входа скоростного счетчика	Флаг выполнения сравнения Флаг переполнения/потери значимости текущего значения
Выход команды PWM(891)	Флаг текущей выдачи импульсов

- Результаты сравнения с диапазонами
- Частота следования импульсов на импульсном выходе 0...импульсном выходе 3
- Частота импульсов на входе скоростного счетчика 0

Символ РКС



P: Указатель порта
C: Управляющие данные
D: Первое слово назначения

Варианты выполнения

Варианты выполнения		
	Выполнение в каждом цикле при включенном условии выполнения	PRV(881)
	Однократное выполнение по положительному фронту	@PRV(881)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

P: Указатель порта

Операнд P указывает порт, для которого выполняется операция.

P	Порт
0000 hex	Импульсный выход 0
0001 hex	Импульсный выход 1
0002 hex	Импульсный выход 2 (только для CP1H)
0003 hex	Импульсный выход 3 (только для CP1H)
0010 hex	Скоростной счетчик 0

P	Порт
0011 hex	Скоростной счетчик 1
0012 hex	Скоростной счетчик 2
0013 hex	Скоростной счетчик 3
0020 hex	Позиционирование на базе ПЧ 0 (только для CP1L)
0021 hex	Позиционирование на базе ПЧ 1 (только для CP1L)
0030 hex	Счетчик ошибки 0, со знаком (только для CP1L)
0031 hex	Счетчик ошибки 1, со знаком (только для CP1L)
0040 hex	Счетчик ошибки 0, без знака (только для CP1L)
0041 hex	Счетчик ошибки 1, без знака (только для CP1L)
0100 hex	Вход прерывания 0 в режиме счетчика
0101 hex	Вход прерывания 1 в режиме счетчика
0102 hex	Вход прерывания 2 в режиме счетчика
0103 hex	Вход прерывания 3 в режиме счетчика
0104 hex	Вход прерывания 4 в режиме счетчика
0105 hex	Вход прерывания 5 в режиме счетчика
0106 hex	Вход прерывания 6 в режиме счетчика
0107 hex	Вход прерывания 7 в режиме счетчика
1000 hex	Выход ШИМ 0
1001 hex	Выход ШИМ 1

Примечание. Прерывания от входов 6 и 7 невозможно использовать в модулях ЦПУ CP1H Y и CP1L. Прерывания от входов 4, 5, 6 и 7 невозможно использовать в модулях ЦПУ CP1L с 14 входами/выходами.

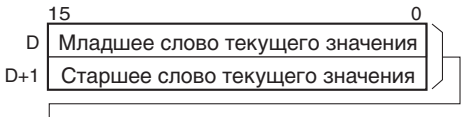
C: Управляющие данные

Функция команды PRV(881) определяется содержанием управляющих слов (C).

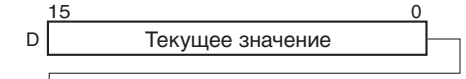
C	Функция команды PRV(881)	Варианты выполнения
0000 hex	Чтение текущего значения.	---
0001 hex	Чтение состояния.	---
0002 hex	Чтение результатов сравнения с диапазонами.	---
00□3 hex	P = 0000 или 0001: Чтение частоты импульсов на импульсном выходе 0 или импульсном выходе 1. P = 0010: Чтение частоты импульсов на входе скоростного счетчика 0.	C = 0003 hex: Стандартный способ вычисления C = 0013 hex: Интервал измерения 10 мс для высокой частоты C = 0023 hex: Интервал измерения 100 мс для высокой частоты C = 0033 hex: Интервал измерения 1 с для высокой частоты

D: Первое слово назначения

Текущее значение выдается в D или в [D, D+1].



Текущее значение (2 слова)
Текущ. знач. имп. выхода, текущ. знач. входа скоростного счетчика, частота на входе скоростного счетчика для входа скоростного счетчика 0



Текущее значение (1 слово)
Текущ. знач. входа прерывания в режиме счетчика, состояние, результаты сравнения с диапазонами

Характеристики операндов

Область	P	C	D
Область CIO	---	---	CIO 0...CIO 6142
Рабочая область	---	---	W0...W510
Область битов хранения	---	---	H0...H510
Область вспомогательных битов	---	---	A448...A958
Область таймеров	---	---	T0000...T4094
Область счетчиков	---	---	C0000...C4094
Область DM	---	---	D0...D32766
Косвенные адреса DM в двоичном формате	---	---	@ D0...@ D32766
Косвенные адреса DM в двоично-десятичном формате (BCD)	---	---	*D0...*D32766
Постоянные	См. описание операнда.	См. описание операнда.	---
Регистры данных	---	---	---
Регистры указателей	---	---	---
Косвенная адресация с помощью регистров указателей	---	---	,IR0...,IR15 -2048...+2047 ,IR0... -2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15

Описание

Команда PRV(881) считывает и возвращает информацию, указанную операндом C, для порта, указанного операндом P. Возможные комбинации запрашиваемой информации и портов приведены в следующей таблице.

P: Указатель порта	C: Управляющие данные			
	0000 hex: Чтение текущего значения	**0001 hex: Чтение состояния**	**0002 hex: Чтение результатов сравнения с диапазо-нами**	**0003 hex: Чтение частоты импульсов на входе скоростного счетчика**
0000...0003 hex: Импульсный выход	OK	OK	Не допустимо.	OK
0010...0013 hex: Вход скоростного счетчика	OK	OK	OK	OK (только скоростной счетчик 0)
0100...0107 hex: Вход прерывания в режиме счетчика	OK	Не допустимо.	Не допустимо.	Не допустимо.
1000 или 1001 hex: Выход ШИМ	Не допустимо.	OK	Не допустимо.	Не допустимо.

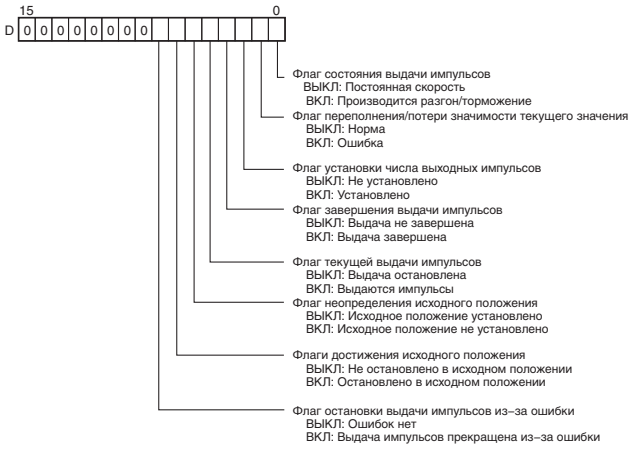
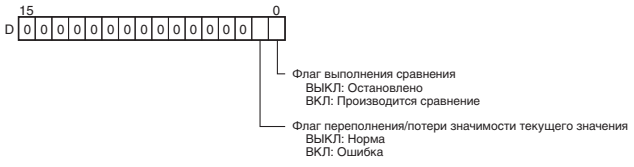
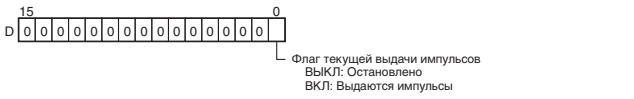
■ Чтение текущего значения (C = 0000 hex)

Если операнд C = 0000 hex, команда PRV(881) считывает и возвращает текущее значение (PV) в соответствии со следующей таблицей.

Порт и режим		Действие	Диапазон значений
Импульсный выход (P = 0000...0003 hex)		В слова D и D+1 записывается текущее значение для импульсного выхода.	8000 0000...7FFF FFFF hex (-2 147 483 648...2 147 483 647)
Вход скоростного счетчика P = 0010... 0013 hex)	Линейный режим	В слова D и D+1 записывается текущее значение для скоростного счетчика.	8000 0000...7FFF FFFF hex (-2 147 483 648...2 147 483 647)
	Кольцевой режим		0000 0000...FFFF FFFF hex (0...4 294 967 295)
Входы прерывания в режиме счетчика (P = 0100...0107 hex)		В слово D записывается текущее значение для входа прерывания.	0000...FFFF hex (0...65 535)

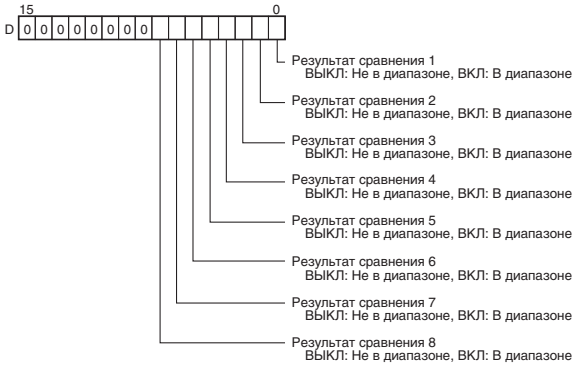
■ Чтение состояния (C = 0001 hex)

Если операнд C = 0001 hex, команда PRV(881) считывает и возвращает состояние в соответствии со следующей таблицей.

Порт и режим	Действие	Результаты чтения
Импульсный выход	В слово D записывается состояние импульсного выхода.	
Вход скоростного счетчика	В слово D записывается состояние скоростного счетчика.	
Выход ШИМ	В слово D записывается состояние выхода ШИМ.	

■ Чтение результатов сравнения с диапазонами (C = 0002 hex)

Если операнд C = 0002 hex, команда PRV(881) считывает результаты сравнения с диапазонами и возвращает их в слово D в соответствии со следующим рисунком.



■ Чтение частоты на импульсном выходе или входе скоростного счетчика (C = 00□3 hex)

Если операнд C = 00□3 hex, команда PRV(881) считывает и возвращает в слова [D, D+1] частоту следования импульсов на импульсном выходе 0...3 или частоту поступления импульсов на вход скоростного счетчика 0.

Диапазоны возвращаемых значений частоты

Значение P	Тип и версия модуля ЦПУ	Тип импульсного выхода/входа счетчика	Результат преобразования
0000... 0003 hex	Модуль ЦПУ CP1H X/XA, версия модуля 1.0	Имп. выход 2 или 3	0000 0000...0000 7530 hex (0...30 000)
	Модуль ЦПУ CP1H X/XA, версия модуля 1.1 и выше	Имп. выход 0 или 1	0000 0000...0001 86A0 hex (0...100 000)
		Имп. выход 2 или 3	
	Модуль ЦПУ CP1H Y, любая версия модуля	Имп. выход 1 или 2	0000 0000...000F 4240 hex (0...1 000 000)
		Имп. выход 2 или 3	0000 0000...0001 86A0 hex (0...100 000)
	Модуль ЦПУ CP1L, любая версия модуля	Импульсный выход 2 или 3	0000 0000...0001 86A0 hex (0...100 000)
0010 hex	Модуль ЦПУ CP1H X/XA или CP1L, любая версия	Любой тип входа счетчика, кроме квадратурного (4×)	0000 0000...0001 86A0 hex (0...100 000) **Примечание.** При частоте следования импульсов более 100 кГц будет постоянно выдаваться максимальное значение 0001 86A0 hex.
		Квадратурный вход (4x)	0000 0000...0003 0D40 hex (0...200 000) **Примечание.** При частоте следования импульсов более 200 кГц будет постоянно выдаваться максимальное значение 0003 0D40 hex.
	Модуль ЦПУ CP1H Y, любая версия	Любой тип входа счетчика, кроме квадратурного (4×)	0000 0000...000F 4240 hex (0...1 000 000) **Примечание.** При частоте следования импульсов более 1 МГц будет постоянно выдаваться максимальное значение 000F 4240 hex.
		Квадратурный вход (4x)	0000 0000...001E 8480 hex (0...2 000 000) **Примечание.** При частоте следования импульсов более 2 МГц будет постоянно выдаваться максимальное значение 001E 8480 hex.

Способы вычисления частоты импульсов

Существует 2 способа вычисления частоты импульсов, подаваемых на импульсный выход 0...3 или поступающих на вход скоростного счетчика 0.

1. Стандартный способ вычисления (прежний способ)

   Для определения количества импульсов подсчитывается каждый импульс, независимо от значения частоты. При высоких значениях частоты форма передних или задних фронтов некоторых импульсов может искажаться, приводя к ошибке измерения (макс. 1% при частоте 100 кГц и максимальная ошибка при частоте 1 МГц).

2. Способ вычисления для высокой частоты

   В этом случае при высоких частотах используется один способ вычисления, а при низких — другой.

   - Подсчет при высокой частоте

     При высоких частотах (свыше 1 кГц) функция подсчитывает число импульсов за фиксированный промежуток времени (время измерения) и вычисляет частоту по найденному количеству. С помощью двух младших разрядов операнда C можно выбрать один из трех следующих интервалов измерения.

Время измерения	Значение C	Описание
10 мс	0013 hex	Количество импульсов подсчитывается каждые 10 мс. Максимальная ошибка при 100 кГц: 0,1%. Максимальная ошибка наблюдается при частоте 1 кГц (10%).
100 мс	0023 hex	Количество импульсов подсчитывается каждые 100 мс. Максимальная ошибка при 100 кГц: 0,01%. Максимальная ошибка наблюдается при частоте 1 кГц (1%).
1 с	0033 hex	Количество импульсов подсчитывается каждые 1 с. Максимальная ошибка при 100 кГц: 0,001%. Максимальная ошибка наблюдается при частоте 1 кГц (0,1%).

   - Подсчет импульсов при низких частотах

     При значениях частоты ниже 1 кГц используется стандартный способ вычисления независимо от заданного времени измерения.

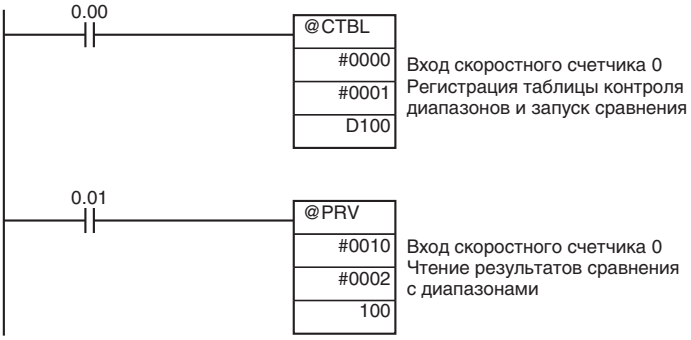
Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	Включен, если значение P или C выходит за допустимый диапазон. Включен при недопустимой комбинации P и C. Включен, если выбрано чтение результатов сравнения с диапазонами, но сравнение с диапазонами в данный момент не выполняется. Включен, если чтение выходной частоты выбрано для порта, не являющегося входом скоростного счетчика 0. Включен, если указанный порт не назначен для работы в режиме скоростного счетчика. Включен при выполнении команды для порта, не выбранного для работы в качестве входа прерывания в режиме счетчика.

Примеры

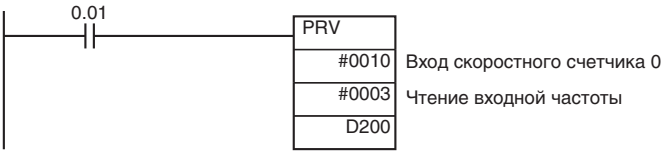
Пример 1

При включении бита CIO 0.00 в приведенном ниже примере программы команда CTBL(882) регистрирует таблицу контроля диапазонов для скоростного счетчика 0 и запускает операцию сравнения. Когда включается бит CIO 0.01, команда PRV(881) считывает результаты сравнения с диапазонами, полученные на данный момент, и записывает их в слово CIO 100.00.



Пример 2

При включении бита CIO 0.01 в приведенном ниже примере программы команда PRV(881) считывает текущее значение частоты поступления импульсов на вход скоростного счетчика 0 и записывает его в шестнадцатеричном формате в слова D201 и D200.

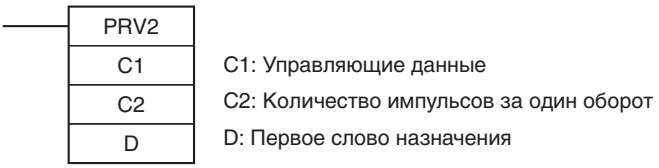


3-20-3 ПРЕОБРАЗОВАНИЕ ЧАСТОТЫ НА ВХОДЕ СЧЕТЧИКА: PRV2(883)

Назначение

Команда PRV2(883) измеряет частоту импульсов, поступающих на вход скоростного счетчика, и либо преобразует измеренную частоту в угловую скорость, либо преобразует текущее значение счетчика в полное число оборотов. 8-разрядный шестнадцатеричный результат выдается в назначенные слова. Данная команда может быть выполнена только для скоростного счетчика 0.

Символ РКС



Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	PRV2(883)
	Однократное выполнение по положительному фронту	@PRV2(883)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

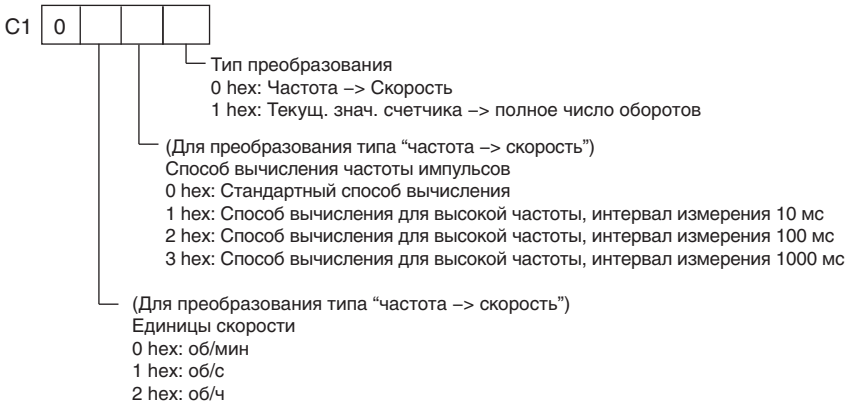
Операнды

C1: Управляющие данные

Функция команды PRV2(883) определяется содержанием управляющего слова C1.

C1	Функция команды PRV2(883)
0□∗0 hex (см. примеч.)	Преобразование частоты в угловую скорость.
0001 hex	Преобразование текущего значения счетчика в общее количество оборотов.

Примечание. Второй разряд C (□) указывает единицы измерения, а третий разряд (∗) указывает способ вычисления частоты.

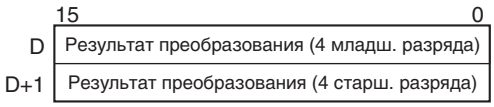


C2: Количество импульсов за один оборот

Указывает количество импульсов, приходящееся на один оборот (0001...FFFF hex).

D: Первое слово назначения

Текущее значение выдается в D или в [D, D+1].



Характеристики операндов

Область	C1	C2	D
Область CIO	---	CIO 0...CIO 6143	CIO 0...CIO 6142
Рабочая область	---	W0...W511	W0...W510
Область битов хранения	---	H0...H511	H0...H510
Область вспомогательных битов	---	A448...A959	A448...A958
Область таймеров	---	T0000...T4095	T0000...T4094
Область счетчиков	---	C0000...C4095	C0000...C4094
Область DM	---	D0...D32767	D0...D32766
Косвенные адреса DM в двоичном формате	---	@ D0... @ D32767	@ D0... @ D32767

Область	C1	C2	D
Косвенные адреса DM в двоично-десятичном формате (BCD)	---	*D0...*D32767	*D0...*D32767
Постоянные	См. описание операнда.	---	---
Регистры данных	---	---	---
Регистры указателей	---	---	---
Косвенная адресация с помощью регистров указателей	---	,IR0...,IR15 -2048...+2047 ,IR0... -2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

Описание

Команда PRV2(883) преобразует значение частоты импульсов, поступающих на вход скоростного счетчика 0, в соответствии со способом преобразования, который указан в C1, и количеством импульсов за один оборот, которое указано в C2. Результат выдается в слова D и D+1.

Тип преобразования определяется содержимым слова C1: 0000 hex или 0001 hex.

Преобразование частоты в угловую скорость (C1 = 0☐✳0 hex)

Если операнд C1 = 0☐✳0 hex, команда PRV2(883) вычисляет угловую скорость (об/мин) по значению частоты и заданному количеству импульсов, приходящемуся на один оборот. Второй разряд C (☐) указывает единицы измерения, а третий разряд (✳) указывает способ вычисления частоты.

1. Единицы измерения угловой скорости
   - Единицы угловой скорости = об/мин

     Если второй разряд операнда C (☐) = 0, команда PRV2(883) вычисляет значение угловой скорости в единицах [об/мин] по значению частоты и заданному числу импульсов за оборот.

     Угловая скорость (об/мин) = (Частота ÷ Импульсов/оборот) × 60
   - Единицы угловой скорости = об/с

     Если второй разряд операнда C (☐) = 1, команда PRV2(883) вычисляет значение угловой скорости в единицах [об/с] по значению частоты и заданному числу импульсов за оборот.

     Угловая скорость (об/с) = Частота ÷ Импульсов/оборот
   - Единицы угловой скорости = об/ч

     Если второй разряд операнда C (☐) = 2, команда PRV2(883) вычисляет значение угловой скорости в единицах [об/ч] по значению частоты и заданному числу импульсов за один оборот.

     Угловая скорость (об/ч) = (Частота ÷ Импульсов/оборот) × 60 × 60
   - Диапазон значений результата преобразования для модулей CP1H X/XA и CP1L
     - Тип входа счетчика: любой тип входа, кроме квадратурного (4×)
       Результат преобразования = 0000 0000...0001 86A0 hex
       (0...100 000)
     - Тип входа счетчика: квадратурный (4×)
       Результат преобразования = 0000 0000...0003 0D40 hex
       (0...200 000)
       (При частоте следования импульсов более 200 кГц будет постоянно выдаваться максимальное значение 0003 0D40 hex.)

- Диапазон значений результата преобразования для модулей ЦПУ CP1H Y
  - Тип входа счетчика: любой тип входа, кроме квадратурного (4×)
    Результат преобразования = 0000 0000...000F 4240 hex
    (0...1 000 000)
  - Тип входа счетчика: квадратурный (4×)
    Результат преобразования = 0000 0000...001E 8480 hex
    (0...2 000 000)
    (При частоте следования импульсов более 2 МГц будет постоянно выдаваться максимальное значение 001E 8480 hex.)

Примечание. Для всех значений угловой частоты, которые превышают максимально допустимую входную частоту, команда выдает максимальное выходное значение. Максимально допустимая входная частота зависит от используемой функции и используемого порта скоростного счетчика. Дополнительную информацию см. в *Руководстве по эксплуатации*. (В случае модуля ЦПУ CP1H Y при переполнении в качестве результата преобразования (об/ч) выдается значение FFFF FFFF hex.)

2. Способ вычисления частоты

   Существуют два способа вычисления частоты импульсов, поступающих на вход скоростного счетчика 0.

   a. Стандартный способ вычисления (C1 = 0☐00)

      Для определения количества импульсов подсчитывается каждый импульс, независимо от значения частоты. При высоких значениях частоты форма передних или задних фронтов некоторых импульсов может искажаться, приводя к ошибке измерения (макс. 1% при частоте 100 кГц).

   b. Способ вычисления для высокой частоты

      В этом случае при высоких частотах используется один способ вычисления, а при низких — другой.

      - Подсчет импульсов при высоких частотах (C1 = 0☐10, 0☐20 или 0☐30)

        При высоких частотах (свыше 1 кГц) функция подсчитывает число импульсов за фиксированный промежуток времени (время измерения) и вычисляет частоту по найденному количеству. С помощью третьего разряда операнда C1 можно выбрать один из трех следующих интервалов измерения.

Время измерения	Значение C1	Описание
10 мс	0☐10 hex	Количество импульсов подсчитывается каждые 10 мс. Максимальная ошибка при 1 кГц: 10%.
100 мс	0☐20 hex	Количество импульсов подсчитывается каждые 100 мс. Максимальная ошибка при 1 кГц: 1%.
1 с	0☐30 hex	Количество импульсов подсчитывается каждые 1 с. Максимальная ошибка при 1 кГц: 0,1%.

      - Подсчет импульсов при низких частотах

        При значениях частоты ниже 1 кГц используется стандартный способ вычисления независимо от заданного времени измерения.

Преобразование текущего значения счетчика в общее количество оборотов (C1 = 0001 hex)

Если операнд C1 = 0001 hex, команда PRV2(883) вычисляет совокупное количество оборотов по текущему значению счетчика и заданному количеству импульсов, приходящемуся на 1 оборот.

Результат преобразования = Текущ. знач. счетчика ÷ Импульсов/оборот

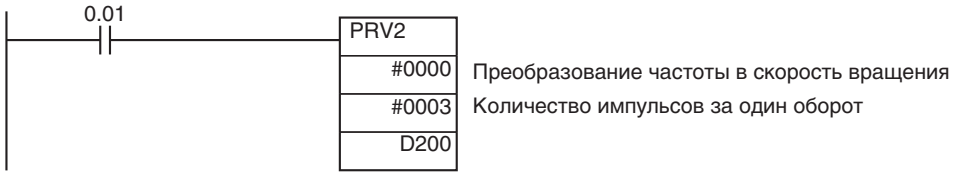
Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	Включен, если скоростной счетчик 0 в настройках отключен. Включен, если значение C1 выходит за допустимый диапазон (0000 или 0001). Включен, если задано нулевое число импульсов за оборот (C2 = 0000).
Флаг переполнения	OF	Включен, если достигнуто максимальное значение для модуля ЦПУ типа Y.

Примеры

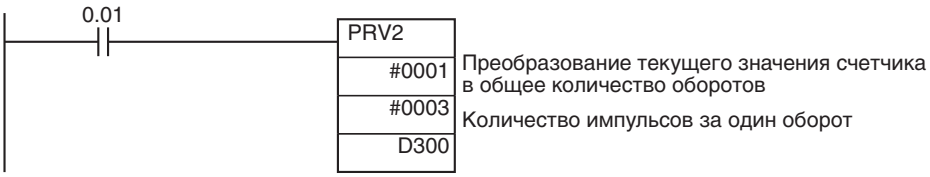
Пример 1

При включенном бите CIO 0.01 команда PRV2(883) измеряет текущую частоту импульсов на входе скоростного счетчика 0, преобразует ее в угловую скорость и выдает полученное шестнадцатеричное значение в слова D201 и D200.



Пример 2

При включенном бите CIO 0.01 команда PRV2(883) считывает текущее значение счетчика, преобразует его в количество оборотов и выдает полученное шестнадцатеричное значение в слова D301 и D300.

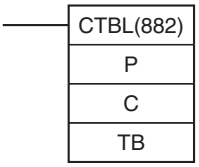


3-20-4 ЗАРЕГИСТРИРОВАТЬ ТАБЛИЦУ СРАВНЕНИЯ: CTBL(882)

Назначение

Команда CTBL(882) регистрирует таблицу сравнения и выполняет сравнение текущего значения скоростного счетчика со значениями таблицы. Возможно сравнение с абсолютными значениями или с диапазонами значений. При выполнении указанного условия выполняется задача обработки прерывания.

Символ РКС



P: Указатель порта
C: Управляющие данные
TB: Первое слово таблицы сравнения

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	CTBL(882)
	Однократное выполнение по положительному фронту	@CTBL(882)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

P: Указатель порта

Операнд P согласно приведенной ниже таблице указывает порт, для которого должен производиться подсчет импульсов.

P	Порт
0000 hex	Скоростной счетчик 0
0001 hex	Скоростной счетчик 1
0002 hex	Скоростной счетчик 2
0003 hex	Скоростной счетчик 3

C: Управляющие данные

Функция команды CTBL(882) определяется содержанием управляющего слова C в соответствии с таблицей ниже.

C	Функция команды CTBL(882)
0000 hex	Зарегистрировать таблицу искомых значений и начать операцию сравнения.
0001 hex	Зарегистрировать таблицу контроля диапазонов и начать операцию сравнения.
0002 hex	Зарегистрировать таблицу искомых значений. Сравнение запускает команда INI(880).
0003 hex	Зарегистрировать таблицу контроля диапазонов. Сравнение запускает команда INI(880).

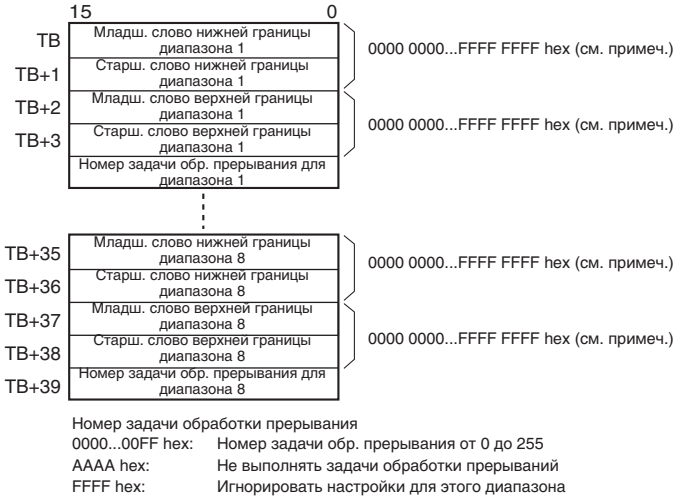
TB: Первое слово таблицы сравнения

Операнд TB указывает адрес первого слова таблицы сравнения. Структура таблицы сравнения зависит от типа выполняемой операции сравнения.

При сравнении с абсолютными значениями длина таблицы сравнения определяется количеством заданных значений, указанных в TB. Таблица может иметь длину от 4 до 145 слов, что показано на приведенном ниже рисунке.



При сравнении с диапазонами значений, таблица сравнения всегда содержит 8 диапазонов. В этом случае длина таблицы составляет 40 слов, как показано ниже. Если некоторые из этих восьми диапазонов не задействуются, для них следует указать номер задачи обработки прерывания FFFF hex.



Примечание. Для каждого диапазона верхнее предельное значение должно быть больше или равно нижнему предельному значению.

Характеристики операндов

Область	P	C	TB
Область CIO	---	---	CIO 0...CIO 6143
Рабочая область	---	---	W0...W511
Область битов хранения	---	---	H0...H511
Область вспомогательных битов	---	---	A448...A959
Область таймеров	---	---	T0000...T4095
Область счетчиков	---	---	C0000...C4095
Область DM	---	---	D0...D32767
Косвенные адреса DM в двоичном формате	---	---	@ D0...@ D32767

Область	P	C	ТВ
Косвенные адреса DM в двоично-десятичном формате (BCD)	---	---	*D0...*D32767
Постоянные	См. описание операнда.	См. описание операнда.	---
Регистры данных	---	---	---
Регистры указателей	---	---	---
Косвенная адресация с помощью регистров указателей	---	---	,IR0...,IR15 -2048...+2047 ,IR0... -2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15

Описание

Команда CTBL(882) либо просто регистрирует таблицу сравнения, либо регистрирует таблицу сравнения и запускает операцию сравнения для порта, указанного операндом P, с использованием способа сравнения, который указан операндом C. Однажды зарегистрированная таблица сравнения остается действительной, пока не регистрируется другая таблица или пока модуль ЦПУ не переключается в режим «Программирование».

Каждый раз, когда выполняется команда CTBL(882), сравнение запускается при наступлении определенных условий. В случае применения команды CTBL(882) для запуска сравнения обычно достаточно использовать вариант однократного выполнения (@CTBL(882)) или включать условие выполнения только на один цикл программы ПЛК.

Примечание. Если будет указана несуществующая задача обработки прерывания, при первом формировании прерывания возникнет критическая ошибка программы.

■ **Регистрация таблицы сравнения (C = 0002 или 0003 hex)**

Если операнд C задан равным 0002 или 0003 hex, регистрируется таблица сравнения, но операция сравнения не запускается. Сравнение запускает команда INI(880).

■ **Регистрация таблицы сравнения и запуск таблицы сравнения (C = 0000 или 0001 hex)**

Если операнд C задан равным 0000 или 0001 hex, регистрируется таблица сравнения и запускается операция сравнения.

■ **Остановка сравнения**

Независимо от того, какая команда использовалась для запуска сравнения, остановка сравнения производится с помощью команды INI(880).

■ **Сравнение с заданным значением**

Когда текущее значение совпадает с заданным значением, вызывается и выполняется соответствующая задача обработки прерывания.

- Один и тот же номер задачи обработки прерывания может быть указан более чем для одного сравниваемого значения.
- Может быть указано направление, в котором должно изменяться текущее значение (возрастать или убывать), чтобы совпадение было признано действительным. Если выключен бит 15 слова, указывающего номер задачи обработки прерывания для некоторого диапазона, сравнение текущего значения с заданным значением будет происходить, только если текущее значение в данный момент

возрастает. А если включен бит 00, сравнение будет производиться только при убывании текущего значения.

- Таблица сравнения может содержать до 48 сравниваемых значений. Количество сравниваемых значений указывается в операнде TB (длина таблицы зависит от количества указанных заданных значений).
- Текущее значение сравнивается со всеми значениями, зарегистрированными в таблице сравнения.

Примечание.

(1) Если в таблице будут зарегистрированы несколько одинаковых значений с одним и тем же направлением изменения, произойдет ошибка.

(2) Если скоростной счетчик сконфигурирован для работы только в режиме прямого счета, выбор отрицательного направления в качестве условия сравнения приведет к ошибке.

(3) Если направление счета изменится в момент равенства текущего значения заданному значению, но прежнее направление счета, в котором оказалось достигнуто заданное значение, не являлось направлением сравнения, условие сравнения для этого заданного значения не будет считаться выполненным. Поэтому сравниваемые значения не должны приходиться на экстремумы (пики или минимумы) текущего значения, в которых меняется направление счета.

■ **Сравнение с диапазоном**

Когда текущее значение оказывается в пределах заданного диапазона, вызывается и выполняется соответствующая задача обработки прерывания.

- Один и тот же номер задачи обработки прерывания может быть указан более чем для одного сравниваемого значения.
- Таблица контроля диапазонов содержит восемь диапазонов, каждый из которых определяется нижним и верхним предельными значениями. Чтобы отключить неиспользуемые диапазоны, для них следует указать номер задачи обработки прерывания FFFF hex.
- Задача обработки прерывания выполняется только один раз, когда текущее значение оказывается в пределах диапазона.
- Если текущее значение оказывается в пределах сразу нескольких диапазонов, зарегистрированных в таблице, первой выполняется задача обработки прерывания для диапазона, расположенного ближе всего к началу таблицы. Остальные задачи обработки прерываний будут выполнены в следующих циклах программы ПЛК.
- Если выполнение задачи обработки прерывания не требуется, в качестве номера задачи следует указать значение AAAA hex. Результаты операции сравнения с диапазонами могут быть прочитаны с помощью команды PRV(881) или с помощью флагов текущего выполнения сравнения.

Примечание. Если верхняя граница для какого-либо диапазона окажется меньше нижней границы, произойдет ошибка.

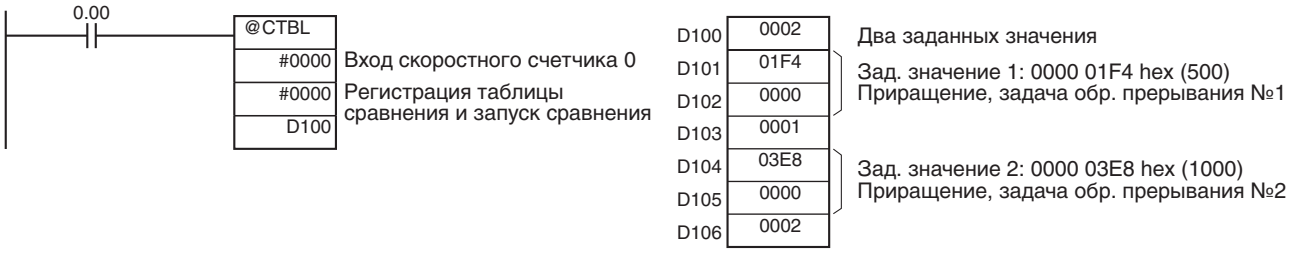
Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	Включен, если значение P или C выходит за допустимый диапазон. Включен, если для операции сравнения с заданными значениями указано нулевое количество заданных значений. Включен, если количество заданных значений, указанное для операции сравнения с заданными значениями, превышает 48. Включен, если в таблице сравнения зарегистрировано несколько одинаковых заданных значений с одинаковым направлением сравнения. Включен, если для любого из диапазонов верхняя граница меньше нижней границы. Включен, если заданные значения для всех диапазонов отключены во время операции сравнения с диапазонами. Включен, если в качестве направления сравнения в таблице выбран обратный счет, тогда как скоростной счетчик сконфигурирован только для счета в прямом направлении. Включен, если при выполнении команды оказалось, что скоростной счетчик сконфигурирован для работы в кольцевом режиме и указанное значение превышает максимальный предел счета в этом режиме. Включен, если указанный порт не назначен для работы в режиме скоростного счетчика. Включен, если во время выполнения сравнения выполняется команда с другим указанным методом сравнения.

Пример

При включении бита CIO 0.00 в следующем примере команда CTBL(882) регистрирует таблицу искомых значений и запускает операцию сравнения для скоростного счетчика 0. Текущее значение скоростного счетчика прирастает и, достигнув значения 500, совпадает с заданным значением 1, что приводит к выполнению задачи обработки прерывания 1. Затем текущее значение доходит до 1000 и становится равным заданному значению 2, что приводит к выполнению задачи обработки прерывания 2.



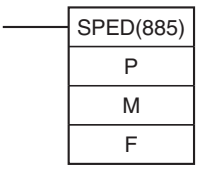
3-20-5 ВЫДАЧА ИМПУЛЬСОВ СКОРОСТИ: SPED(885)

Назначение

Команда SPED(885) устанавливает заданное значение частоты следования выходных импульсов для указанного порта и запускает выдачу импульсов без разгона или торможения. Она подходит как для позиционирования в независимом режиме, так и для регулирования скорости в непрерывном режиме. Для позиционирования в независимом режиме должно быть задано количество импульсов с помощью команды PULS(886).

Команду SPED(885) также можно выполнить непосредственно во время выдачи импульсов для изменения выходной частоты, что позволяет ступенчато изменять скорость вращения двигателя.

Символ РКС



P: Указатель порта
M: Режим работы выхода
F: Первое слово частоты импульсов

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	SPED(885)
	Однократное выполнение по положительному фронту	@SPED(885)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

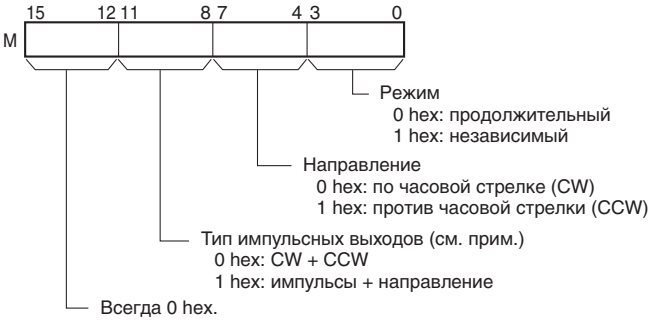
P: Указатель порта

Указатель порта указывает порт (т. е. выход), на который должны выдаваться импульсы.

P	Порт
0000 hex	Импульсный выход 0
0001 hex	Импульсный выход 1
0002 hex	Импульсный выход 2 (только для CP1H)
0003 hex	Импульсный выход 3 (только для CP1H)
0020 hex	Позиционирование на базе ПЧ 0 (только для CP1L)
0021 hex	Позиционирование на базе ПЧ 1 (только для CP1L)

M: Режим работы выхода

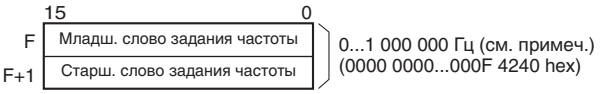
Значение операнда M определяет режим работы импульсного выхода.



Прим.: Один и тот же тип должен быть указан для импульсных выходов 0 и 1 (имеются только в CP1H).

F: Первое слово частоты импульсов

Двойное слово [F, F+1] задает частоту следования импульсов в [Гц].



Примечание. Максимальное значение частоты, которое может быть указано, зависит от модели и поддержки выдачи импульсов. См. *Руководство по эксплуатации ЦПУ CP1H*.

Характеристики операндов

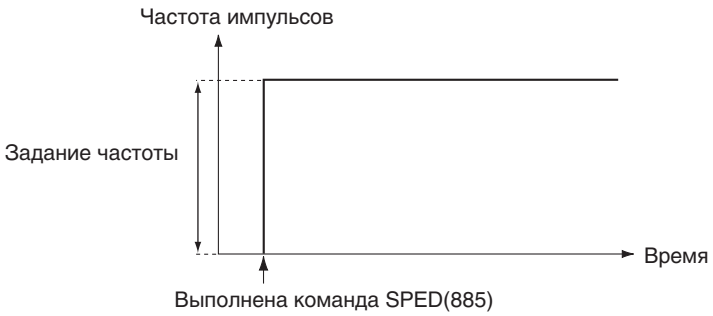
Область	P	M	F
Область CIO	---	---	CIO 0...CIO 6142
Рабочая область	---	---	W0...W510
Область битов хранения	---	---	H0...H510
Область вспомогательных битов	---	---	A448...A958
Область таймеров	---	---	T0000...T4094
Область счетчиков	---	---	C0000...C4094
Область DM	---	---	D0...D32766
Косвенные адреса DM в двоичном формате	---	---	@ D0...@ D32767
Косвенные адреса DM в двоично-десятичном формате (BCD)	---	---	*D0...*D32767
Постоянные	См. описание операнда.	См. описание операнда.	См. описание операнда.
Регистры данных	---	---	---
Регистры указателей	---	---	---
Косвенная адресация с помощью регистров указателей	---	---	,IR0...,IR15 -2048...+2047 ,IR0... -2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15

Верхние предельные значения задаваемой частоты

Порт	CP1H			CP1L
	X40/XA40, версия 1.0	X40/XA40 версия 1.1 и выше	Y20	
Порт 0	100 кГц	100 кГц	1 МГц	100 кГц
Порт 1	100 кГц	100 кГц	1 МГц	100 кГц
Порт 2	30 кГц	100 кГц	100 кГц	---
Порт 3	30 кГц	100 кГц	100 кГц	---

Описание

Команда SPED(885) инициирует выдачу импульсов с порта, указанного операндом P, с использованием метода, указанного операндом M, с частотой, указанной операндом F. Выдача импульсов будет запускаться при каждом выполнении команды SPED(885). Поэтому в общем случае должен использоваться вариант однократного выполнения (@SPED(885)), либо условие выполнения должно включаться только на один цикл программы ПЛК.



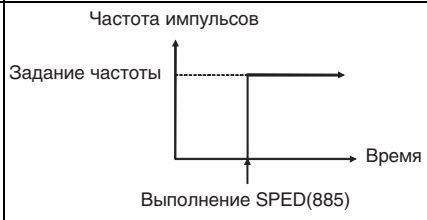
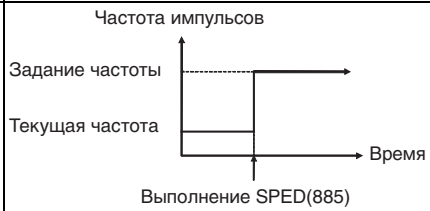
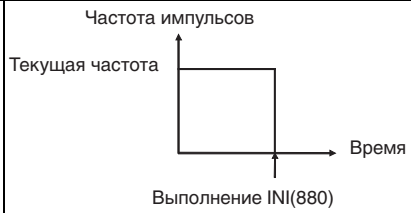
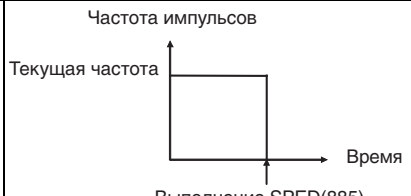
В независимом режиме выдача импульсов будет прервана автоматически, когда будет выдано количество импульсов, заранее заданное командой PULS(886). В непрерывном режиме выдача импульсов будет продолжаться, пока она не будет остановлена другой командой программы.

При попытке перехода из независимого режима в непрерывный или наоборот непосредственно во время выдачи импульсов произойдет ошибка.

■ **Непрерывный режим (регулирование скорости)**

Импульсный выход, запущенный в непрерывном режиме, продолжает работать, пока его не останавливает другая команда программы.

Примечание. Переключение модуля ЦПУ в режим «Программирование» немедленно остановит выдачу импульсов.

Действие	Назначение	Применение	Изменение частоты	Описание	Операция/ команда
Запуск выдачи импульсов	Вывод с указанной скоростью	Мгновенное изменение скорости (частоты)		Выдача импульсов с указанной частотой.	SPED(885) (непрерывный)
Изменение параметров	Ступенчатое изменение скорости	Изменение скорости во время работы		Ступенчатое изменение частоты (увеличение или уменьшение) выходных импульсов.	SPED(885) (непрерывный) ↓ SPED(885) (непрерывный)
Прекращение выдачи импульсов	Прекращение выдачи импульсов	Мгновенная остановка		Немедленное прекращение выдачи импульсов.	SPED(885) (непрерывный) ↓ INI(880)
	Прекращение выдачи импульсов	Мгновенная остановка		Немедленное прекращение выдачи импульсов.	SPED(885) (непрерывный) ↓ SPED(885) (непрерывный, заданная частота 0 Гц)

■ **Независимый режим (позиционирование)**

Импульсный выход, запущенный в независимом режиме, продолжает работать, пока не оказывается выдано указанное количество импульсов.

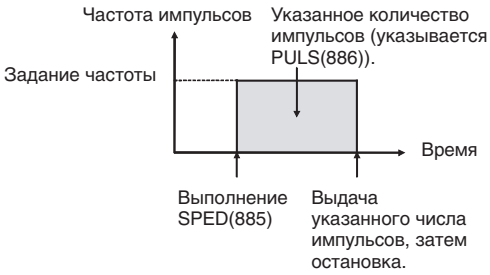
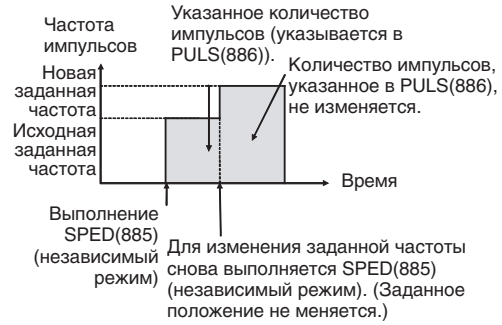
Примечание.

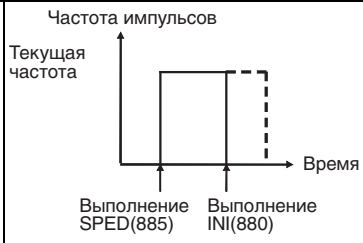
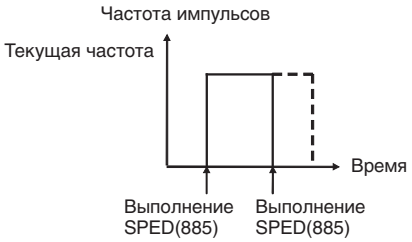
(1) Переключение модуля ЦПУ в режим «Программирование» немедленно остановит выдачу импульсов.

(2) Количество выходных импульсов должно задаваться при каждом перезапуске выхода.

(3) Количество выходных импульсов должно быть заранее задано командой PULS(886). Команда SPED(885) не запустит выдачу импульсов, если перед ней не будет выполнена команда PULS(886).

(4) Если количество импульсов, заданное командой PULS(886), является абсолютным (т. е. не относительным), направление, заданное операндом SPED(885), не учитывается.

Действие	Назначение	Применение	Изменение частоты	Описание	Операция/ команда
Запуск выдачи импульсов	Вывод с указанной скоростью	Позиционирование без разгона или торможения		Запуск выдачи импульсов с указанной частотой и немедленное прекращение после выдачи указанного количества импульсов. **Примечание.** Целевое положение (указанное количество импульсов) невозможно изменить непосредственно во время позиционирования.	PULS(886) ↓ SPED(885) (независимый)
Изменение параметров	Ступенчатое изменение скорости	Ступенчатое изменение скорости во время работы		Во время позиционирования можно мгновенно изменить (увеличить или уменьшить) частоту выходных импульсов, выполнив команду SPED(885). Целевое положение (указанное количество импульсов) не изменяется.	PULS(886) ↓ SPED(885) (независимый) ↓ SPED(885) (независимый)

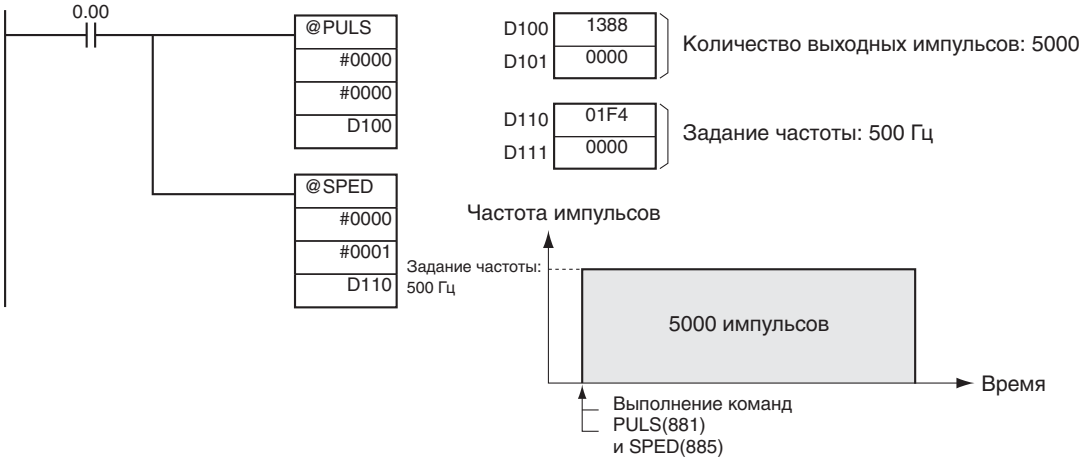
Действие	Назначение	Применение	Изменение частоты	Описание	Операция/ команда
Прекращение выдачи импульсов	Прекращение выдачи импульсов (заданное количество импульсов не сохраняется).	Мгновенная остановка		Немедленное прекращение выдачи импульсов и сброс заданного количества выходных импульсов.	PULS(886) ↓ SPED(885) (независимый) ↓ INI(880)
					PLS2(887) ↓ INI(880)
	Прекращение выдачи импульсов (заданное количество импульсов не сохраняется).	Мгновенная остановка		Немедленное прекращение выдачи импульсов и сброс заданного количества выходных импульсов.	PULS(886) ↓ SPED(885) (независимый) ↓ SPED(885) (независимый, заданная частота 0 Гц)

Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	Включен, если значение P, M или F выходит за допустимый диапазон. Включен, если в данный момент уже выполняется команда PLS2(887) или ORG(889), управляющая выдачей импульсов с указанного порта. Включен, если команда SPED(885) или INI(880) использована для переключения между непрерывным и независимым режимами во время выдачи импульсов. Включен, если команда SPED(885) была выполнена в задаче обработки прерывания, в то время как циклической задачей уже выполнялась команда, управляющая выдачей импульсов. Включен, если команда SPED(885) была выполнена в независимом режиме с абсолютным количеством импульсов, но исходное положение не было установлено.

Пример

При включении бита CIO 0.00 команда PULS(886) задает количество выходных импульсов для импульсного выхода 0. В данном примере задается абсолютное число импульсов 5000. Затем выполняется команда SPED(885), запускающая выдачу импульсов по способу «CW + CCW» в направлении по часовой стрелке в независимом режиме с заданной частотой 500 Гц.

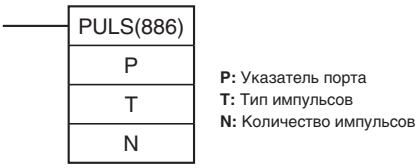


3-20-6 ЗАДАТЬ КОЛ-ВО ИМПУЛЬСОВ: PULS(886)

Назначение

Команда PULS(886) позволяет предварительно задать требуемое количество выходных импульсов. Позже, когда с помощью команды SPED(885) или ACC(888) будет запущена генерация импульсов в независимом режиме, будет выдано именно это количество импульсов.

Символ РКС



Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	PULS(886)
	Однократное выполнение по положительному фронту	@PULS(886)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

P: Указатель порта

Указатель порта указывает порт. Параметры, заданные в операндах D и N, будут применены к следующей выполненной команде SPED(885) или ACC(888), в которой будет указан такой же номер выходного порта.

P	Порт
0000 hex	Импульсный выход 0
0001 hex	Импульсный выход 1
0002 hex	Импульсный выход 2 (только для CP1H)
0003 hex	Импульсный выход 3 (только для CP1H)
0020 hex	Позиционирование на базе ПЧ 0 (только для CP1L)
0021 hex	Позиционирование на базе ПЧ 1 (только для CP1L)

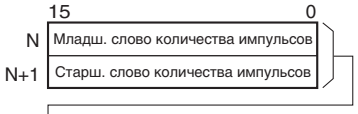
T: Тип импульсов

Операнд T указывает, какое количество импульсов задает команда: относительное или абсолютное.

T	Тип импульсов
0000 hex	Относительные
0001 hex	Абсолютные

N и N+1: Количество импульсов

Слова [N, N+1] содержат 8-разрядное шестнадцатеричное значение количества импульсов. Количество импульсов может быть относительным (указывается относительно исходного положения) или абсолютным (указывает абсолютные координаты целевого положения).



Относительное кол–во импульсов:
0...2 147 483 647 (0000 0000...7FFF FFFF hex)

Абсолютное кол–во импульсов:
–2 147 483 648...2 147 483 647 (8000 0000...7FFF FFFF hex)

Фактическое количество импульсов перемещения, которое будет выдано, определяется описанным ниже образом.

При выдаче относительного количества импульсов количество импульсов перемещения = заданное количество импульсов. При выдаче абсолютного количества импульсов количество импульсов перемещения = заданное количество импульсов - текущее значение.

Характеристики операндов

Область	P	T	N
Область CIO	---	---	CIO 0...CIO 6142
Рабочая область	---	---	W0...W510
Область битов хранения	---	---	H0...H510
Область вспомогательных битов	---	---	A448...A958
Область таймеров	---	---	T0000...T4094
Область счетчиков	---	---	C0000...C4094
Область DM	---	---	D0...D32766
Косвенные адреса DM в двоичном формате	---	---	@ D0...@ D32767
Косвенные адреса DM в двоично-десятичном формате (BCD)	---	---	*D0...*D32767
Постоянные	См. описание операнда.	См. описание операнда.	См. описание операнда.
Регистры данных	---	---	---
Регистры указателей	---	---	---
Косвенная адресация с помощью регистров указателей	---	---	,IR0...,IR15 -2048...+2047 ,IR0... -2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15

Описание

Команда PULS(886) задает тип и количество импульсов, указанные операндами T и N, для порта, указанного операндом P. Сама же выдача импульсов запускается позже с помощью команды SPED(885) или ACC(888) в независимом режиме.

Флаги

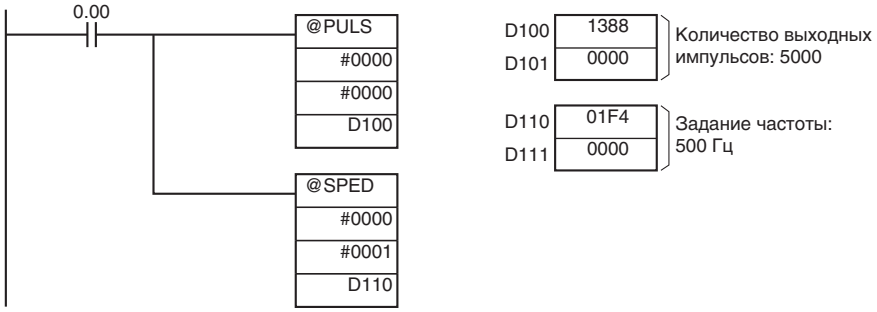
Название	Обозначение	Действие
Флаг ошибки	ER	Включен, если значение P, T или N выходит за допустимый диапазон. Включен, если команда PULS(886) выполнена для порта, который уже выдает импульсы. Включен, если команда PULS(886) была выполнена в задаче обработки прерывания, в то время как в циклической задаче уже выполнялась команда, управляющая выдачей импульсов.

Меры предосторожности

- Если команда PULS(886) будет выполнена для порта, который в данный момент уже выдает импульсы, произойдет ошибка. В связи с этим следует использовать вариант однократного выполнения команды (@PULS(886)) или включать условие выполнения команды только на один цикл программы ПЛК.
- Рассчитанное количество импульсов, выдаваемое в соответствии с командой PULS(886), не изменится, даже если будет выполнена команда INI(880) с целью изменения текущего значения импульсного выхода.
- Если количество импульсов, заданное командой PULS(886), является абсолютным (т. е. не относительным), направление, заданное командой SPED(885) или ACC(888), не учитывается.
- Количество выходных импульсов может превысить допустимый диапазон текущих значений (-2 147 483 648...2 147 483 647).

Пример

При включении бита CIO 0.00 команда PULS(886) задает количество выходных импульсов для импульсного выхода 0. В данном примере задается абсолютное число импульсов 5000. Затем выполняется команда SPED(885), запускающая выдачу импульсов по способу «CW + CCW» в направлении по часовой стрелке в независимом режиме с заданной частотой 500 Гц.



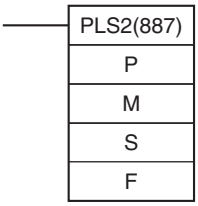
3-20-7 ВЫДАЧА ИМПУЛЬСОВ: PLS2(887)

Назначение

Команда PLS2(887) выдает указанное количество импульсов на указанный порт. Импульсы начинают выдаваться с указанной начальной частотой, разгоняются до заданной частоты с указанным темпом разгона, замедляются с указанным темпом торможения и прекращают выдаваться при частоте, которая практически совпадает с начальной частотой. Поддерживается только позиционирование в независимом режиме.

Команду PLS2(887) также можно выполнить во время выдачи импульсов, чтобы изменить количество выходных импульсов, заданную частоту, темп разгона или темп торможения. Таким образом, с помощью команды PLS2(887) можно поэтапно изменять скорость вращения, используя различные тепы разгона и торможения, а также изменять конечное положение (и одновременно скорость) и направление вращения.

Символ РКС



P: Указатель порта
M: Режим работы выхода
S: Первое слово таблицы настроек
F: Первое слово начальной частоты

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	PLS2(887)
	Однократное выполнение по положительному фронту	@PLS2(887)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

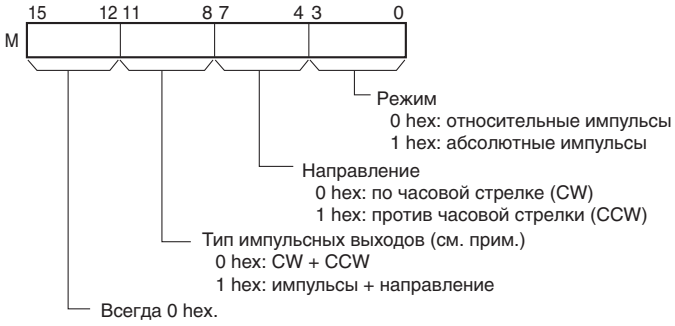
P: Указатель порта

Указатель порта указывает порт.

P	Порт
0000 hex	Импульсный выход 0
0001 hex	Импульсный выход 1
0002 hex	Импульсный выход 2 (только для CP1H)
0003 hex	Импульсный выход 3 (только для CP1H)
0020 hex	Позиционирование на базе ПЧ 0 (только для CP1L)
0021 hex	Позиционирование на базе ПЧ 1 (только для CP1L)

M: Режим работы выхода

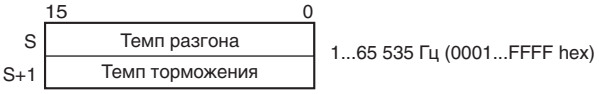
Операнд M содержит параметры работы импульсного выхода, которые показаны на рисунке ниже.



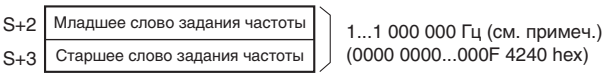
Прим.: Один и тот же тип должен быть указан для импульсных выходов 0 и 1.

S: Первое слово таблицы настроек

На приведенных ниже рисунках показано содержание слов S...S+5, управляющих работой импульсного выхода.

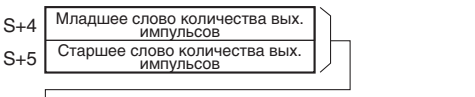


Укажите, на сколько должна возрастать или уменьшаться частота за один интервал управления импульсным сигналом (4 мс).



Укажите значение частоты по завершении разгона (Гц).

Прим.: Максимальное значение частоты, которое может быть указано, зависит от модели и поддержки выдачи импульсов. См. р*уководство по эксплуатации ЦПУ CP1H.*



Относительное кол–во импульсов:
0...2 147 483 647 (0000 0000...7FFF FFFF hex)

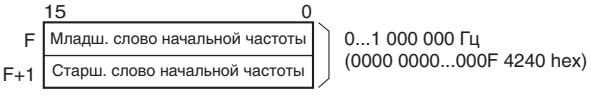
Абсолютное кол–во импульсов:
–2 147 483 648...2 147 483 647 (8000 0000...7FFF FFFF hex)

Фактическое количество импульсов перемещения, которое будет выдано, определяется описанным ниже образом.

При выдаче относительного количества импульсов количество импульсов перемещения = заданное количество импульсов. При выдаче абсолютного количества импульсов количество импульсов перемещения = заданное количество импульсов - текущее значение.

F: Первое слово начальной частоты

Значение начальной частоты (частоты запуска) задается словами F и F+1.



Укажите значение начальной частоты (Гц).

Характеристики операндов

Область	P	M	S	F
Область CIO	---	---	CIO 0...CIO 6138	CIO 0...CIO 6142
Рабочая область	---	---	W0...W506	W0...W510
Область битов хранения	---	---	H0...H506	H0...H510
Область вспомогательных битов	---	---	A448...A954	A448...A958
Область таймеров	---	---	T0000...T4090	T0000...T4094
Область счетчиков	---	---	C0000...C4090	C0000...C4094
Область DM	---	---	D0...D32762	D0...D32766
Косвенные адреса DM в двоичном формате	---	---	@ D0...@ D32767	@ D0...@ D32767
Косвенные адреса DM в двоично-десятичном формате (BCD)	---	---	*D0...*D32767	*D0...*D32767
Постоянные	См. описание операнда.	См. описание операнда.	---	См. описание операнда.
Регистры данных	---	---	---	---
Регистры указателей	---	---	---	---
Косвенная адресация с помощью регистров указателей	---	---	,IR0...,IR15 -2048...+2047 ,IR0... -2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	,IR0...,IR15 -2048...+2047 ,IR0... -2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15

Верхние предельные значения целевой и начальной частоты

Порт	CP1H			CP1L
	X40/XA40, версия 1.0	X40/XA40 версия 1.1 и выше	Y20	
Порт 0	100 кГц	100 кГц	1 МГц	100 кГц
Порт 1	100 кГц	100 кГц	1 МГц	100 кГц
Порт 2	30 кГц	100 кГц	100 кГц	---
Порт 3	30 кГц	100 кГц	100 кГц	---

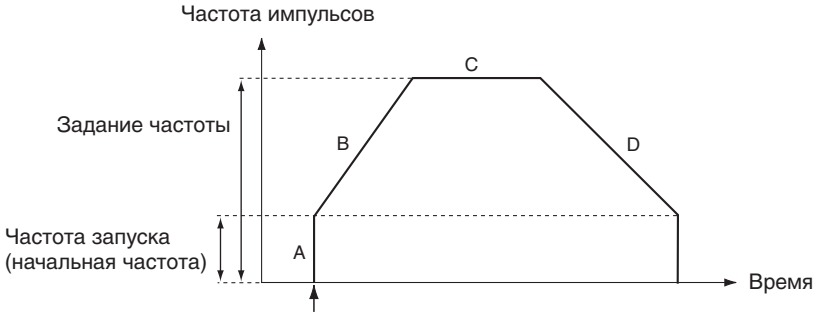
Описание

Команда PLS2(887) запускает выдачу импульсов с порта, который указан операндом P, в режиме, который указан операндом M, с начальной частотой, которая указана операндом F (1 на рисунке ниже). По истечении каждого интервала управления импульсным сигналом (4 мс) частота возрастает на величину, указанную операндом S (темп разгона), пока не оказывается достигнута заданная (целевая) частота, указанная операндом S (2 на рисунке ниже). По достижении заданной частоты разгон прекращается и далее импульсы выдаются с постоянной частотой (скоростью) (3 на рисунке ниже).

Точка начала торможения определяется по количеству выходных импульсов и темпу торможения, заданному в S. Когда эта точка оказывается достигнута, частота начинает снижаться по истечении каждого интервала управления импульсным сигналом (4 мс) на величину, указанную в S (темп торможения), пока не оказывается достигнута начальная частота, также указанная в S. В этой точке выдача импульсов прекращается (4 на рисунке ниже).

Выдача импульсов запускается при каждом выполнении команды PLS2(887). Поэтому в большинстве случаев должен использоваться

вариант однократного выполнения команды (@PLS2(887)), либо условие выполнения должно включаться только на один цикл программы ПЛК.



Команду PLS2(887) можно использовать только для позиционирования.

Команда PLS2(887) может быть выполнена во время выдачи импульсов для команды ACC(888) в независимом или непрерывном режиме, а также во время разгона, вращения с постоянной скоростью или во время торможения. (См. примечание). Команда ACC(888) также может быть выполнена во время выдачи импульсов для PLS2(887) во время разгона, вращения с постоянной скоростью или во время торможения.

Примечание.

(1) Выполнение команды PLS2(887) во время регулирования скорости командой ACC(888) (в непрерывном режиме) с точно такой же целевой частотой, что и у команды ACC(888), позволяет реализовать так называемую фиксированную подачу по прерыванию (перемещение на фиксированное расстояние при наступлении указанных условий). Разгон в этом случае командой PLS2(887) не выполняется, но если будет задан нулевой темп разгона, установится флаг ошибки и команда PLS2(887) выполнена не будет. Поэтому обязательно должно быть задано ненулевое значение темпа разгона.

(2) Может быть задано значение темпа разгона/торможения 1 Гц и выше. Время разгона/торможения сверху не ограничено. Если разница между скоростью запуска и целевой скоростью составляет более 100 кГц, темп разгона/торможения будет автоматически увеличен.

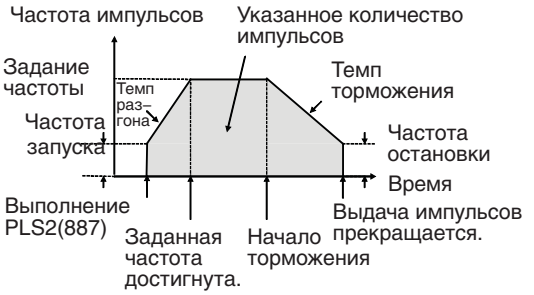
- Если разница между скоростью запуска и целевой скоростью находится в интервале 100...200 кГц, темп разгона/торможения будет составлять 2 Гц или выше.
- Если разница между скоростью запуска и целевой скоростью находится в интервале 200...300 кГц, темп разгона/торможения будет составлять 3 Гц или выше.

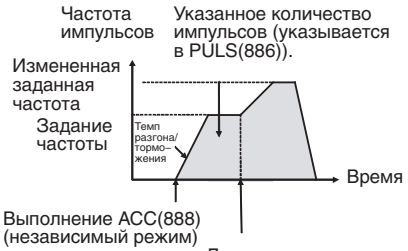
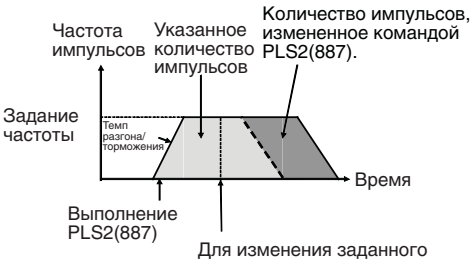
.
.
.
.

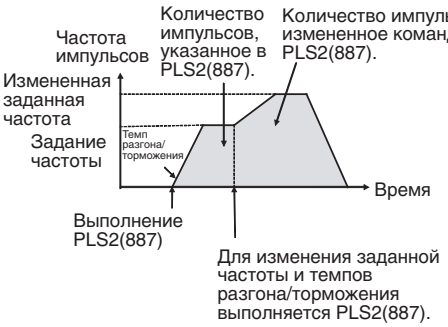
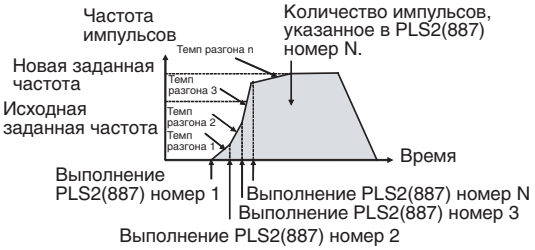
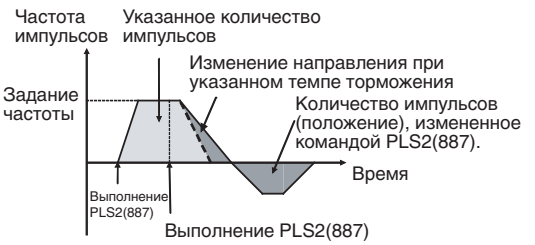
- Если разница между скоростью запуска и целевой скоростью находится в интервале 900...1000 кГц, темп разгона/торможения будет составлять 10 Гц или выше.

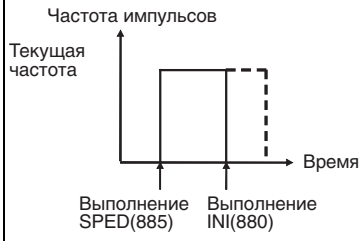
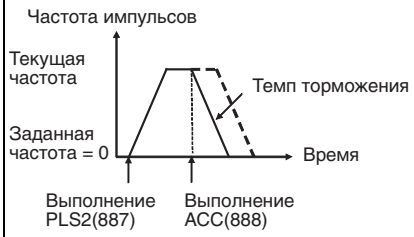
■ **Независимый режим (позиционирование)**

Примечание. Переключение модуля ЦПУ в режим «Программирование» немедленно остановит выдачу импульсов.

Действие	Назначение	Применение	Изменение частоты	Описание	Операция/ команда
Запуск выдачи импульсов	Сложное управление по трапецеидальному закону	Скорость при позиционировании изменяется по трапецеидальному закону (используются разные темпы разгона и торможения, не нулевая начальная скорость) Количество импульсов можно изменять во время позиционирования.		Разгон и торможение с фиксированными темпами (разными). Выдача импульсов прекращается после того, как выдано указанное количество импульсов. (См. примечание). **Примечание.** Целевое положение (указанное количество импульсов) можно изменить непосредственно во время позиционирования.	PLS2(887)

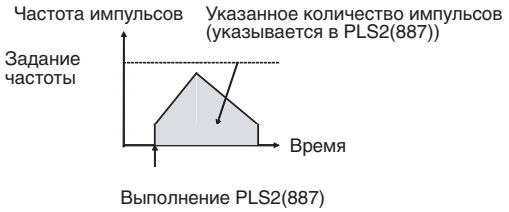
Действие	Назначение	Применение	Изменение частоты	Описание	Операция/команда
Изменение параметров	Плавное изменение скорости (с разными темпами разгона и торможения)	Изменение заданной скорости (частоты) во время позиционирования (разные темпы разгона и торможения)	 Для изменения заданной частоты и темпов разгона/торможения выполняется PLS2(887). (Заданное положение не меняется. Вновь указывается первоначальное заданное положение.)	Во время позиционирования можно изменить темп разгона, темп торможения и заданную частоту, выполнив команду PLS2(887). **Примечание.** Во избежание непреднамеренного изменения целевого положения последнее должно быть изначально указано в абсолютных координатах.	PLS2(887) ↓ PLS2(887) PULS(886) ↓ ACC(888) (независимый) ↓ PLS2(887)
	Изменение заданного положения	Изменение заданного положения во время позиционирования (функция многократного пуска)	 Для изменения заданного положения выполняется PLS2(887). (Заданная частота и темпы разгона/торможения не меняются.)	Во время позиционирования можно изменить целевое положение (количество импульсов), скорость разгона, скорость торможения и заданную частоту, выполнив команду PLS2(887). **Примечание.** Если после изменения параметров поддерживать постоянную скорость невозможно, генерируется ошибка и продолжается выполнение операции с первоначальным целевым положением.	PLS2(887) ↓ PLS2(887) PULS(886) ↓ ACC(888) (независимый) ↓ PLS2(887)

Действие	Назначение	Применение	Изменение частоты	Описание	Операция/ команда
Изменение параметров (продолжение)	Плавное изменение заданного положения и скорости	Изменение заданного положения и заданной скорости (частоты) во время позиционирования (функция многократного пуска)		Во время позиционирования можно изменить целевое положение (количество импульсов), скорость разгона, скорость торможения и заданную частоту, выполнив команду PLS2(887). **Примечание.** Если после изменения параметров поддерживать постоянную скорость невозможно, генерируется ошибка и продолжается выполнение операции с первоначальным целевым положением.	PULS(886) ↓ ACC(888) (независимый) ↓ PLS2(887)
		Изменение темпов разгона и торможения во время позиционирования (функция многократного пуска)		Во время позиционирования (разгона или торможения) можно изменить скорость разгона или скорость торможения, выполнив команду PLS2(887).	PLS2(887) ↓ PLS2(887) PULS(886) ↓ ACC(888) (независимый) ↓ PLS2(887)
	Изменение направления	Изменение направления во время позиционирования		Во время позиционирования с использованием относительного количества импульсов можно перейти к абсолютному количеству импульсов и изменить направление вращения, выполнив команду PLS2(887).	PLS2(887) ↓ PLS2(887) PULS(886) ↓ ACC(888) (независимый) ↓ PLS2(887)

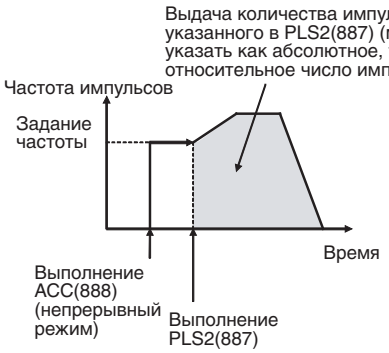
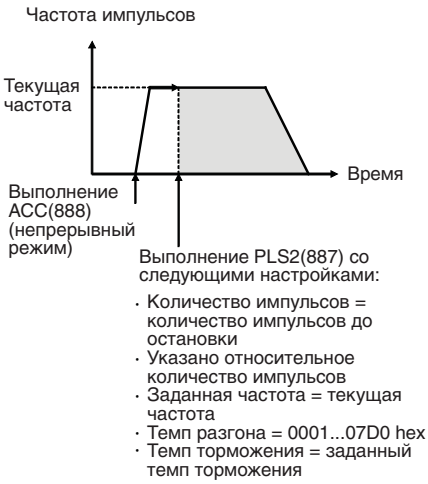
Действие	Назначение	Применение	Изменение частоты	Описание	Операция/ команда
Прекращение выдачи импульсов	Прекращение выдачи импульсов (заданное количество импульсов не сохраняется).	Мгновенная остановка		Немедленное прекращение выдачи импульсов и сброс заданного количества выходных импульсов.	PLS2(887) ↓ INI(880)
	Плавное прекращение выдачи импульсов (заданное количество импульсов не сохраняется).	Торможение до остановки		Прекращение выдачи импульсов с постепенным снижением частоты.	PLS2(887) ↓ ACC(888) (независимый, заданная частота 0 Гц)

Примечание. Треугольный закон изменения скорости

Если указанного количества импульсов недостаточно для достижения заданной частоты и возврата к нулевой частоте, функция автоматически сокращает время разгона/торможения и реализует управление по треугольному закону (осуществляет только разгон и торможение). К возникновению ошибки это не приводит.



■ **Переключение из непрерывного режима (регулирование скорости) в независимый режим (позиционирование)**

Пример применения	Изменение частоты	Описание	Операция/ команда
Переключение из режима регулирования скорости в режим перемещения на фиксированное расстояние во время работы		Во время операции регулирования скорости, начатой командой ACC(888), можно перейти к операции позиционирования, выполнив команду PLS2(887).	ACC(888) (непрерывный) ↓ PLS2(887)
Подача на фиксированное расстояние по прерыванию			

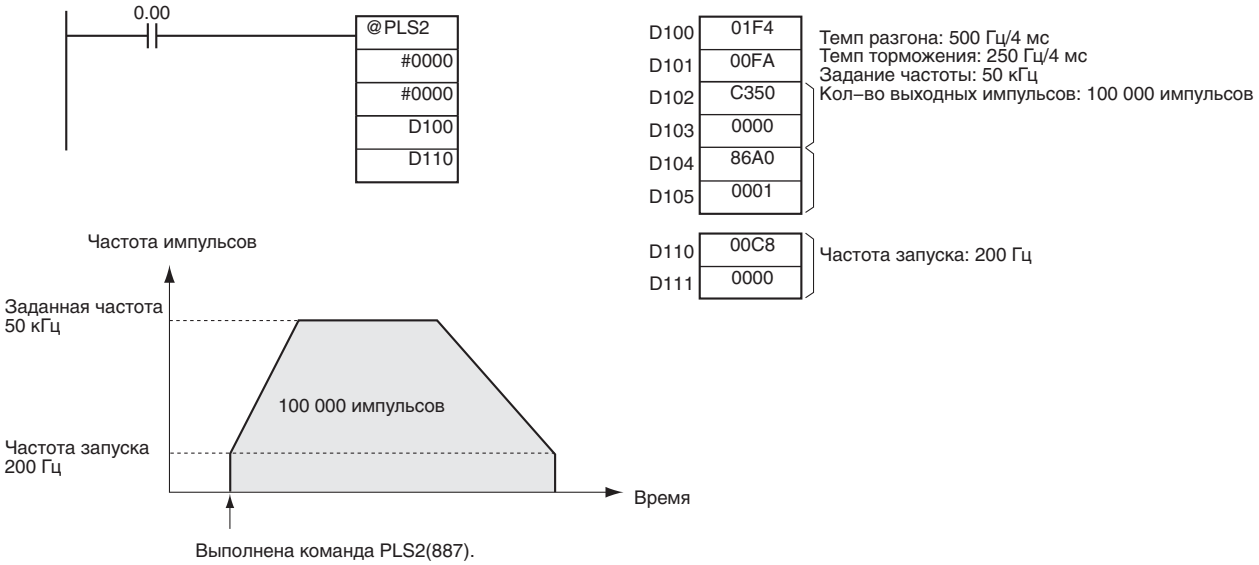
Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	Включен, если значение P, M, S или F выходит за допустимый диапазон. Включен, если команда PLS2(887) выполнена для порта, который уже выдает импульсы по инициативе команды SPED(885) или ORG(889). Включен, если команда PLS2(887) была выполнена в задаче обработки прерывания, в то время как циклической задачей уже выполнялась команда, управляющая выдачей импульсов. Включен, если в команде PLS2(887) указано абсолютное количество импульсов, но исходное положение не установлено.

Пример

При включении бита CIO 0.00 в приведенном далее примере программы команда PLS2(887) инициирует выдачу импульсов с импульсного выхода 0 с абсолютным количеством импульсов 100 000. Частота выдачи импульсов возрастает на 500 Гц каждые 4 мс, начиная с 200 Гц, пока не достигается заданная частота (скорость) 50 кГц. Достигнув точки начала торможения, частота выходных импульсов начинает уменьшаться на 250 Гц каждые 4 мс, пока не достигается начальная частота (скорость) 200 Гц, после чего выдача импульсов прекращается.

Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

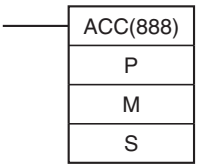


3-20-8 УПРАВЛЕНИЕ РАЗГОНОМ: ACC(888)

Назначение

Команда ACC(888) выдает импульсную последовательность на указанный порт с указанной частотой и указанным темпом разгона/торможения (темп разгона равен темпу торможения). Она подходит как для позиционирования в независимом режиме, так и для управления в режиме постоянной скорости. В случае позиционирования ACC(888) используется в комбинации с командой PULS(886). Команду ACC(888) также можно выполнить непосредственно во время выдачи импульсов для изменения заданной частоты или темпа разгона/торможения, что позволяет плавно (линейно) изменять скорость во время вращения двигателя.

Символ РКС



P: Указатель порта
M: Режим работы выхода
S: Первое слово таблицы настроек

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	ACC(888)
	Однократное выполнение по положительному фронту	@ACC(888)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	**Области пошаговых программ**	**Подпрограммы**	**Задачи обработки прерываний**
OK	OK	OK	OK

Операнды

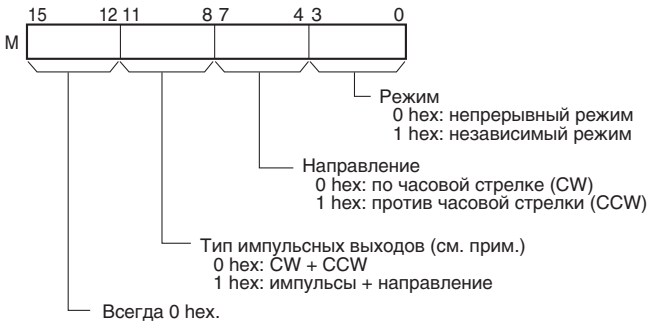
P: Указатель порта

Указатель порта указывает порт (т. е. выход), на который должны выдаваться импульсы.

P	Порт
0000 hex	Импульсный выход 0
0001 hex	Импульсный выход 1
0002 hex	Импульсный выход 2 (только для CP1H)
0003 hex	Импульсный выход 3 (только для CP1H)
0020 hex	Позиционирование на базе ПЧ 0 (только для CP1L)
0021 hex	Позиционирование на базе ПЧ 1 (только для CP1L)

M: Режим работы выхода

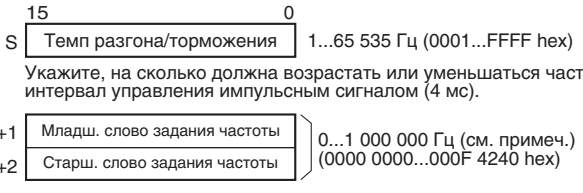
Операнд M содержит параметры работы импульсного выхода, которые показаны на рисунке ниже.



Прим.: Один и тот же тип должен быть указан для импульсных выходов 0 и 1 (имеются только в CP1H).

S: Первое слово таблицы настроек

На приведенных ниже рисунках показано содержание слов S...S+2, управляющих работой импульсного выхода.



Прим.: Максимальное значение частоты, которое может быть указано, зависит от модели и поддержки выдачи импульсов. См. *Руководство по эксплуатации ЦПУ CP1H*.

Характеристики операндов

Область	P	M	S
Область CIO	---	---	CIO 0...CIO 6141
Рабочая область	---	---	W0...W509
Область битов хранения	---	---	H0...H509
Область вспомогательных битов	---	---	A448...A957
Область таймеров	---	---	T0000...T4093
Область счетчиков	---	---	C0000...C4093
Область DM	---	---	D0...D32765
Косвенные адреса DM в двоичном формате	---	---	@ D0...@ D32767

Область	P	M	S
Косвенные адреса DM в двоично-десятичном формате (BCD)	---	---	*D0...*D32767
Постоянные	См. описание операнда.	См. описание операнда.	---
Регистры данных	---	---	---
Регистры указателей	---	---	---
Косвенная адресация с помощью регистров указателей	---	---	,IR0...,IR15 -2048...+2047 ,IR0... -2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15

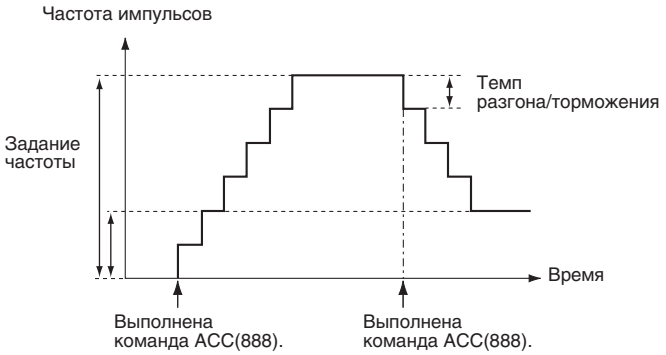
Верхние предельные значения задаваемой частоты

Порт	CP1H			CP1L
	X40/XA40, версия 1.0	X40/XA40, версия 1.1 и выше	Y20	
Порт 0	100 кГц	100 кГц	1 МГц	100 кГц
Порт 1	100 кГц	100 кГц	1 МГц	100 кГц
Порт 2	30 кГц	100 кГц	100 кГц	---
Порт 3	30 кГц	100 кГц	100 кГц	---

Описание

Команда ACC(888) инициирует выдачу импульсов с порта, который указан операндом P, в режиме, который указан операндом M, с использованием заданной частоты и темпов разгона/торможения, которые указаны операндом S. Частота повышается по истечении каждого интервала управления импульсным сигналом (4 мс) на величину, указанную операндом S (темп разгона), пока не оказывается достигнута заданная частота, также указанная операндом S.

Выдача импульсов запускается при каждом выполнении команды ACC(888). Поэтому в большинстве случаев должен использоваться вариант однократного выполнения команды (@ACC(888)), либо условие выполнения должно включаться только на один цикл программы ПЛК.



В независимом режиме выдача импульсов прекращается автоматически, когда оказывается выдано указанное количество импульсов. В непрерывном режиме выдача импульсов продолжается, пока ее не останавливает другая команда программы.

Попытка переключения из независимого режима в непрерывный или наоборот непосредственно во время выдачи импульсов приведет к ошибке.

Команда PLS2(887) может быть выполнена во время выдачи импульсов для команды ACC(888) в независимом или непрерывном режиме, а также во время разгона, вращения с постоянной скоростью или во время торможения. (См. примечание). Команда ACC(888) также может быть выполнена во время выдачи импульсов для PLS2(887) во время разгона, вращения с постоянной скоростью или во время торможения.

Примечание.

(1) Выполнение команды PLS2(887) во время регулирования скорости командой ACC(888) (в непрерывном режиме) с точно такой же целевой частотой, что и у команды ACC(888), позволяет реализовать так называемую фиксированную подачу по прерыванию (перемещение на фиксированное расстояние при наступлении указанных условий). Разгон в этом случае командой PLS2(887) не выполняется, но если будет задан нулевой темп разгона, установится флаг ошибки и команда PLS2(887) выполнена не будет. Поэтому обязательно должно быть задано ненулевое значение темпа разгона.

(2) Может быть задано значение темпа разгона/торможения 1 Гц и выше. Время разгона/торможения сверху не ограничено. Если разница между скоростью запуска и целевой скоростью составляет более 100 кГц, темп разгона/торможения будет автоматически увеличен.

- Если разница между скоростью запуска и целевой скоростью находится в интервале 100...200 кГц, темп разгона/торможения будет составлять 2 Гц или выше.
- Если разница между скоростью запуска и целевой скоростью находится в интервале 200...300 кГц, темп разгона/торможения будет составлять 3 Гц или выше.

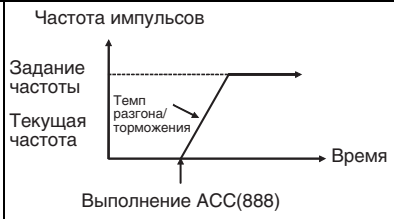
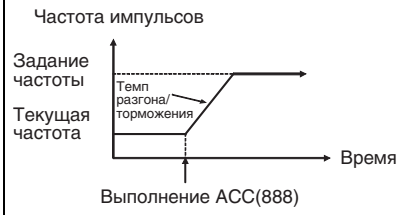
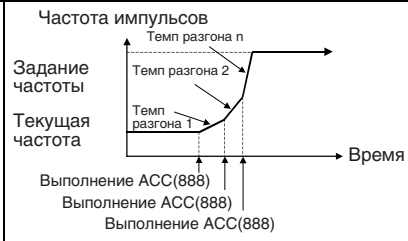
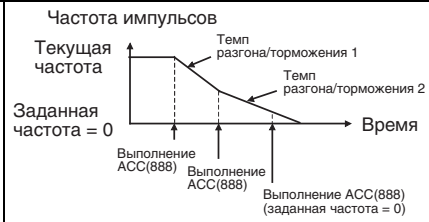
.
.
.
.

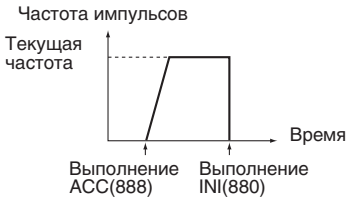
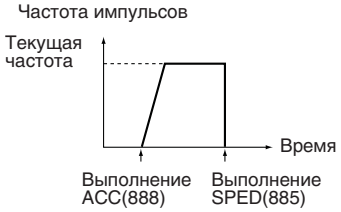
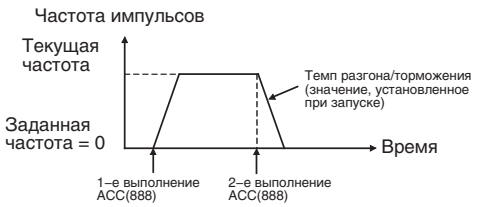
- Если разница между скоростью запуска и целевой скоростью находится в интервале 900...1000 кГц, темп разгона/торможения будет составлять 10 Гц или выше.

■ **Непрерывный режим (регулирование скорости)**

Выдача импульсов продолжается непрерывно, пока ее не прекращает другая команда программы.

Примечание. Переключение модуля ЦПУ в режим «Программирование» немедленно остановит выдачу импульсов.

Действие	Назначение	Применение	Изменение частоты	Описание	Операция/ команда
Запуск выдачи импульсов	Вывод с указанным разгоном и скоростью	Рост скорости (частоты) с указанным темпом		Выдача импульсов и изменение частоты с фиксированной скоростью.	ACC(888) (непрерывный)
Изменение параметров	Плавное изменение скорости	Плавное изменение скорости во время работы		Изменение текущего значения частоты с фиксированной скоростью. Возможно увеличение (разгон) или уменьшение (торможение) частоты.	ACC(888) или SPED(885) (продолжительный) ↓ ACC(888) непрерывный)
		Поэтапное изменение скорости во время работы		Изменение темпа разгона или торможения во время разгона или торможения.	ACC(888) (непрерывный) ↓ ACC(888) (непрерывный)
		Остановка с плавным изменением скорости		В процессе остановки меняется темп торможения. **Примечание.** После того как задание скорости принимает значение 0 Гц, все оставшееся время используется текущий темп торможения.	ACC(888) (непрерывный) ↓ ACC(888) (непрерывный) ↓ ACC(888) (непрерывный, нулевое задание частоты)

Действие	Назначение	Применение	Изменение частоты	Описание	Операция/ команда
Прекращение выдачи импульсов	Прекращение выдачи импульсов	Мгновенная остановка		Выдача импульсов немедленно прекращается.	ACC(888) (непрерывный) ↓ INI(880) (непрерывный)
	Прекращение выдачи импульсов	Мгновенная остановка		Выдача импульсов немедленно прекращается.	ACC(888) (непрерывный) ↓ SPED(885) (непрерывный, заданная частота 0 Гц)
	Плавное прекращение выдачи импульсов	Торможение до остановки		Частота выходных импульсов постепенно снижается вплоть до полной остановки. **Примечание.** Если задание частоты при втором выполнении ACC(888) имеет значение 0 Гц, продолжает использоваться темп торможения, действовавший при первом выполнении ACC(888).	ACC(888) (непрерывный) ↓ ACC(888) (непрерывный, нулевое задание частоты)

■ **Независимый режим (позиционирование)**

Импульсный выход, запущенный в независимом режиме, продолжает работать, пока не оказывается выдано указанное количество импульсов.

Точка начала торможения определяется по количеству выходных импульсов и темпу торможения, заданному в S. Когда эта точка оказывается достигнута, частота начинает снижаться по истечении каждого интервала управления импульсным сигналом (4 мс) на величину, указанную в S (темп торможения), пока не оказывается выдано указанное количество импульсов. В этой точке выдача импульсов прекращается (4 на рисунке ниже).

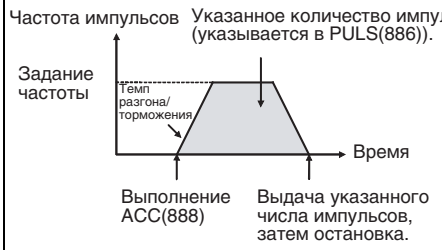
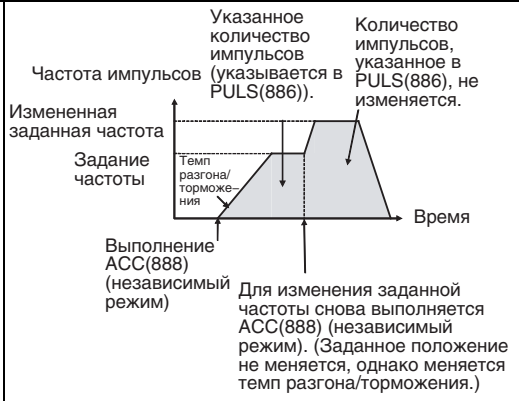
Примечание.

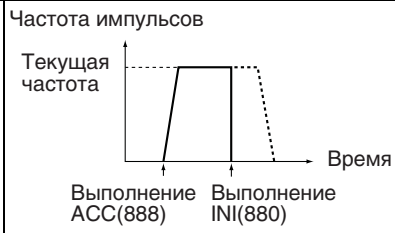
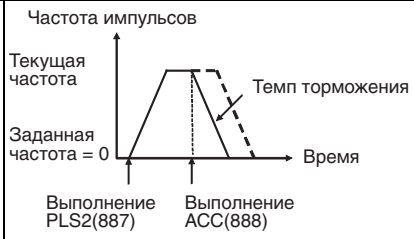
(1) Переключение модуля ЦПУ в режим «Программирование» немедленно остановит выдачу импульсов.

(2) Количество выходных импульсов должно задаваться при каждом перезапуске выхода.

(3) Количество выходных импульсов должно быть заранее задано командой PULS(886). Команда ACC(888) не запустит выдачу импульсов, если перед ней не будет выполнена команда PULS(886).

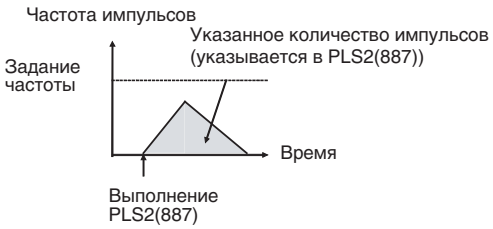
(4) Если количество импульсов, заданное командой PULS(886), является абсолютным (т. е. не относительным), направление, заданное операндом ACC(888), не учитывается.

Действие	Назначение	Применение	Изменение частоты	Описание	Операция/ команда
Запуск выдачи импульсов	Простой трапецеидальный закон управления	Скорость при позиционировании изменяется по трапецеидальному закону (одинаковый темп разгона и торможения, нулевая начальная скорость). Количество импульсов невозможно изменить во время позиционирования.		Разгон и торможение с одним и тем же фиксированным темпом, мгновенное прекращение после выдачи указанного количества импульсов. (См. примечание). **Примечание.** Целевое положение (указанное количество импульсов) невозможно изменить непосредственно во время позиционирования.	PULS(886) ↓ ACC(888) (независимый)
Изменение параметров	Плавное изменение скорости (с одинаковым темпом разгона и торможения)	Изменение заданной скорости (частоты) во время позиционирования (темп разгона = темп торможения)		Во время позиционирования можно изменить темп разгона/торможения и заданную частоту, выполнив команду ACC(888). Целевое положение (указанное количество импульсов) не изменяется.	PULS(886) ↓ ACC(888) или SPED(885) (независимый) ↓ ACC(888) (независимый)

Действие	Назначение	Применение	Изменение частоты	Описание	Операция/ команда
Прекращение выдачи импульсов	Для прекращения выдачи импульсов (заданное количество импульсов не сохраняется).	Мгновенная остановка		Выдача импульсов немедленно прекращается, оставшееся количество выходных импульсов обнуляется.	PULS(886) ↓ ACC(888) (независимый) ↓ INI(880)
	Плавное прекращение выдачи импульсов (заданное количество импульсов не сохраняется).	Торможение до остановки		Прекращение выдачи импульсов с постепенным снижением частоты. **Примечание.** Если работа импульсного выхода запускается командой ACC(888), остается в силе первоначальный темп разгона/торможения. Если работу запускает команда SPED(885), темп разгона/торможения будет недействительным и выдача импульсов будет немедленно остановлена.	PULS(886) ↓ ACC(888) или SPED(885) (независимый) ↓ ACC(888) (независимый, заданная частота 0 Гц) PLS2(887) ↓ ACC(888) (независимый, заданная частота 0 Гц)

Примечание. Треугольный закон изменения скорости

Если указанного количества импульсов недостаточно для достижения заданной частоты и возврата к нулевой частоте, функция автоматически сокращает время разгона/торможения и реализует управление по треугольному закону (осуществляет только разгон и торможение). К возникновению ошибки это не приводит.

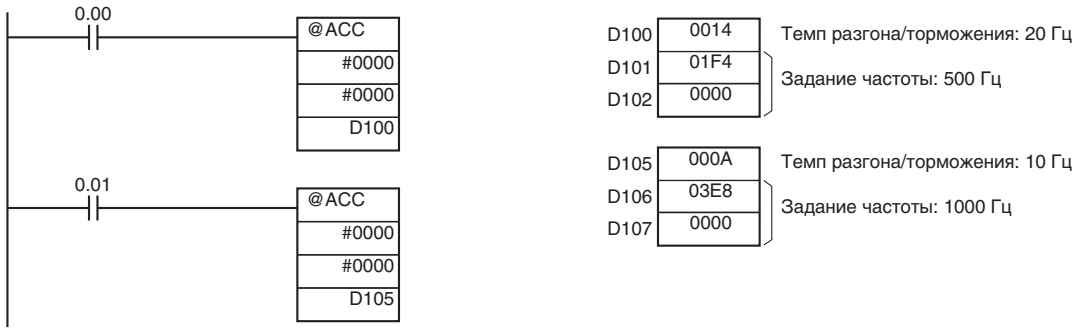


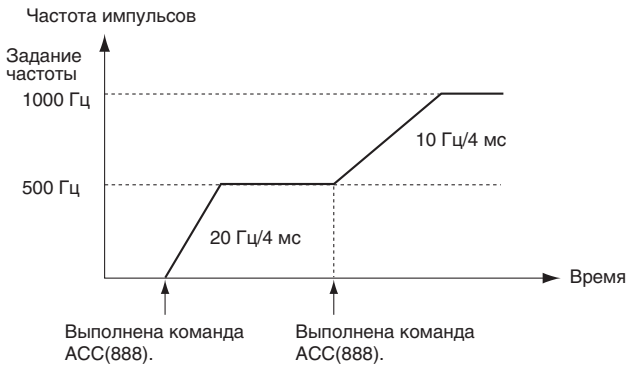
Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	Включен, если значение P, M или S выходит за допустимый диапазон. Включен, если импульсы с указанного порта уже выдаются командой ORG(889). Включен, если команда ACC(888) выполняется с целью переключения между независимым и непрерывным режимами для порта, с которого уже выдаются импульсы командой SPED(885), ACC(888) или PLS2(887). Включен, если команда ACC(888) была выполнена в задаче обработки прерывания, в то время как циклической задачей уже выполнялась команда, управляющая выдачей импульсов. Включен, если команда ACC(888) выполнена для выдачи абсолютного количества импульсов в независимом режиме, но исходное положение не установлено.

Пример

При включении бита CIO 0.00 команда ACC(888) запускает выдачу импульсов с импульсного выхода 0 в непрерывном режиме, в направлении по часовой стрелке, по способу «CW + CCW». Частота выходных импульсов возрастает на 20 Гц каждые 4 мс, пока не достигается заданная частота 500 Гц. Когда включается бит CIO 0.01, команда ACC(888) устанавливает новый темп разгона. Частота выходных импульсов возрастает на 10 Гц каждые 4 мс, пока не достигается заданная частота 1000 Гц.





3-20-9 ПОИСК ИСХОДНОГО ПОЛОЖЕНИЯ: ORG(889)

Назначение

Команда ORG(889) выполняет операцию поиска или возврата в исходное положение.

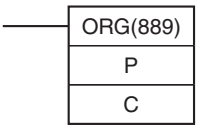
■ **Поиск исходного положения**

Выдается импульсная последовательность с применением указанного метода, вызывающая вращение двигателя. На основании сигнала приближения к исходному положению и сигнала достижения исходного положения определяются координаты исходного положения.

■ **Возврат в исходное положение**

Система позиционирования возвращается в предварительно установленное исходное положение.

Символ РКС



P: Указатель порта
C: Управляющие данные

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	ORG(889)
	Однократное выполнение по положительному фронту	@ORG(889)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

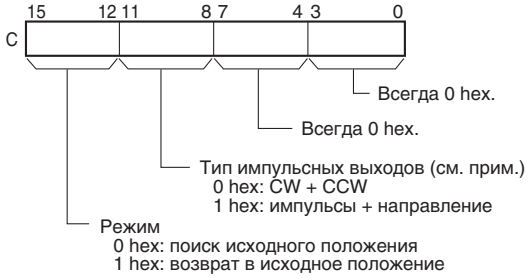
P: Указатель порта

Указатель порта указывает порт (т. е. выход), на который должны выдаваться импульсы.

P	Порт
0000 hex	Импульсный выход 0
0001 hex	Импульсный выход 1
0002 hex	Импульсный выход 2 (только для CP1H)
0003 hex	Импульсный выход 3 (только для CP1H)
0020 hex	Позиционирование на базе ПЧ 0 (только для CP1L)
0021 hex	Позиционирование на базе ПЧ 1 (только для CP1L)

C: Управляющие данные

Значение операнда C определяет используемый метод поиска исходного положения.



Прим.: Один и тот же тип должен быть указан для импульсных выходов 0 и 1.

Характеристики операндов

Область	P	C
Область CIO	---	---
Рабочая область	---	---
Область битов хранения	---	---
Область вспомогательных битов	---	---
Область таймеров	---	---
Область счетчиков	---	---
Область DM	---	---
Косвенные адреса DM в двоичном формате	---	---
Косвенные адреса DM в двоично-десятичном формате (BCD)	---	---
Постоянные	См. описание операнда.	См. описание операнда.
Регистры данных	---	---
Регистры указателей	---	---
Косвенная адресация с помощью регистров указателей	---	---

Описание

Команда ORG(889) выполняет операцию поиска исходного положения или возврата в исходное положение для порта, указанного операндом P, с использованием метода, указанного операндом C.

Перед выполнением команды ORG(889) в настройках ПЛК должны быть заданы следующие параметры. Подробную информацию см. в *Руководстве по эксплуатации ЦПУ CP1H*.

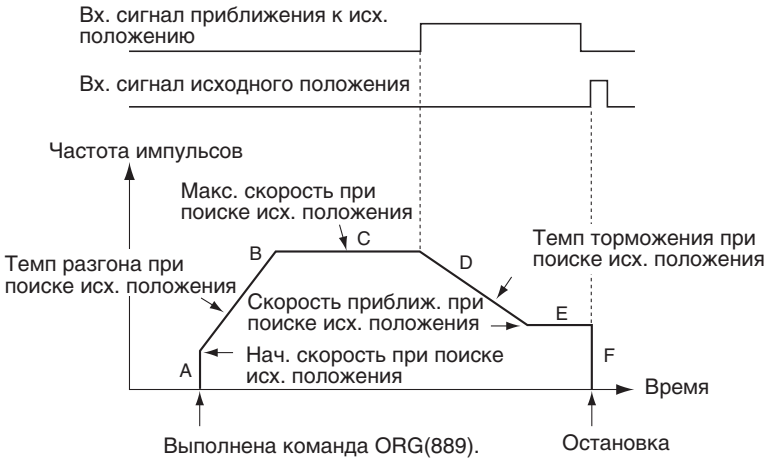
Поиск исходного положения	Возврат в исходное положение
Включение/выключение функции поиска исходного положения Режим работы для поиска исходного положения Режим поиска исходного положения Способ обнаружения исходного положения Направление поиска исходного положения Начальная скорость поиска/возврата в исходное положение Максимальная скорость поиска исходного положения Скорость приближения при поиске исходного положения Компенсация исходного положения Темп разгона при поиске исходного положения Темп торможения при поиске исходного положения Тип входного сигнала предельного хода Тип входного сигнала приближения к исходному положению Тип входного сигнала исходного положения	Начальная скорость поиска/возврата в исходное положение Заданная скорость возврата Темп разгона при возврате в исходное положение Темп торможения при возврате в исходное положение

Операция поиска или возврата в исходное положение запускается каждый раз, когда выполняется команда ORG(889). Поэтому в общем случае должен использоваться вариант однократного выполнения (@ORG(889)), либо условие выполнения должно включаться только на один цикл программы ПЛК.

■ **Поиск исходного положения (биты 12...15 слова C = 0 hex)**

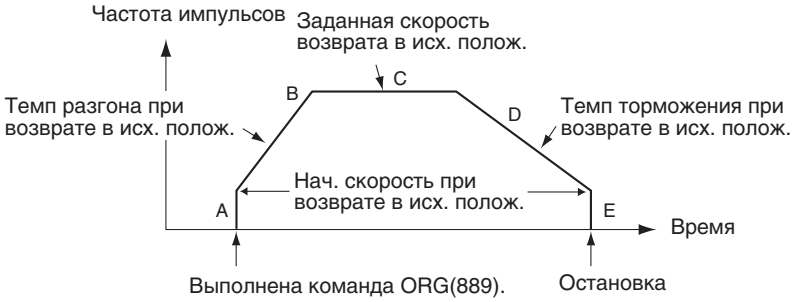
Команда ORG(889) запускает выдачу импульсов с использованием указанного метода с заданной начальной скоростью поиска исходного положения (1 на рисунке ниже). Частота выходных импульсов «разгоняется» до заданной максимальной скорости поиска с заданным темпом разгона при поиске (2 на рисунке ниже). Далее импульсы выдаются с постоянной частотой (т. е. двигатель вращается с постоянной скоростью) до тех пор, пока не включается входной сигнал приближения к исходному положению (3 на рисунке ниже), после чего частота выходных импульсов «тормозится» до заданной скорости приближения к исходному положению с заданным темпом торможения при поиске (4 на рисунке ниже). Затем импульсы вновь выдаются с постоянной частотой, пока не включается входной сигнал достижения исходного положения (5 на рисунке ниже). Когда включается сигнал достижения исходного положения (6 на рисунке ниже), выдача импульсов прекращается.

После того как операция поиска исходного положения будет завершена, включится выход сброса счетчика ошибки. На описанный выше ход поиска также влияет выбранный режим работы, способ обнаружения исходного положения и другие параметры. Подробную информацию см. в *Руководстве по эксплуатации ЦПУ CP1H*.



■ **Возврат в исходное положения (биты 12...15 слова C = 1 hex)**

Команда ORG(889) запускает выдачу импульсов с использованием указанного метода с заданной начальной скоростью возврата в исходное положение (1 на рисунке ниже). Частота выходных импульсов «разгоняется» до заданной постоянной скорости возврата в исходное положение с темпом разгона при возврате (2 на рисунке ниже), после чего импульсы выдаются с постоянной частотой (3 на рисунке ниже). Положение точки начала торможения определяется по количеству импульсов, остающемуся до исходного положения, и заданному темпу торможения. По достижении этой точки частота выходных импульсов начинает «тормозиться» (4 на рисунке ниже) с заданным темпом торможения при возврате, пока не достигается заданная начальная скорость возврата в исходное положение. На этом возврат в исходное положение завершается и выдача импульсов прекращается (5 на рисунке ниже).

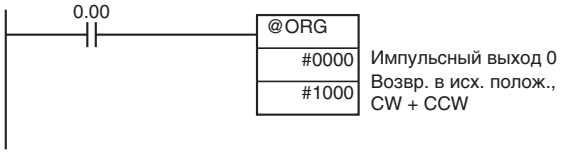


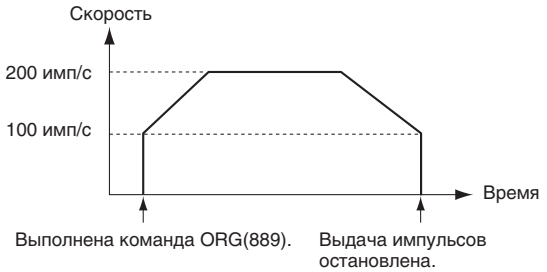
Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	Включен, если значение P или C выходит за допустимый диапазон. Включен, если команда ORG(889) выполнена для порта, с которого уже выдаются импульсы командой SPED(885), ACC(888) или PLS2(887). Включен, если команда ORG(889) была выполнена в задаче обработки прерывания, в то время как в циклической задаче уже выполнялась команда, управляющая выдачей импульсов. Включен, если значения параметров поиска или возврата в исходное положение в настройках ПЛК выходят за допустимый диапазон. Включен, если максимальная скорость поиска исходного положения меньше или равна скорости приближения к исходному положению, либо если скорость приближения к исходному положению меньше или равна начальной скорости поиска исходного положения. Включен, если заданная постоянная скорость при возврате в исходное положение меньше или равна начальной скорости возврата в исходное положение. Включен, если была запущена операция возврата в исходное положение, но не было заранее определено исходное положение.

Пример

При включении бита CIO 0.00 команда ORG(889) запускает операцию возврата в исходное положение для импульсного выхода 0 с использованием двух импульсных сигналов направления (CW + CCW). В настройках ПЛК заданы: начальная скорость 100 имп/с, целевая постоянная скорость 200 имп/с и темп разгона/торможения 50 Гц/4 мс.





Значения параметров в настройках ПЛК указаны в следующей таблице.

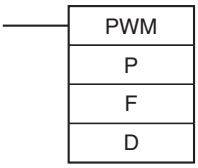
Параметр	Значение
Начальная скорость поиска/возврата для имп. выхода 0	0000 0064 hex: 100 имп/с
Заданная скорость возврата для имп. выхода 0	0000 00C8 hex: 200 имп/с
Темп разгона при возврате в исх. положение для имп. выхода 0	0032 hex: 50 hex/4 мс
Темп торможения при возврате в исх. положение для имп. выхода 0	0032 hex: 50 hex/4 мс

3-20-10 ИМПУЛЬСЫ С ПЕРЕМЕННОЙ СКВАЖНОСТЬЮ: PWM(891)

Назначение

Команда PWM(891) предназначена для выдачи импульсной последовательности с указанным коэффициентом заполнения с указанного порта.

Символ РКС



Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	PWM(891)
	Однократное выполнение по положительному фронту	@PWM(891)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

P: Указатель порта

Указатель порта указывает порт (т. е. выход), на который должны выдаваться импульсы.

P	Порт
0000 hex	Импульсный выход 0 Коэффициент заполнения с шагом 1%, частота с шагом 0,1 Гц
0001 hex	Импульсный выход 1 Коэффициент заполнения с шагом 1%, частота с шагом 0,1 Гц
1000 hex	Импульсный выход 0 Коэффициент заполнения с шагом 0,1%, частота с шагом 0,1 Гц
1001hex	Импульсный выход 1 Коэффициент заполнения с шагом 0,1%, частота с шагом 0,1 Гц
0100 hex	Импульсный выход 0 (только CP1L) Коэффициент заполнения с шагом 1%, частота с шагом 1 Гц
0101 hex	Импульсный выход 1 (только CP1L) Коэффициент заполнения с шагом 1%, частота с шагом 1 Гц
0100 hex	Импульсный выход 0 (только CP1L) Коэффициент заполнения с шагом 0,1%, частота с шагом 1 Гц
1101 hex	Импульсный выход 1 (только CP1L) Коэффициент заполнения с шагом 0,1%, частота с шагом 1 Гц

F: Частота

Операнд F указывает частоту следования выходных импульсов. Значение F должно находиться в одном из указанных ниже диапазонов.

- При указании с шагом 0,1 Гц: 0001...FFFF hex (от 0,1 до 6553,5 Гц)
- При указании с шагом 1 Гц (только CPL1): 0001...8020 hex (от 1 до 32 800 Гц)

Указанная погрешность формы импульсной последовательности, создаваемой командой PMW(891) (коэфф. заполн. +5%/-0%), применима только для диапазона частот от 0,1 до 1000 Гц, что связано с ограниченными техническими возможностями выходной цепи.

D: Коэффициент заполнения

Операнд D указывает коэффициент заполнения последовательности выходных импульсов (т. е. отношение времени включенного состояния выхода к периоду) в процентах. Значение D должно находиться в одном из указанных ниже диапазонов.

- При указании с шагом 0,1%: 0000...03E8 hex (от 0,0% до 100,0%)
- При указании с шагом 1%: 0000...0064 hex (от 0% до 100%)

Характеристики операндов

Область	P	F	D
Область CIO	---	CIO 0...CIO 6143	CIO 0...CIO 6143
Рабочая область	---	W0...W511	W0...W511
Область битов хранения	---	H0...H511	H0...H511
Область вспомогательных битов	---	A448...A959	A448...A959
Область таймеров	---	T0000...T4095	T0000...T4095
Область счетчиков	---	C0000...C4095	C0000...C4095
Область DM	---	D0...D32767	D0...D32767
Косвенные адреса DM в двоичном формате	---	@ D0...@ D32767	@ D0...@ D32767
Косвенные адреса DM в двоично-десятичном формате (BCD)	---	*D0...*D32767	*D0...*D32767
Постоянные	См. описание операнда.	0000...FFFF hex	0000...03E8 hex
Регистры данных	---	DR0...DR15	DR0...DR15
Регистры указателей	---	---	---
Косвенная адресация с помощью регистров указателей	---	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

Описание

Команда PWM(891) выдает с порта, указанного операндом P, последовательность импульсов с частотой, которая указана в F, и коэффициентом заполнения, который указан в D. Команду PWM(891) можно выполнять во время выдачи импульсов с целью изменения текущего коэффициента заполнения, не прерывая выдачу импульсов. Попытки изменения частоты при этом игнорируются.

Выдача импульсов запускается при каждом выполнении команды PWM(891). Поэтому в общем случае должен использоваться вариант однократного выполнения (@PWM(891)), либо условие выполнения должно включаться только на один цикл программы ПЛК.

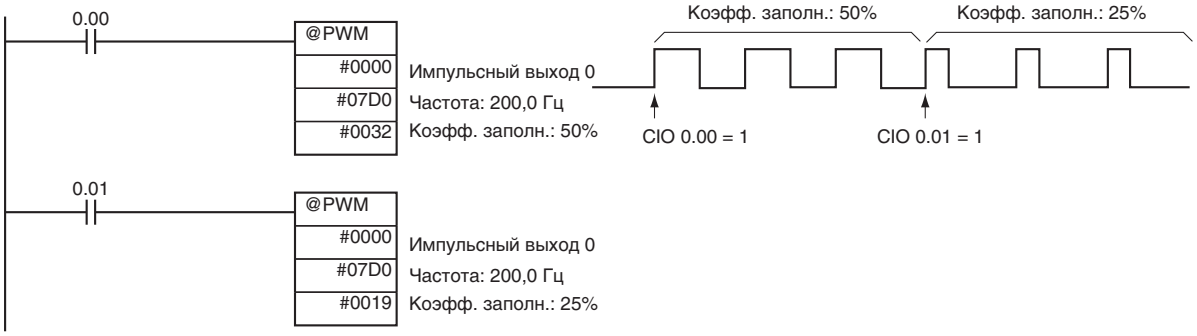
Выдача импульсов будет продолжаться до тех пор, пока она не будет остановлена с помощью команды INI(880) (C = 0003 hex: остановить выдачу импульсов) или пока модуль ЦПУ не будет переключен в режим «Программирование».

Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	Включен, если значение P, F или D выходит за допустимый диапазон. Включен, если импульсы с указанного порта уже выдаются командой ORG(889). Включен, если команда PWM(891) была выполнена в задаче обработки прерывания, в то время как в циклической задаче уже выполнялась команда, управляющая выдачей импульсов.

Пример

При включении бита CIO 0.00 команда PWM(891) запускает выдачу импульсов с импульсного выхода 0 с частотой 200 Гц и коэффициентом заполнения 50%. Когда включается бит CIO 0.01, коэффициент заполнения принимает значение 25%.



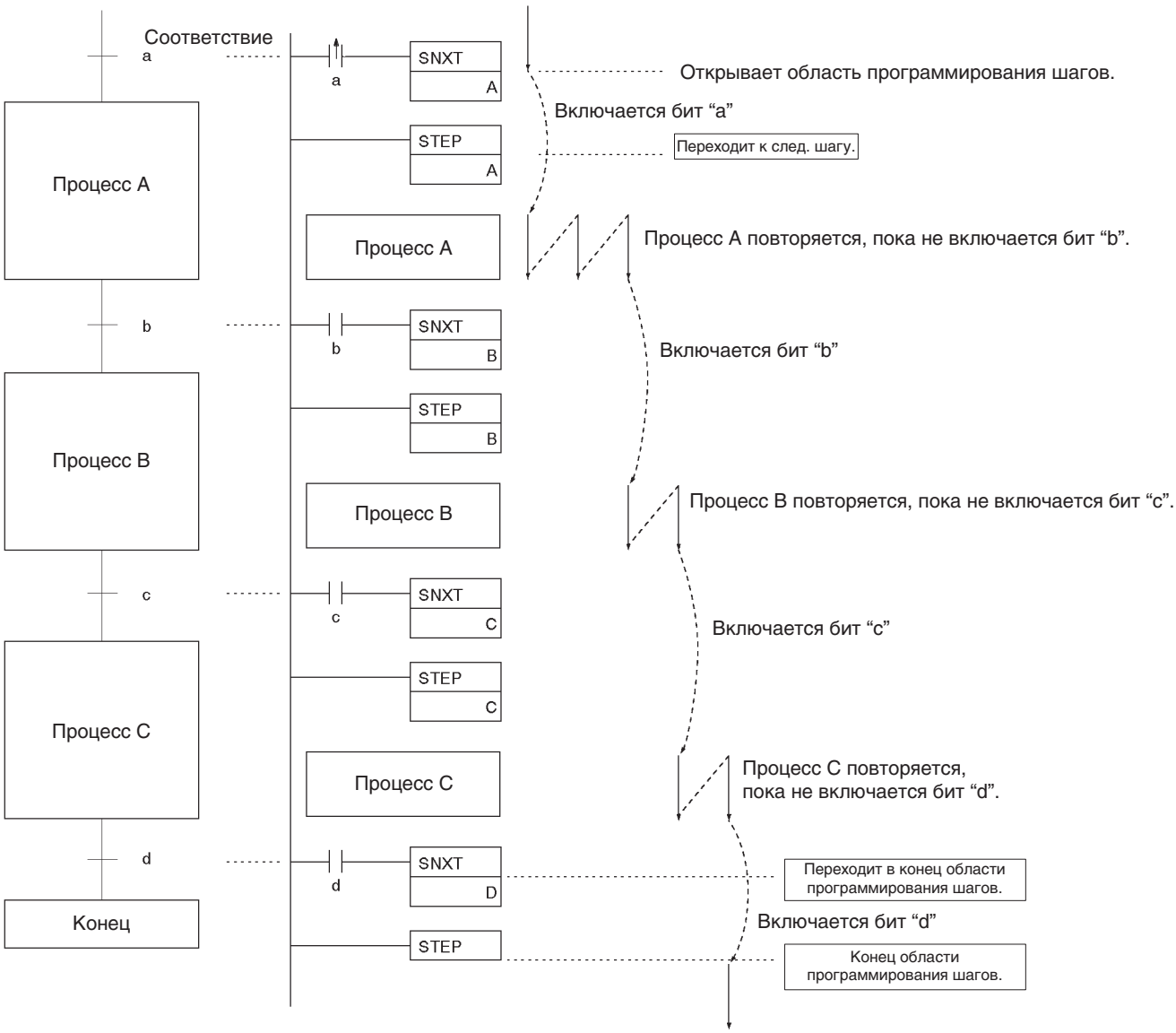
3-21 Команды для пошагового выполнения

В данном разделе описаны команды, предназначенные для программирования пошаговых операций. Эти команды используются как «точки останова», разбивающие программу сравнительно большого размера на отдельные сегменты, которые могут выполняться как отдельные функциональные модули и сбрасываются по завершении выполнения.

Команда	Мнемоническое обозначение	Код функции	Стр.
ОПРЕДЕЛИТЬ ШАГ	STEP	008	825
ЗАПУСТИТЬ ШАГ	SNXT	009	825

Команды STEP(008)/SNXT(009) могут использоваться вместе для программирования пошаговых операций.

Команда	Действие	Диаграмма
SNXT(009): ЗАПУСТИТЬ ШАГ	Управляет переходом к следующему шагу программы.	Соответствие
STEP(008): ОПРЕДЕЛИТЬ ШАГ	Указывает на начало шага. Выполнение программы шага повторяется до тех пор, пока не удовлетворяются условия для перехода к следующему шагу.	Соответствие



Примечание. В качестве битов, управляющих выполнением процессов A, B, C и D, используются рабочие биты.

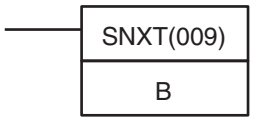
3-21-1 ОПРЕДЕЛИТЬ ШАГ и ЗАПУСТИТЬ ШАГ: STEP(008)/SNXT(009)

Назначение

Команда SNXT(009) размещается непосредственно перед командой STEP(008). Она управляет выполнением шага, включая указанный бит управления. Если команде SNXT(009) предшествует программа другого шага, команда SNXT(009) также выключает бит управления этого предыдущего шага.

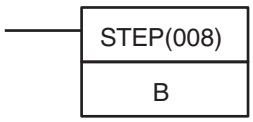
Команда STEP(008) размещается сразу за командой SNXT(009), перед программой шага. Она обозначает начало программы данного шага (процесса) и указывает бит, управляющий выполнением этого шага. Команда STEP(008) также размещается в конце области программирования шагов после самой последней команды SNXT(009) с целью обозначения конца области программирования шагов. Если команда STEP(008) используется для завершения области программирования шагов, бит управления в ней указывать не требуется.

Символы РКС



B: Бит

При обозначении начала программы шага указывается бит управления и РКС-символ выглядит следующим образом:



B: Бит

При обозначении конца программы шага бит управления не указывается и РКС-символ выглядит следующим образом:



Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	STEP(008)/ SNXT(009)
	Однократное выполнение по положительному фронту	Не предусмотрено
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
Не допускается	OK	Не допускается	Не допускается

Характеристики операндов

Область	B
Область CIO	---
Рабочая область	W0.00...W511.15
Область битов хранения	---
Область вспомогательных битов	---
Область таймеров	---
Область счетчиков	---
Область DM	---
Косвенные адреса DM в двоичном формате	---
Косвенные адреса DM в двоично-десятичном формате (BCD)	---
Постоянные	---
Регистры данных	---
Регистры указателей	---
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15

Описание

SNXT(009)

Команда SNXT(009) используется в программе для выполнения одной из трех следующих операций.

1,2,3...

1. Для запуска выполнения пошаговой программы.
2. Для перехода к следующему шагу (биту управления).
3. Для завершения выполнения пошаговой программы.

Область программы, заключенная между самой первой командой STEP(008) (всегда использующей бит управления) и самой последней командой STEP(008) (никогда не использующей бит управления) является областью программирования шагов (областью пошаговой программы).

Запуск выполнения пошаговой программы

Команда SNXT(009) размещается в начале области программирования шагов с целью запуска выполнения пошаговой программы. Она включает указанный операндом B бит управления для следующей команды STEP(008) и передает управление этому шагу (т. е. программе, команды которой располагаются после команды STEP(008), указанной битом управления B). Для команды SNXT(009), обозначающей начало области программирования шагов, должен использоваться вариант однократного выполнения, иначе выполнение шага будет длиться только один цикл.

Переход к следующему шагу

Любая команда SNXT(009), примененная не в начале и не в конце области программирования шагов, служит для перехода к следующему шагу. Она выключает предыдущий бит управления и включает следующий бит управления (указанный операндом B) для следующего шага, тем самым запуская этот шаг (т. е. программу, команды которой находятся после команды STEP(008) с битом управления, который указан операндом B).

Завершение области программирования шагов

Если команда SNXT(009) располагается в самом конце области программирования шагов, она завершает выполнение пошаговой программы и выключает предыдущий бит управления. Хотя бит управления, указываемый операндом B, в данном случае является формальным, он будет включен при выполнении команды, поэтому здесь следует указывать бит, который больше нигде не используется и не приведет к проблемам.

STEP(008)

Команда STEP(008) выполняет одну из двух следующих операций, в зависимости от того, в каком месте программы она используется и указывается ли для нее бит управления.

1,2,3...

1. Запуск определенного шага.
2. Завершение области программирования шагов (т. е. выполнения пошаговой программы).

Запуск шага

Команда STEP(008) размещается в начале каждого шага. Она используется с операндом B, указывающим адрес бита для управления данным шагом.

После того как команда SNXT(009) включает бит управления B, выполняются команды шага, начиная с команды, следующей сразу за

командой STEP(008). С началом выполнения шага также устанавливается флаг выполнения шага (A200.12).

Шаг, выполненный в одном цикле программы, продолжает выполняться в других циклах до тех пор, пока не оказываются соблюдены условия перехода к следующему шагу (т. е. пока команда SNXT(009) не включит бит управления для следующей команды STEP(008)).

Когда команда SNXT (009) включает бит управления для шага, бит управления B текущей команды сбрасывается (выключается), а шаг, управляемый битом B, блокируется.

Состояние и работа выходов и команд в пределах программы шага зависят от состояния (1 или 0) бита управления B. (Состоянием бита управления управляет команда SNXT(009)). Если бит управления B выключен, команды внутри шага находятся в сброшенном и заблокированном состоянии. Взаимосвязь состояний и режимов работы поясняется в следующих таблицах.

Состояние бита управления	Биты и команды
ВКЛ	Команды шага выполняются в обычном порядке.
ВКЛ→ВЫКЛ	Биты и команды внутри шага заблокированы (см. следующую таблицу).
ВЫКЛ	Все команды шага обрабатываются как NOP.

Состояние блокировки (IL)

Выход команды		Состояние
Биты, указанные для команды OUT, OUT NOT		Все выключены
Команды таймера: TIM, TIMX(551), TIMH(015), TIMHX(551), TMHH(540), TIMHHX(552), TIML(542) и TIMLX(553)	Текущее значение	0000 hex (сброс)
	Флаг завершения	ВЫКЛ (сброс)
Биты или слова, указанные для других команд (см. примеч.)		Сохраняют прежнее состояние (но команды не выполняются)

Примечание. Подразумеваются все остальные команды, такие как TTIM(087), TTIMX(555), MTIM(543), MTIMX(554), SET, REST, CNT, CNTX(546), CNTR(012), CNTRX(548), SFT(010) и KEEP(011).

Команда STEP(008) должна размещаться в начале каждого шага. Команда STEP(008) размещается в начале области шага и указывает, что после нее начинается программа шага.

Завершение области программирования шагов

Команда STEP(008) размещается в конце области программирования шагов без операнда. Она обозначает конец пошаговой программы. Команда SNXT(009) выключает предыдущий бит управления и прекращает выполнение пошаговой программы.

Флаги: STEP(008)

Название	Обозначение	Действие
Флаг ошибки	ER	Включен, если указанный бит B находится не в области WR. Включен, если команда STEP(008) используется в программе обработки прерывания. Выключен во всех остальных случаях.

Флаги: SNXT(009)

Название	Обозначение	Действие
Флаг ошибки	ER	Включен, если указанный бит B находится не в области WR. Включен, если команда SNXT(009) используется в программе обработки прерывания. Выключен во всех остальных случаях.

Меры предосторожности

В качестве бита управления для команд STEP(008)/SNXT(009) должен использоваться бит рабочей области.

Бит управления, указанный для команды STEP(008) или SNXT(009), не должен использоваться ни в одном другом месте прикладной программы. В противном случае произойдет ошибка дублирования бита.

Если в программе шага использована команда SBS(091) для вызова подпрограммы, выходы и команды подпрограммы не будут заблокированы при выключении бита управления.

Биты управления в пределах одной области программирования шагов должны использоваться по порядку и находиться в одном слове.

Команда SNXT(009) выполняется только один раз (т. е. по положительному фронту условия выполнения).

Обязательно предусматривайте команду SNXT(009) в конце области программирования шагов и используйте в качестве формального бита управления бит рабочей области. Если в последней команде SNXT(009), завершающей область программирования шагов, будет указан бит управления, который ранее использовался для какого-либо шага, при выполнении команды SNXT(009) будет запущен соответствующий шаг.

При указанных нарушениях произойдет ошибка и будет установлен флаг ошибки: для команды SNXT(009) или STEP(008) указан бит управления (операнд B), не являющийся битом рабочей области; программа шага размещена не в циклической задаче, а в каком-либо другом месте прикладной программы.

При выполнении команды STEP(008) включается на один цикл флаг выполнения шага (A200.12). Этот флаг можно использовать для процедуры инициализации сразу после запуска выполнения шага.

Требования к расположению областей программирования шагов

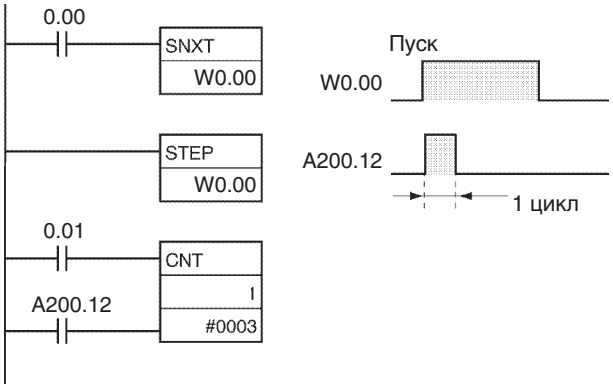
Команды STEP(008) и SNXT(009) нельзя использовать внутри подпрограмм, программ обработки прерываний или программных блоков.

Позаботьтесь о том, чтобы два шага не выполнялись в пределах одного цикла ПЛК.

Команды, которые невозможно использовать в программах шагов

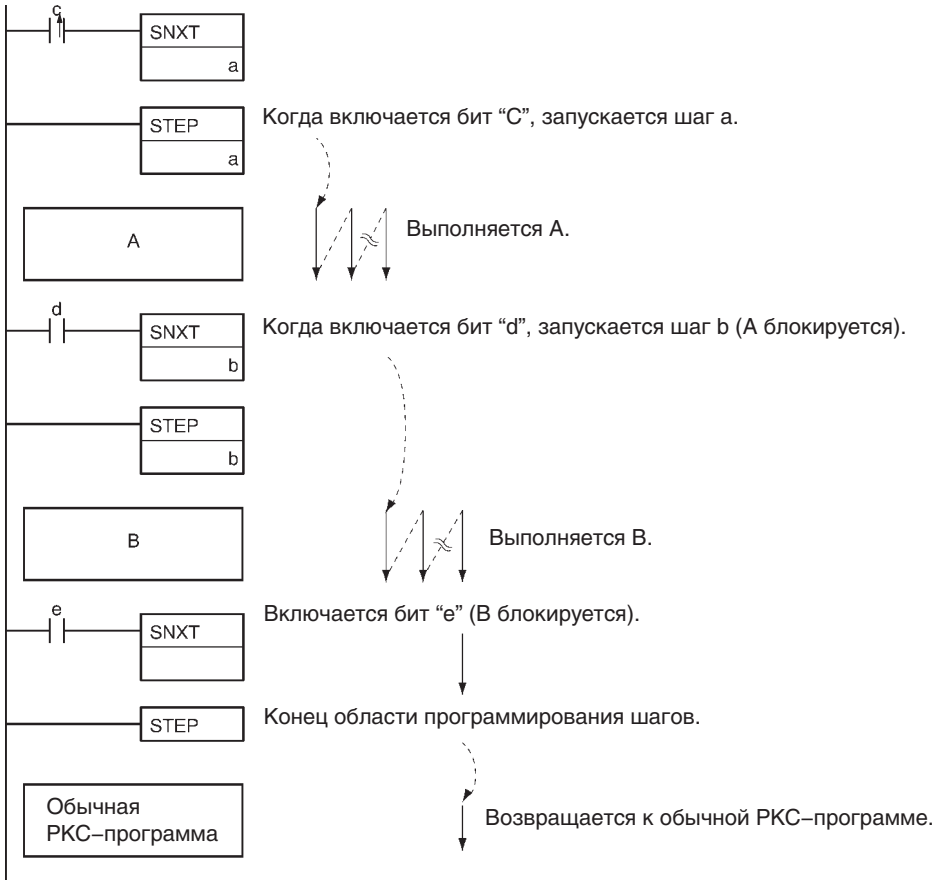
В следующей таблице перечислены команды, которые невозможно использовать в программах шагов.

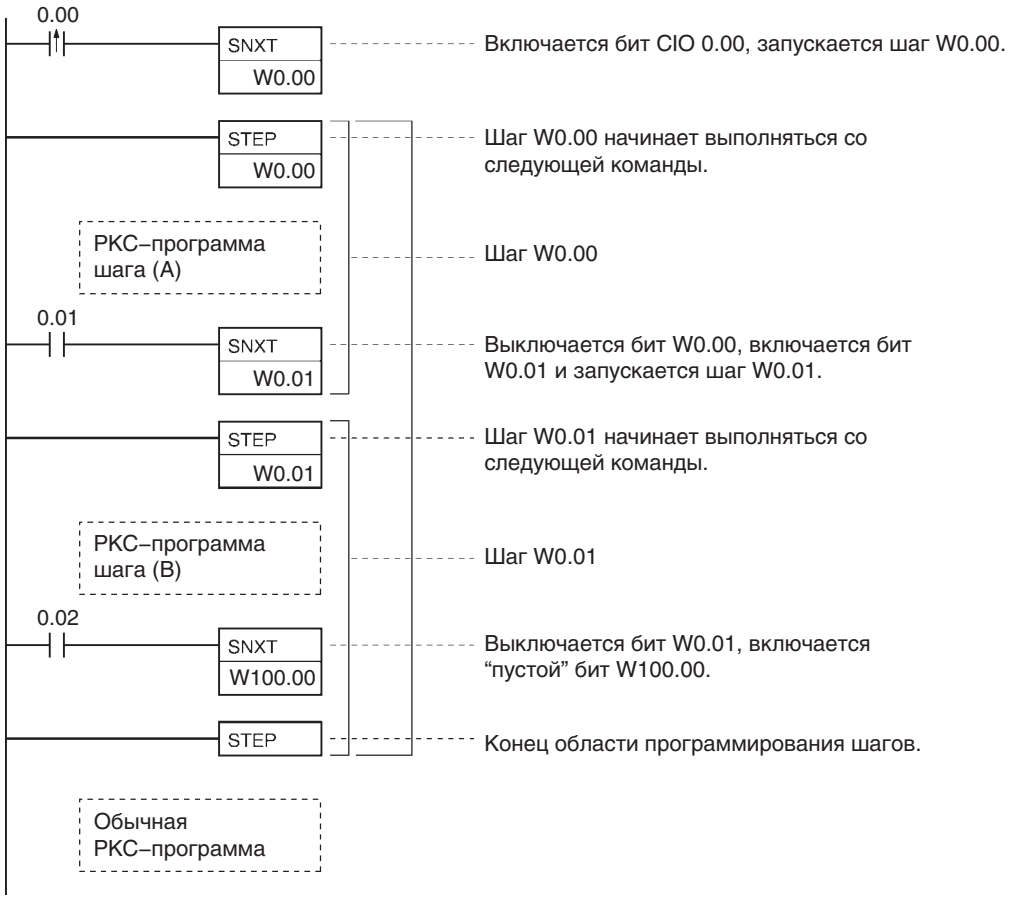
Функция	Мнемоническое обозначение	Название
Команды управления последовательностью выполнения	END(001)	КОНЕЦ
	IL(002)	БЛОКИРОВКА
	ILC(003)	ОТМЕНА БЛОКИРОВКИ
	JMP(004)	ПЕРЕХОД
	JME(005)	КОНЕЦ ПЕРЕХОДА
	CJP(510)	УСЛОВНЫЙ ПЕРЕХОД
	CJPN(511)	УСЛОВНЫЙ ПЕРЕХОД НЕ
	JMP0(515)	МНОЖЕСТВЕННЫЙ ПЕРЕХОД
	JME0(516)	КОНЕЦ МНОЖЕСТВЕННОГО ПЕРЕХОДА
Команды управления подпрограммами	SBN(092)	ВХОД В ПОДПРОГРАММУ
	RET(093)	ВЫХОД ИЗ ПОДПРОГРАММЫ



Сопутствующие биты

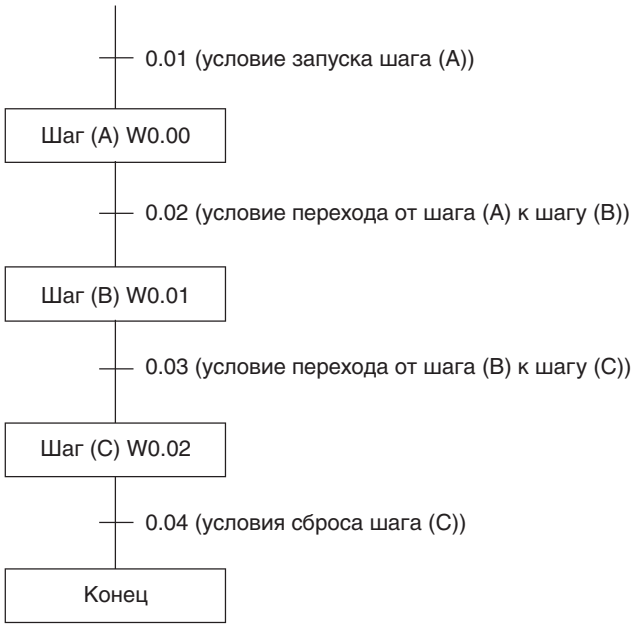
Название	Адрес	Подробная информация
Флаг шага	A200.12	Включен в течение одного цикла, когда команда STEP(008) запускает программу шага. Может использоваться для сброса таймеров и выполнения других операций при запуске нового шага.

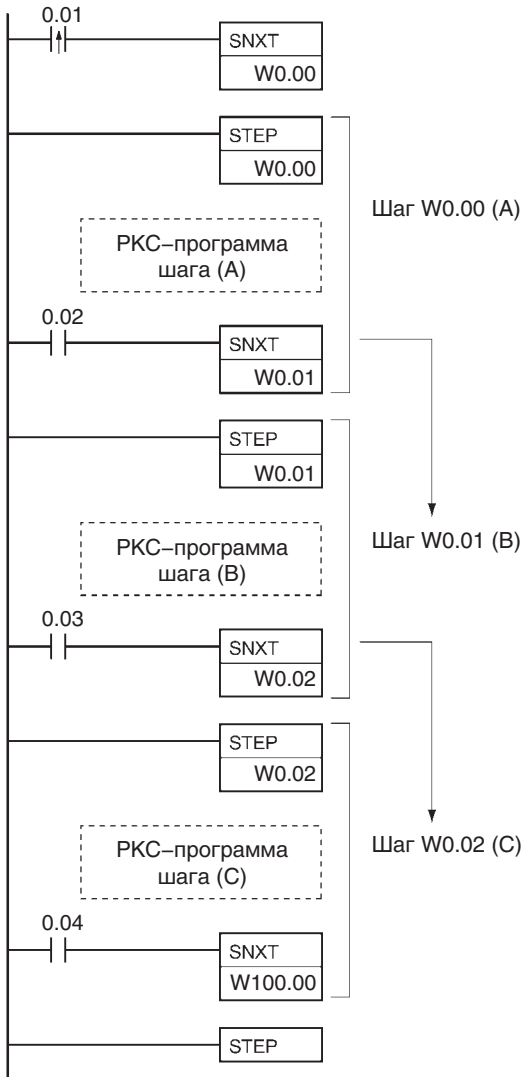

c
SNXT
a
STEP
a
Когда включается бит “C”, запускается шаг a.
A
Выполняется A.
d
SNXT
b
Когда включается бит “d”, запускается шаг b (A блокируется).
STEP
b
B
Выполняется B.
e
SNXT
Включается бит “e” (B блокируется).
STEP
Конец области программирования шагов.
Обычная
РКС–программа
Возвращается к обычной РКС–программе.



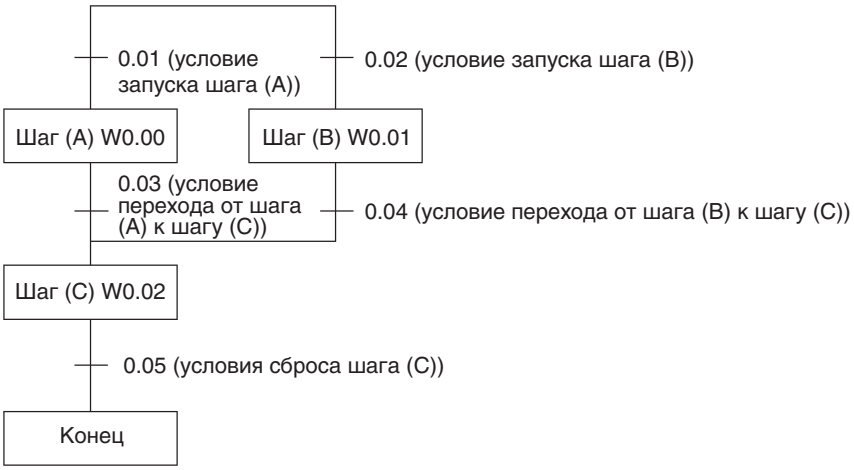
Примеры

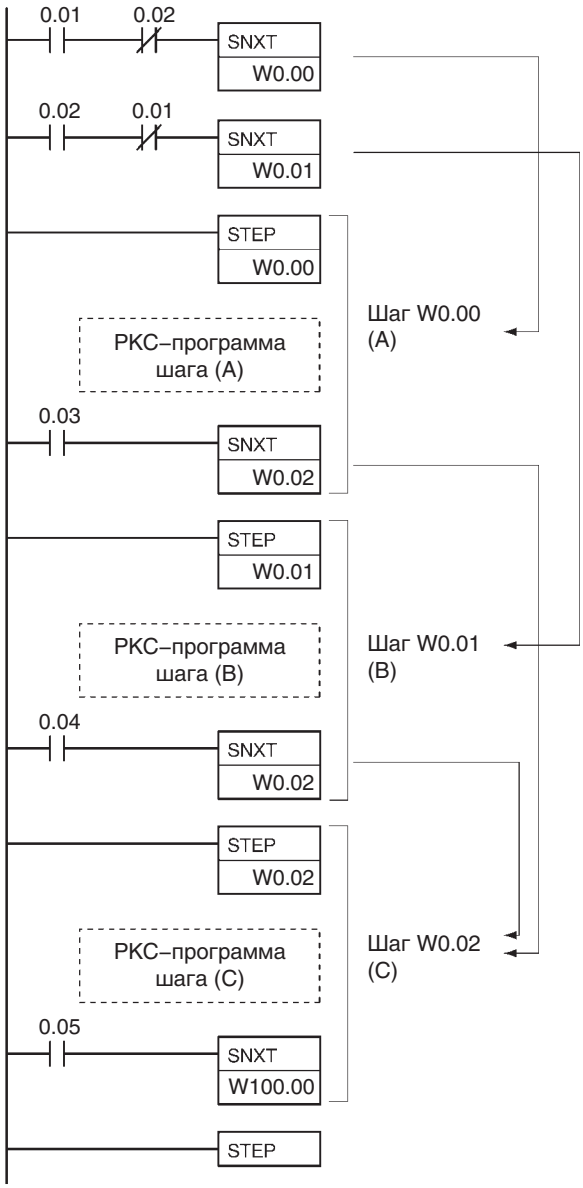
Последовательное выполнение операций



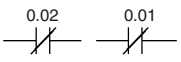


Разветвленное выполнение операций



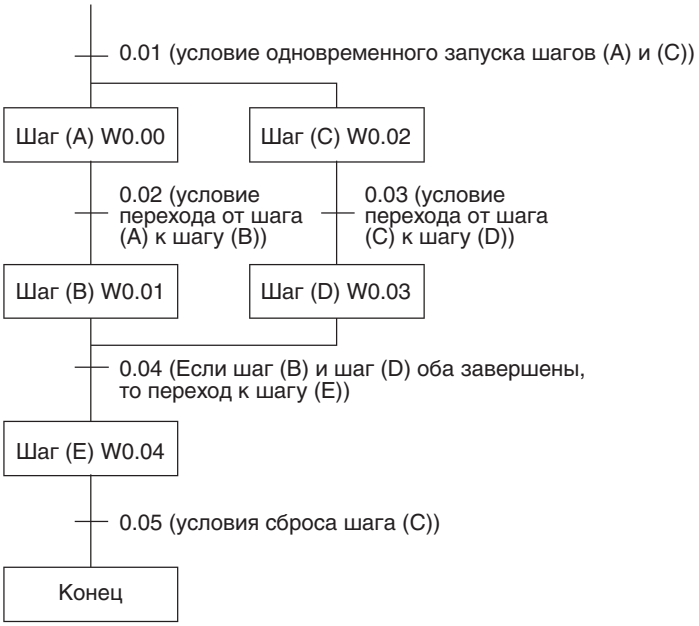


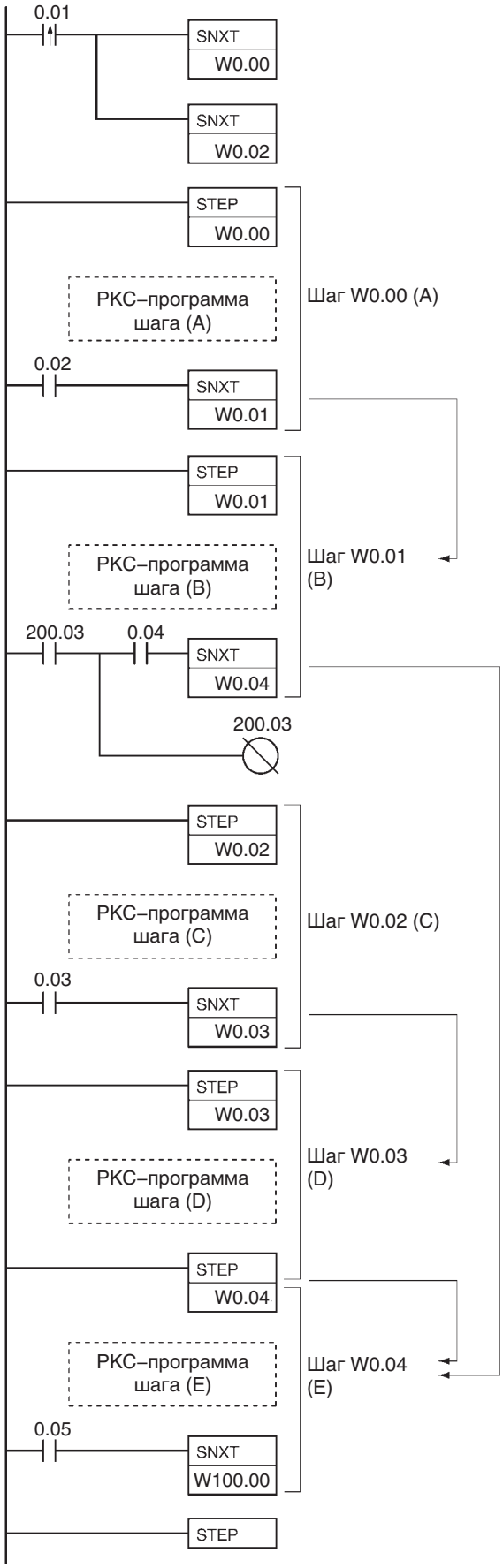
Приведенный выше пример программы используется в том случае, когда шаги A и B не могут выполняться одновременно. Для одновременного выполнения шагов A и B следует удалить условие выполнения, показанное на рисунке ниже.



Примечание. В точке программы (см. пример выше), где команда SNXT(009) выполняется с битом W0.02, разветвление завершается и происходит переход к следующему шагу, хотя один и тот же бит управления используется дважды. При проверке программы в CX-Programmer это не воспринимается как ошибка. Ошибка дублирования бита происходит лишь в том случае, когда некоторый бит, использованный для управления выполнением шага, также используется в каком-либо другом месте прикладной программы.

Параллельное выполнение операций



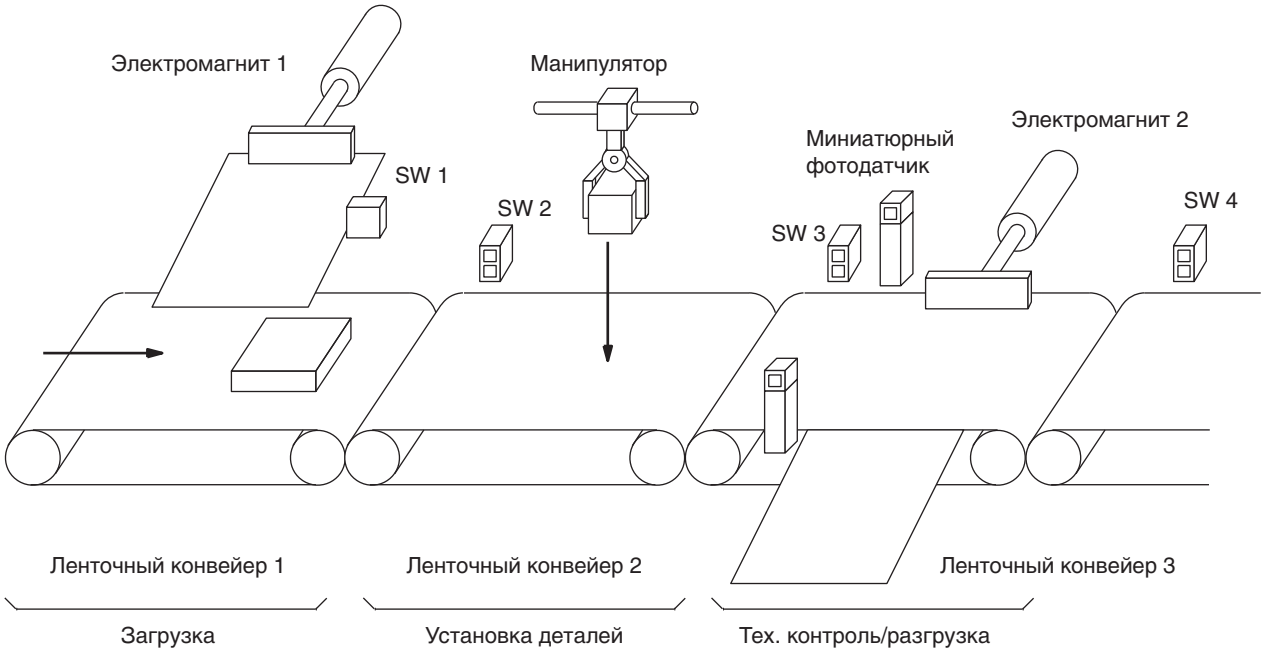

0.01
SNXT
W0.00
SNXT
W0.02
STEP
W0.00
РКС–программа шага (A)
Шаг W0.00 (A)
0.02
SNXT
W0.01
STEP
W0.01
РКС–программа шага (B)
Шаг W0.01 (B)
200.03
0.04
SNXT
W0.04
200.03
STEP
W0.02
РКС–программа шага (C)
Шаг W0.02 (C)
0.03
SNXT
W0.03
STEP
W0.03
РКС–программа шага (D)
Шаг W0.03 (D)
STEP
W0.04
РКС–программа шага (E)
Шаг W0.04 (E)
0.05
SNXT
W100.00
STEP

Примеры использования

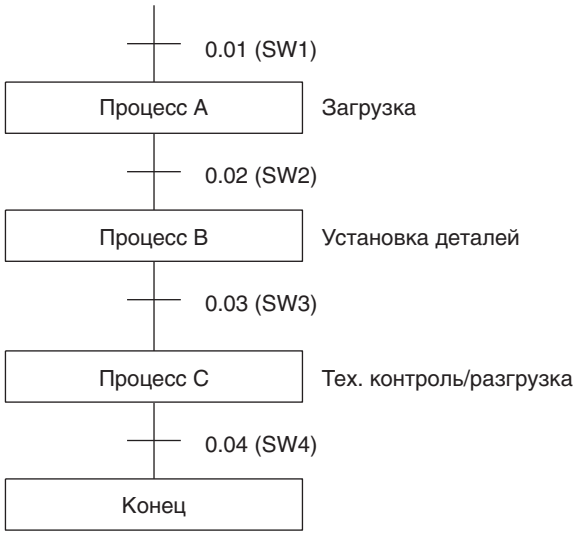
Ниже рассмотрены примеры трех разных способов организации пошагового выполнения технологических операций. *Пример 1* демонстрирует последовательное выполнение, в *Примере 2* показано разветвленное выполнение, *Пример 3* иллюстрирует параллельное выполнение.

Пример 1. Последовательное выполнение

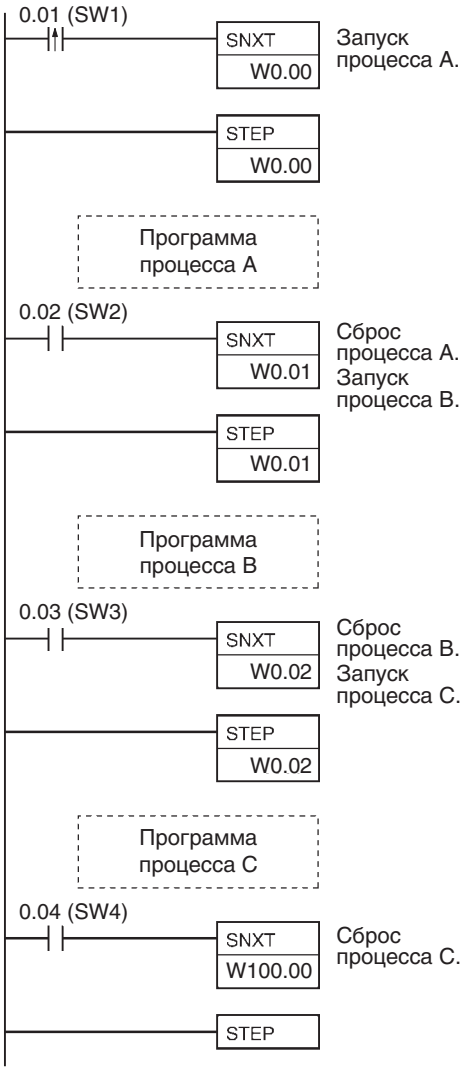
Показанный ниже технологический процесс состоит из трех операций: загрузки, установки деталей и технического контроля/разгрузки. Эти операции выполняются последовательно друг за другом, при переходе к следующей операции предыдущая операция сбрасывается. На конвейере установлены датчики (SW1, SW2, SW3 и SW4), сигнализирующие начало и завершение тех или иных операций.



На следующем рисунке показана последовательность выполнения операций и выключатели, которые используются для управления выполнением.

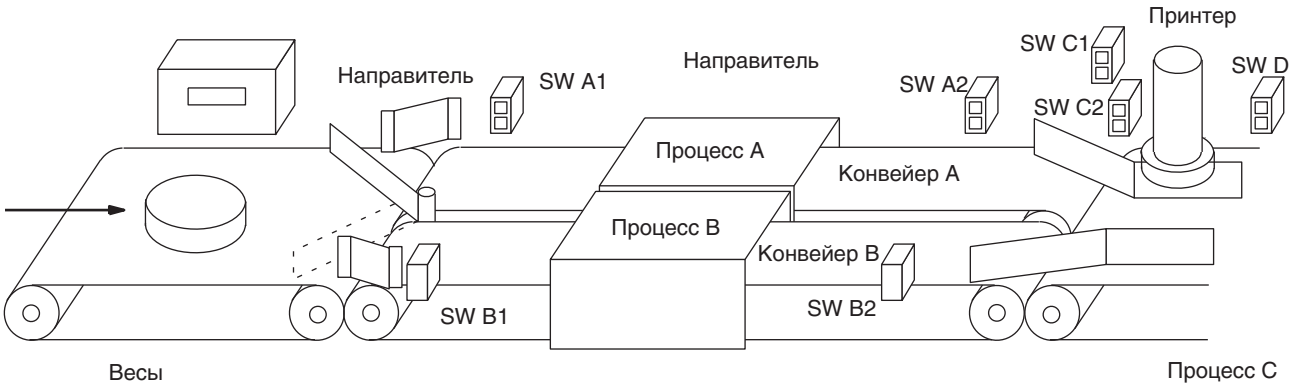


Программа, реализующая данный технологический процесс, имеет наиболее простую структуру: каждый шаг запускается одной уникальной командой SNXT(009). Каждый следующий шаг запускается, когда поступает сигнал от соответствующего датчика, сообщающий о завершении предыдущего шага.

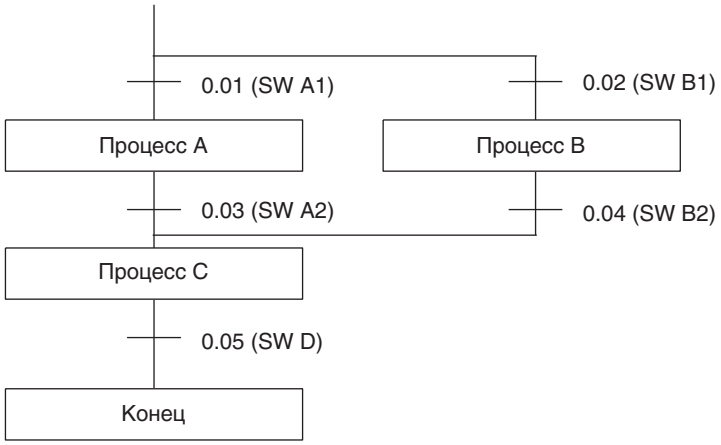


Пример 2. Разветвленное выполнение

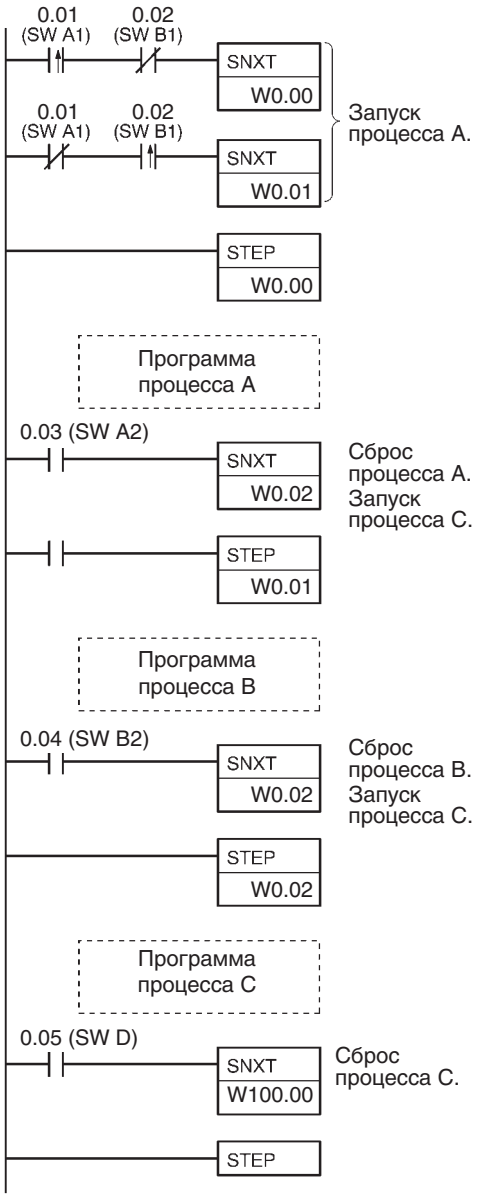
Ниже показан пример технологической линии, на которой продукт перед нанесением маркировки обрабатывается одним из двух возможных способов, при чем выбор способа обработки зависит от веса продукта. Какой бы вид обработки ни был выбран, операция нанесения маркировки всегда одна и та же. В разных местах конвейерной системы установлены датчики, сигнализирующие начало и завершение тех или иных операций.



На следующем рисунке показана последовательность выполнения операций и выключатели, которые используются для управления выполнением. В данном случае, в зависимости от состояния выключателей SW A1 и SW B1, выполняется либо операция A, либо операция B.

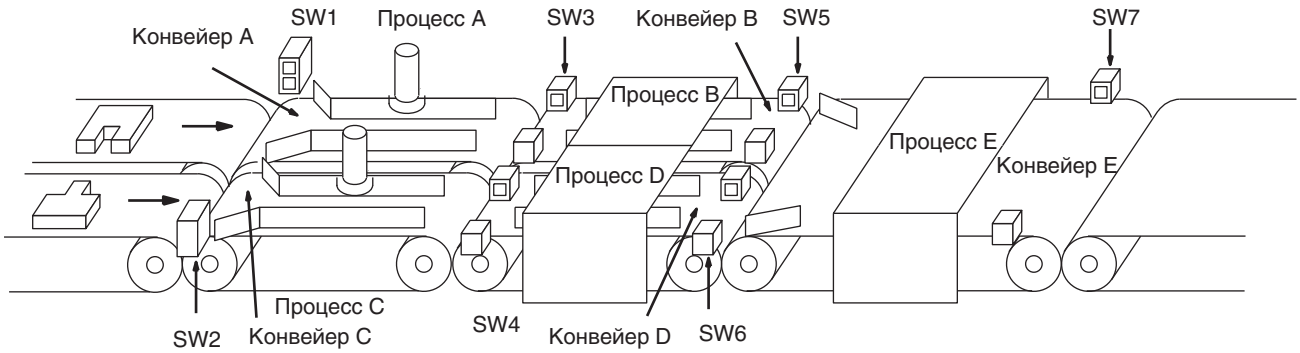


Программа, реализующая данный технологический процесс, начинается с двух команд SNXT(009), запускающих операции A и B. Биты CIO 0.01 (SW A1) и CIO 0.02 (SW B1) применены в программе таким образом, что при включении условия выполнения, запускающего операцию A или операцию B, будет выполнена только одна из этих операций. Каждый из этих шагов завершается командой SNXT(009), которая запускает шаг (операцию) C.

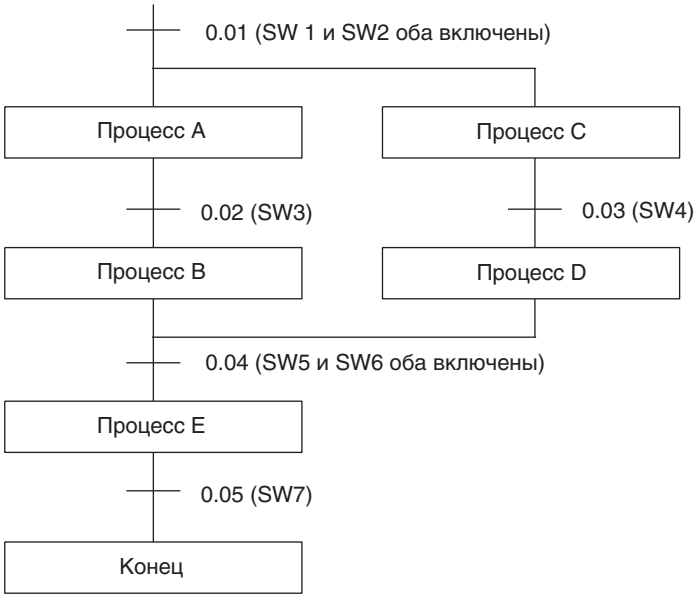


Пример 3. Параллельное выполнение

В приведенном ниже примере две разные детали конечного изделия обрабатываются одновременно. Каждая из них последовательно проходит через две технологические операции, после чего детали соединяются во время пятой по счету операции. В разных местах конвейерной системы установлены датчики, сигнализирующие начало и завершение тех или иных операций.

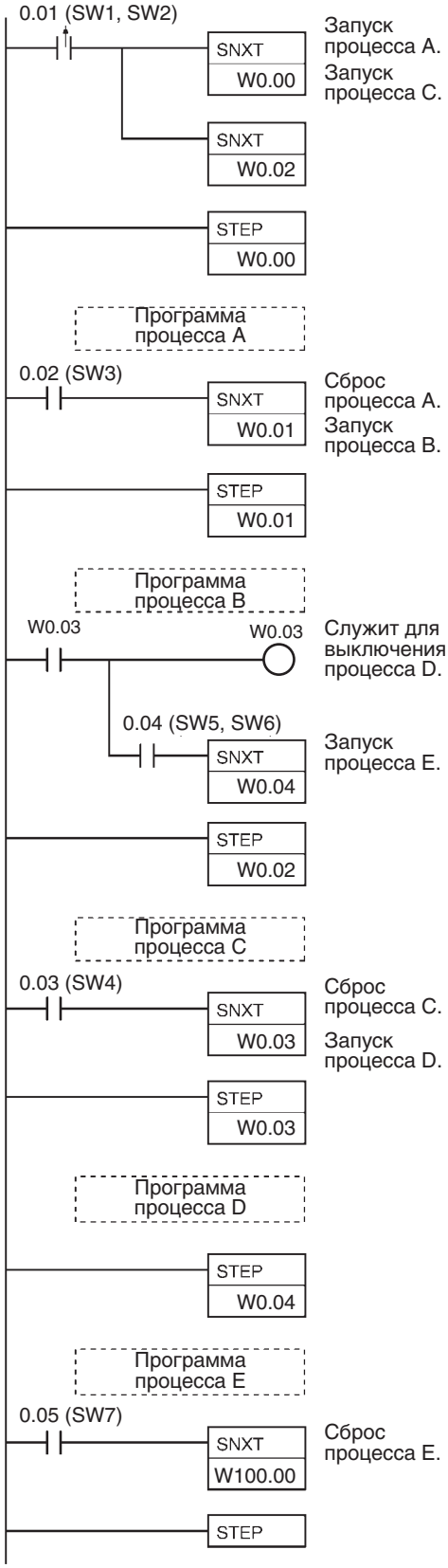


На следующем рисунке показана последовательность выполнения операций и выключатели, которые используются для управления выполнением. В данном случае операция A и операция C запускаются одновременно. После завершения операции A запускается операция B; после завершения операции C запускается операция D. После того как обе операции, B и D, завершены, запускается операция E.



Программа, реализующая данный технологический процесс, начинается с двух команд SNXT(009), запускающих операции A и C. Эти команды ответвляются от одной общей линии, имеют общее условие выполнения и всегда выполняются вместе, запуская программы шагов A и C. По завершении выполнения шагов A и C сразу же начинают выполняться шаги B и D.

После того как обе операции, B и D, завершаются (т. е. включаются выключатели SW5 и SW6), операции B и D одновременно сбрасываются командой SNXT(009), расположенной в конце программы операции B. Хотя в конце шага D команда SNXT(009) отсутствует, ее бит управления выключается в результате выполнения команды SNXT(009) W0.04. Это обусловлено тем, что бит W0.03 применен в команде выхода (OUT) в шаге, который сбрасывается командой SNXT(009) W0.04 (бит W0.03 выключается, когда выполняется команда SNXT(009) W0.04). Таким образом, перед выполнением шага E операция B сбрасывается непосредственно, а операция D сбрасывается косвенным образом.


0.01 (SW1, SW2)
SNXT
W0.00
Запуск процесса A.
Запуск процесса C.
SNXT
W0.02
STEP
W0.00
Программа процесса A
0.02 (SW3)
SNXT
W0.01
Сброс процесса A.
Запуск процесса B.
STEP
W0.01
Программа процесса B
W0.03
W0.03
Служит для выключения процесса D.
0.04 (SW5, SW6)
SNXT
W0.04
Запуск процесса E.
STEP
W0.02
Программа процесса C
0.03 (SW4)
SNXT
W0.03
Сброс процесса C.
Запуск процесса D.
STEP
W0.03
Программа процесса D
STEP
W0.04
Программа процесса E
0.05 (SW7)
SNXT
W100.00
Сброс процесса E.
STEP

3-22 Команды базовых модулей ввода/вывода

В данном разделе описаны команды, предназначенные для работы с модулями ввода/вывода.

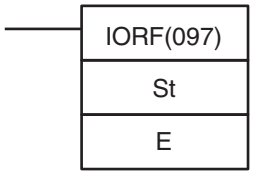
Команда	Мнемоническое обозначение	Код функции	Стр.
ОБНОВЛЕНИЕ СЛОВ ВВОДА/ ВЫВОДА	IORF	097	843
ПРЕОБРАЗОВАТЬ В 7-СЕГМЕНТНЫЙ КОД	SDEC	078	846
ЧТЕНИЕ ИНТЕЛЛЕКТУАЛЬНЫХ ВХОДОВ/ВЫХОДОВ	IORD	222	870
ЗАПИСЬ ИНТЕЛЛЕКТУАЛЬНЫХ ВХОДОВ/ВЫХОДОВ	IOWR	223	873
ЧТЕНИЕ ЦИФРОВОГО ПЕРЕКЛЮЧАТЕЛЯ	DSW	210	849
ЧТЕНИЕ ДЕСЯТИЧНОЙ КЛАВИАТУРЫ	TKY	211	853
ЧТЕНИЕ ШЕСТНАДЦАТЕРИЧНОЙ КЛАВИАТУРЫ	HKY	212	857
ЧТЕНИЕ МАТРИЦЫ	MTR	213	861
ВЫВОД НА 7-СЕГМЕНТНЫЙ ИНДИКАТОР	7SEG	214	865

3-22-1 ОБНОВЛЕНИЕ СЛОВ ВВОДА/ВЫВОДА: IORF(097)

Назначение

Обновление содержимого указанных слов ввода/вывода.

Символ РКС



St: Начальное слово

E: Конечное слово

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	IORF(097)
	Однократное выполнение по положительному фронту	@IORF(097)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограм-мы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

St: Начальное слово

CIO 0...CIO 999 (область битов входов/выходов) или
CIO 2000...CIO 2959 (область битов спец. модулей ввода/вывода)

E: Конечное слово

CIO 0...CIO 999 (область битов входов/выходов) или
CIO 2000...CIO 2959 (область битов спец. модулей ввода/вывода)

Примечание. St и E должны находиться в одной области памяти.

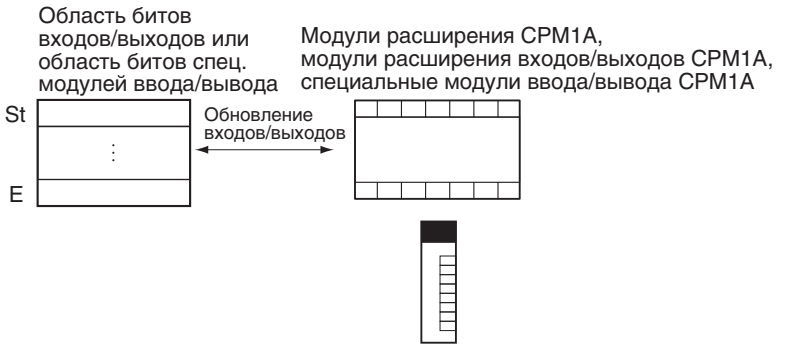
Характеристики операндов

Область	St	E
Область CIO	CIO 0...CIO 999 CIO 2000...CIO 2959	
Вспомогательная область	---	
Область битов хранения	---	
Область специальных битов	---	
Область таймеров	---	
Область счетчиков	---	
Область DM	---	
Косвенные адреса DM в двоичном формате	---	
Косвенные адреса DM в двоично-десятичном формате (BCD)	---	
Постоянные	---	
Регистры данных	---	
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

Описание

Команда IORF(097) обновляет содержимое группы слов ввода/вывода в диапазоне от St до E (включительно). С помощью команды IORF(097) можно обновить содержимое слов, отведенных для модулей расширения CPM1A, модулей расширения входов/выходов CPM1A или специальных модулей ввода/вывода.

Если для обновления указаны слова из области битов специального модуля ввода/вывода, обновляются все 10 слов, отведенных для этого модуля, если первое слово из этих 10 слов входит в диапазон обновляемых слов, заданный в команде.



Команду IORF(097) невозможно использовать для встроенных входов/выходов модуля ЦПУ или модулей шины ЦПУ. Для этих целей следует использовать команды с префиксом мгновенного обновления. Команду IORF(097) и команды с префиксом мгновенного обновления невозможно использовать для встроенных аналоговых входов/выходов модулей ЦПУ линейки XA.

Если в указанном диапазоне слов от St до E присутствуют слова, не принадлежащие каким-либо модулям в текущей конфигурации ПЛК, никаких действий над этими словами не производится, обновляются только слова, принадлежащие реально существующим модулям.

Применимые модули

В следующей таблице перечислены модули, поддерживаемые командой IORF(097).

Входы/выходы	Слова	Поддержка IORF(097)
Встроенные входы/выходы	Входы: CIO 0 и CIO 1 Выходы: CIO 100 и CIO 101	Нет
Встроенные аналоговые входы	CIO 200...CIO 203	Нет
Встроенные аналоговые выходы	CIO 210 и CIO 211	Нет
Модули расширения вх./вых. CPM1A	Входы: CIO 2...CIO 99 Выходы: CIO 102...CIO 199	Да
Модули расширения CPM1A		
Спец. модули ввода/вывода	CIO 2000...CIO 2959	Да
Модули шины ЦПУ	CIO 1500...CIO 1899	Нет

Примечание. Если некоторый модуль поддерживает обновление с помощью команды IORF(097), это не значит, что он также поддерживает команды с префиксом мгновенного обновления (!). Информацию о поддержке тем или иным модулем команд с префиксом мгновенного обновления следует уточнять отдельно.

Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	Включен, если St больше E. Включен, если St и E находятся в разных областях памяти. Выключен во всех остальных случаях.

Меры предосторожности

Если в одной команде будут одновременно указаны слова из области битов входов/выходов (CIO 0...CIO 999) и области битов специальных модулей ввода/вывода (CIO 2000...CIO 2959), произойдет ошибка.

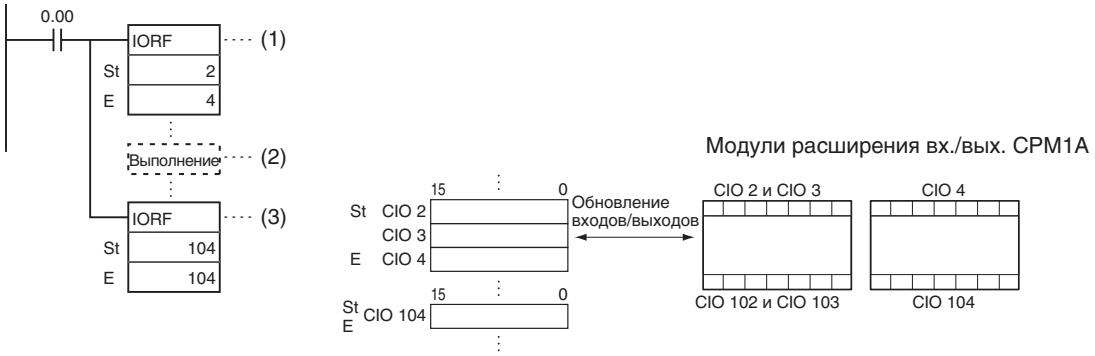
Если во время обновления слов ввода/вывода произойдет ошибка шины ввода/вывода, обновление данных ввода/вывода, инициированное командой IORF(097), сразу же прекратится, не завершившись до конца.

Если команда IORF(097) применяется в задаче обработки прерывания, в настройках ПЛК обязательно следует выключить циклическое обновление для специальных модулей ввода/вывода. Если циклическое обновление для специальных модулей ввода/вывода не будет выключено и во время его выполнения будет инициировано обновление входов/выходов с помощью команды IORF(097), произойдет некритическая ошибка дублированного обновления и установится флаг ошибки задачи обработки прерывания (A402.13).

Примеры

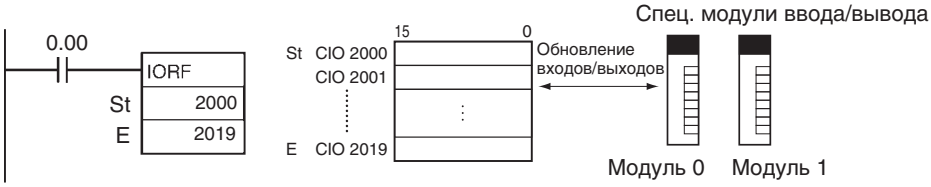
Обновление слов в области битов входов/выходов

При включении бита CIO 0.00 обновляется содержимое слов CIO 2...CIO 4 (36 входов) (1), после чего выполняются требуемые операции (2), и обновляется содержимое слова CIO 104 (8 выходов).



Обновление слов области битов специального модуля ввода/вывода

Ниже показан пример программы, обновляющей 20 слов в диапазоне от CIO 2000 до CIO 2019 (входы/выходы специальных модулей ввода/вывода с номерами модулей 0 и 1) при включении бита CIO 0.00.

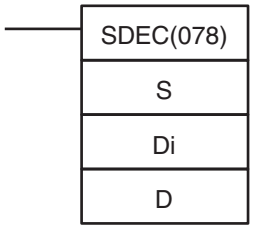


3-22-2 ПРЕОБРАЗОВАТЬ В 7-СЕГМЕНТНЫЙ КОД: SDEC(078)

Назначение

Преобразование шестнадцатеричного содержимого указанного разряда(-ов) в 8-битовый код для отображения на 7-сегментном индикаторе и запись этого кода в младшие или старшие октеты (8 битов) указанных слов назначения.

Символ РКС



S: Исходное слово

Di: Указатель разрядов

D: Первое слово назначения

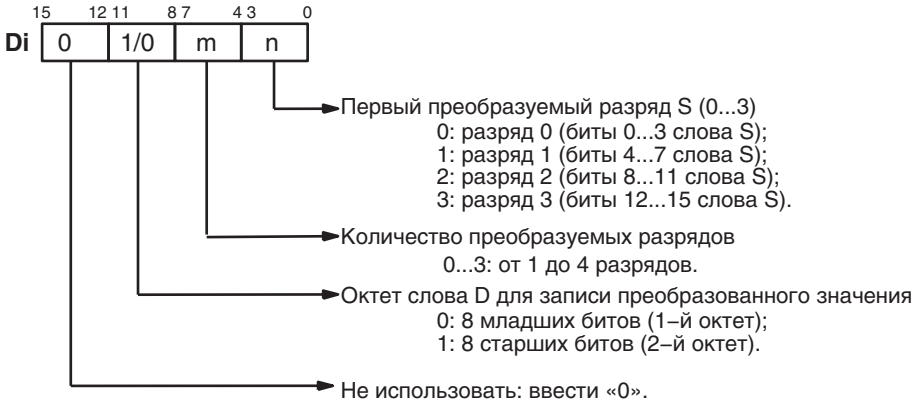
Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	SDEC(078)
	Однократное выполнение по положительному фронту	@SDEC(078)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды: указатель разрядов

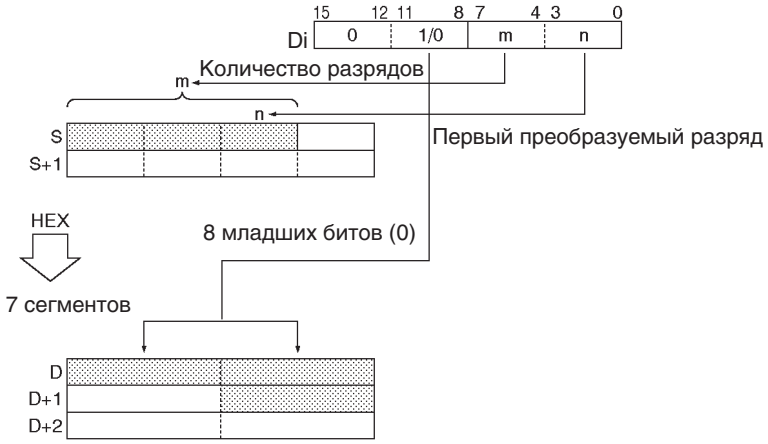


Характеристики операндов

Область	S	Di	D
Область CIO	CIO 0...CIO 6143		
Рабочая область	W0...W511		
Область битов хранения	H0...H511		
Область вспомогательных битов	A0...A959		A448...A959
Область таймеров	T0000...T4095		
Область счетчиков	C0000...C4095		
Область DM	D0...D32767		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767		
Постоянные	---	Только указанные значения	---
Регистры данных	DR0...DR15		---
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда SDEC(078) воспринимает содержимое слова S как четыре шестнадцатеричных разряда, преобразует разряды, выбранные операндом Di (первый разряд и количество разрядов), в коды для отображения на 7-сегментном индикаторе и выдает результаты в биты слов D, указанные операндом Di.



Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	Включен, если значения параметров в Di выходят за допустимые диапазоны. Выключен во всех остальных случаях.

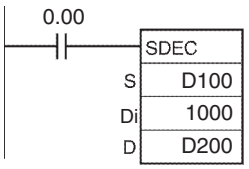
Меры предосторожности

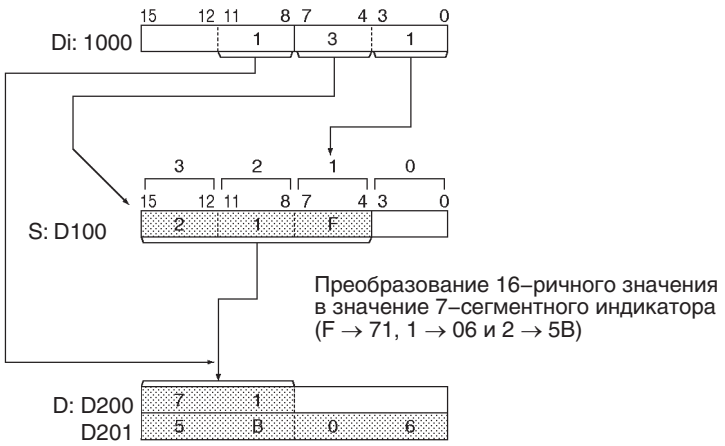
Если в операнде Di выбрано преобразование нескольких разрядов, разряды преобразуются в порядке от младшего разряда к старшему. После разряда 3 следующим является разряд 0.

Результаты записываются в слова назначения (D), начиная с указанного байта (октета) слова с наименьшим адресом и далее в порядке возрастания адресов слов. Если преобразованное значение записывается только в один из байтов слова назначения, содержимое второго байта этого слова не изменяется.

Примеры

При включении бита CIO 0.00 содержимое трех разрядов слова D100, начиная с разряда 1, преобразуется из шестнадцатеричного формата в коды для отображения на 7-сегментном индикаторе. Полученные коды записываются в старший байт слова D200 и в оба байта слова D201. Выбор преобразуемых байтов и расположения выходных байтов произведен с помощью слова CIO 1000.

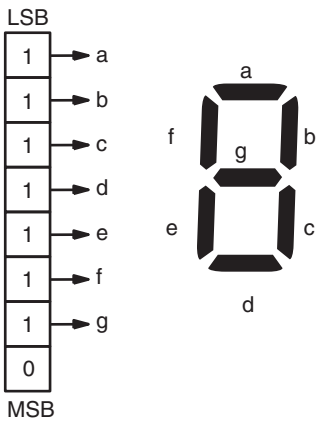




7-сегментные коды

В следующей таблице для каждого значения шестнадцатеричного разряда (4 бита) приведен соответствующий 8-битовый код 7-сегментного индикатора.

Исходное значение					7-сегментый код									Индикатор Исходное значение
Раз-ряд	Биты				–	g	f	e	d	c	b	a	Hex	
0	0	0	0	0	0	0	1	1	1	1	1	1	3F	0
1	0	0	0	1	0	0	0	0	0	1	1	0	06	1
2	0	0	1	0	0	1	0	1	1	0	1	1	5B	2
3	0	0	1	1	0	1	0	0	1	1	1	1	4F	3
4	0	1	0	0	0	1	1	0	0	1	1	0	66	4
5	0	1	0	1	0	1	1	0	1	1	0	1	6D	5
6	0	1	1	0	0	1	1	1	1	1	0	1	7D	6
7	0	1	1	1	0	0	1	0	0	1	1	1	27	7
8	1	0	0	0	0	1	1	1	1	1	1	1	7F	8
9	1	0	0	1	0	1	1	0	1	1	1	1	6F	9
A	1	0	1	0	0	1	1	1	0	1	1	1	77	A
B	1	0	1	1	0	1	1	1	1	1	0	0	7C	b
C	1	1	0	0	0	0	1	1	1	0	0	1	39	C
D	1	1	0	1	0	1	0	1	1	1	1	0	5E	d
E	1	1	1	0	0	1	1	1	1	0	0	1	79	E
F	1	1	1	1	0	1	1	1	0	0	0	1	71	F

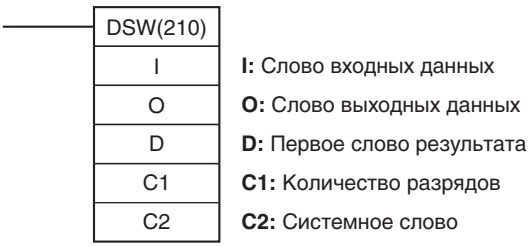


3-22-3 ЧТЕНИЕ ЦИФРОВОГО ПЕРЕКЛЮЧАТЕЛЯ — DSW(210)

Назначение

Чтение значения, выставленного на внешнем цифровом (или барабанном) переключателе, который подключен к модулю входов-выходов, и запись 4-разрядного или 8-разрядного значения в указанные слова.

Символ РКС



Варианты выполнения

Варианты выполнения	Выполнение в каждом цикле при включенном условии выполнения	DSW(210)
	Однократное выполнение по положительному фронту	Не предусмотрено
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

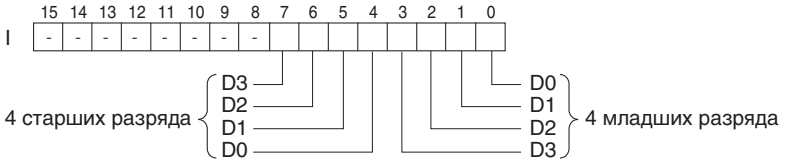
Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
Не допускается	OK	OK	Не допускается

Операнды

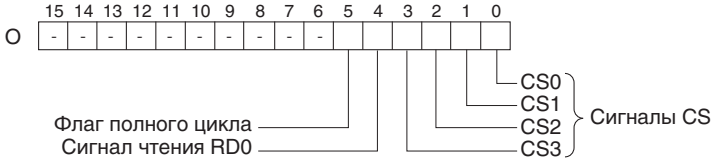
I: Слово входов (входы линий данных D0...D3)

Укажите слово входов, отведенное для модуля входов, и подсоедините линии данных D0...D3 цифрового переключателя к модулю входов, как показано на рисунке ниже.



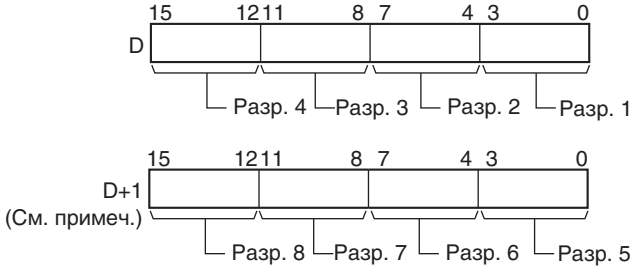
O: Слово выходов (выходы сигналов управления CS/RD)

Укажите слово выходов, отведенное для модуля выходов, и подсоедините сигналы управления цифрового переключателя (сигналы CS и RD) к модулю выходов, как показано на рисунке ниже.



D: Первое слово результата

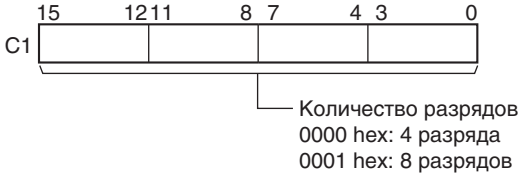
Указывает адрес первого слова для записи значений, выставленных внешним цифровым переключателем



Примеч. Только если C1 = 0001 hex (чтение 8 разрядов).

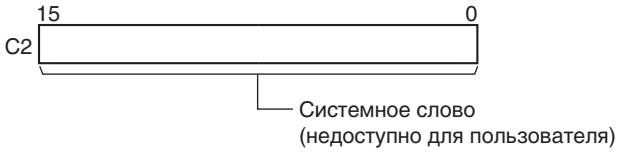
C1: Количество разрядов

Указывает количество разрядов внешнего цифрового переключателя, которые должны быть считаны. C1 может иметь одно из двух значений: 0000 hex — для чтения 4 разрядов, 0001 hex — для чтения 8 разрядов.



C2: Системное слово

Указывает рабочее слово, используемое командой. Это слово будет невозможно использовать для каких-либо иных целей.



Характеристики операндов

Область	I	O	D	C1	C2
Область CIO	CIO 0...CIO 6143			---	CIO 0...CIO 6143
Рабочая область	W0...W511			---	W0...W511
Область битов хранения	H0...H511			---	H0...H511
Область вспомогательных битов	A0...A959	A448...A953		---	A448...A959
Область таймеров	T0000...T4095			---	T0000...T4095
Область счетчиков	C0000...C4095			---	C0000...C4095
Область DM	D0...D32767			---	D0...D32767
Косвенные адреса DM в двоичном формате	@ D0...@ D32767			---	@ D0... @ D32767
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767			---	---
Постоянные	---			0000 или 0001 hex	---
Регистры данных	DR0...DR15		---	---	DR0...DR15
Регистры указателей	---				
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)..., IR15+(++) ,–(– –)IR0..., –(– –)IR15		---	---	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)..., IR15+(++) ,–(– –)IR0..., –(– –)IR15

Описание

Команда DSW(210) выдает управляющие сигналы в биты 00...04 выходного слова O, считывает указанное количество разрядов (4-х или 8-ми разрядов, указывается операндом C1) цифрового переключателя, используя входное слово I, и записывает полученный результат в слова D и D+1. (При чтении 4-х разрядов результат записывается в слово D. При чтении 8-ми разрядов результат записывается в слова D и D+1.)

Каждые 16 циклов команда DSW(210) считывает одно 4-разрядное или 8-разрядное значение цифрового переключателя, после чего начинает выполняться сначала и считывает следующее значение. В каждом 16-м по счету цикле модуля ЦПУ включается флаг завершения полного цикла (бит 05 слова O).

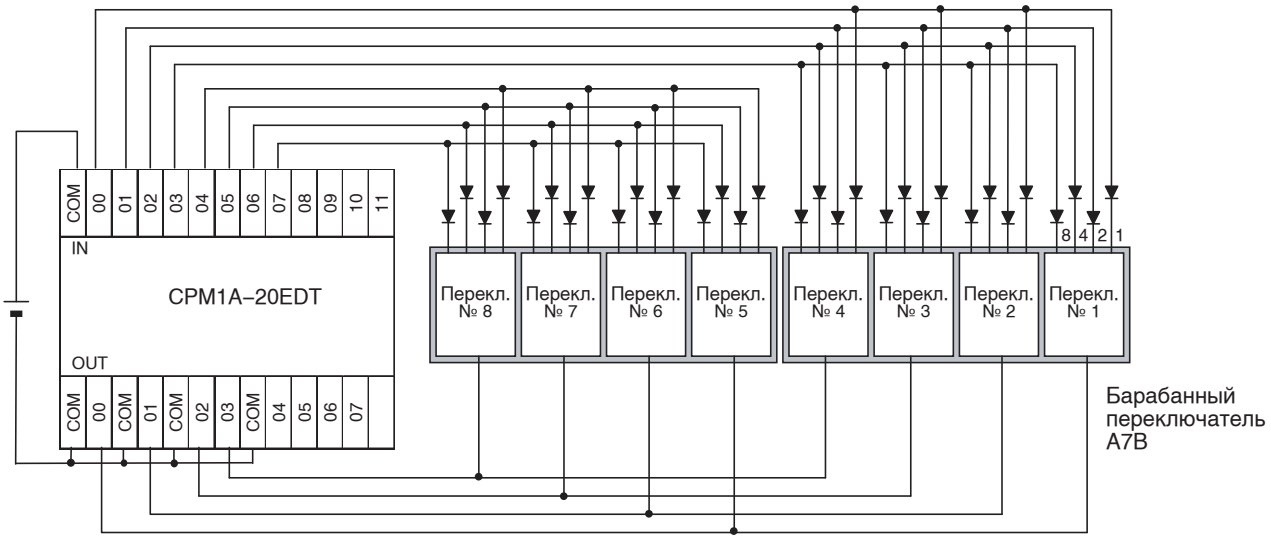
Команда DSW(210) считывает одно 4-разрядное или 8-разрядное значение за 16 циклов, после чего начинает выполняться сначала и вновь считывает значение за следующие 16 циклов.

При выполнении команда DSW(210) начинает чтение установленного переключателем значения с самого начала (т. е., начиная с 1-го из 16-ти циклов), независимо от того, на каком этапе прекратилось выполнение предыдущей команды.

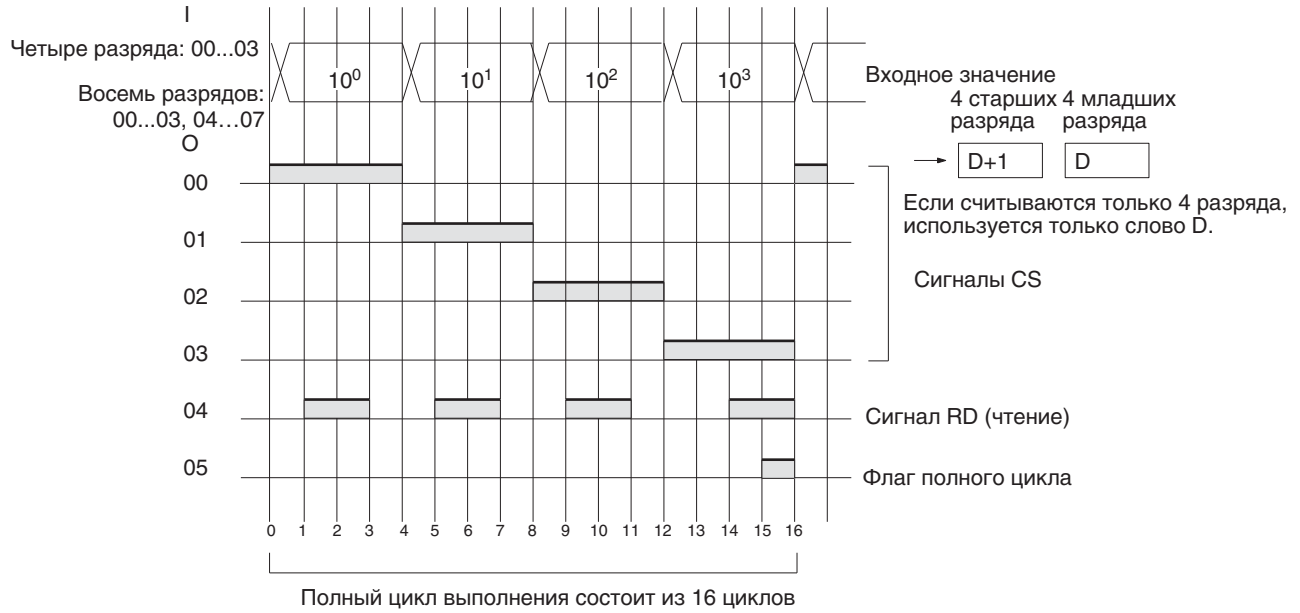
Команда DSW(210) может применяться в программе любое число раз.

Подключение внешних цепей

Подключите линии данных и линии сигналов управления цифрового или барабанного переключателя к входам 0...7 модуля входов и выходам 0...4 модуля выходов в соответствии со схемой, приведенной ниже. На схеме показан пример подключения для переключателя барабанного типа A7B.



Временная диаграмма



Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	ВЫКЛ

Меры предосторожности

Никакая другая команда не должна обращаться к системному слову (C2) с целью чтения или записи. При обращении другой команды к системному слову команда DSW(210) будет работать неправильно. В самом первом цикле выполнения прикладной программы команда DSW(210) не инициализирует содержимое системного слова. Поэтому, если команда DSW(210) должна выполняться в самом первом цикле ПЛК, в программе должна быть предусмотрена очистка содержимого системного слова.

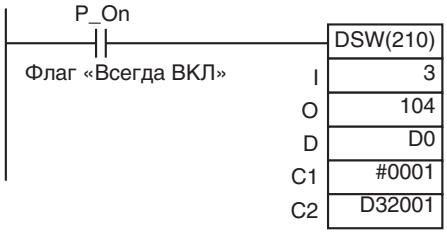
После выполнения команды DSW(210) обязательно должно производиться обновление входов/выходов для модуля входов и модуля выходов, к которым подключен цифровой или барабанный переключатель. В противном случае команда DSW(210) будет работать неправильно. Поэтому не следует подключать цифровой или барабанный переключатель к следующим модулям:

- ведомые коммуникационные устройства (ведомые узлы DeviceNet или CompoBus/S).

Пример

В приведенном ниже примере команда DSW(210) считывает 8-разрядное число, установленное на цифровом переключателе, и постоянно выдает полученное значение в слова D0...D3. Для связи с цифровым переключателем используются слова CIO 3 и CIO 104.

Для того чтобы считывались 8 разрядов, в операнд C1 (в данном случае D32000) введено значение 0001 hex. В качестве системного слова используется слово D32001.



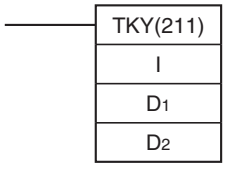
3-22-4 ЧТЕНИЕ ДЕСЯТИЧНОЙ КЛАВИАТУРЫ — TKY(211)

Назначение

Чтение числового значения, набираемого на десятиклавишной клавиатуре, и сохранение до 8 введенных цифр (разрядов) в указанные слова в двоично-десятичном коде.

Для этой команды требуется модуль с 10 входами или больше.

Символ РКС



I: Слово входных данных

D1: Первое слово регистра

D2: Слово, введенное с клавиатуры

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	TKY(211)
	Однократное выполнение по положительному фронту	@TKY(211)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

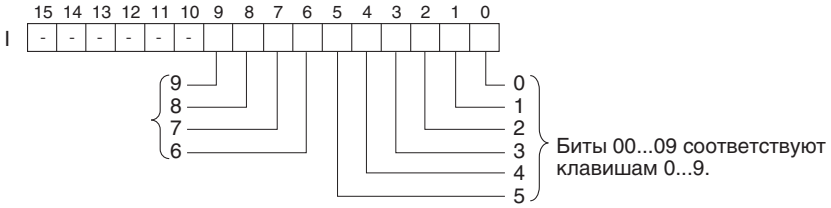
Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
Не допускается	OK	OK	Не допускается

Операнды

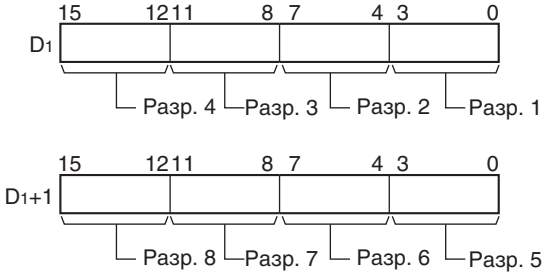
I: Слово входов (входы линий данных)

Укажите слово входов, отведенное для модуля входов, и подсоедините линии данных 0...9 десятиклавишной клавиатуры к модулю входов, как показано на следующем рисунке.



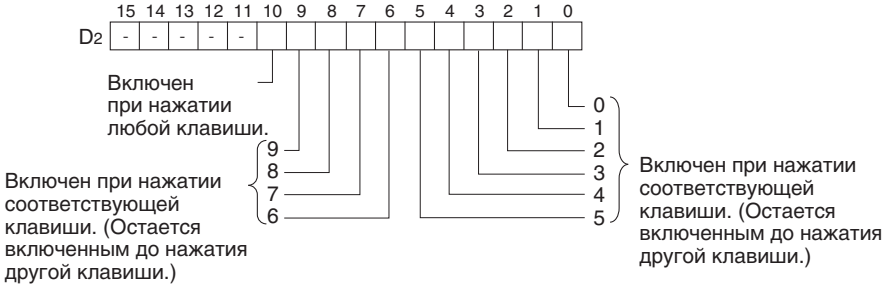
D_1: Первое слово регистра

Указывает адрес первого слова для записи числового значения (до 8 разрядов), вводимого с помощью десятиклавишной клавиатуры.



D_2: Слово индикации нажатия клавиш

Биты 00...10 слова D_2 сигнализируют нажатие клавиш. При нажатии какой-либо клавиши на клавиатуре (0...9) включается соответствующий бит в слове D_2 (0...9). Бит 10 слова D_2 сигнализирует нажатие любой клавиши. Он включен, пока удерживается нажатой какая-нибудь из клавиш.



Характеристики операндов

Область	I	D_1	D_2
Область CIO	CIO 0...CIO 6143	CIO 0...CIO 6142	CIO 0...CIO 6143
Рабочая область	W0...W511	W0...W510	W0...W511
Область битов хранения	H0...H511	H0...H510	H0...H511
Область вспомогательных битов	A0...A959	A448...A958	A448...A959
Область таймеров	T0000...T4095	T0000...T4094	T0000...T4095
Область счетчиков	C0000...C4095	C0000...C4094	C0000...C4095
Область DM	D0...D32767	D0...D32766	D0...D32767
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767		
Постоянные	---		
Регистры данных	DR0...DR15	---	DR0...DR15
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда TKY(211) считывает числовое значение из входного слова I, отведенного для десятиклавишной клавиатуры, которая подключена к модулю входов, и сохраняет до 8 введенных цифр (разрядов) в слова регистра D_1 и D_1+1 в двоично-десятичном коде. Помимо этого при каждом нажатии клавиши включается соответствующий бит в слове D_2 (0...9) (остается во включенном состоянии до нажатия другой клавиши). Бит 10 слова D_2 включен во время нажатия любой клавиши и выключен, если ни одна из клавиш в данный момент не нажата.

Два слова [D_1 и D_1+1] работают как 8-разрядный регистр сдвига. При нажатии клавиши на десятиклавишной клавиатуре в самый младший двоично-десятичный разряд (4 бита) слова D_1 записывается двоичное значение, соответствующее нажатой цифре. Остальные разряды [D_1, D_1+1] сдвигаются влево, при этом самый старший разряд слова D_1+1 утрачивается.

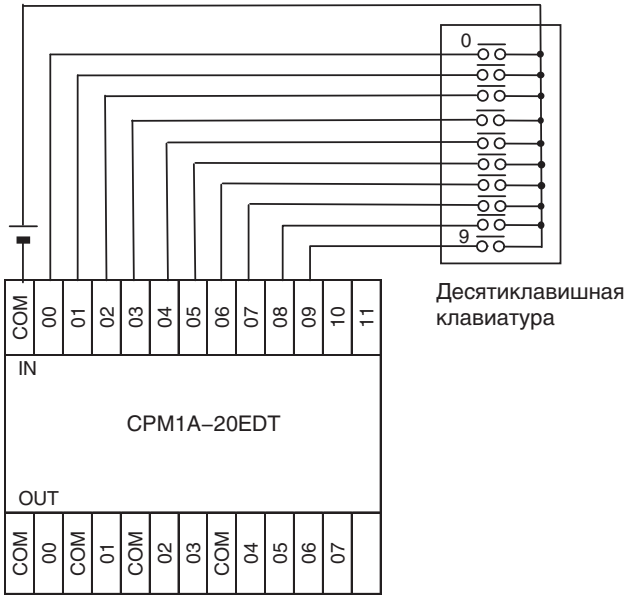
При выполнении команда TKY(211) начинает ввод значения с клавиатуры с самого начала (т. е. с первого цикла), независимо от того, на каком этапе было остановлено выполнение последней команды.

Пока удерживается нажатой одна из клавиш клавиатуры, нажатие любой другой клавиши игнорируется.

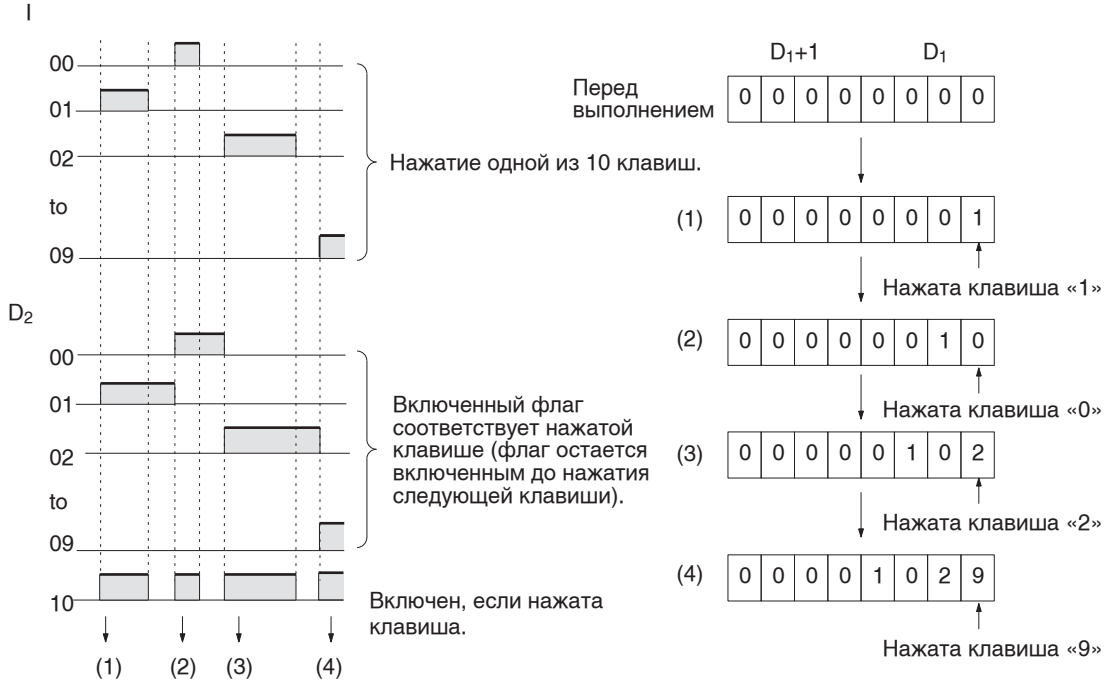
Команда TKY(211) может использоваться в программе любое число раз.

Подключение внешних цепей

Подсоедините цепи десятиклавишной клавиатуры таким образом, чтобы клавишам 0...9 соответствовали входы 0...9 модуля входов, как показано на рисунке ниже.



Временная диаграмма



Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	ВЫКЛ

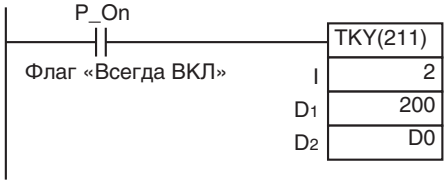
Меры предосторожности

После выполнения команды TKY(211) обязательно должно производиться обновление входов для модуля входов, к которому подключена десятичная клавиатура. В противном случае команда TKY(211) будет работать неправильно. Поэтому не следует подключать десятичную клавиатуру к следующим модулям:

- ведомые коммуникационные устройства (ведомые узлы DeviceNet или CompoBus/S).

Пример

В приведенном примере команда TKY(211) считывает числовое значение, набираемое на десятиклавишной клавиатуре, и записывает это значение в слова CIO 200 и CIO 201. Для связи с десятиклавишной клавиатурой используется слово CIO 2.

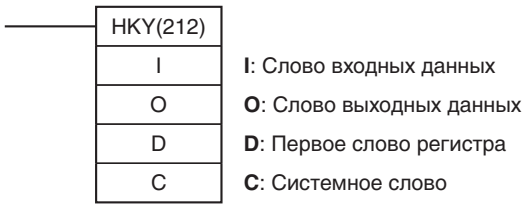


3-22-5 ЧТЕНИЕ ШЕСТНАДЦАТЕРИЧНОЙ КЛАВИАТУРЫ — HKY(212)

Назначение

Чтение числового значения, набираемого на 16-клавишной клавиатуре, которая подключена к модулю входов и модулю выходов, и сохранение до 8 введенных цифр (разрядов) в указанные слова в шестнадцатеричном коде.

Символ РКС



Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	HKY(212)
	Однократное выполнение по положительному фронту	Не предусмотрено
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

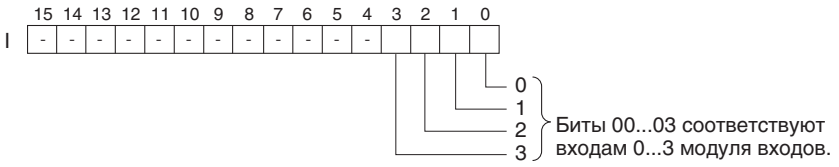
Доступные области программы

Области программных блоков	**Области пошаговых программ**	**Подпрограммы**	**Задачи обработки прерываний**
Не допускается	OK	OK	Не допускается

Операнды

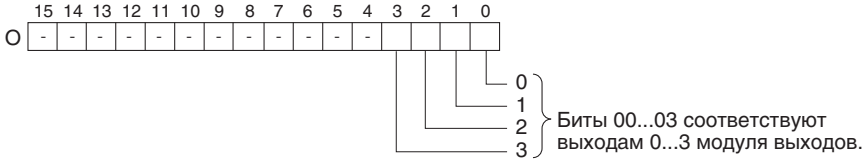
I: Слово входов (входы линий данных D0...D3)

Укажите слово входов, отведенное для модуля входов, и подсоедините линии данных D0...D3 16-клавишной клавиатуры к модулю входов, как показано на рисунке ниже.



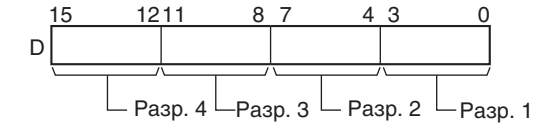
O: Слово выходов (выходы сигналов селекции)

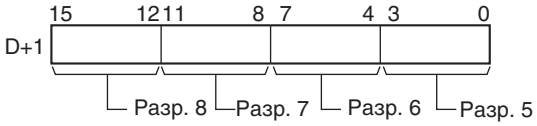
Укажите слово выходов, отведенное для модуля выходов, и подсоедините сигналы селекции 16-клавишной клавиатуры к модулю выходов, как показано на рисунке ниже.

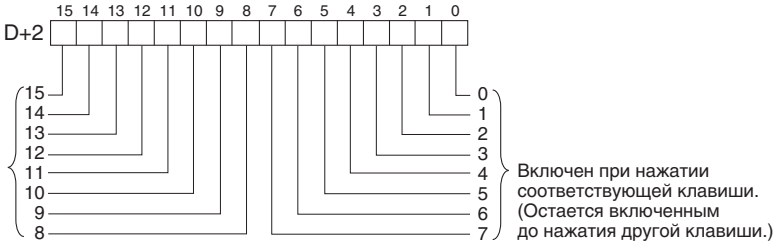


D: Первое слово регистра

Указывает адрес первого слова для записи числового значения (до 8 разрядов), вводимого с помощью 16-клавишной клавиатуры. (Помимо этого при каждом нажатии клавиши включается соответствующий бит в слове D+2 (0...F) (остается во включенном состоянии до нажатия другой клавиши).)

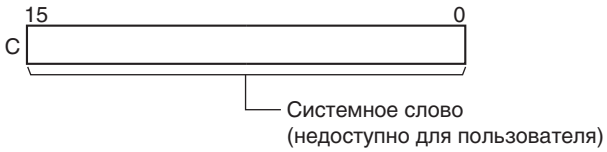






C: Системное слово

Указывает рабочее слово, используемое командой. Это слово будет невозможно использовать для каких-либо иных целей.



Характеристики операндов

Область	I	O	D	C
Область CIO	CIO 0...CIO 6143		CIO 0...CIO 6141	CIO 0...CIO 6143
Рабочая область	W0...W511		W0...W509	W0...W511
Область битов хранения	H0...H511		H0...H509	H0...H511
Область вспомогательных битов	A0...A957	A448... A959	A448...A957	A448...A959
Область таймеров	T0000...T4095		T0000...T4093	T0000...T4095
Область счетчиков	C0000...C4095		C0000...C4093	C0000...C4095

Область	I	O	D	C
Область DM	D0...D32767		D0...D32765	D0...D32767
Косвенные адреса DM в двоичном формате	@ D0...@ D32767			
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767			
Постоянные	---			
Регистры данных	DR0...DR15		---	DR0...DR15
Регистры указателей	---			
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15			

Описание

Команда HKY(212) выдает сигналы селекции в биты 00...03 выходного слова O, считывает вводимое значение, используя биты 00...03 входного слова I, и сохраняет до 8 введенных шестнадцатеричных цифр (разрядов) в слова регистра D и D+1.

Команда HKY(212) вводит каждый разряд за 3...12 циклов ПЛК, после чего выполнение начинается сначала и ввод продолжается. Помимо этого при каждом нажатии клавиши включается соответствующий бит в слове D+2 (0...F) (остается во включенном состоянии до нажатия другой клавиши).

Для того чтобы определить, какая именно клавиша нажата, команда HKY(212) поочередно включает сигналы селекции (биты 0...3 слова O) и находит включенный вход. Таким образом, чтение одного шестнадцатеричного разряда может занять от 3 до 12 циклов. После того как нажатая клавиша считана, команда HKY(212) начинает выполняться сначала и считывает другой разряд за следующие 3...12 циклов.

При выполнении команда HKY(212) начинает цикл считывания разряда с самого начала (т. е. с первого сигнала селекции), независимо от того, на каком этапе было остановлено выполнение последней команды.

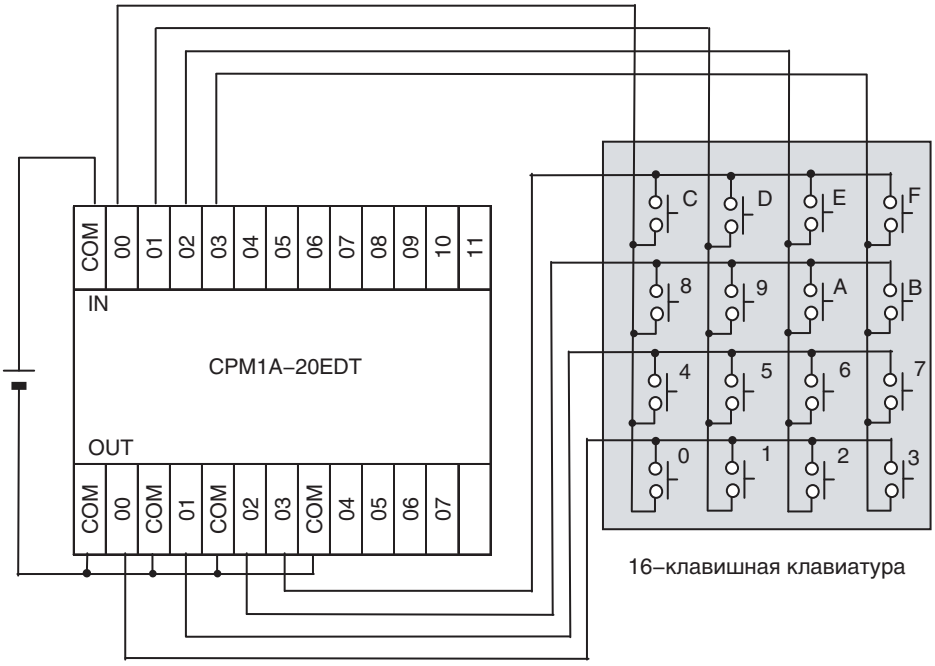
Два слова [D_1 и D_1+1] работают как 8-разрядный регистр сдвига. При нажатии клавиши на 16-клавишной клавиатуре в самый младший шестнадцатеричный разряд (4 бита) слова D_1 записывается значение, соответствующее нажатой клавише. Остальные разряды [D_1, D_1+1] сдвигаются влево, при этом самый старший разряд слова D_1+1 утрачивается.

Пока удерживается нажатой одна из клавиш клавиатуры, нажатие любой другой клавиши игнорируется.

Команда HKY(212) может использоваться в программе любое число раз.

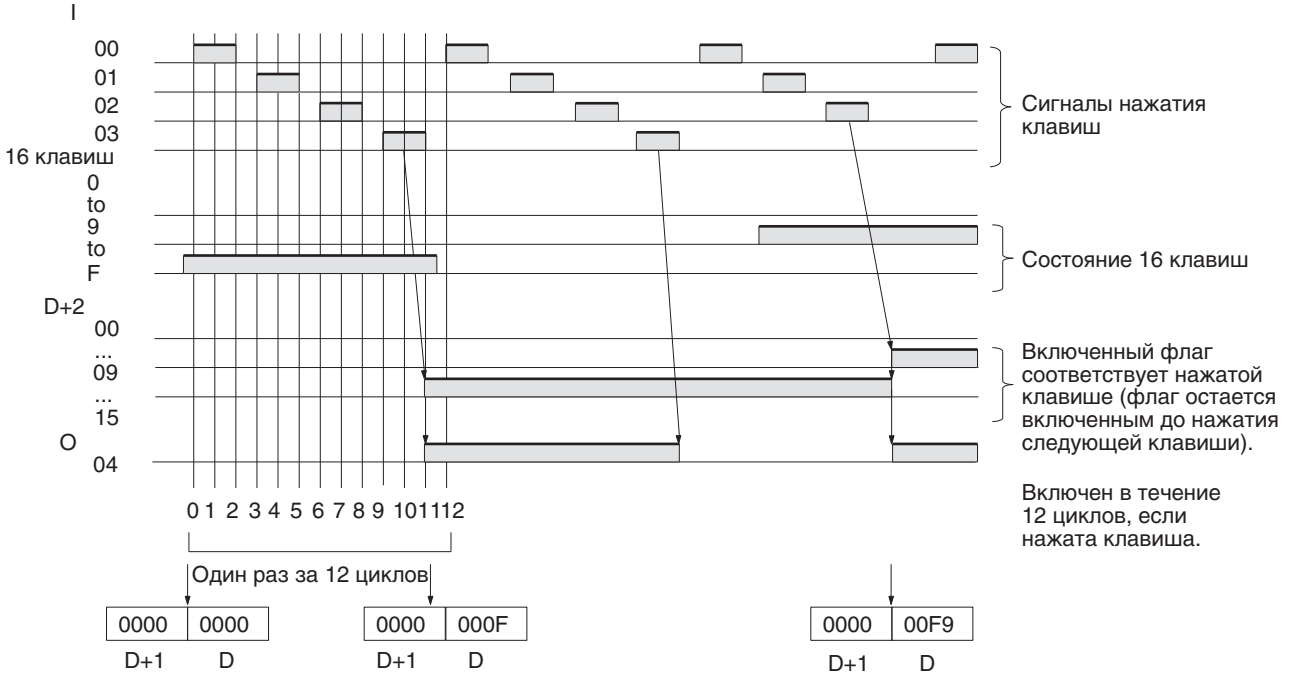
Подключение внешних цепей

Подключите шестнадцатеричную клавиатуру к контактам 0...3 модуля входов и контактам 0...3 модуля выходов, как показано на следующем рисунке.



16–клавишная клавиатура

Временная диаграмма



Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	ВЫКЛ

Меры предосторожности

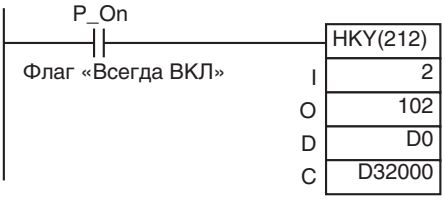
Никакая другая команда не должна обращаться к системному слову (C) с целью чтения или записи. При обращении другой команды к системному слову команда HKY(212) будет работать неправильно. В самом первом цикле выполнения прикладной программы команда HKY(212) не инициализирует содержимое системного слова. Поэтому, если команда HKY(212) должна выполняться в самом первом цикле ПЛК, в программе должна быть предусмотрена очистка содержимого системного слова.

После выполнения команды HKY(212) обязательно должно производиться обновление входов/выходов для модуля входов и модуля выходов, к которым подключена 16-клавишная клавиатура. В противном случае команда HKY(212) будет работать неправильно. Поэтому не следует подключать 16-клавишную клавиатуру к следующим модулям:

- ведомые коммуникационные устройства (ведомые узлы DeviceNet или CompoBus/S).

Пример

В приведенном ниже примере команда HKY(212) считывает до 8 шестнадцатеричных цифр (разрядов) с 16-клавишной клавиатуры и сохраняет введенное значение в слова D0 и D1. Для связи с 16-клавишной клавиатурой используются слова CIO 2 и CIO 102. В качестве системного слова используется слово D32000.

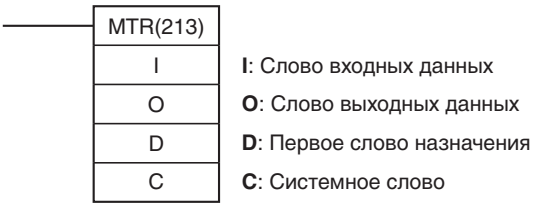


3-22-6 ЧТЕНИЕ МАТРИЦЫ: MTR(213)

Назначение

Ввод до 64 сигналов с матрицы размером 8 × 8, подсоединенный к 8 входам модуля входов и 8 выходам модуля выходов, и сохранение введенного 64-битового значения в 4 слова назначения.

Символ РКС



Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	MTR(213)
	Однократное выполнение по положительному фронту	Не предусмотрено
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

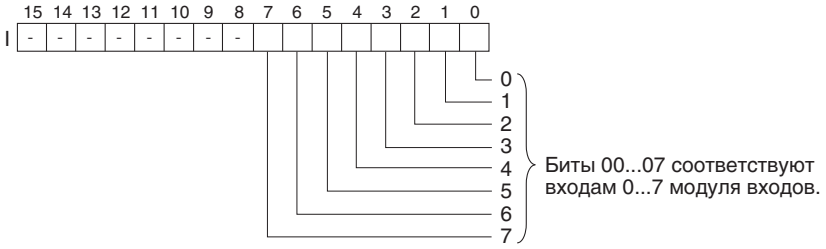
Доступные области программы

Области программных блоков	**Области пошаговых программ**	**Подпрограммы**	**Задачи обработки прерываний**
Не допускается	OK	OK	Не допускается

Операнды

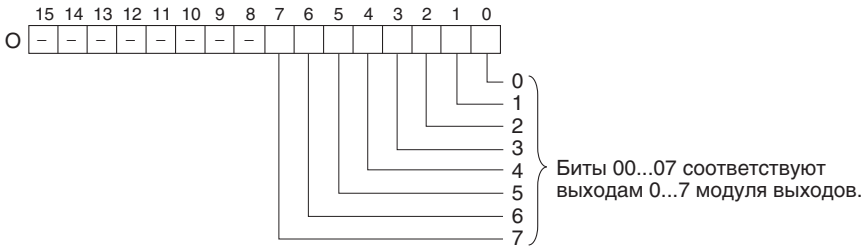
I: Слово входов

Укажите слово входов, отведенное для модуля входов, и подсоедините 8 линий входных сигналов к модулю входов, как показано на рисунке ниже.



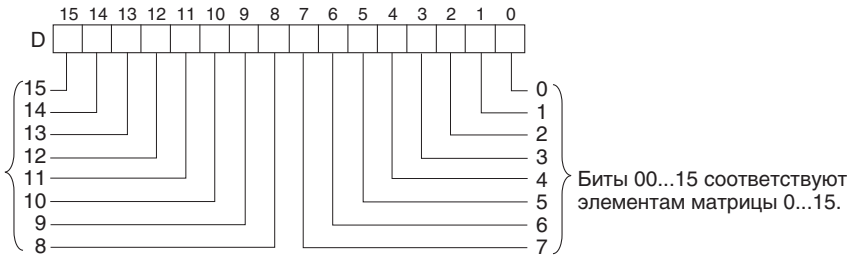
O: Слово выходов (выходы сигналов селекции)

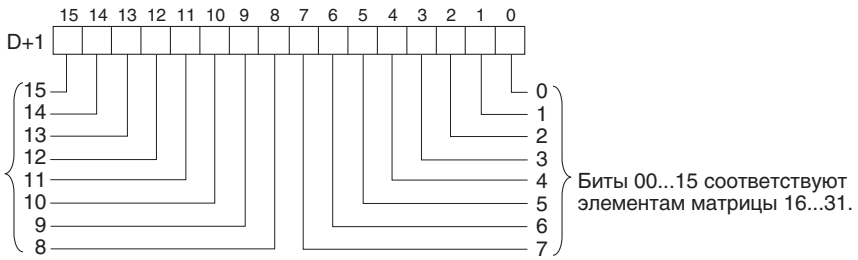
Укажите слово выходов, отведенное для модуля выходов, и подсоедините 8 линий сигналов селекции к модулю выходов, как показано на рисунке ниже.

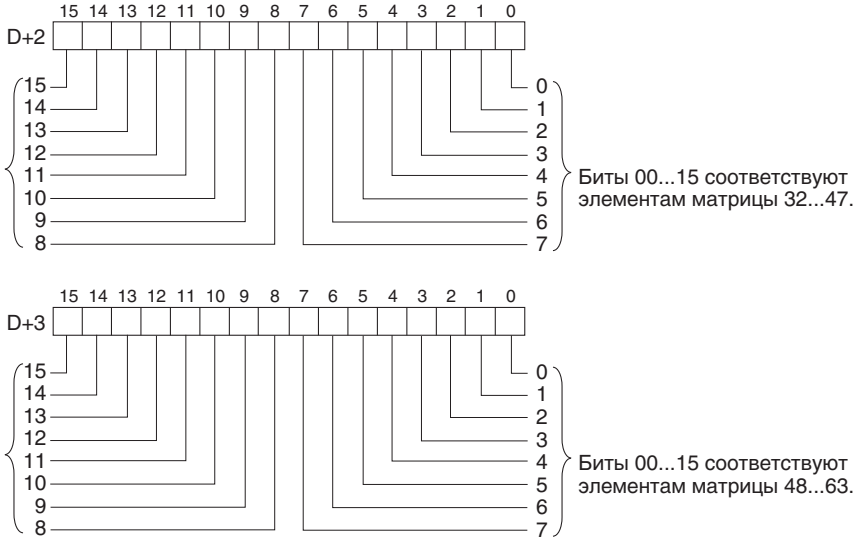


D: Первое слово регистра

Указывает адрес первого из четырех слов, содержащих значение, введенное с матрицы 8 × 8.

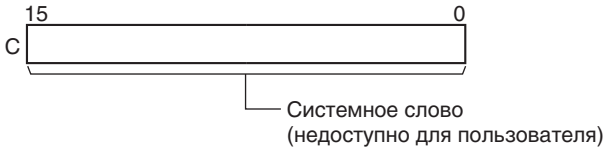






C: Системное слово

Указывает рабочее слово, используемое командой. Это слово будет невозможно использовать для каких-либо иных целей.



Характеристики операндов

Область	I	O	D	C
Область CIO	CIO 0...CIO 6143		CIO 0...CIO 6140	CIO 0...CIO 6143
Рабочая область	W0...W511		W0...W508	W0...W511
Область битов хранения	H0...H511		H0...H508	H0...H511
Область вспомогательных битов	A0...A959	A448...A959	A448...A956	A448...A959
Область таймеров	T0000...T4095		T0000...T4092	T0000...T4095
Область счетчиков	C0000...C4095		C0000...C4092	C0000...C4095
Область DM	D0...D32767		D0...D32764	D0...D32767
Косвенные адреса DM в двоичном формате	@ D0...@ D32767			
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767			
Постоянные	---			
Регистры данных	DR0...DR15		---	DR0...DR15
Регистры указателей	---			
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15			

Описание

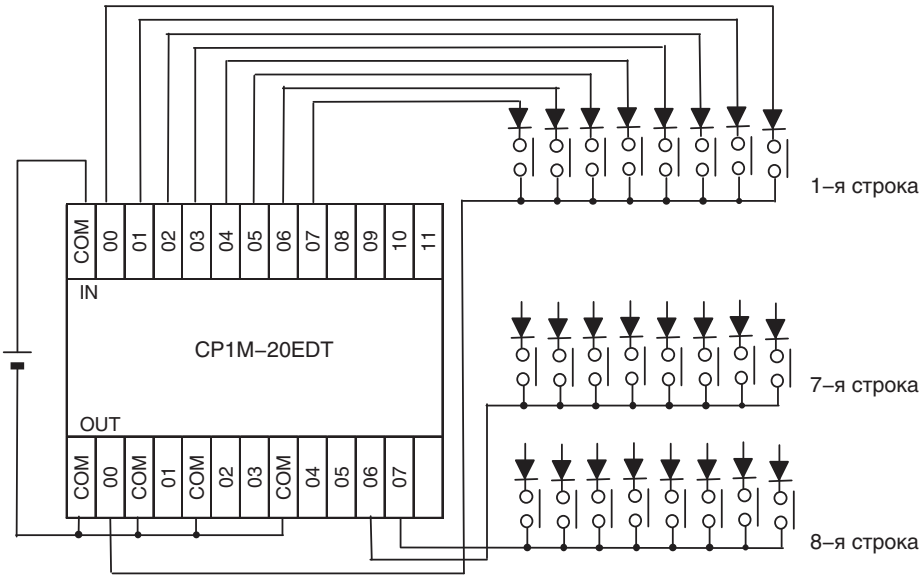
Команда MTR(213) подает сигналы селекции, используя биты 00...07 выходного слова O, считывает по очереди вводимые данные, используя биты 00...07 входного слова I, и сохраняет введенное 64-битовое значение в слова D...D+3 (4 слова). Команда MTR(213) считывает одно состояние 64-битовой матрицы за 24 цикла модуля ЦПУ. По завершении 24-х циклов (после последовательного включения каждого из восьми сигналов селекции) включается на 1 цикл флаг завершения полного цикла (бит 08 слова O).

При выполнении команда MTR(213) приступает к чтению состояния матрицы с самого начала (т. е. с первого сигнала селекции), независимо от того, на каком этапе было остановлено выполнение последней команды.

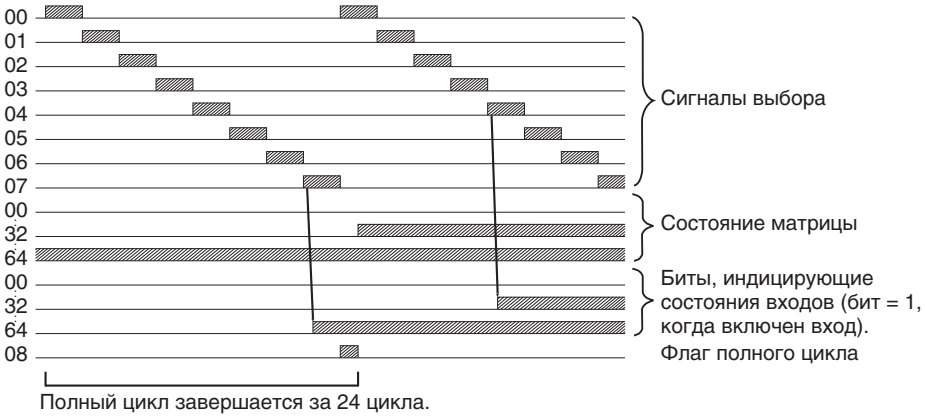
Команда MTR(213) может использоваться в программе любое число раз.

Подключение внешних цепей

Подключите 64-битовую матрицу к контактам 0...7 модуля входов и контактам 0...7 модуля выходов, как показано на следующем рисунке.



Временная диаграмма



Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	ВЫКЛ

Меры предосторожности

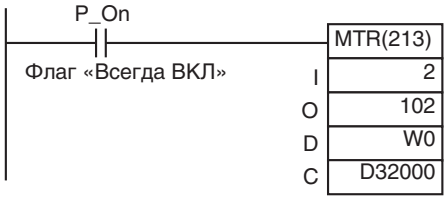
Никакая другая команда не должна обращаться к системному слову (C) с целью чтения или записи. При обращении другой команды к системному слову команда MTR(213) будет работать неправильно. В самом первом цикле выполнения прикладной программы команда MTR(213) не инициализирует содержимое системного слова. Поэтому, если команда MTR(213) должна выполняться в самом первом цикле ПЛК, в программе должна быть предусмотрена очистка содержимого системного слова.

После выполнения команды MTR(213) обязательно должно производиться обновление входов/выходов для модуля входов и модуля выходов, к которым подключена внешняя матрица. В противном случае команда MTR(213) будет работать неправильно. Поэтому не следует подключать внешнюю матрицу к следующим модулям:

- ведомые коммуникационные устройства (ведомые узлы DeviceNet или CompoBus/S).

Пример

В приведенном ниже примере команда MTR(213) считывает 64-битовое значение с матрицы 8 × 8 и сохраняет это значение в слова W0...W3. Для связи с матрицей 8 × 8 используются слова CIO 2 и CIO 102. В качестве системного слова используется слово D32000.

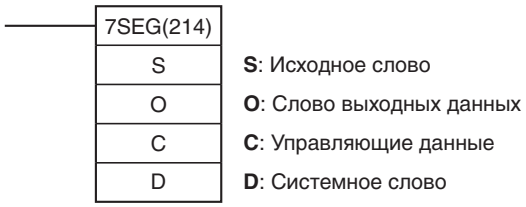


3-22-7 ВЫВОД НА 7-СЕГМЕНТНЫЙ ИНДИКАТОР — 7SEG(214)

Назначение

Преобразование исходного значения (4- или 8-разрядного в двоично-десятичном формате) в значение для отображения на 7-сегментном индикаторе и вывод этого значения в указанное выходное слово.

Символ РКС



Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	7SEG(214)
	Однократное выполнение по положительному фронту	Не предусмотрено
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

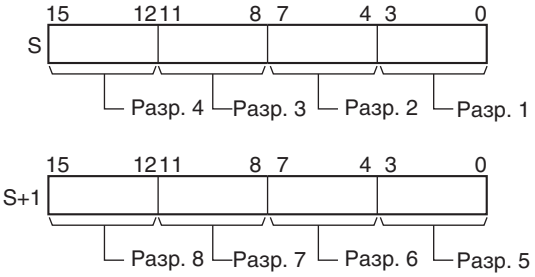
Доступные области программы

Области программных блоков	**Области пошаговых программ**	**Подпрограммы**	**Задачи обработки прерываний**
Не допускается	OK	OK	Не допускается

Операнды

S: Исходное слово

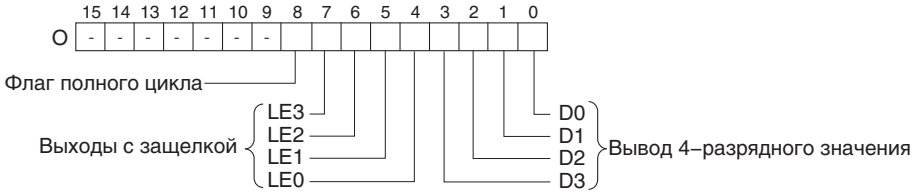
Укажите первое исходное слово, содержащее значение, которое должно быть преобразовано в значение для 7-сегментного индикатора.



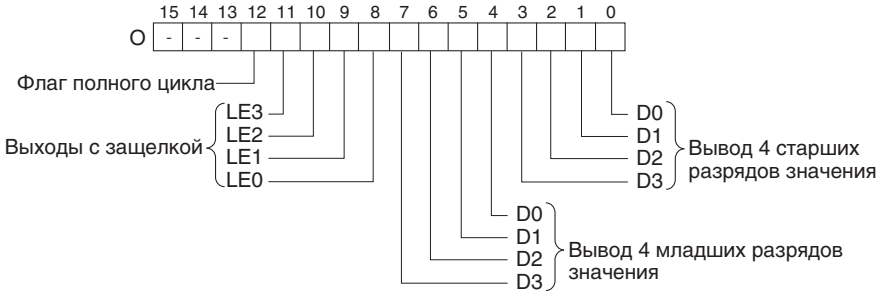
O: Слово выходов (выходы данных и выходы защелкивания)

Укажите слово выходов, отведенное для модуля выходов, и подсоедините 7-сегментный индикатор к модулю выходов, как показано на рисунке ниже.

- Преобразование 4 разрядов



- Преобразование 8 разрядов



C: Управляющие данные

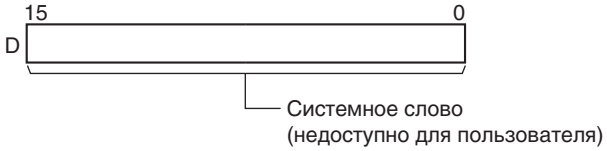
Значение операнда C указывает количество разрядов преобразуемого значения и тип входных и выходных цепей в соответствии со следующей таблицей (под типом цепи понимается тип транзисторного выхода: NPN или PNP).

Исходное значение	Тип входов данных индикатора	Тип входов защелкивания индикатора	C
4 разряда (S)	Такой же, как у модуля выходов	Такой же, как у модуля выходов	0000
		Отличный от модуля выходов	0001
	Отличный от модуля выходов	Такой же, как у модуля выходов	0002
		Отличный от модуля выходов	0003

Исходное значение	Тип входов данных индикатора	Тип входов защелкивания индикатора	C
8 разрядов (S, S+1)	Такой же, как у модуля выходов	Такой же, как у модуля выходов	0004
		Отличный от модуля выходов	0005
	Отличный от модуля выходов	Такой же, как у модуля выходов	0006
		Отличный от модуля выходов	0007

D: Системное слово

Указывает рабочее слово, используемое командой. Это слово будет невозможно использовать для каких-либо иных целей.



Характеристики операндов

Область	S	O	C	D
Область CIO	CIO 0...CIO 6143		---	CIO 0...CIO 6143
Рабочая область	W0...W511		---	W0...W511
Область битов хранения	H0...H511		---	H0...H511
Область вспомогательных битов	A0...A959	A448...A959	---	A448...A959
Область таймеров	T0000...T4095		---	T0000...T4095
Область счетчиков	C0000...C4095		---	C0000...C4095
Область DM	D0...D32767		---	D0...D32767
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		---	@ D0...@ D32767
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767			
Постоянные	---	---	0000... 0007	---
Регистры данных	---	DR0...DR15	---	DR0...DR15
Регистры указателей	---			
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047, IR0...-2048...+2047, IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		---	,IR0...,IR15 -2048...+2047 , IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)..., IR15+(++) ,–(– –)IR0... ,–(– –)IR15

Описание

Команда 7SEG(214) принимает исходное значение, преобразует его в значение для отображения на 7-сегментном индикаторе и выдает это значение (старшие 4 разряда D0...D3, младшие 4 разряда D0...D3, сигналы выходов защелкивания LE0...LE3) на 7-сегментный индикатор, подключенный к выходу, который указан операндом O. Значение операнда C указывает количество разрядов исходного значения (4 или 8 разрядов) и тип цепи (NPN или PNP) модулей входов и выходов.

Команда 7SEG(214) отображает одно 4-разрядное или 8-разрядное значение за 12 циклов ПЛК, после чего выполняется с самого начала и отображает следующее значение.

По истечении 12 циклов (после того, как команда 7SEG(214) поочередно включит каждый из четырех выходных сигналов защелкивания) на один цикл включается флаг завершения полного цикла (бит 08 слова O при преобразовании 4 разрядов, бит 12 слова O при преобразовании 8 разрядов). Завершив вывод значения на 7-сегментный индикатор за 12 циклов, команда 7SEG(214) начинает выполняться сначала и преобразует текущее значение исходных слов за следующие 12 циклов.

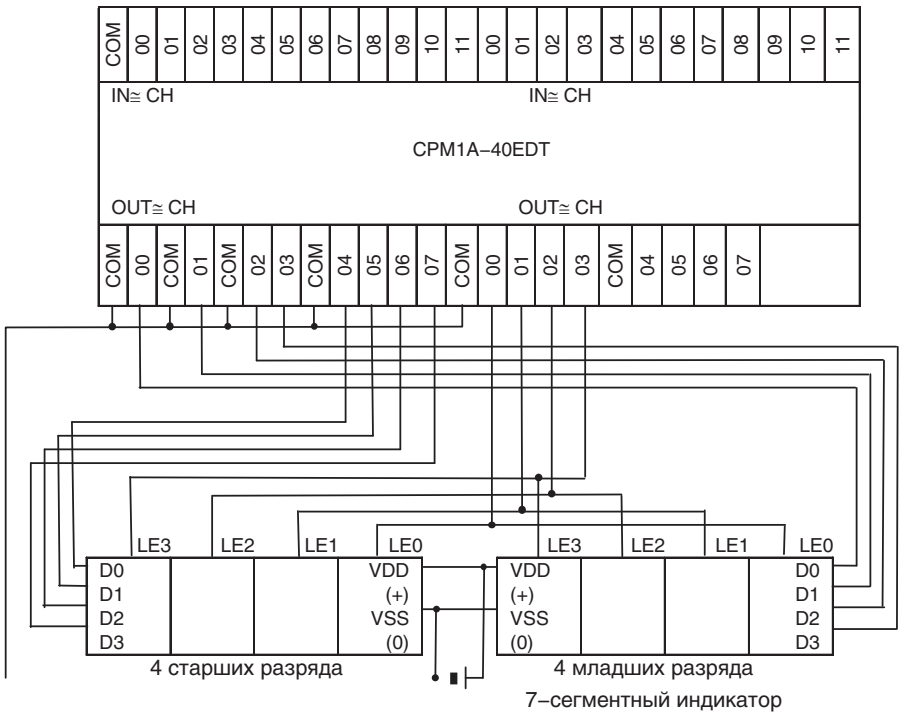
При выполнении команда 7SEG(214) начинает цикл вывода значения с самого начала (т. е. с выхода защелкивания 0), независимо от того, на каком этапе было остановлено выполнение последней команды.

Даже если в подключенном 7-сегментном индикаторе имеется меньше четырех или восьми разрядов, команда 7SEG(214) все равно выдает на индикатор 4-разрядное или 8-разрядное значение.

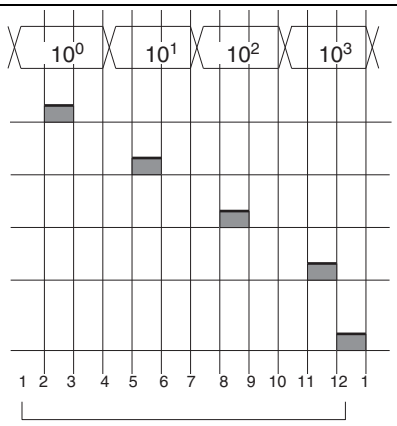
Подключение внешних цепей

Выполните подключение 7-сегментного индикатора к модулю выходов в соответствии с приведенной ниже схемой. Приведена схема для 8-разрядного индикатора. В случае 4-разрядного индикатора выходы данных (D0...D3) должны подключаться к выходам 0...3, а выходы защелкивания (LE0...LE3) должны подключаться к выходам 4...7.

По завершении полного цикла вывода одного значения на выход 12 (для 8-разрядного индикатора) или выход 8 (для 4-разрядного индикатора) подается сигнал завершения цикла, однако этот сигнал использовать не обязательно.



Временная диаграмма

Функция	Бит(-ы) слова O		Состояние выходов (тип цепи (NPN или PNP) и, соответственно, активный уровень («0» или «1») определяются операндом C)
	(4 разряда, 1 блок)	(4 разряда, 2 блока)	
Выдача данных	00...03	00...03 04...07	 **Прим.** 0...3: вывод значения слова S; 4...7: вывод значения слова S+1. Для завершения полного цикла требуется 12 циклов.
Выход защелк. 0	04	08	
Выход защелк. 1	05	09	
Выход защелк. 2	06	10	
Выход защелк. 3	07	11	
Флаг полного цикла	08	12	

Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	ВЫКЛ

Меры предосторожности

Никакая другая команда не должна обращаться к системному слову (D) с целью чтения или записи. При обращении другой команды к системному слову команда 7SEG(214) будет работать неправильно. В самом первом цикле выполнения прикладной программы команда 7SEG(214) не инициализирует содержимое системного слова. Поэтому, если команда 7SEG(214) должна выполняться в самом первом цикле ПЛК, в программе должна быть предусмотрена очистка содержимого системного слова.

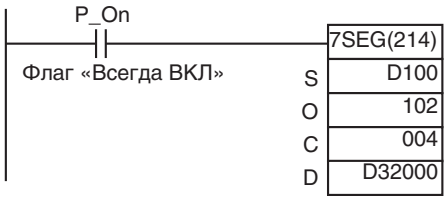
После выполнения команды 7SEG(214) обязательно должно производиться обновление выходов для модуля выходов, к которому подключен 7-сегментный индикатор. В противном случае команда 7SEG(7) будет работать неправильно. Поэтому не следует подключать 7-сегментный индикатор к следующим модулям:

- ведомые коммуникационные устройства (ведомые узлы DeviceNet или CompoBus/S).

Пример

Ниже приведен пример программы, в котором команда 7SEG(214) преобразует 8 двоично-десятичных разрядов слов D100 и D101 и выдает преобразованное значение с помощью слова CIO 102.

Отображается 8-разрядное значение, при этом 7-сегментный индикатор и модуль выходов имеют входы и выходы одного типа, поэтому управляющие слова (C) содержат значение 0004. В качестве системного слова D используется слово D32000.



Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

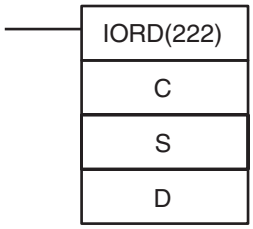
3-22-8 ЧТЕНИЕ ИНТЕЛЛЕКТУАЛЬНЫХ ВХОДОВ/ВЫХОДОВ: IORD(222)

Назначение

Чтение содержимого области памяти специального модуля ввода/вывода или модуля шины ЦПУ.

Эту команду возможно использовать только для модулей ЦПУ CP1H. Для модулей ЦПУ CP1L она недоступна. Если команда IORD(222) будет использована для модуля ЦПУ CP1L, будет установлен флаг ошибки (ER).

Символ РКС



C: Управляющие данные

S: Источник передачи и количество слов

D: Получатель данных

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	IORD(222)
	Однократное выполнение по положительному фронту	@IORD(222)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

C: Зависит от специального модуля ввода/вывода или модуля шины ЦПУ.

S: Специальный модуль ввода/вывода:0000...005F hex
(соответствует номерам модуля от 0 до 95)
Модуль шины ЦПУ:8000...800F hex
(соответствует номерам модуля от 0 до F hex)

S+1: Количество передаваемых слов
(0001...0080 hex, зависит от спец. модуля ввода/вывода или модуля шины ЦПУ)



S+1: 4 старших разряда
S: 4 младших разряда

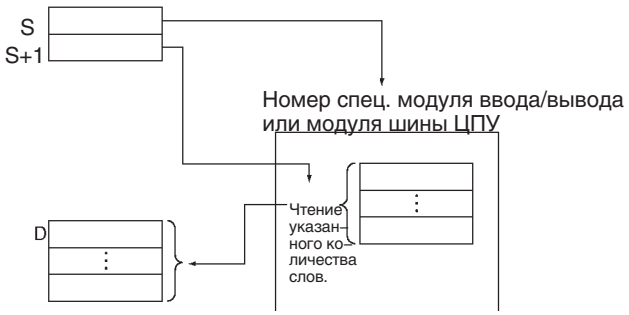
Характеристики операндов

Область	C	S	D
Область CIO	CIO 0...CIO 6143	CIO 0...CIO 6142	CIO 0...CIO 6143
Рабочая область	W0...W511	W0...W510	W0...W511
Область битов хранения	H0...H511	H0...H510	H0...H511
Область вспомогательных битов	A0...A959	A0...A958	A448...A959
Область таймеров	T0000...T4095	T0000...T4094	T0000...T4095
Область счетчиков	C0000...C4095	C0000...C4094	C0000...C4095
Область DM	D0...D32767	D0...D32766	D0...D32767
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		

Область	C	S	D
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767		
Постоянные	#0000...#FFFF (двоичн.)	Только указанные значения	---
Регистры данных	DR0...DR15	---	
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда IORD(222) считывает содержимое указанного операндом S+1 количества слов из области памяти специального модуля ввода/вывода или модуля шины ЦПУ, номер модуля которого указан операндом S, и выдает считанные данные по адресу, указанному в D. Подробную информацию о содержании считываемых данных для каждого модуля можно найти в руководстве по эксплуатации для соответствующего специального модуля ввода/вывода или модуля шины ЦПУ.



Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	Включен, если количество передаваемых слов (S) выходит за диапазон 0001...0080 hex. Включен, если номер модуля (S) выходит за диапазон 0000...005F hex или 8000...800F hex. Включен, если в команде IORD(222) указан неподдерживаемый специальный модуль ввода/вывода или модуль шины ЦПУ. Включен при наличии ошибки настройки или ошибки в указанном спец. модуле ввода/вывода. Включен при наличии ошибки настройки или ошибки в указанном модуле шины ЦПУ. Выключен во всех остальных случаях.
Флаг равенства	=	Включен при успешном завершении операции чтения. Выключен при завершении операции чтения с ошибкой.

Меры предосторожности

Если операция чтения завершится успешно (т. е. без ошибок), будет установлен флаг равенства.

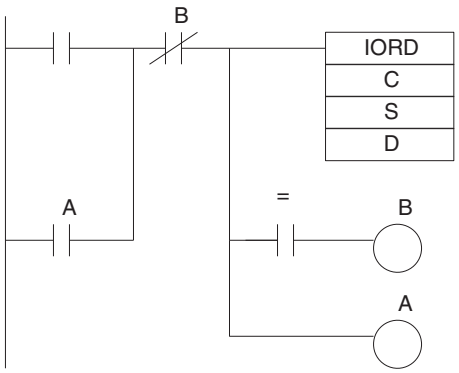
Если операция чтения не будет завершена успешно из-за занятости специального модуля ввода/вывода или модуля шины ЦПУ, флаг равенства будет сброшен.

Любое из перечисленных ниже нарушений вызовет ошибку и установку флага ошибки.

- Количество слов для передачи (S) выходит за допустимый диапазон 0001...0080 (hex).
- Номер модуля (S) выходит за допустимый диапазон 0000...005F hex или 8000...800F hex.
- Указан специальный модуль ввода/вывода или модуль шины ЦПУ, который не поддерживается командой IORD(222).
- В указанном спец. модуле ввода/вывода имеется ошибка настройки или ошибка.
- В указанном модуле шины ЦПУ имеется ошибка настройки или ошибка.

Результаты выполнения команды IORD(222) отражаются во флагах условий. В частности, по завершении чтения устанавливается флаг равенства. Для использования флага условия (например, флага равенства) в разветвленном выходе флаг условия следует присоединять к тем же входным условиям, что используются для команды IORD(222).

Если специальный модуль ввода/вывода или модуль шины ЦПУ будет занят, операция чтения выполнена не будет. Ниже показан пример программы, в которой, за счет использования флага равенства, команда IORD(222) выполняется в каждом цикле, пока не оказывается выполнена операция чтения.

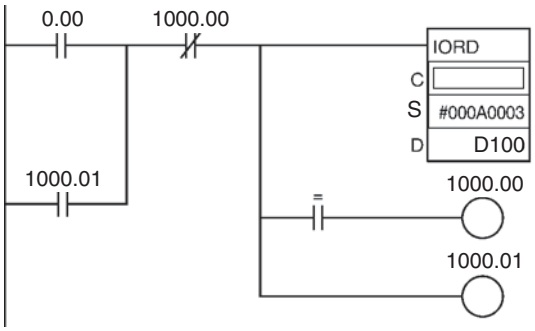


При включении входного условия включается выход A, за счет чего команда IORD(222) продолжает выполняться в последующих циклах, пока не устанавливается флаг равенства. Когда чтение завершится и установится флаг равенства, включится выход B, и выполнение команды IORD(222) прекратится.

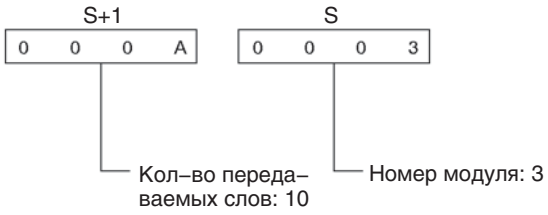
Обязательно размещайте флаги условий сразу за командой IORD(222), а не после каких-либо других команд. Если флаг условия будет размещен после другой команды, на его состояние будет влиять выполнение этой другой команды.

Пример

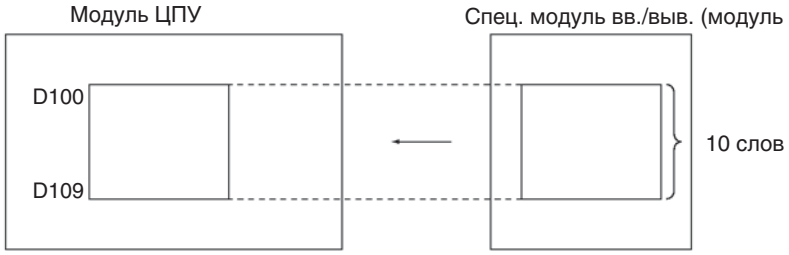
Ниже показан пример использования команды IORD(222) для чтения данных.



Если CIO 0.00 = 1, читается содержимое 10 слов специального модуля ввода/вывода с номером модуля 3 и записывается в слова D100...D109.



Содержание управляющих данных (C) зависит от модели специального модуля ввода/вывода.



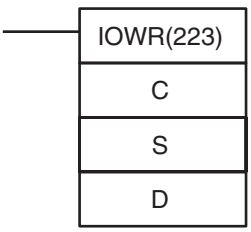
3-22-9 ЗАПИСЬ ИНТЕЛЛЕКТУАЛЬНЫХ ВХОДОВ/ВЫХОДОВ: IOWR(223)

Назначение

Вывод содержимого области памяти ввода/вывода модуля ЦПУ в память специального модуля ввода/вывода или модуля шины ЦПУ.

Эту команду возможно использовать только для модулей ЦПУ CP1H. Для модулей ЦПУ CP1L она недоступна. Если команда IOWR(223) будет использована для модуля ЦПУ CP1L, будет установлен флаг ошибки (ER).

Символ РКС



C: Управляющие данные

S: Источник передачи и количество слов

D: Адресат передачи и количество слов

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	IOWR(223)
	Однократное выполнение по положительному фронту	@IOWR(223)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

C: Зависит от специального модуля ввода/вывода или модуля шины ЦПУ.

D: Специальный модуль ввода/вывода:0000...005F hex (соответствует номерам модулей от 0 до 95) Модуль шины ЦПУ:8000...800F hex (соответствует номерам модулей от 0 до F hex)

D+1: Количество передаваемых слов (0001...0080 hex, зависит от спец. модуля ввода/вывода или модуля шины ЦПУ)



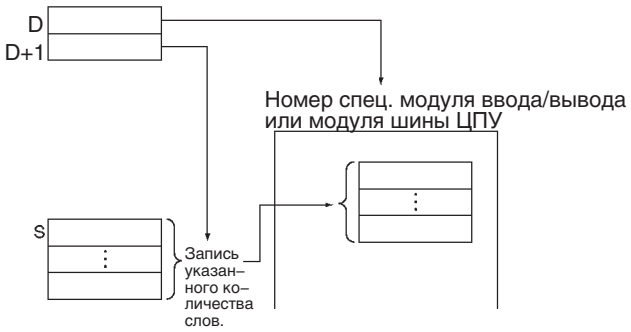
D+1: 4 старших разряда
D: 4 младших разряда

Характеристики операндов

Область	C	S	D
Область CIO	CIO 0...CIO 6143		CIO 0...CIO 6142
Рабочая область	W0...W511		W0...W510
Область битов хранения	H0...H511		H0...H510
Область вспомогательных битов	A0...A959		A0...A958
Область таймеров	T0000...T4095		T0000...T4094
Область счетчиков	C0000...C4095		C0000...C4094
Область DM	D0...D32767		D0...D32766
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767		
Постоянные	#0000...#FFFF (двоичн.)		Только указанные значения
Регистры данных	DR0...DR15	---	---
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда IOWR(223) записывает содержимое указанного операндом D количества слов, начиная с первого исходного слова (указанного операндом S), в специальный модуль ввода/вывода или модуль шины ЦПУ, номер модуля которого указан операндом D.



Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	Включен, если количество передаваемых слов (D) выходит за диапазон 0001...0080 hex. Включен, если номер модуля (D) выходит за диапазон 0000...005F hex или 8000...800F hex. Включен, если в S введена константа, тогда как количество слов для передачи (D+1) не равно 0001 hex. Включен, если в команде IOWR(223) указан неподдерживаемый специальный модуль ввода/вывода или модуль шины ЦПУ. Включен при наличии ошибки настройки или ошибки в указанном спец. модуле ввода/вывода. Включен при наличии ошибки настройки или ошибки в указанном модуле шины ЦПУ. Выключен во всех остальных случаях.
Флаг равенства	=	Включен при успешном завершении операции записи. Выключен при завершении операции записи с ошибкой.

Меры предосторожности

При выборе значения «0001» для количества передаваемых слов (D+1) в операнд S можно ввести константу. Однако, если будет указано другое количество слов (не 0001), а в S будет введена константа, произойдет ошибка и будет установлен флаг ошибки.

Если операция записи завершится успешно (т. е. без ошибок), будет установлен флаг равенства.

Если операция записи не будет завершена успешно из-за занятости специального модуля ввода/вывода или модуля шины ЦПУ, флаг равенства будет сброшен.

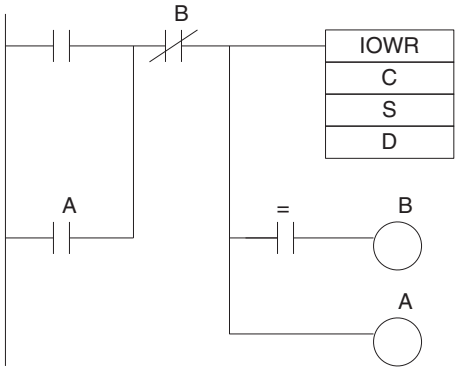
Любое из перечисленных ниже нарушений вызовет ошибку и установку флага ошибки.

- Имеется ошибка проверки модуля ввода/вывода, ошибка настройки специального модуля ввода/вывода или другая ошибка в самом специальном модуле ввода/вывода.
- Имеется ошибка проверки модуля шины ЦПУ, ошибка настройки модуля шины ЦПУ или другая ошибка в самом модуле шины ЦПУ.
- Количество слов для передачи (D) выходит за допустимый диапазон 0001...0080 (hex).

- Номер модуля (D) выходит за допустимый диапазон 0000...005F hex или 8000...800F hex.
- Указан специальный модуль ввода/вывода или модуль шины ЦПУ, который не поддерживается командой IOWR(223).
- В указанном спец. модуле ввода/вывода имеется ошибка настройки или ошибка.
- В указанном модуле шины ЦПУ имеется ошибка настройки или ошибка.

Результаты выполнения команды IOWR(223) отражаются во флагах условий. В частности, по завершении чтения устанавливается флаг равенства. Для использования флага условия (например, флага равенства) в разветвленном выходе флаг условия следует присоединять к тем же входным условиям, что используются для команды IOWR(223).

Если специальный модуль ввода/вывода или модуль шины ЦПУ будет занят, операция записи выполнена не будет. Ниже показан пример программы, в которой, за счет использования флага равенства, команда IOWR(223) выполняется в каждом цикле, пока не оказывается выполнена операция записи.

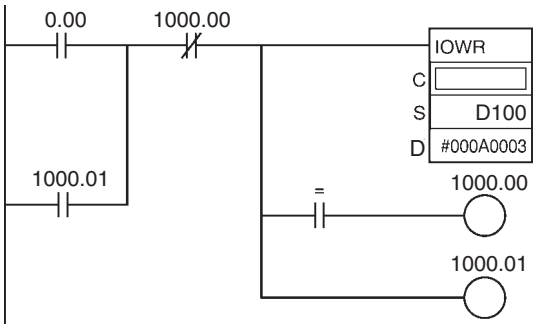


При включении входного условия включается выход A, за счет чего команда IOWR(223) продолжает выполняться в последующих циклах, пока не устанавливается флаг равенства. Когда запись завершится и установится флаг равенства, включится выход B, и выполнение команды IOWR(223) прекратится.

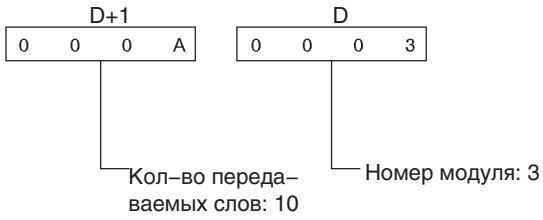
Обязательно размещайте флаги условий сразу за командой IOWR(223), а не после каких-либо других команд. Если флаг условия будет размещен после другой команды, на его состояние будет влиять выполнение этой другой команды.

Пример

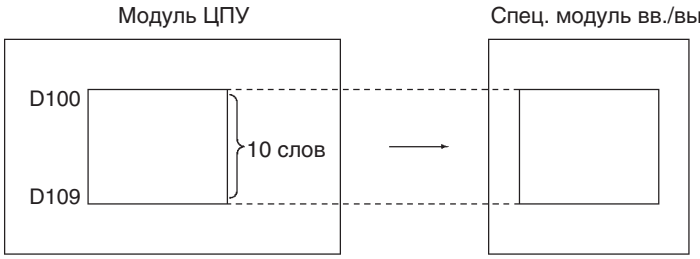
Ниже показан пример использования команды IOWR(223) для записи данных.



Если CIO 0.00 = 1, содержимое 10 слов D100...D109 записывается в специальный модуль ввода/вывода.



Содержание управляющих данных (C) зависит от модели специального модуля ввода/вывода.



3-22-10 ОБНОВЛЕНИЕ ШИНЫ ВВОДА/ВЫВОДА ЦПУ: DLNK(226)

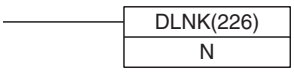
Назначение

Немедленное обновление слов ввода/вывода для модуля шины ЦПУ с указанным номером модуля. Обновляются указанные ниже данные.

- Слова, отведенные для модуля шины ЦПУ в областях модулей шины ЦПУ ПЛК (25 слов в области CIO и 100 слов в области DM)
- Особые виды данных, например, данные, которыми модули обмениваются через логические связи.

Эту команду возможно использовать только для модулей ЦПУ CP1H. Для модулей ЦПУ CP1L она недоступна. Если команда DLNK(226) будет использована для модуля ЦПУ CP1L, будет установлен флаг ошибки (ER).

Символ РКС



N: Номер модуля

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	DLNK(226)
	Однократное выполнение по положительному фронту	@DLNK(226)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограм-мы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

N: Номер модуля

Указывает номер модуля для модуля шины ЦПУ (0000...000F hex или 0...15 десятичн.).

Характеристики операндов

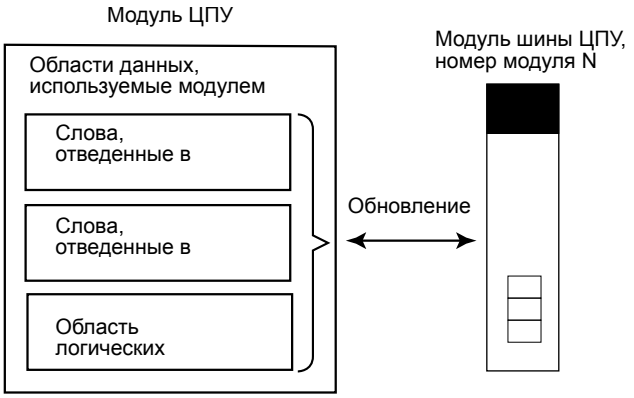
Область	N
Область CIO	CIO 0...CIO 6143
Рабочая область	W0...W511
Область битов хранения	H0...H511
Область вспомогательных битов	A448...A959
Область таймеров	T0000...T4095
Область счетчиков	C0000...C4095
Область DM	D0...D32767
Косвенные адреса DM в двоичном формате	@ D0...@ D32767
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767
Постоянные	#0000...#000F (двоичн.) или 0...15 (десятичн.)
Регистры данных	DR0...DR15
Регистры указателей	---
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15

Описание

Команда DLNK(226) инициирует немедленное обновление данных ввода/вывода для модуля шины ЦПУ с указанным номером модуля. Обновляется содержимое данных, указанных ниже. Подробную информацию об условиях выполнения, используемых для мгновенного обновления, см. ниже в пункте *Меры предосторожности*.

1. Слова, отведенные для модуля шины ЦПУ в областях модулей шины ЦПУ ПЛК (25 слов в области CIO и 100 слов в области DM).
2. Особые данные, например, данные, которыми модуль шины ЦПУ обменивается с другими модулями через логические связи (data link) или по сети децентрализованнного ввода/вывода DeviceNet (обновление происходит одновременно с данными в областях модуля шины ЦПУ).

Модуль шины ЦПУ	Обновление особых данных модуля
Модуль интерфейса Controller Link	Обновление логических связей
Модуль интерфейса DeviceNet	Обновление данных удаленного ввода/вывода



Следующая таблица демонстрирует различия между командами DLNK(226) и IORF(097).

Команда	Действие
DLNK(226)	• Обновление данных ввода/вывода области модуля шины ЦПУ в области CIO (25 слов) • Обновление данных ввода/вывода области модуля шины ЦПУ в области DM (100 слов) • Обновление особых данных модуля шины ЦПУ, например, данных при обмене через логические связи или данных удаленного ввода/вывода при обмене по сети DeviceNet
IORF(097)	• Обновление слов ввода/вывода, используемых модулями расширения входов/выходов CPM1A или модулями расширения CPM1A • Обновление слов ввода/вывода (10 слов CIO, выделенных для специального модуля ввода/вывода)

Команда DLNK(226) обновляет данные, используемые совместно модулем ЦПУ и модулем шины ЦПУ. Ниже описаны два существенных фактора, которые необходимо учитывать при использовании команды DLNK(226).

1,2,3...

1. Во время обмена данными через логические связи или по сети удаленного ввода/вывода DeviceNet, инициированного командой DLNK(226), обмен данными с другими модулями не производится. Обмен данными происходит, когда указанный модуль получает право на прием/передачу данных в очередном коммуникационном (сетевом) цикле и фактически принимает/передает данные. Таким образом, между моментом выполнения команды DLNK(226) и фактическим обменом данными может существовать задержка, определяемая длительностью коммуникационного цикла в данной сети.
2. Команда DLNK(226) не может обновить данные ввода/вывода для модуля шины ЦПУ, если тот в данный момент уже осуществляет обмен данными. Таким образом, при слишком частом выполнении команды DLNK(226) в программе обновление данных ввода/вывода производиться не будет. Рекомендуется выдерживать паузу между командами DLNK(226), превышающую по длительности коммуникационный цикл.

Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	Включен, если указанный номер модуля выходит за диапазон 0000...000F hex (0...15 десятичн.). Включен, если в ПЛК отсутствует модуль шины ЦПУ с указанным номером. Выключен во всех остальных случаях.
Флаг равенства	=	Выключен, если обновление данных ввода/вывода не удалось выполнить из-за того, что модуль шины ЦПУ был занят обновлением данных. Выключен при наличии ошибки настройки или ошибки в указанном модуле шины ЦПУ. Выключен, если командой DLNK(226) была выполнена задача обработки прерывания, что привело к конфликту с общей (циклической) процедурой обновления входов/выходов. Включен, если обновление данных ввода/вывода завершилось успешно.

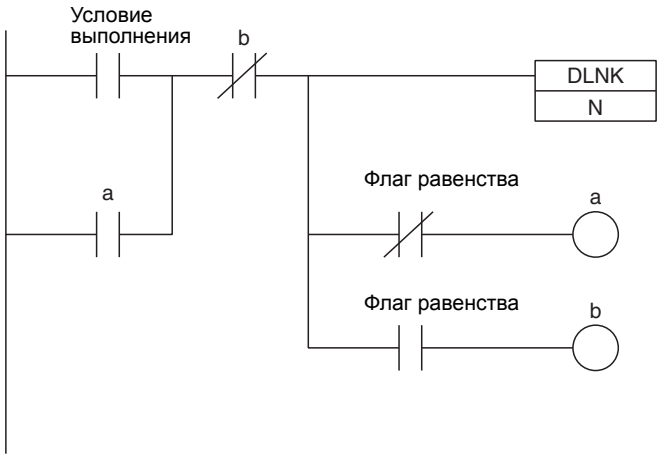
Меры предосторожности

Обновление данных ввода/вывода не будет выполнено, если в указанном модуле шины ЦПУ имеется ошибка модуля шины ЦПУ (A402.07) или ошибка настройки модуля шины ЦПУ (A402.03).

Если во время обновления данных ввода/вывода, запущенного командой DLNK(226), произойдет ошибка шины ввода/вывода, обновление данных ввода/вывода будет остановлено.

Команда DLNK(226) обновляет данные, используемые совместно модулем ЦПУ и модулем шины ЦПУ. Процедура связи, осуществляемая модулем шины ЦПУ (например, обмен данными через логические связи, осуществляемый модулем Controller Link), требует определенного времени.

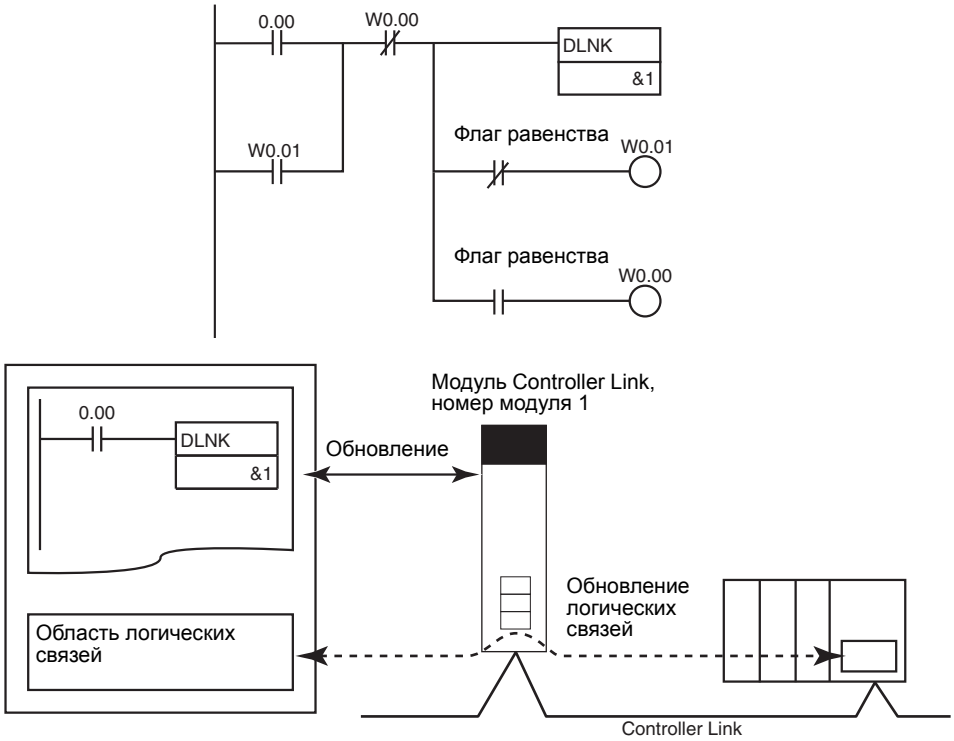
Если указанный модуль шины ЦПУ в данный момент уже производит обмен данными, команда DLNK(226) не выполняется, вместо этого сбрасывается флаг равенства. Рекомендуется использовать условие выполнения таким образом, чтобы выполнение команды DLNK(226) повторялось автоматически (см. пример программы ниже).



Пример

Если бит CIO 0.00 включен, команда DLNK(226) выполняет немедленное обновление данных ввода/вывода (в данном случае обновление логических связей в пределах ПЛК) для модуля шины ЦПУ с номером модуля 1 (в данном случае для модуля интерфейса Controller Link). Если обновление данных ввода/вывода не может быть выполнено из-за того,

что модуль Controller Link уже обновляет данные, сбрасывается флаг равенства, что вызывает включение бита W0.01 и повторное выполнение команды в следующем цикле. После успешного завершения обновления данных ввода/вывода флаг равенства будет установлен и команда в следующем цикле больше выполнена не будет.

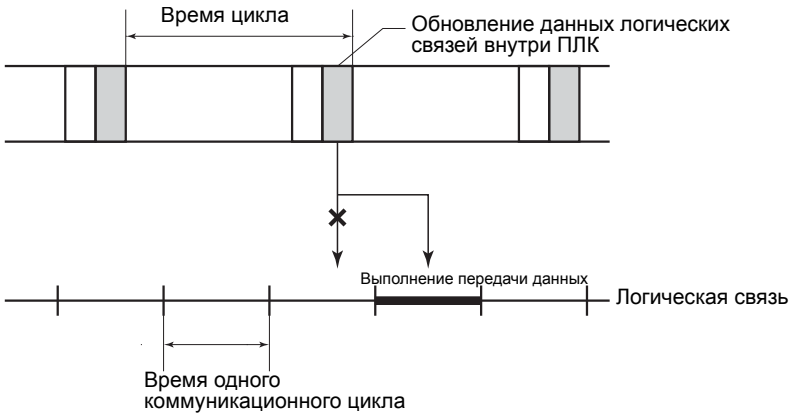


Фактическое время обновления содержимого области логических связей в данном примере определяется из следующих соображений.

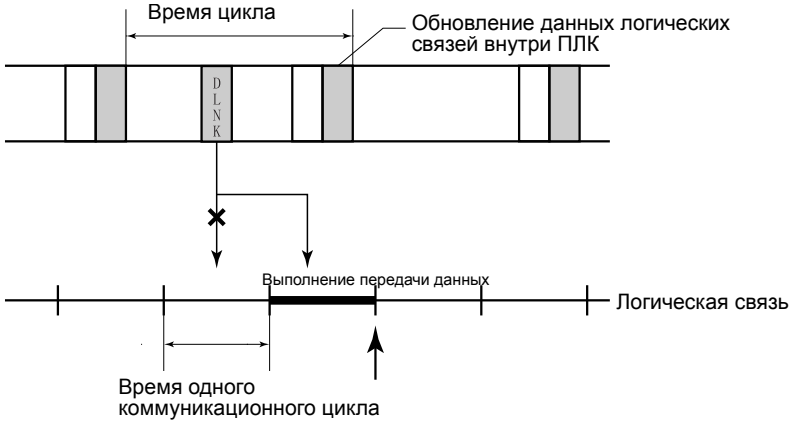
- Передача. Модуль передает данные по сети после получения очередного маркера (права на передачу) (передача данных запаздывает максимум на длительность одного коммуникационного цикла).
- Прием. Вводятся данные, полученные по сети в последний раз, когда модуль получил маркер (право на прием) (прием данных запаздывает максимум на длительность одного коммуникационного цикла).

Примеры процедуры передачи данных:

- Передача данных ввода/вывода, полученных при предыдущем обновлении



• Передача данных при выполнении команды DLNK(226)



3-23 Команды последовательного интерфейса

В данном разделе описаны команды, используемые для обмена данными по последовательному интерфейсу.

Команда	Мнемоническое обозначение	Код функции	Стр.
МАКРОС ПРОТОКОЛА	PMCR	260	884
ОТПРАВИТЬ	TXD	236	892
ПРИНЯТЬ	RXD	235	898
ОТПРАВИТЬ ЧЕРЕЗ МОДУЛЬ ПОСЛЕД. ИНТЕРФЕЙСА	TXDU	256	904
ПРИНЯТЬ ЧЕРЕЗ МОДУЛЬ ПОСЛЕД. ИНТЕРФЕЙСА	RXDU	255	912
ИЗМЕНИТЬ ПАРАМЕТРЫ ПОСЛЕД. ПОРТА	STUP	237	921

3-23-1 Последовательный интерфейс

Команды для связи по последовательному интерфейсу бывают двух типов. Команды TXD(236), RXD(235), TXDU(256) и RXDU(255) осуществляют обмен данными с внешним устройством без какого-либо определенного протокола передачи данных (свободно-программируемый режим). Команда PMCR(260) осуществляет связь с внешним устройством с использованием процедур (протоколов) приема и передачи данных, предварительно запрограммированных пользователем. Отличия между этими двумя типами команд поясняются в таблицах ниже.

Примечание.

(1) Команды TXD(236) и RXD(235) передают данные только через порт дополнительной платы последовательного интерфейса.

(2) Команды TXDU(256) и RXDU(255) передают данные только через модуль последовательного интерфейса серии CJ (версии 1.2 или выше).

(3) Команда PMCR(260) передает данные только через модуль последовательного интерфейса серии CJ.

Команды	Коммуникационные кадры	Функция
TXD(236), RXD(235), TXDU(256) и RXDU(255)	Могут использоваться следующие виды кадров: Без кодов начала и конца: Данные Код начала и код конца: ST | Данные | ED Только код начала: ST | Данные Код конца CR+LF: Данные | CR | LF Только код конца: Данные | ED Код начала и код конца CR+LF: ST | Данные | CR | LF	Передает или принимает данные только в одном направлении. Может быть задана задержка передачи.
PMCR(260)	Для выполнения требований внешнего устройства может быть сформирован кадр (сообщение) следующего типа. Заголовок | Адрес | Данные | Проверка ошибок | Код завершен. Возможно создание коммуникационных последовательностей. Память ввода/вывода Чтение/запись	Может быть сконфигурировано 16 шагов передачи и приема данных. Возможно изменение шагов и повторное выполнение в соответствии с поступающими ответами. Могут задаваться контрольные значения времени для слежения за работой системы связи. Возможно использование символьных имен ПЛК для чтения и записи данных. Возможно использование повторяющихся символов. Другие возможности.

Команды	Режим	Порты связи
TXD(236) и RXD(235)	Без протокола (свободно программируемый обмен)	Последовательный порт дополнительной платы последовательного интерфейса Модуль ЦПУ CP1H Доп. плата последовательного интерфейса RXD TXD
TXDU(256) и RXDU(255)	Без протокола (свободно программируемый обмен)	Последовательный порт модуля последовательного интерфейса серии CJ (версии 1.2 и выше) Модуль ЦПУ CP1H TXDU/RXDU Модуль последовательного интерфейса серии CJ RXD TXD
PMCR(260)	Макросы протоколов	Последовательный порт модуля последовательного интерфейса серии CJ Модуль ЦПУ CP1H PMCR Модуль последовательного интерфейса серии CJ Прием Передача

3-23-2 МАКРОС ПРОТОКОЛА: PMCR(260)

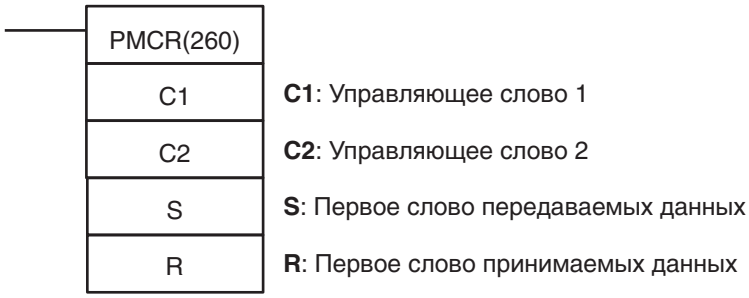
Назначение

Вызов и выполнение процедуры обмена данными, зарегистрированной в модуле последовательного интерфейса серии CJ.

Эту команду возможно использовать только для модулей ЦПУ CP1H. Для модулей ЦПУ CP1L она недоступна. Если команда PMCR(260) будет использована для модуля ЦПУ CP1L, будет установлен флаг ошибки (ER).

Примечание. Для использования модулей последовательного интерфейса серии CJ требуется адаптер для модулей CJ.

Символ РКС



Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	PMCR(260)
	Однократное выполнение по положительному фронту	@PMCR(260)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

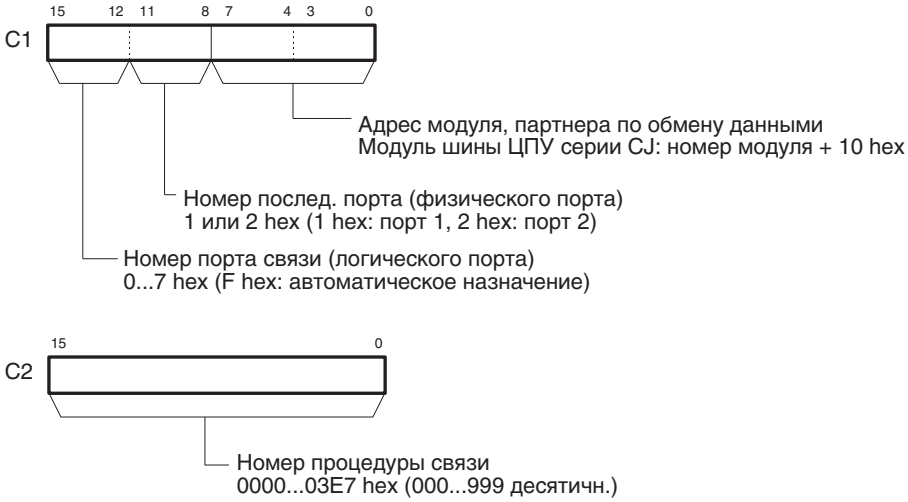
Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

C1: управляющее слово 1; C2: управляющее слово 2

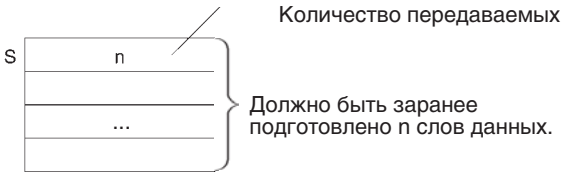
Содержание двух управляющих слов представлено ниже.



S: первое слово области передачи

Этот операнд указывает адрес первого слова области, используемой для передачи данных. Первое слово (слово по адресу S) содержит значение количества передаваемых слов +1 (т. е., включая само слово S), а передаваемые данные начинаются со слова по адресу S+1. Может быть передано от 0000 до 00FA hex (от 0 до 250 десятичн.) слов.

Если в используемой процедуре связи не указан операнд (т. е. при непосредственном указании адреса слова или использовании слова связи), в S следует ввести константу #0000. Если вместо константы вводится адрес слова или регистр, это слово или регистр всегда должны содержать значение 0000. При любом другом значении константы или содержимого слова/регистра произойдет ошибка, будет установлен флаг ошибки и команда PMCR(260) не будет выполнена.



R: первое слово области приема

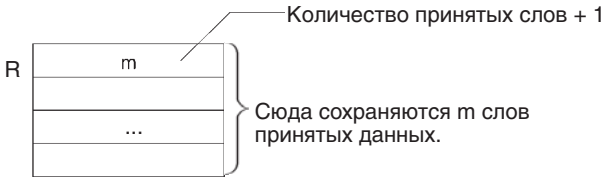
В группу слов, начиная со слова R+1, автоматически записываются принятые данные. В первое слово этой группы, адрес которого указан в R, записывается значение количества принятых слов + 1 (т. е., включая само слово R): от 0000 до 00FA hex (от 0 до 250 десятичн.).

Подготовка к выполнению команды PMCR

Задайте m слов данных (начиная с D) в качестве начальных значений для буфера приема (подстановочные данные на случай неуспешного приема). Значение m может находиться в диапазоне от 0002 до 00FA (hex) (2...255). Если для m будет указано значение 0000 (hex) или 0001 (hex), исходное содержимое буфера приема будет обнулено.

Операнд R всегда должен содержать адрес слова, даже если принимать данные не предполагается. Если в нем будет указана константа, произойдет ошибка, будет установлен флаг ошибки и команда PMCR(260) не будет выполнена. При отсутствии принимаемых данных операнд R не будет задействован и может быть использован для других целей.

Если в используемой процедуре связи не указан операнд (т. е. при непосредственном указании адреса слова или использовании слова связи), в R следует ввести константу #0000. Если вместо константы вводится адрес слова или регистр, это слово или регистр всегда должны содержать значение «0000».



Характеристики операндов

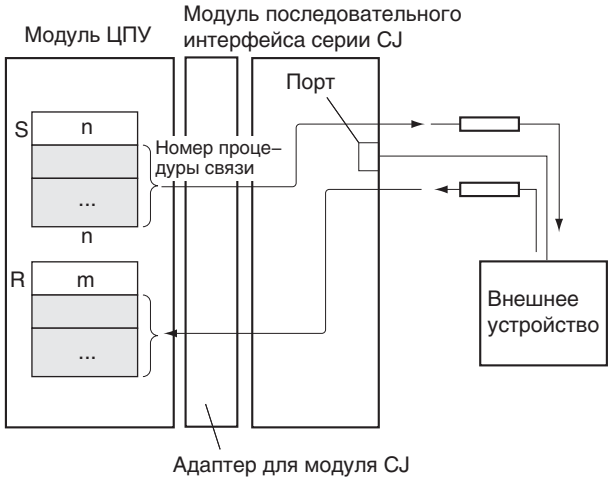
Область	C1	C2	S	R
Область CIO	CIO 0...CIO 6143			
Рабочая область	W0...W511			
Область битов хранения	H0...H511			
Область вспомогательных битов	A0...A447 A448...A959			A448...A959
Область таймеров	T0000...T4095			
Область счетчиков	C0000...C4095			
Область DM	D0...D32767			
Косвенные адреса DM в двоичном формате	@ D0...@ D32767			
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767			
Постоянные	Только указанные значения	0000...03E7 hex (0...999)	#0000 (двоичн.)	
Регистры данных	DR0...DR15		---	
Регистры указателей	---			
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15			

Описание

Команда PMCR(260) запускает выполнение коммуникационной последовательности (процедуры связи), номер которой указан в C2, используя логический порт, указанный битами 12...15 слова C1, и физический порт, указанный битами 8...11 слова C1, для модуля, адрес которого указан битами 0...7 слова C1.

Если в передаваемом сообщении использован символьный операнд, будет передано содержимое слов области передачи (т. е. S слов, начиная со слова S+1). Если принимаемое сообщение будет содержать символьный операнд, принятые данные будут размещены в памяти, начиная со слова R+1, а в слово R, при условии успешного завершения передачи, будет автоматически записано количество принятых слов.

Если передача данных завершится ошибкой, в область приема (начиная с R+1) будут вновь записаны данные, которые были помещены в буфер приема перед выполнением команды PMCR(260).



Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	Включен, если при выполнении PMCR(260) выключен флаг доступности порта связи для указанного логического порта. Включен, если C1 выходит за указанные диапазоны. (При выходе значения C2 за установленные диапазоны флаг ошибки не включается. В слово кода завершения порта связи (A203...A210) вспомогательной области будет записан код завершения кадра связи.) Включен, если количество слов в S или R превышает 249 (если указываются слова). Выключен во всех остальных случаях.

Меры предосторожности

Данные из области передачи, указанной операндом S, передаются, если в передаваемом сообщении присутствует символ чтения: R( ).

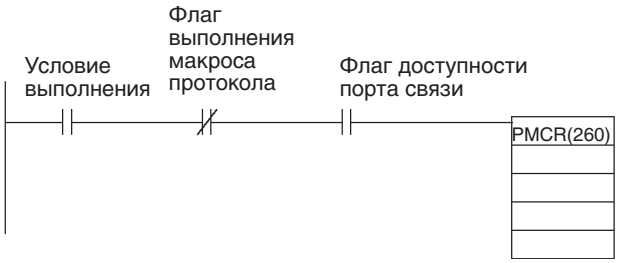
Данные принимаются в область приема, указанную операндом R, если в принятом сообщении присутствует символ записи: W( ).

Команда PMCR(260) может быть выполнена для последовательного порта связи модуля последовательного интерфейса. К модулю ЦПУ могут быть подключены максимум 2 модуля последовательного интерфейса (подключаются в качестве модулей расширения серии CJ). В битах 0...7 слова C1 должен быть указан адрес используемого модуля (партнера по обмену данными), а в битах 8...11 указывается номер последовательного порта. Адреса модулей указываются в соответствии со следующей таблицей.

Порядок использования символов R( ) и W( ) в коммуникационных сообщениях описан в *Руководстве по работе с программой CX-Protocol (W344)*.

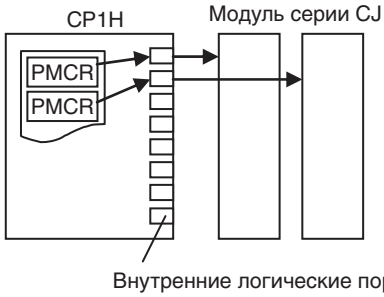
Модуль/Плата	Адрес модуля
Модуль последовательного интерфейса	Номер модуля + 10 hex

В начале выполнения команды PMCR(260) включается соответствующий флаг выполнения макропрограммы протокола. После того как процедура связи завершена и данные записаны в указанную область приема, этот флаг сбрасывается. Во избежание одновременного выполнения нескольких процедур связи для одного и того же физического порта в условия выполнения команды PMCR(260) обязательно должен входить соответствующий флаг выполнения макропрограммы протокола (в качестве входного нормально замкнутого условия). Пример программы приведен ниже.



Команды SEND(090), RECV(098) и CMND(490) также используют логические порты 0...7 для выполнения процедур связи через модуль последовательного интерфейса (внутри ПЛК используются команды FINS). Если некоторый логический порт уже используется командой SEND(090), RECV(098), CMND(490) или PMCR(260), для другой команды

PMCR(260) он в этот момент не доступен. Во избежание одновременного выполнения нескольких процедур связи для одного и того же логического порта в условия выполнения команды PMCR(260) должен входить соответствующий флаг доступности порта связи (A202.00...A202.07). Пример программы приведен выше.



В перечисленных ниже случаях включается флаг ошибки.

- Во время выполнения PMCR(260) выключен соответствующий флаг доступности порта связи для используемого логического порта (0...7).
- Значение C1 выходит за допустимый диапазон.

Инициализация области приема

Перед выполнением команды PMCR(260) в область приема следует поместить резервные значения на случай ошибки приема. После выполнения PMCR(260) в область приема автоматически записываются данные из буфера приема. Практическое значение инициализации области приема можно пояснить на следующем примере. Во время чтения текущего значения из регулятора температуры с использованием макроса протокола может произойти ошибка передачи данных. Для того чтобы в такой ситуации не было возвращено нулевое текущее значение, в область приема заранее помещают некоторое ненулевое (подстановочное) значение.

Сопутствующие флаги и слова

В следующей таблице перечислены флаги и слова, которые могут использоваться по мере необходимости при выполнении команды PMCR(260).

Вспомогательная область

Название	Адрес	Содержание
Флаг доступности порта связи	A202.00... A202.07	Включен, если данный порт доступен для связи по сети (в т. ч. для команды PMCR(260)). Биты 00...07 соответствуют логическим портам 0...7. Когда начинается обмен данными по сети, флаг доступности порта связи сбрасывается. По завершении связи (неважно, с ошибкой или без ошибки) он вновь устанавливается.

Предварительная версия
Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

Название	Адрес	Содержание
Флаг ошибки порта связи	A219.00... A219.07	Включается, если во время обмена данными по сети происходит ошибка. Биты 00...07 соответствуют логическим портам 0...7. Состояние данного флага сохраняется вплоть до следующего сеанса сетевого обмена данными. Даже если во время предыдущей операции (выполнения команды связи) произошла ошибка, в начале следующего сеанса связи флаг сбрасывается.
Коды завершения для портов связи	A203...A210	В данное слово записывается код завершения, отражающий результаты выполнения операции сетевого обмена данными. Слова A203...A210 соответствуют логическим портам связи 0...7. Непосредственно во время выполнения команды связи слово кода завершения содержит значение «00». По завершении выполнения в него записывается новый код ответа. В начале выполнения программы содержимое этих слов очищается.

Ответные коды

Код	Содержание
1106 (hex)	Отсутствует программа с соответствующим номером Процедура передачи/приема данных с указанным номером не зарегистрирована. Измените номер процедуры передачи/приема данных или создайте программу с таким номером с помощью CX-Programmer.
2201 (hex)	Выполнение невозможно из-за выполнения другого протокола В данный момент уже выполняется макропрограмма протокола, поэтому выполнение другой макропрограммы невозможно. Используйте в программе флаг выполнения макропрограммы протокола в качестве НЗ условия.
2202 (hex)	Выполнение невозможно из-за остановки выполнения В данный момент происходит переключение протоколов, поэтому выполнение макропрограммы протокола невозможно. Используйте в программе флаг изменения параметров последовательного порта в качестве НЗ условия.
2401 (hex)	Отсутствует регистрационная таблица В данных макропрограммы протокола или передаваемых данных произошла ошибка. Передайте данные макроса протокола с помощью CX-Programmer.
Прочие коды	Информацию о других ответных кодах см. в *Справочном руководстве по командам связи для серии CJ (W342)*.

Область модуля шины ЦПУ

n = 1500 + 25 x номер модуля

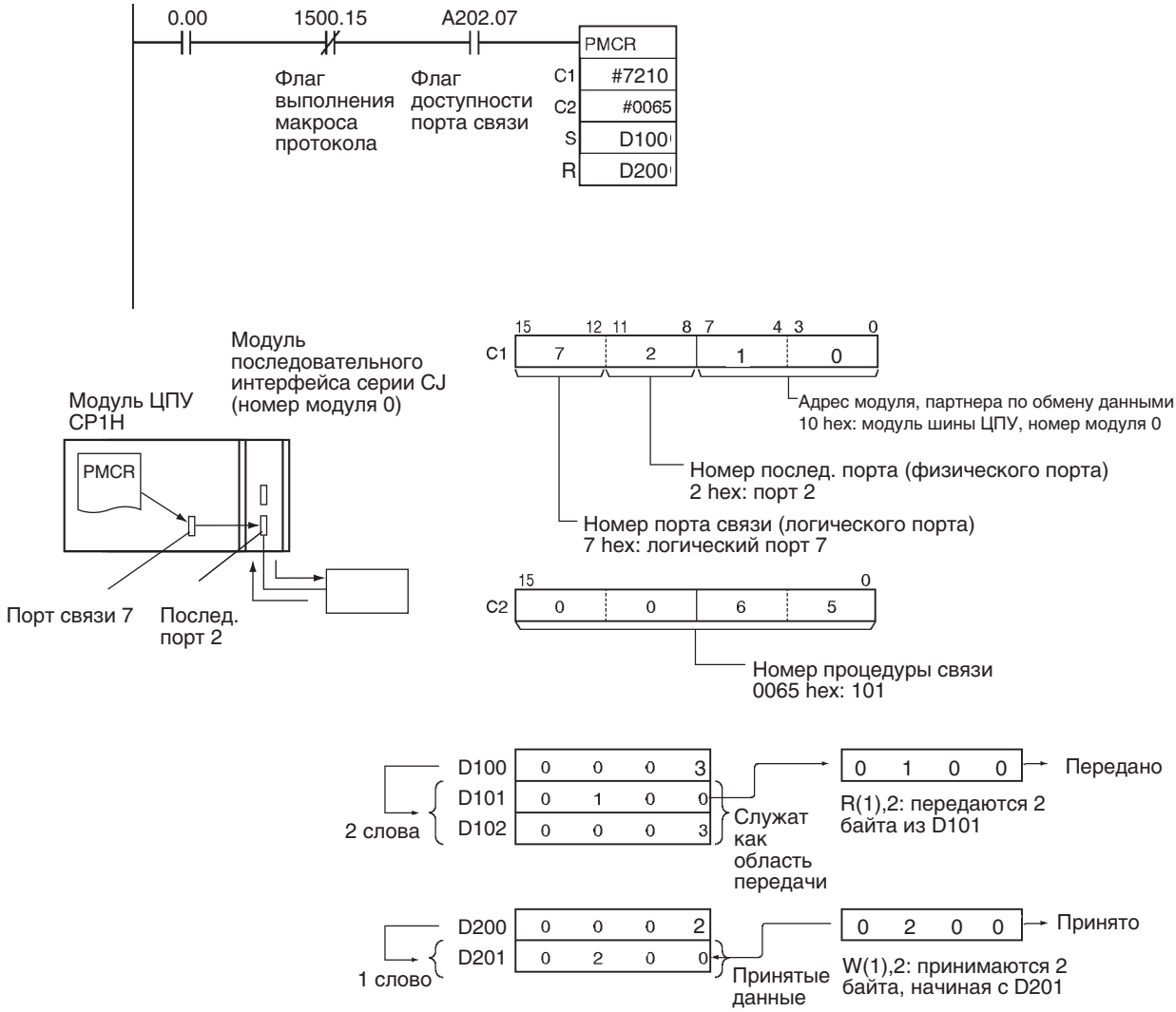
Название	Адрес	Содержание
Флаг выполнения макроса протокола порта 1	Бит 15 слова CIO n+9	Включен во время выполнения PMCR(260). При неуспешном выполнении флаг останется выключенным. Данный флаг выключается по завершении процедуры связи (с ошибкой или без ошибки).
Флаг выполнения макроса протокола порта 2	Бит 15 слова CIO n+19	

Примеры

При включении бита CIO 0.00 в приведенном ниже примере программы выполняется процедура связи с номером 101 (0065 hex) с использованием порта 2 модуля последовательного интерфейса с номером 0, при условии, что включен флаг доступности порта связи для порта 7 (A202.07) и выключен флаг выполнения макропрограммы протокола (CIO 1500.15).

При наличии в передаваемом сообщении символьного операнда будут переданы 2 слова из области передачи, начинающейся со слова D101 (так как D100 = #0003).

При наличии символьного операнда в принятом сообщениии в область приема, начиная со слова D201, будут записаны 2 слова данных, а в слово D200 будет записано количество принятых слов +1.



Примеч. Как видно из рисунка выше, вид операции (передача или прием данных) фактически определяется символом в сообщении: символ чтения R( ) — передача, символ записи W( ) — прием.

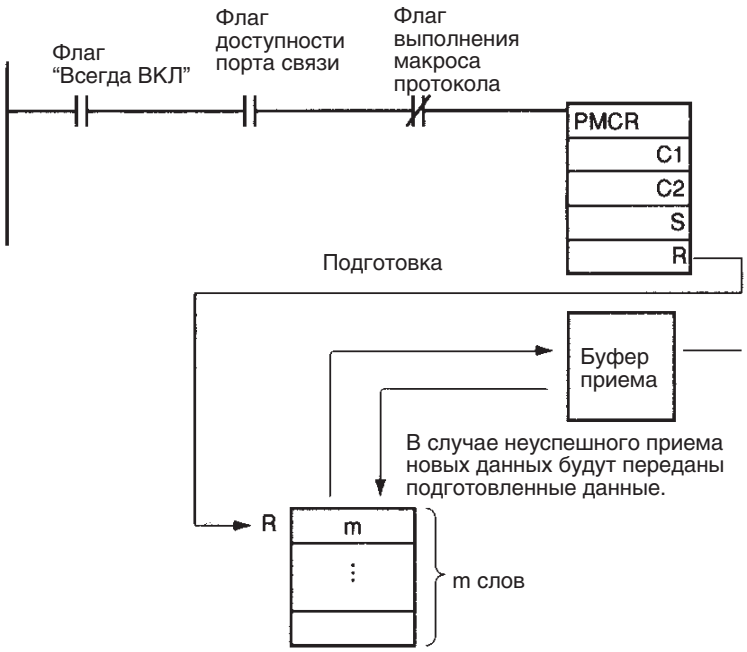
Сохранение содержимого области приема

В начале выполения команды PMCR(260), то есть непосредственно перед выполнением процедуры связи, буфер приема обнуляется. Если некоторая программа (например, показанная ниже) периодически считывает текущие значения технологических параметров или другие данные и в некоторый момент времени из-за ошибки приема или по другой причине эти данные прочесть не удается, область приема (т. е. возвращаемые значения) будет обнулена и команда будет возвращать нулевые значения вплоть до следующего успешного выполнения операции чтения.

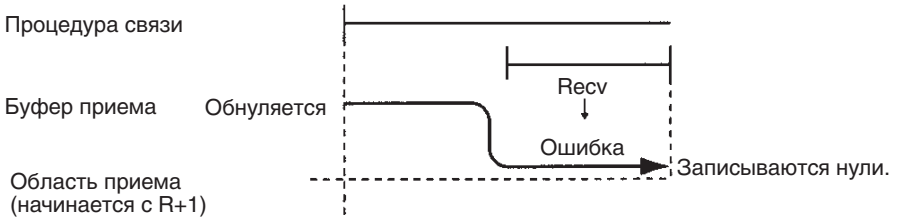
На случай возникновения ошибок приема предусмотрена функция, которая позволяет сохранять предыдущие данные в области приема. Если используется эта функция, после обнуления содержимого буфера приема (но до выполнения процедуры связи) в него переносится содержимое первых m слов области приема. Таким образом, даже если новые данные не будут получены из-за ошибки связи, в область приема будут записаны последние успешно принятые данные и она не будет содержать нулевые значения.

В m следует указать количество слов области приема, содержимое которых должно сохраняться. Если будет указано значение «0» или «1», функция сохранения значений будет отключена и содержимое области приема будет обнуляться.

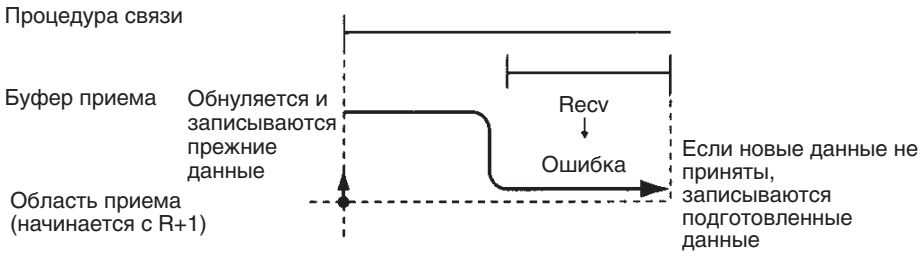
В приведенном ниже примере программы производится постоянное или периодическое выполнение команды PMCR(260) с целью чтения данных за одну операцию приема данных.



Данные в области приема не сохраняются



Данные в области приема сохраняются

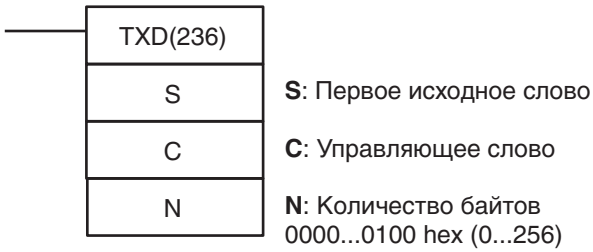


3-23-3 ОТПРАВИТЬ: TXD(236)

Назначение

Выдает указанное количество байтов данных через последовательный порт дополнительной платы последовательного интерфейса, установленной в модуль ЦПУ.

Символ РКС



S: Первое исходное слово

C: Управляющее слово

N: Количество байтов
0000...0100 hex (0...256)

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	TXD(236)
	Однократное выполнение по положительному фронту	@TXD(236)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

Содержание управляющего слова C представлено ниже.



Примечание.

Модули ЦПУ CP1H и CP1L M (30 или 40 вх./вых.):
1 hex: порт 1;
2 или 0 hex: порт 2.
Модули ЦПУ CP1L L (14 или 20 вх./вых.):
1 или 0 hex: порт 1

Характеристики операндов

Область	S	C	N
Область CIO	CIO 0...CIO 6143		
Рабочая область	W0...W511		
Область битов хранения	H0...H511		
Область вспомогательных битов	A0...A447 A448...A959		
Область таймеров	T0000...T4095		
Область счетчиков	C0000...C4095		
Область DM	D0...D32767		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767		
Постоянные	---	Только указанные значения	#0000...#0100 (двоичн.) или &0...&256 (десятичн.)
Регистры данных	---	DR0...DR15	
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда TXD(236) считывает содержимое N байтов данных из слов S...S+(N÷2)-1 и выдает это содержимое, никак его не обрабатывая, в беспротокольном режиме (режиме RS-232C) через последовательный порт дополнительной платы последовательного интерфейса, установленной в модуль ЦПУ (используемый порт указывается битами 8...11 слова C).

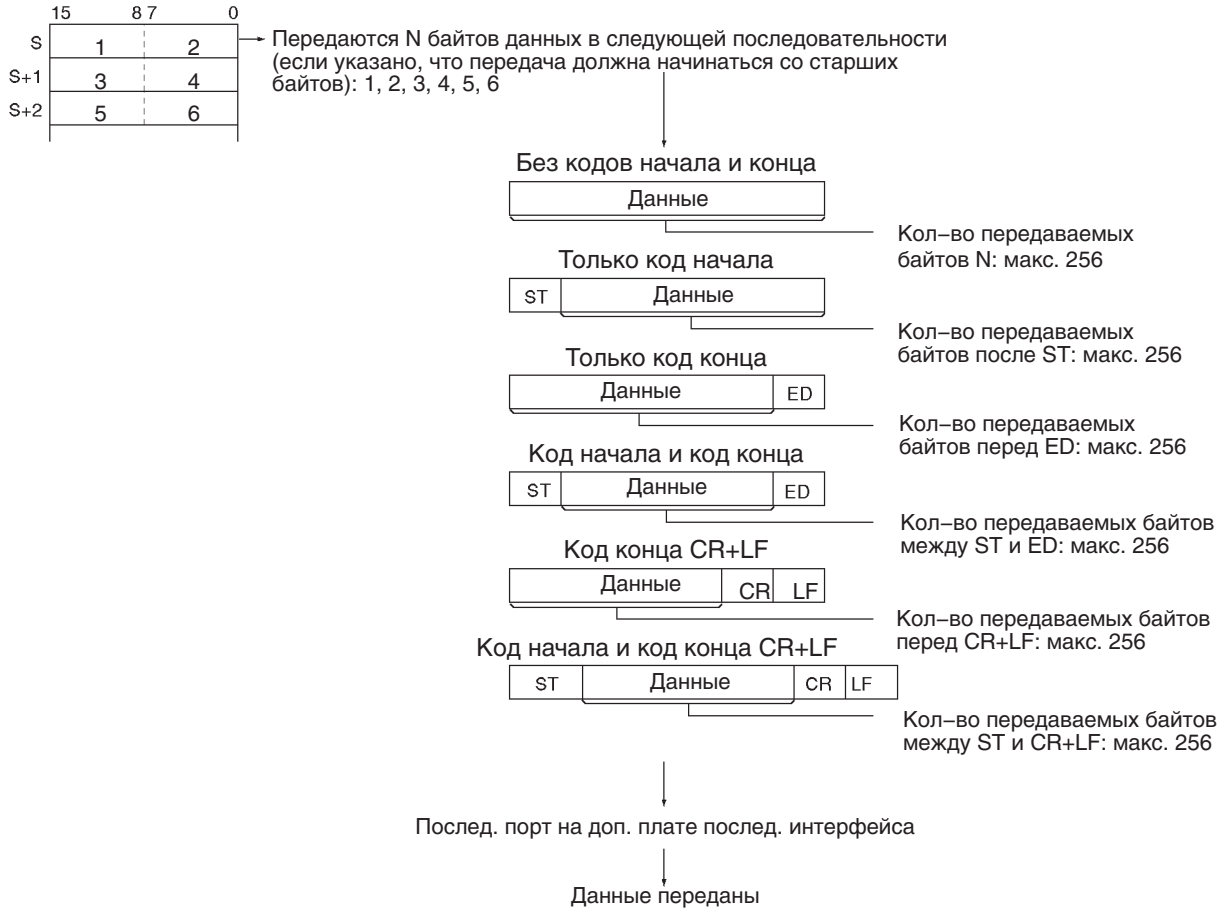
Перед выдачей данных к ним добавляются коды начала и конца кадра, указанные для беспротокольного режима. Коды начала и конца кадра задаются в настройках ПЛК.

Передача данных возможна, только если для порта включен флаг готовности к передаче. Флаги готовности к передаче: A392.13 — для последовательного порта 1, A392.05 — для последовательного порта 2.

Может быть передано до 259 байтов, в том числе полезные данные (N = макс. 256 байт), код начала и код конца кадра.

Примечание. Последовательный порт 1 — это порт доп. платы последовательного интерфейса, установленной в гнездо для доп. платы 1. Последовательный порт 2 — это порт доп. платы последовательного интерфейса, установленной в гнездо для доп. платы 2.

Следующий рисунок демонстрирует очередность передачи байтов данных, а также содержание передаваемого кадра для разных конфигураций кодов начала и конца кадра.



Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	Включен, если в настройках ПЛК не выбран беспротокольный режим (режим RS-232C). Включен, если значение C выходит за указанный диапазон. Включен, если значение N выходит за диапазон 0000...0100 hex. Включен, если была инициирована передача при выключенном флаге готовности к передаче. Выключен во всех остальных случаях.

Меры предосторожности

Команда TXD(236) может использоваться только для тех последовательных портов дополнительной платы последовательного интерфейса, которые переведены в беспротокольный режим (по-другому, режим свободно программируемого обмена или режим RS-232C).

В настройках ПЛК может быть задан один из следующих форматов кадра сообщения передачи данных.

- Код начала: нет или 00...FF hex.
- Код конца: нет, CR+LF или 00...FF hex.

Данные будут переданы с любыми кодами начала и/или конца кадра, которые указаны в настройках ПЛК. Если коды начала и кадра указаны, они добавляются к передаваемым данным (N). В данном случае в N может быть указано максимум 256 байтов.

Данные передаются в том порядке, который указан в C.

Если для N выбрано значение «0», ничего передано не будет.

Если в C выбрано управление сигналом RS, бит 15 слова S используется в качестве сигнала RS.

Если в C выбрано управление сигналом ER, бит 15 слова S используется в качестве сигнала ER.

Если в C выбрано управление сигналами RS и ER, бит 15 слова S используется в качестве сигнала RS, а бит 14 слова S используется в качестве сигнала ER.

Если вторая тетрада слова C (управление сигналом RS и ER) содержит значение 1, 2 или 3 hex, команда TXD(236) выполняется независимо от состояния флага готовности к передаче.

Любое из следующих нарушений вызовет ошибку и установку флага ошибки.

- В настройках ПЛК для порта не выбран беспротокольный режим.
- Значение C выходит за допустимый диапазон.
- Значение N выходит за диапазон 0000...0100 hex.
- Предпринята попытка передачи при выключенном флаге готовности к передаче.

Сопутствующие параметры в настройках ПЛК

После перевода последовательного порта 1 или последовательного порта 2 (для которого выполняется команда TXD(236)) в беспротокольный режим связи (режим RS-232C) для этого порта также необходимо настроить следующие параметры в области настроек ПЛК.

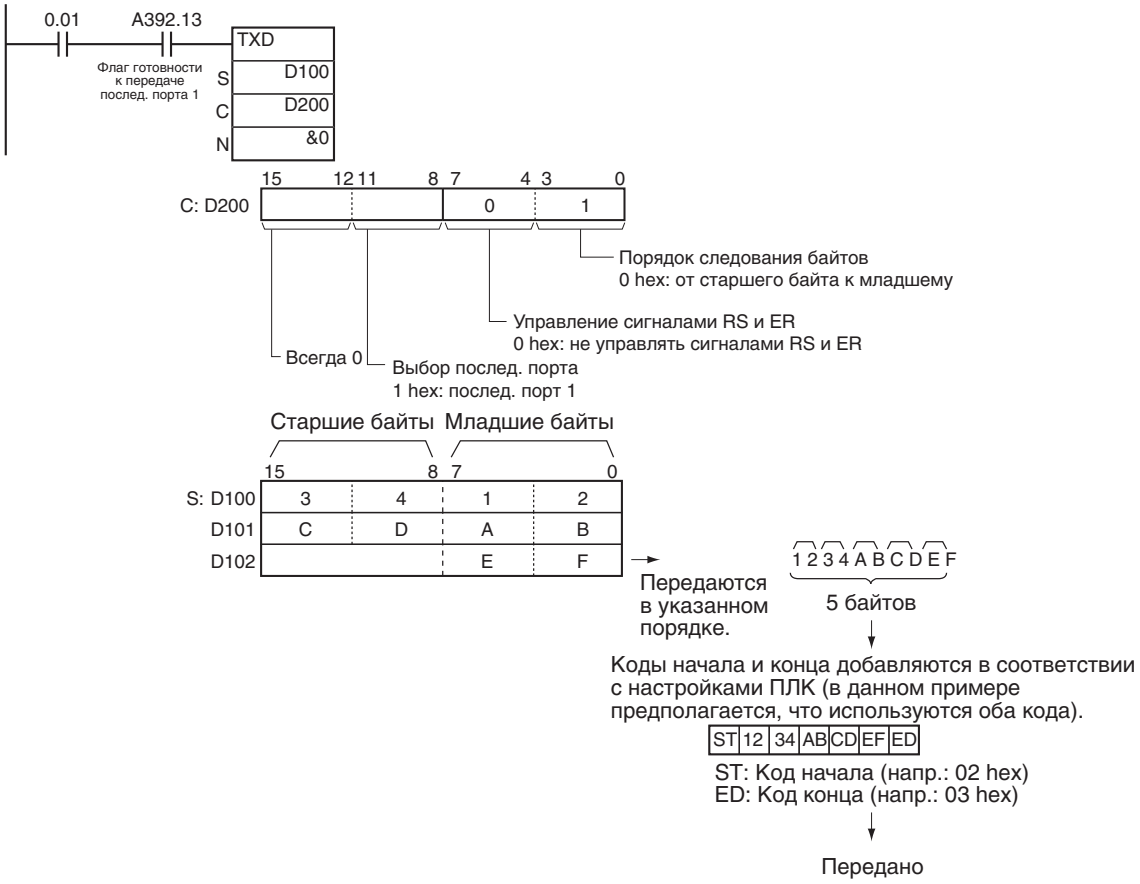
Значение		Описание	
Параметры связи		Стандартные	Стандартные параметры имеют следующие значения: 9600 бод, 1 старт-бит, 7 битов данных, проверка четности, 2 стоп-бита.
		Настройки пользователя	Скорость: 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600 или 115200 бит/с Формат: 7 или 8 битов данных; 1 или 2 стоп-бита; чет, нечет или не проверять
Код начала		Отключен	---
		Задан	00...FF hex
Код конца	Принимаемые байты	Нет	---
		Да	00 hex: 256 байт 01...FF hex: 1...255 байт
	Код конца	Нет	---
		CR/LF	---
		Задан	00...FF hex
Задержка		0000...210F hex: 0...99 990 мс (с шагом 10 мс)	

Вспомогательная область

Название	Адрес	Содержание
Флаг готовности к передаче послед. порта 1	A392.13	Включен, если есть возможность передачи данных в беспротокольном режиме.
Флаг готовности к передаче послед. порта 2	A392.05	

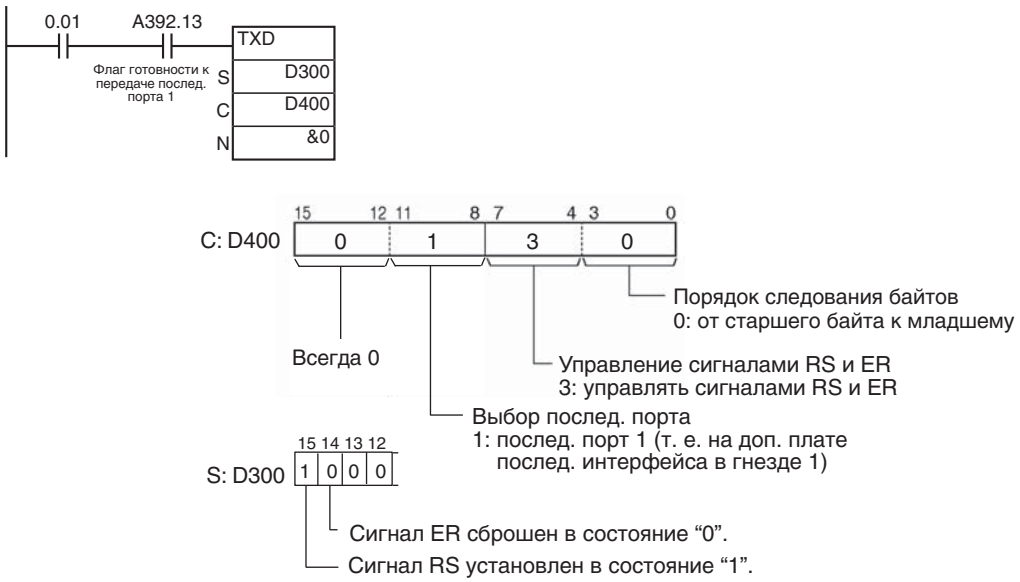
Пример: передача данных

Если в приведенной ниже программе включен бит CIO 0.01 и включен флаг готовности к передаче последовательного порта 1 (A392.13), на дополнительную плату последовательного интерфейса, которая установлена в гнездо для доп. платы 1, передаются пять байтов данных, начиная с младшего байта слова D100.



Пример: управление сигналами

Если в приведенной ниже программе включен бит CIO 0.01 и включен флаг готовности к передаче последовательного порта 1 (A392.13), сигнал RS принимает состояние бита 15 слова D300, а сигнал ER принимает состояние бита 14 слова D300.

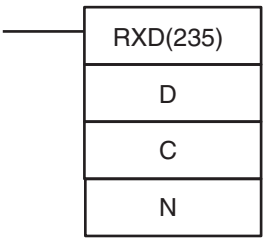


3-23-4 ПРИНЯТЬ: RXD(235)

Назначение

Чтение указанного количества байтов данных через последовательный порт дополнительной платы последовательного интерфейса, установленный в модуль ЦПУ.

Символ РКС



D: Первое слово назначения

C: Управляющее слово

N: Количество сохраняемых байтов
0000...0100 hex (0...256 десятичн.)

Варианты выполнения

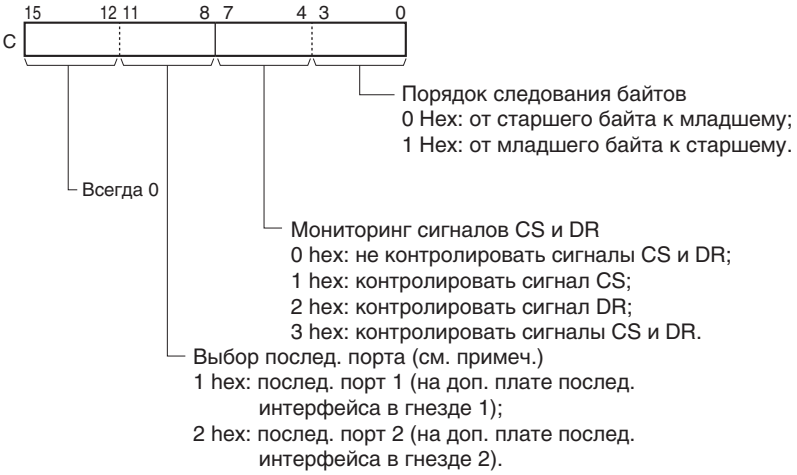
Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	RXD(235)
	Однократное выполнение по положительному фронту	@RXD(235)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

Содержание управляющего слова C представлено ниже.



Примечание. Модули ЦПУ CP1H и CP1L M (30 или 40 вх./вых.):
1 hex: порт 1;
2 или 0 hex: порт 2.
Модули ЦПУ CP1L L (14 или 20 вх./вых.):
1 или 0 hex: порт 1.

Характеристики операндов

Область	D	C	N
Область CIO	CIO 0...CIO 6143		
Рабочая область	W0...W511		

Область	D	C	N
Область битов хранения	H0...H511		
Область вспомогательных битов	A448...A959	A0...A447 A448...A959	
Область таймеров	T0000...T4095		
Область счетчиков	C0000...C4095		
Область DM	D0...D32767		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767		
Постоянные	---	Только указанные значения	#0000...#0100 (двоичн.) или &0...&256 (десятичн.)
Регистры данных	---	DR0...DR15	
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда RXD(235) считывает данные, которые были приняты в беспротокольном режиме через последовательный порт дополнительной платы последовательного интерфейса, и записывает N байтов данных в слова D...D+(N÷2)-1. Если принятых байтов данных окажется меньше, чем N, будет записано столько байтов данных, сколько их фактически было принято.

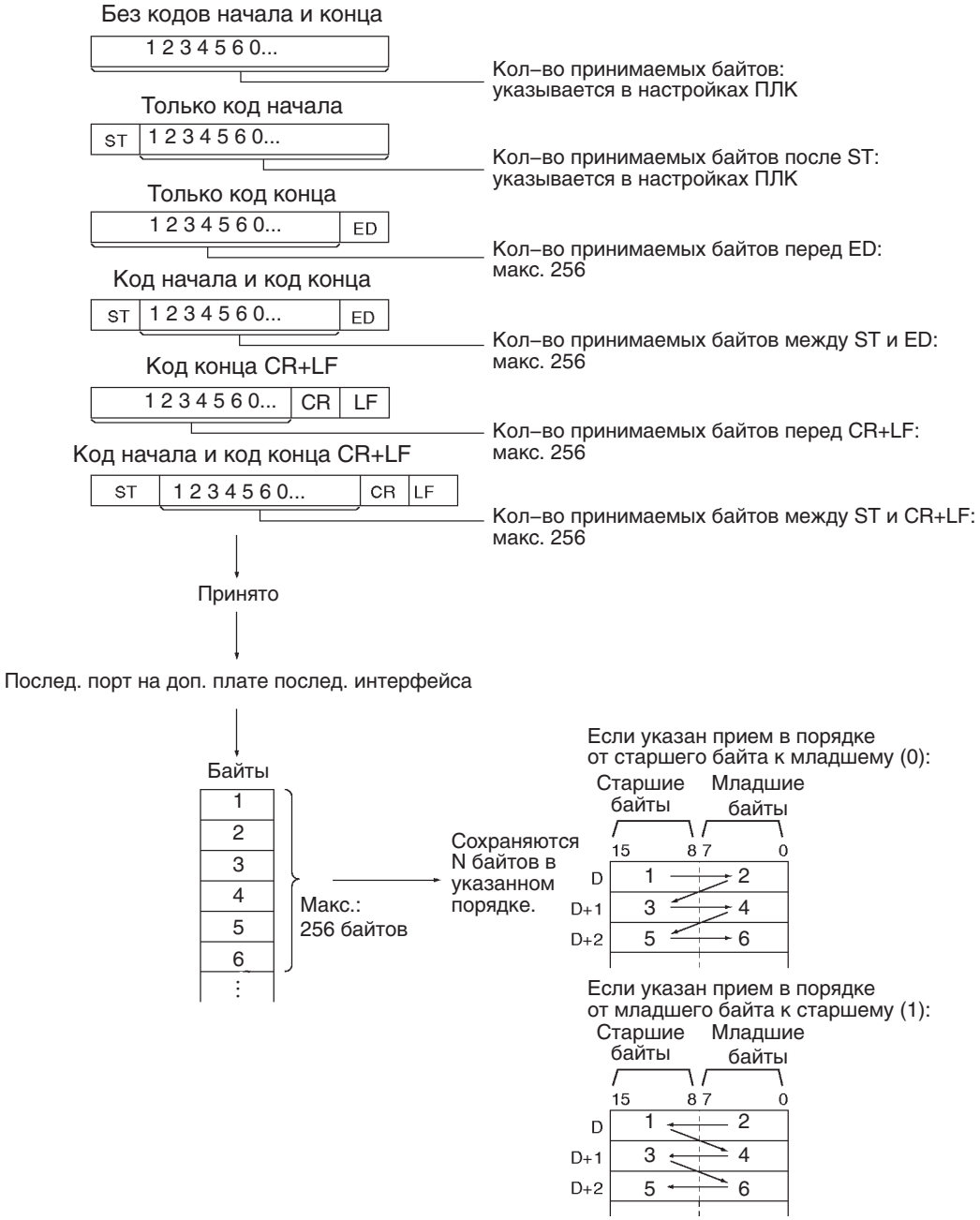
Прием данных возможен, только если для порта включен флаг готовности к приему. Флаги готовности к приему: A392.14 — для последовательного порта 1, A396.06 — для последовательного порта 2. Команда RXD(235) должна выполняться, только если в данный момент установлен соответствующий флаг готовности к приему.

Может быть принято до 259 байтов, в том числе полезные данные (N = макс. 256 байт), код начала и код конца кадра.

Следующий рисунок демонстрирует очередность приема байтов данных, а также содержание принимаемого кадра для разных конфигураций кодов начала и конца кадра.

Примечание. Последовательный порт 1 — это порт доп. платы последовательного интерфейса, установленной в гнездо для доп. платы 1.

Последовательный порт 2 — это порт доп. платы последовательного интерфейса, установленной в гнездо для доп. платы 2.



Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	Включен, если в настройках ПЛК не выбран беспротокольный режим. Включен, если значение C выходит за указанный диапазон. Включен, если значение N выходит за диапазон 0000...0100 hex. Выключен во всех остальных случаях.

Меры предосторожности

Команда RXD(235) может использоваться только для последовательных портов дополнительных плат последовательного интерфейса, установленных в модуль ЦПУ. Кроме того, порт должен быть переведен в так называемый беспротокольный режим (режим, в котором не используется какой-либо конкретный протокол передачи/приема данных).

В настройках ПЛК может быть задан один из следующих форматов кадра сообщения приема данных.

- Код начала: нет или 00...FF hex.
- Код конца: нет, CR+LF или 00...FF hex. Если код конца кадра не используется (т. е. выбрано «нет»), для количества принимаемых байтов может быть указано значение от 00 до FF hex (от 1 до 255 десятичн.; 00 = 256 байтов).

Когда количество принятых байтов достигает указанного в настройках ПЛК значения, устанавливается флаг завершения приема. Когда устанавливается флаг завершения приема, количество байтов, указываемое счетчиком приема, имеет такое же значение, что и количество принимаемых байтов, которое указано в настройках ПЛК или в выделенной области настроек DM. Если принятых байтов окажется больше, чем указано, установится флаг переполнения при приеме данных.

Если в настройках ПЛК указан код конца кадра, флаг завершения приема будет установлен, когда будет получен код конца кадра или будет принято 256 байт данных. Если после включения флага завершения приема будут приняты еще данные, установится флаг переполнения при приеме данных.

По завершении выполнения команды RXD(235) в память, начиная с адреса D, будут записаны полученные данные, флаг завершения приема будет сброшен, счетчик приема будет обнулен. Флаг переполнения при приеме данных, если он был установлен, также будет сброшен.

Данные сохраняются в память в том порядке, который указан в C.

Если для N было указано значение «0», то будет сброшен флаг завершения приема, будет обнулен счетчик приема, но в память ничего записано не будет.

Если в C выбран мониторинг сигнала CS, в бит 15 слова D записывается состояние сигнала CS.

Если в C выбран мониторинг сигнала DR, в бит 15 слова D записывается состояние сигнала DR.

Если в C выбран мониторинг сигналов CS и DR, в бит 15 слова D записывается состояние сигнала CS, а в бит 14 слова D записывается состояние сигнала DR.

Если указан мониторинг сигнала CS или DR, принятые данные в память не сохраняются.

Если вторая тетрада слова C (управление сигналами RS и ER) содержит значение 1, 2 или 3 hex, команда RXD(235) выполняется независимо от состояния флага завершения приема.

Любое из следующих нарушений вызовет ошибку и установку флага ошибки.

- Для данного порта в настройках ПЛК не выбран беспротокольный режим (режим RS-232C).
- Значение C выходит за допустимый диапазон.
- Значение N выходит за диапазон 0000...0100 hex.

Чтение данных из буфера приема с помощью команды RXD(235) следует производить по возможности сразу после установки флага завершения приема. Если данные считаны не будут, а прием продолжится, буфер приема окажется переполнен и прием данных прекратится. В этом случае порт должен быть перезапущен для возобновления работы.

При выполнении команды RXD(235) буфер приема не очищается. Следовательно, команда RXD(235) может использоваться для чтения одних и тех же данных любое число раз.

Сопутствующие параметры в настройках ПЛК

После перевода последовательного порта 1 или последовательного порта 2 (для которого выполняется команда RXD(235)) в беспротокольный режим связи (режим RS-232C) для этого порта также необходимо настроить следующие параметры в области настроек ПЛК.

Значение		Описание	
Параметры связи		Стандартные	Стандартные параметры имеют следующие значения: 9600 бод, 1 старт-бит, 7 битов данных, проверка четности, 2 стоп-бита.
		Настройки пользователя	Скорость: 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600 или 115200 бит/с Формат: 7 или 8 битов данных; 1 или 2 стоп-бита; чет, нечет или не проверять
Код начала		Отключен	---
		Задан	00...FF hex
Код конца	Принимаемые байты	Нет	---
		Да	00 hex: 256 байт 01...FF hex: 1...255 байт
	Код конца	Нет	---
		CR/LF	---
		Задан	00...FF hex
Задержка		0000...210F hex: 0...99 990 мс (с шагом 10 мс)	

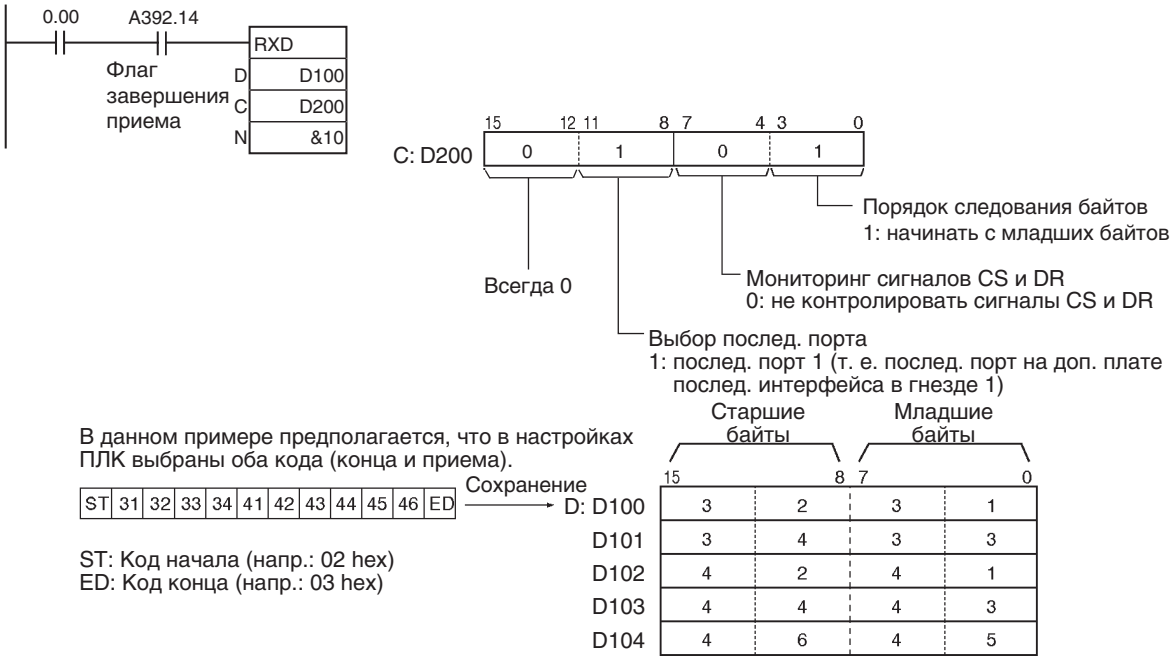
Флаги вспомогательной области

Название	Адрес	Содержание
Флаг завершения приема последовательным портом 1	A392.14	Включен, если последовательный порт завершил прием в беспротокольном режиме. Если указано количество байтов: включается, когда принято указанное количество байтов. Если указан код конца кадра: включается, когда принят код конца кадра или получено 256 байт данных.
Флаг завершения приема последовательным портом 2	A392.06	

Название	Адрес	Содержание
Флаг переполнения при приеме данных последовательным портом 1	A392.15	Включен, если во время приема данных через последовательный порт в беспротокольном режиме произошло переполнение буфера приема. Если указано количество байтов: включается, если после завершения приема до выполнения команды RXD(235) были приняты другие данные. Если указан код конца кадра: включается, если после получения кода конца кадра до выполнения команды RXD(235) были приняты другие данные. Включен, если до получения кода конца кадра было принято 257 байт данных.
Флаг переполнения при приеме данных последовательным портом 2	A392.07	
Счетчик приема последовательного порта 1	A394	Указывает (в шестнадцатеричном формате) количество байтов данных, принятое через последовательный порт в беспротокольном режиме.
Счетчик приема последовательного порта 2	A393	
Бит перезапуска последовательного порта 1	A526.01	Для перезапуска последовательного порта следует включить этот бит. При этом будут сброшены флаг завершения приема и флаг переполнения при приеме, а также будет обнулен счетчик приема. Этот бит выключается автоматически по завершении процедуры перезапуска.
Бит перезапуска последовательного порта 2	A526.00	

Примеры

Если включен бит CIO 0.00 и установлен флаг завершения приема для последовательного порта 1 (A392.14), в память ПЛК, начиная с адреса D100, сохраняются 10 байтов данных, принятых через последовательный порт 1.



3-23-5 ОТПРАВИТЬ ЧЕРЕЗ МОДУЛЬ ПОСЛЕД. ИНТЕРФЕЙСА: TXDU(256)

Назначение

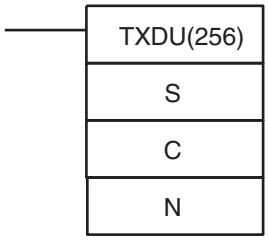
Выводит указанное количество байтов данных через один из последовательных портов модуля последовательного интерфейса серии CJ.

Примечание.

Для использования модулей последовательного интерфейса серии CJ требуется адаптер для модулей CJ.

Эту команду возможно использовать только для модулей ЦПУ CP1H. Для модулей ЦПУ CP1L она недоступна. Если команда TXDU(256) будет использована для модуля ЦПУ CP1L, будет установлен флаг ошибки (ER).

Символ РКС



S: Первое исходное слово

C: Первое управляющее слово

N: Количество байтов
0000...0100 hex (0...256)

Варианты выполнения

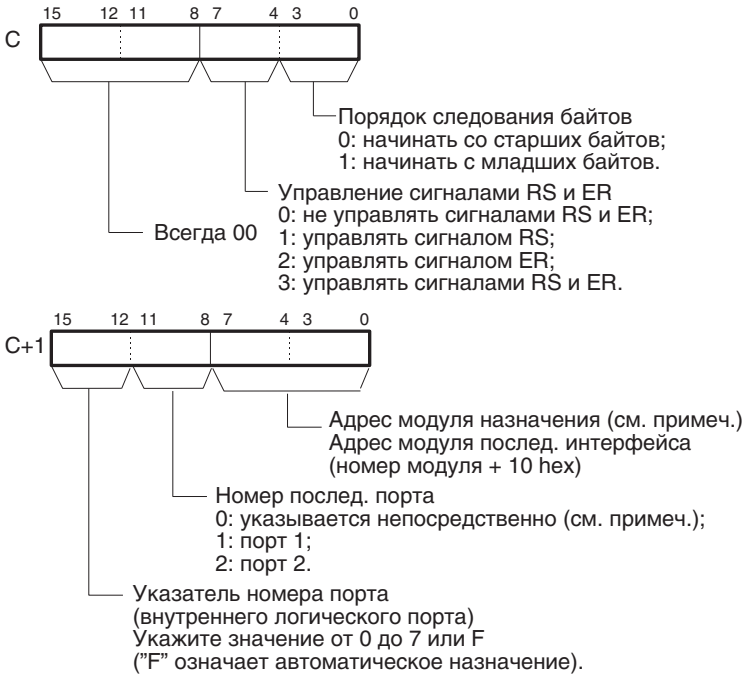
Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	TXDU(256)
	Однократное выполнение по положительному фронту	@TXDU(256)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

Содержание управляющих слов C и C+1 представлено ниже.



Примечание. Можно также непосредственно ввести адрес модуля для последовательного порта. Для этого номер последовательного порта следует задать равным «0» и указать в качестве адреса модуля последовательного интерфейса адрес модуля, принадлежащий последовательному порту (80 hex + 4 × номер модуля — для порта 1; 81 hex + 4 × номер модуля — для порта 2).

Характеристики операндов

Область	S	C	D
Область CIO	CIO 0...CIO 6143	CIO 0...CIO 6142	CIO 0...CIO 6143
Рабочая область	W0...W511	W0...W510	W0...W511
Область битов хранения	H0...H511	H0...H510	H0...H511
Область вспомогательных битов	A0...A959	A0...A958	A0...A959
Область таймеров	T0000...T4095	T0000...T4094	T0000...T4095
Область счетчиков	C0000...C4095	C0000...C4094	C0000...C4095
Область DM	D0...D32767	D0...D32766	D0...D32767
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767		
Постоянные	---	Только указанные значения	#0000...#0100 (двоичн.) или &0...&256 (десятичн.)
Регистры данных	---	---	DR0...DR15
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

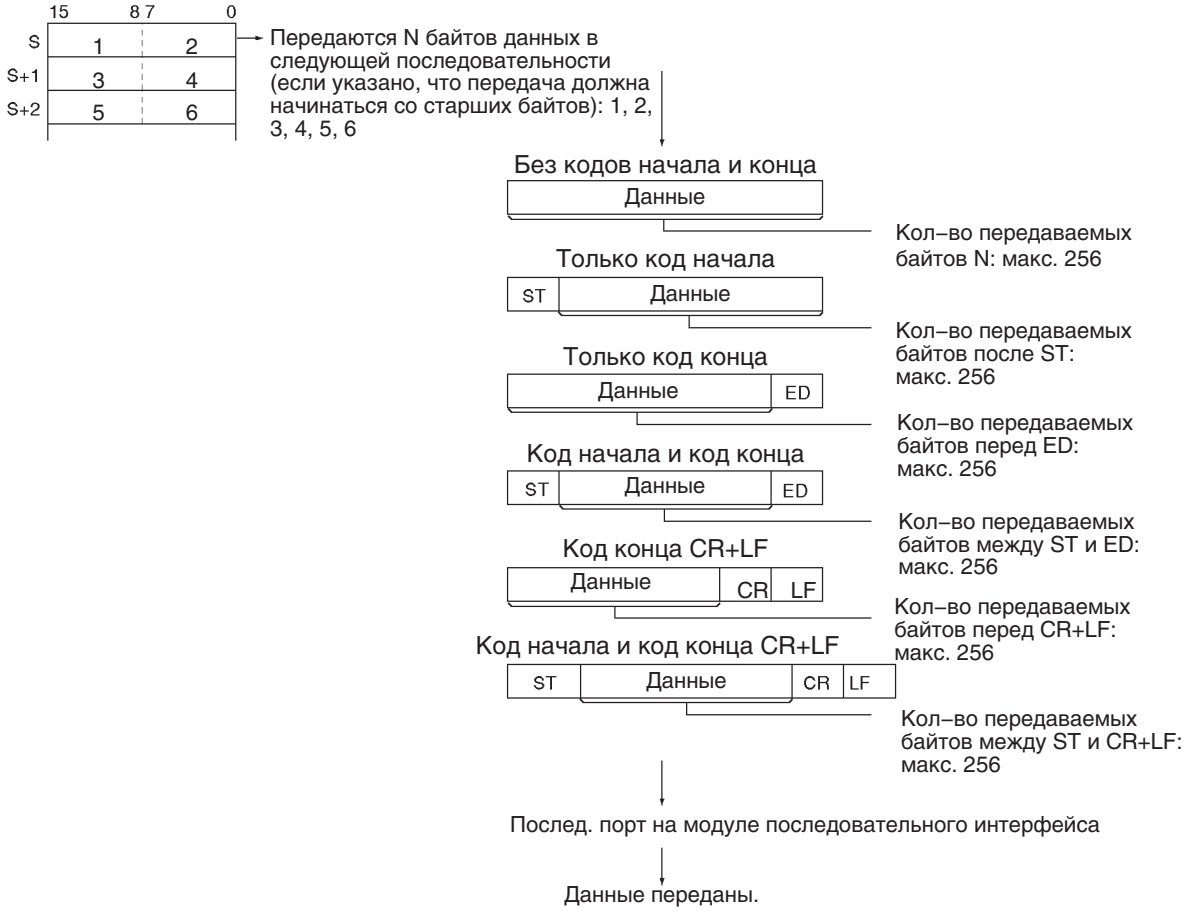
Команда TXDU(256) считывает N байтов данных из слов S...S+(N÷2)-1 и выдает эти данные в беспротокольном режиме, никак их не обрабатывая, через модуль последовательного интерфейса, номер которого указан битами 0...7 слова C+1, используя порт, указанный битами 8...11 слова C+1. В битах 12...15 слова C+1 может быть указано любое значение номера логического порта в диапазоне от 0 до 7.

Перед выводом данных к ним добавляются коды начала и конца кадра, заданные для беспротокольного режима в выделенной области настроек DM. Может быть передано до 259 байтов, в том числе полезные данные (N = макс. 256 байт), код начала и код конца кадра.

Передача данных возможна, только если для указанного логического порта включен флаг доступности порта связи (биты A202.00...A202.07 для портов 0...7) и выключен флаг выполнения команды TXDU (в выделенной области настроек DM).

Примечание. Для того чтобы номер логического порта назначался автоматически, в биты 12...15 слова C+1 следует ввести «F». Более подробно это описано в пункте *О нумерации портов связи* на стр. 930.

Следующий рисунок демонстрирует очередность передачи байтов данных, а также содержание передаваемого кадра для разных конфигураций кодов начала и конца кадра.



Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	Включен, если в момент выполнения команды все логические порты уже используются или выключен флаг доступности порта связи для указанного логического порта. Включен, если значение C выходит за указанный диапазон. Включен, если значение N выходит за диапазон 0000...0200 hex. Выключен во всех остальных случаях.

Меры предосторожности

Команда TXDU(256) может использоваться только для тех последовательных портов модуля последовательного интерфейса, которые были переведены в беспротокольный режим.

В выделенной области настроек DM может быть сконфигурирован один из следующих форматов кадра сообщения передачи данных.

- Код начала: нет или 00...FF hex.
- Код конца: нет, CR+LF или 00...FF hex.

Данные будут передаваться с любой комбинацией кодов начала или конца кадра, которая указана в выделенной области настроек DM. Если коды начала и конца кадра указаны, они добавляются к передаваемым данным (N). В данном случае в N может быть указано максимум 256 байтов.

Передача данных возможна, только если для порта включен флаг готовности к передаче.

Данные передаются в том порядке, который указан в C.

Если для N выбрано значение «0», ничего передано не будет.

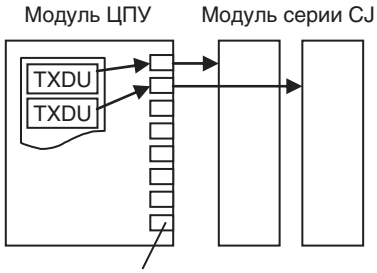
Если в C выбрано управление сигналом RS, бит 15 слова S используется в качестве сигнала RS.

Если в C выбрано управление сигналом ER, бит 15 слова S используется в качестве сигнала ER.

Если в C выбрано управление сигналами RS и ER, бит 15 слова S используется в качестве сигнала RS, а бит 14 слова S используется в качестве сигнала ER.

Фактически команда TXDU(256) внутри ПЛК взаимодействует с модулем последовательного интерфейса (с номером версии 1.2 и выше) посредством команды FINS. Она посылает ему команду передачи данных и использует для этого логический порт. Поскольку команды SEND(090), RECV(098), CMND(490), PMCR(260) и RXDU(255) также пользуются логическими портами 0...7, команду TXDU(256) будет невозможно выполнить для логического порта, который уже используется другой командой связи или другой командой TXDU(256).

Чтобы команда TXDU(256) не оказалась выполненной, когда логический порт занят, в программе следует использовать флаг доступности порта связи (A202.00...A202.07) в качестве нормально разомкнутого условия.



Команда TXDU(256) не может быть выполнена, пока включен флаг выполнения команды TXDU (бит 5 слова n+9 или n+19, где n = CIO 1500 + 25 × номер модуля). Чтобы исключить выполнение команды TXDU(256) для порта, с которым еще не завершила работу предыдущая команда TXDU(256), в программе следует использовать флаг выполнения команды TXDU в качестве нормально замкнутого условия.

Любое из следующих нарушений вызовет ошибку и установку флага ошибки.

- При выполнении команды TXDU(256) был выключен флаг доступности порта связи для указанного логического порта.
- Значение C выходит за допустимый диапазон.
- Значение N выходит за диапазон 0000...0100 hex.

Примечание. Для некоторых внешних устройств, в зависимости от особенностей их работы, может потребоваться задержка передачи данных при использовании команды TXDU(256). При необходимости задержки следует задать или скорректировать время задержки в выделенной области настроек DM.

Сопутствующие флаги и слова

В следующих таблицах перечислены параметры области настройки ПЛК и флаги вспомогательной области, которые могут использоваться по мере необходимости при работе с командой TXDU(256).

Параметры в области настроек DM

(m = D30000 + 100 × номер модуля)

Слово области настроек		Бит	Название	Значение
Порт 1	Порт 2			
m+2	m+12	15	Выбор задержки передачи в беспротокольном режиме	0: по умолч. (0 мс); 1: использовать значение в битах 1...14.
		0...14	Время задержки передачи в беспротокольном режиме	0000...7530 hex 0...300 000 мс десятичн. (с шагом 10 мс)
m+4	m+14	8...15	Код начала кадра в беспротокольном режиме	00...FF hex
		0...7	Код конца кадра в беспротокольном режиме	00...FF hex
m+5	m+15	12...15	Выбор кода начала кадра в беспротокольном режиме	0: нет; 1: использовать код начала.
		8...11	Выбор кода конца кадра в беспротокольном режиме	0: нет; 1: использовать код конца кадра; 2: использовать CR+LF.

Вспомогательная область

Название	Адрес	Описание
Флаги доступности портов связи	A202.00... A202.07	Включен, если порт с соответствующим номером доступен для выполнения команды связи (в т. ч. TXDU(256)). Биты 00...07 соответствуют портам связи 0...7. Этот флаг выключен во время выполнения команды связи и включается по завершении выполнения (успешного или неуспешного).
Коды завершения для портов связи	A203... A210	После выполнения команд связи эти слова содержат коды завершения для портов связи с соответствующими номерами. Слова A203...A210 соответствуют портам связи 0...7. Слово кода завершения содержит «00» во время выполнения команды связи и содержит соответствующий код по завершении выполнения. В начале работы ПЛК все эти слова обнуляются (0000).
Флаги ошибок портов связи	A219.00... A219.07	Включен, если при выполнении команды связи произошла ошибка. Если этот флаг включен, для выяснения причины ошибки следует прочитать код завершения в словах A203...A210. Этот флаг выключен, если выполнение команды завершилось успешно (т. е. без ошибок). Биты 00...07 соответствуют портам связи 0...7. Состояние этого флага сохраняется вплоть до выполнения следующей команды связи. Даже если предыдущая операция завершилась ошибкой, при следующем выполнении команды связи флаг будет сброшен.

Коды завершения

Код	Значение
0205 hex	Превышено время ожидания ответа (эта ошибка может возникать в режиме связи по протоколу Host Link).
0401 hex	Неизвестная команда (эта ошибка может возникать в режимах связи: макрос протокола, NT Link, проверка отклика или шлюз последовательного интерфейса).
1001 hex	Слишком длинная команда.
1002 hex	Слишком короткая команда.
1003 hex	Указанное количество элементов данных не соответствует фактическому объему передаваемых данных.
1004 hex	Неверный формат команды.
110C hex	Другая ошибка параметра.
2201 hex	Выполнение операции невозможно во время выполнения другой операции (в выполнении отказано, так как модуль в данный момент передает данные).
2202 hex	Выполнение операции невозможно при остановленном выполнении (в выполнении отказано, так как модуль в данный момент переключает протоколы связи).

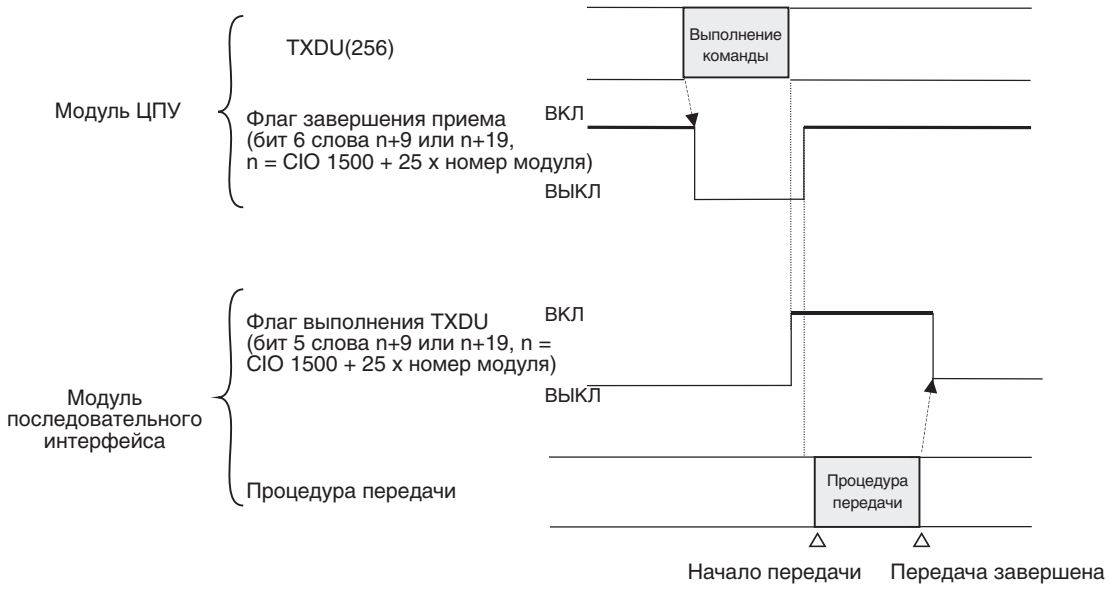
Сопутствующие флаги в области модулей шины ЦПУ

(n = CIO 1500 + 25 × номер модуля)

Слово		Бит	Название	Состояние
Порт 1	Порт 2			
n+9	n+19	05	Флаг выполнения команды TXDU	0: TXDU(256) не выполняется; 1: TXDU(256) выполняется.

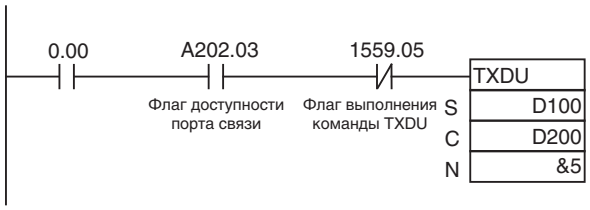
Пример: работа флагов

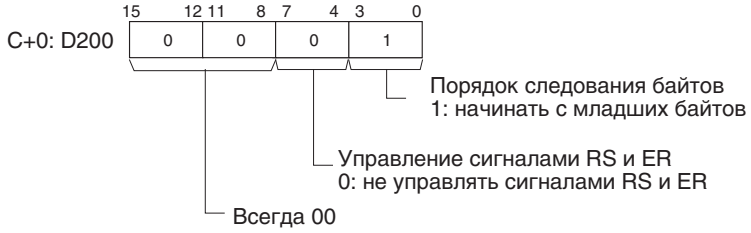
Следующая временная диаграмма демонстрирует работу флага доступности порта связи и флага выполнения команды TXDU.

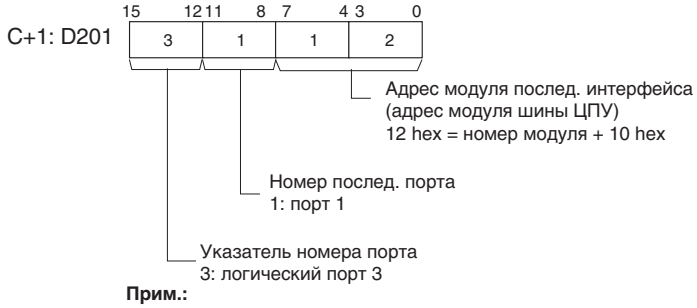


Пример: передача данных

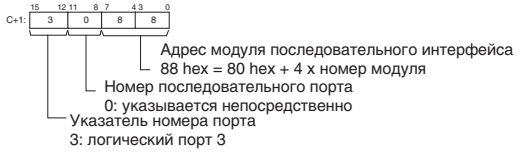
Если в приведенном ниже примере программы включен бит CIO 0.00, включен бит A202.03 (флаг доступности порта связи) и выключен бит CIO 1559.05 (флаг выполнения команды TXDU для порта 1): команда TXDU(256) осуществляет передачу данных через последовательный порт 1 модуля последовательного интерфейса с номером модуля 2. Пять байтов выходных данных считываются из области DM, начиная с младшего байта слова D100, и выдаются через логический порт 3 на некоторое устройство общего назначения (например, на принтер).

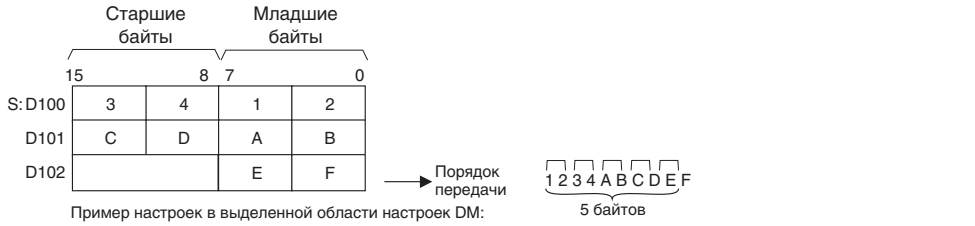






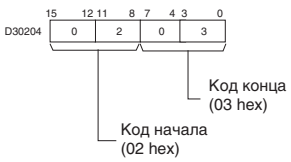
Прим.:
Можно непосредственно ввести адрес модуля для последовательного порта. Для этого следует задать номер последовательного порта равным “0” и указать в качестве адреса модуля последовательного интерфейса адрес модуля, принадлежащий последовательному порту (80 hex + 4 x номер модуля — для порта 1; 81 hex + 4 x номер модуля — для порта 2).





Пример настроек в выделенной области настроек DM:

Значения кодов начала и конца



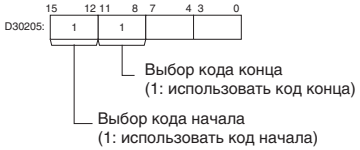
В данном примере в словах настроек области DM выбрано использование и кода начала, и кода конца.

ST	12	34	AB	CD	EF	ED

ST: Код начала (напр.: 02 hex)
ED: Код конца (напр.: 03 hex)

Данные переданы.

Выбор использования кодов начала и конца



3-23-6 ПРИНЯТЬ ЧЕРЕЗ МОДУЛЬ ПОСЛЕД. ИНТЕРФЕЙСА: RXDU(255)

Назначение

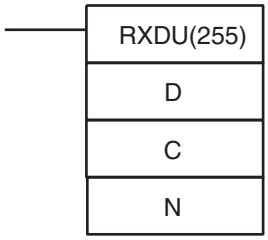
Чтение указанного количества байтов данных через один из последовательных портов модуля последовательного интерфейса серии CJ.

Примечание.

Для использования модулей последовательного интерфейса серии CJ требуется адаптер для модулей CJ.

Эту команду возможно использовать только для модулей ЦПУ CP1H. Для модулей ЦПУ CP1L она недоступна. Если команда RXDU(255) будет использована для модуля ЦПУ CP1L, будет установлен флаг ошибки (ER).

Символ РКС



D: Первое слово назначения

C: Первое управляющее слово

N: Количество байтов
0000...0100 hex (0...256)

Варианты выполнения

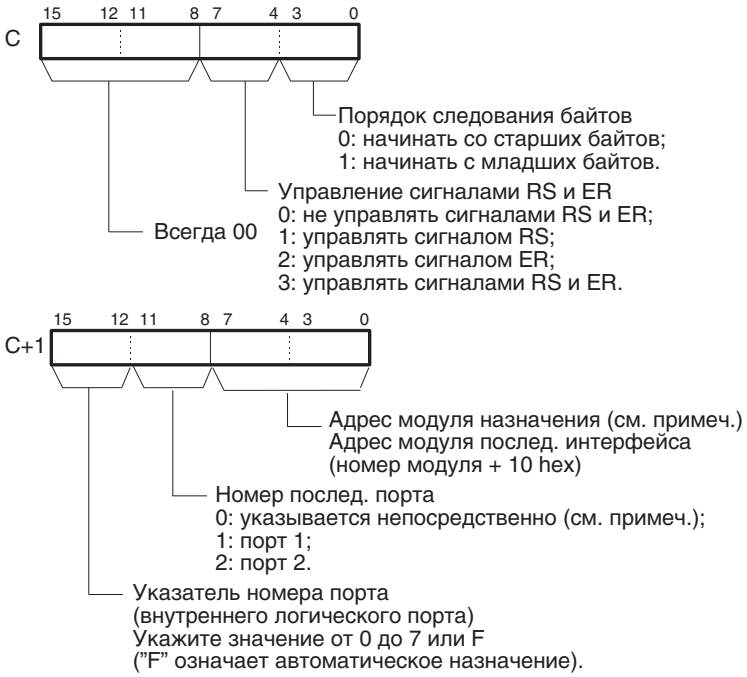
Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	RXDU(255)
	Однократное выполнение по положительному фронту	@RXDU(255)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

Содержание управляющих слов C и C+1 представлено ниже.



Примечание. Можно также непосредственно ввести адрес модуля для последовательного порта. Для этого номер последовательного порта следует задать равным «0» и указать в качестве адреса модуля последовательного интерфейса адрес модуля, принадлежащий последовательному порту (80 hex + 4 × номер модуля — для порта 1; 81 hex + 4 × номер модуля — для порта 2).

Характеристики операндов

Область	D	C	D
Область CIO	CIO 0...CIO 6143	CIO 0...CIO 6142	CIO 0...CIO 6143
Рабочая область	W0...W511	W0...W510	W0...W511
Область битов хранения	H0...H511	H0...H510	H0...H511
Область вспомогательных битов	A448...A959	A0...A958	A0...A959
Область таймеров	T0000...T4095	T0000...T4094	T0000...T4095
Область счетчиков	C0000...C4095	C0000...C4094	C0000...C4095
Область DM	D0...D32767	D0...D32766	D0...D32767
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767		
Постоянные	---	Только указанные значения	#0000...#0100 (двоичн.) или &0...&256 (десятичн.)
Регистры данных	---	---	DR0...DR15
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда RXDU(255) считывает данные, которые были приняты в беспротокольном режиме модулем последовательного интерфейса с адресом модуля, указанным в битах 0...7 слова C+1, через порт, указанный в битах 8...11 слова C+1, и сохраняет эти данные в память модуля ЦПУ, начиная с адреса D. Если через порт будет принято меньше байтов данных, чем указано в N, в память будет сохранено столько байтов данных, сколько их будет принято. В битах 12...15 слова C+1 может быть указано любое значение номера логического порта в диапазоне от 0 до 7.

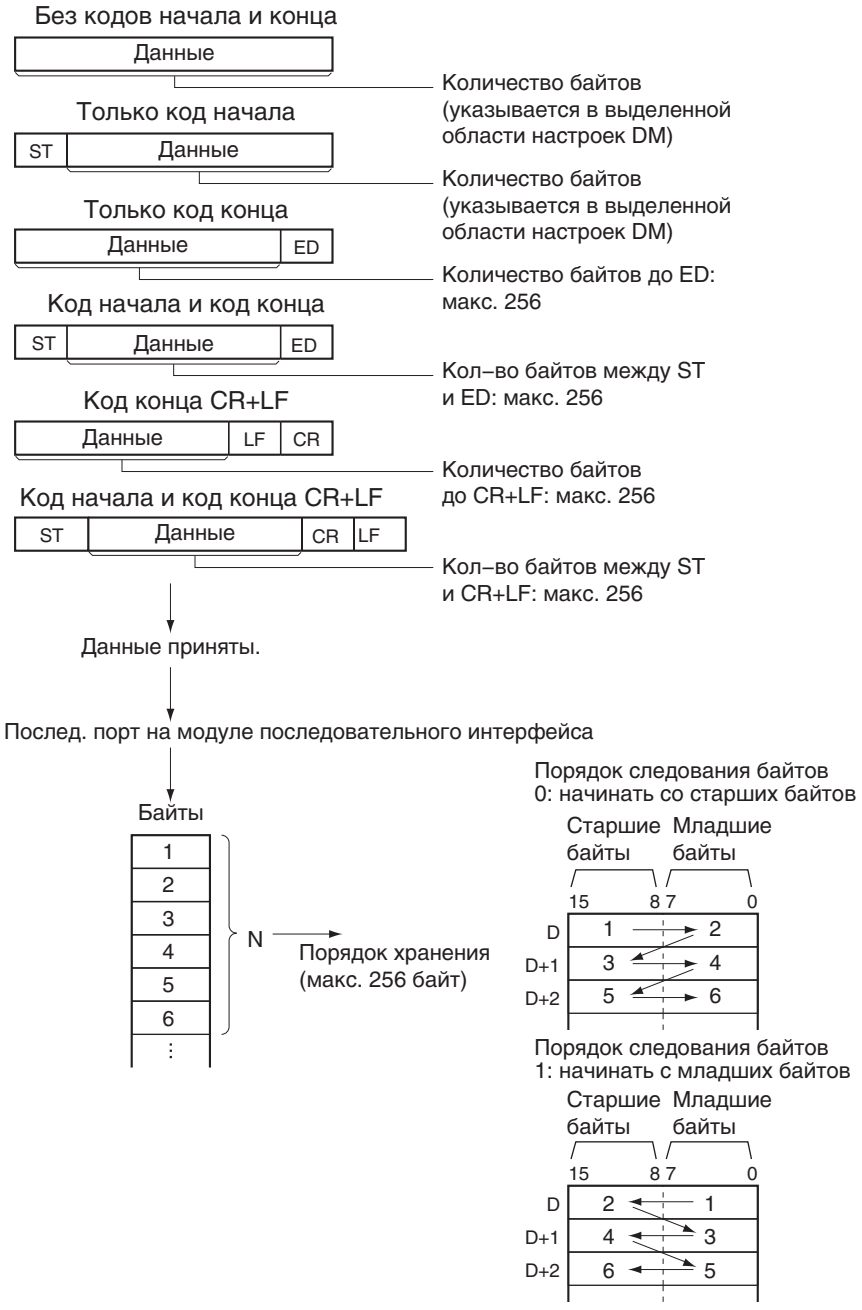
Команду RXDU(255) следует использовать для считывания принятых данных из буфера приема после включения флага завершения приема (в выделенной области настроек DM).

Может быть принято до 259 байтов, в том числе полезные данные (N = макс. 256 байт), код начала и код конца кадра.

Следующий рисунок демонстрирует очередность приема байтов данных, а также содержание принимаемого кадра для разных конфигураций кодов начала и конца кадра.

Примечание. Для того чтобы номер логического порта назначался автоматически, в биты 12...15 слова C+1 следует ввести «F». Более подробно это описано в пункте *О нумерации портов связи* на стр. 930.

Следующий рисунок демонстрирует очередность передачи байтов данных, а также содержание передаваемого кадра для разных конфигураций кодов начала и конца кадра.



Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	Включен, если в момент выполнения команды все логические порты уже используются или выключен флаг доступности порта связи (A202.00...A202.07) для указанного логического порта. Включен, если значение C выходит за указанный диапазон. Включен, если значение N выходит за диапазон 0000...0100 hex. Выключен во всех остальных случаях.

Меры предосторожности

Команда RXDU(255) может использоваться только для тех последовательных портов модуля последовательного интерфейса, которые были переведены в беспротокольный режим.

В выделенной области настроек DM может быть сконфигурирован один из следующих форматов кадра сообщения приема данных.

- Код начала: нет или 00...FF hex.
- Код конца: нет, CR+LF или 00...FF hex. Если код конца кадра не используется (т. е. выбрано «нет»), для количества принимаемых байтов может быть указано значение от 01 до FF hex (от 1 до 255 десятичн.; 00 = 256 байтов).

Когда количество принятых байтов достигает значения, указанного в выделенной области настроек DM, устанавливается флаг завершения приема. Когда устанавливается флаг завершения приема, количество байтов, указываемое счетчиком приема, имеет такое же значение, что и количество принимаемых байтов, которое указано в выделенной области настроек DM. Если принятых байтов окажется больше, чем указано, установится флаг переполнения при приеме данных.

Если в выделенной области настроек DM указан код конца кадра, флаг завершения приема будет установлен, когда будет получен код конца кадра или будет принято 256 байт данных. Если после включения флага завершения приема будут приняты еще данные, установится флаг переполнения при приеме данных.

После получения 259 байтов данных дальнейший прием данных прекращается. Если после этого поступят еще данные, будут установлены флаг ошибки избытка данных и флаг ошибки передачи.

Если последовательным портом модуля последовательного интерфейса будет принято больше данных, чем указано в N, при следующем выполнении команды RXDU(255) эти избыточные данные будут отброшены.

При выполнении команды RXDU(255) принятые данные сохраняются в память модуля ЦПУ, начиная с адреса D, сбрасывается флаг завершения приема (даже если установлен флаг переполнения при приеме (примеч. 3)), обнуляется счетчик приема.

Данные сохраняются в память в том порядке, который указан в C.

Если для N было указано значение «0», то будут сброшены флаг завершения приема и флаг переполнения при приеме данных (если он был установлен), будет обнулен счетчик приема, но в память ничего записано не будет.

Если в C выбран мониторинг сигнала CS, в бит 15 слова D записывается состояние сигнала CS.

Если в C выбран мониторинг сигнала DR, в бит 15 слова D записывается состояние сигнала DR.

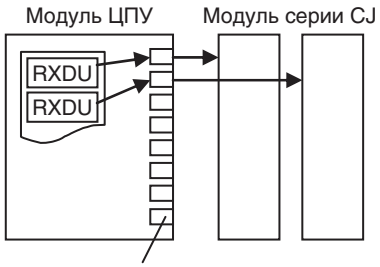
Если в C выбран мониторинг сигналов CS и DR, в бит 15 слова D записывается состояние сигнала CS, а в бит 14 слова D записывается состояние сигнала DR.

Если указан мониторинг сигнала CS или DR, принятые данные в память не сохраняются.

Если вторая тетрада слова C (управление сигналами RS и DR) содержит значение 1, 2 или 3 hex, команда RXDU(255) выполняется независимо от состояния флага завершения приема.

Фактически команда RXDU(255) внутри ПЛК взаимодействует с модулем последовательного интерфейса посредством команды FINS. Она посылает ему команду приема данных и использует для этого логический порт. Поскольку команды SEND(090), RECV(098), CMND(490), PMCR(260) и TXDU(255) также пользуются логическими портами 0...7, команду RXDU(256) будет невозможно выполнить для логического порта, который уже используется другой командой связи или другой командой RXDU(255).

Чтобы команда RXDU(255) не оказалась выполненной, когда логический порт занят, в программе следует использовать флаг доступности порта связи (A202.00...A202.07) в качестве нормально разомкнутого условия.



Внутренние логические порты (8 портов)

Команда RXDU(255) не может быть выполнена, пока не включен флаг завершения приема (бит 6 слова n+9 или n+19, где n = CIO 1500 + 25 × номер модуля). Используйте флаг завершения приема в качестве нормально разомкнутого условия для команды RXDU(255).

Любое из следующих нарушений вызовет ошибку и установку флага ошибки.

- При выполнении команды RXDU(255) был выключен флаг доступности порта связи для указанного логического порта.
- Значение C выходит за допустимый диапазон.
- Значение N выходит за диапазон 0000...0100 hex.

Чтение данных из буфера приема с помощью команды RXDU(255) следует производить по возможности сразу после установки флага завершения приема. Если данные считаны не будут, а прием продолжится, буфер приема (емкость: 260 байт) окажется переполнен и прием данных прекратится. В этом случае порт должен быть перезапущен для возобновления работы.

При выполнении команды RXDU(255) для последовательного порта модуля последовательного интерфейса буфер приема очищается. Таким образом, командой RXDU(255) можно воспользоваться для чтения текущего содержимого буфера приема только один раз.

Сопутствующие флаги и слова

В следующих таблицах перечислены слова, имеющие отношение к работе команды RXDU(255).

Параметры в области настроек DM

(m = D30000 + 100 × номер модуля)

Слово области настроек		Бит	Название	Значение
Порт 1	Порт 2			
m+4	m+14	8...15	Код начала кадра в беспротокольном режиме	00...FF hex
		0...7	Код конца кадра в беспротокольном режиме	00...FF hex

Слово области настроек		Бит	Название	Значение
Порт 1	Порт 2			
m+5	m+15	12...15	Выбор кода начала кадра в беспротокольном режиме	0: нет; 1: использовать код начала.
		8...11	Выбор кода конца кадра в беспротокольном режиме	0: нет; 1: использовать код конца кадра; 2: использовать CR+LF.

Вспомогательная область

Название	Адрес	Описание
Флаги доступности портов связи	A202.00... A202.07	Включен, если порт с соответствующим номером доступен для выполнения команды связи (в т. ч. RXDU(255)). Биты 00...07 соответствуют портам связи 0...7. Этот флаг выключен во время выполнения команды связи и включается по завершении выполнения (успешного или неуспешного).
Коды завершения для портов связи	A203... A210	После выполнения команд связи эти слова содержат коды завершения для портов связи с соответствующими номерами. Слова A203...A210 соответствуют портам связи 0...7. Слово кода завершения содержит «00» во время выполнения команды связи и содержит соответствующий код по завершении выполнения. В начале работы ПЛК все эти слова обнуляются (0000).
Флаги ошибок портов связи	A219.00... A219.07	Включен, если при выполнении команды связи произошла ошибка. Если этот флаг включен, для выяснения причины ошибки следует прочитать код завершения в словах A203...A210. Этот флаг выключен, если выполнение команды завершилось успешно (т. е. без ошибок). Биты 00...07 соответствуют портам связи 0...7. Состояние этого флага сохраняется вплоть до выполнения следующей команды связи. Даже если предыдущая операция завершилась ошибкой, при следующем выполнении команды связи флаг будет сброшен.

Коды завершения

Код	Значение
0205 hex	Превышено время ожидания ответа (эта ошибка может возникать в режиме связи по протоколу Host Link).
0401 hex	Неизвестная команда (эта ошибка может возникать в режимах связи: макрос протокола, NT Link, проверка отклика или шлюз последовательного интерфейса).
1001 hex	Слишком длинная команда.
1002 hex	Слишком короткая команда.
1004 hex	Неверный формат команды.
110C hex	Другая ошибка параметра.
2201 hex	Выполнение операции невозможно во время выполнения другой операции (в выполнении отказано, так как модуль в данный момент передает данные).
2202 hex	Выполнение операции невозможно при остановленном выполнении (в выполнении отказано, так как модуль в данный момент переключает протоколы связи).

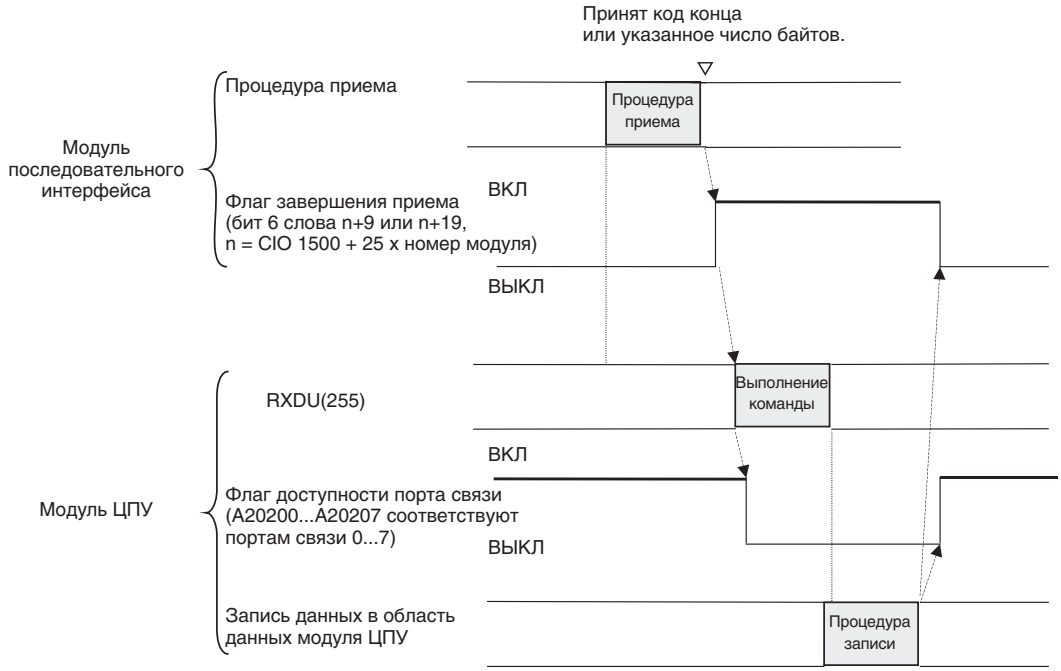
Сопутствующие флаги в области модулей шины ЦПУ

(n = CIO 1500 + 25 × номер модуля)

Слово		Бит	Функция
Порт 1	Порт 2		
n+8	n+18	04	Флаг ошибки избытка данных 1: До выполнения команды RXDU(255) буфер приема содержал больше, чем 259 байтов данных. Прим. После установки данный флаг ошибки может быть сброшен только путем выключения и повторного включения питания либо путем перезапуска платы.
n+9	n+19	06	Флаг завершения приема 0: Данные не были приняты или принимаются в данный момент. 1: Прием завершен. 0 → 1: Платой или модулем принято указанное количество байтов. 1 → 0: Была выполнена команда RXDU(255) с целью записи данных из буфера в область данных модуля ЦПУ.
n+9	n+19	07	Флаг переполнения при приеме данных 0: Количество байтов, принятое модулем, не превысило указанное количество байтов. 1: Количество байтов, принятое модулем, превысило указанное количество байтов. 0 → 1: После завершения приема данных на модуль поступили еще данные. 1 → 0: Была выполнена команда RXDU(255) с целью записи данных из буфера в область данных модуля ЦПУ.
n+10	n+20	05	Счетчик приема Указывает количество принятых данных в шестнадцатеричном формате в диапазоне от 0000 до 0100 hex (от 0 до 256 десятичн.).

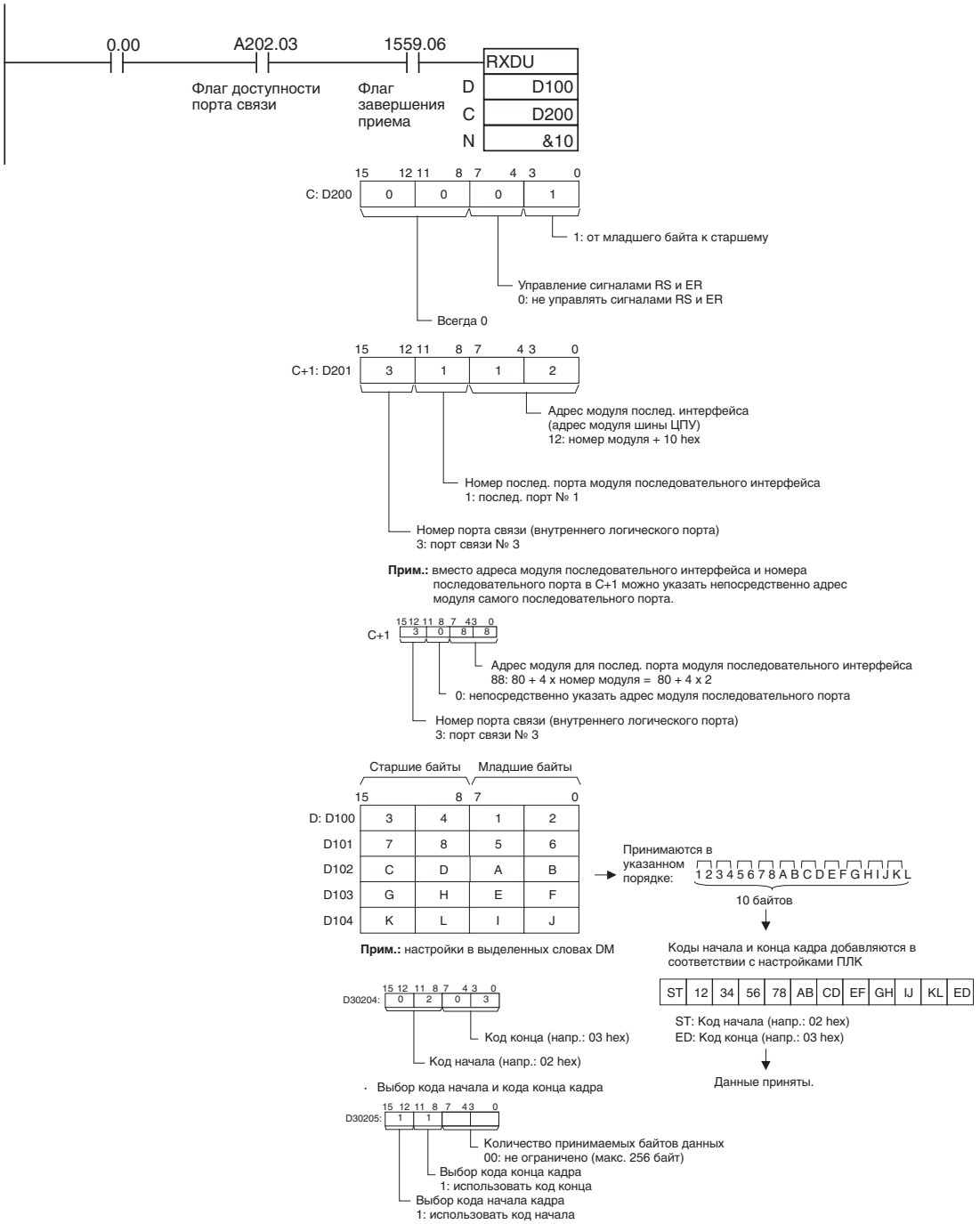
Пример: работа флагов

Следующая временная диаграмма иллюстрирует работу команды RXDU(255) и связанных с нею флагов.



Пример: прием данных

Если включен бит CIO 0.00, включен бит A202.03 (флаг доступности порта связи) и выключен бит CIO 1559.06 (флаг завершения приема для порта 1), то команда RXDU(255) считывает данные, которые были приняты через последовательный порт 1 модуля последовательного интерфейса с номером модуля 2. (Производится прием данных от устройства общего назначения, например от считывателя штрих-кодов, с использованием логического порта связи с номером 3.) Десять байтов принятых данных записываются в область DM, начиная с младшего байта слова по адресу D100.

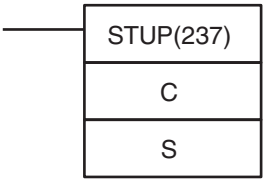


3-23-7 ИЗМЕНИТЬ ПАРАМЕТРЫ ПОСЛЕД. ПОРТА: STUP(237)

Назначение

Изменение параметров связи для последовательного порта дополнительной платы последовательного интерфейса или модуля последовательного интерфейса серии CJ (модуля шины ЦПУ). Команда STUP(237) позволяет изменить режим (протокол) связи непосредственно во время работы ПЛК.

Символ РКС



C: Управляющее слово (порт)

S: Первое исходное слово

Варианты выполнения

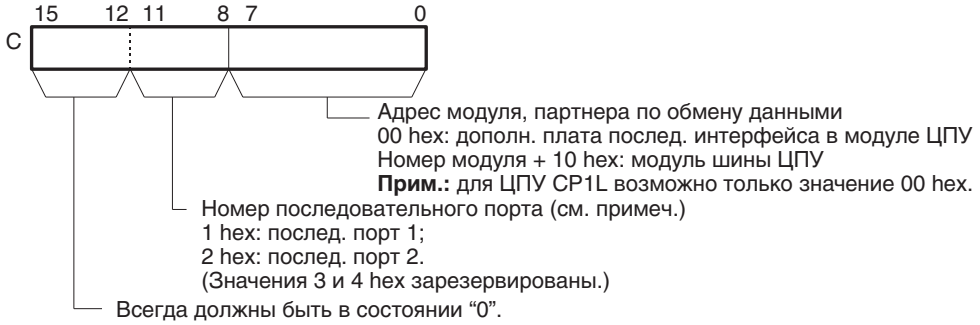
Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	STUP(237)
	Однократное выполнение по положительному фронту	@STUP(237)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	Не допускается

Операнды

Содержание управляющего слова C представлено ниже.



Примечание.

Модули ЦПУ CP1H и CP1L M (30 или 40 вх./вых.):
1 hex: порт 1;
2 hex: порт 2.
Модули ЦПУ CP1L L (14 или 20 вх./вых.):
1 или 2 hex: порт 1;
0 hex: недопустимое значение
(при значении 0 hex будет установлен флаг ошибки).

Характеристики операндов

Область	C	S
Область CIO	CIO 0...CIO 6143	CIO 0...CIO 6134
Рабочая область	W0...W511	W0...W502
Область битов хранения	H0...H511	H0...H502

Область	C	S
Область вспомогательных битов	A0...A438 A448...A959	A0...A438 A448...A950
Область таймеров	T0000...T4095	T0000...T4086
Область счетчиков	C0000...C4095	C0000...C4086
Область DM	D0...D32767	D0...D32758
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767	
Постоянные	Только указанные значения	#0000
Регистры данных	DR0...DR15	---
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

Описание

Команда STUP(237) записывает 10 слов данных (S...S+9) в область параметров связи модуля с указанным адресом модуля в соответствии с таблицей, представленной ниже. Если в S указывается константа #0000, параметры соответствующего порта сбрасываются к принимаемым по умолчанию значениям.

Адрес модуля	Модуль	Номер порта	Послед. порт	Область параметров связи послед. порта
00 hex	Модуль ЦПУ	1 hex	Порт 1	Параметры связи для последовательного порта 1 в настройках ПЛК
		2 hex	Порт 2	Параметры связи для последовательного порта 2 в настройках ПЛК
Номер модуля + 10 hex	Модуль последовательного интерфейса серии CJ (модуль шины ЦПУ)	1 hex	Порт 1	10 слов, начиная с D30000 + 100 x номер модуля.
		2 hex	Порт 2	10 слов, начиная с D30000 + 100 x номер модуля + 10

Примечание.

Последовательный порт 1 — это порт доп. платы последовательного интерфейса, установленной в гнездо для доп. платы 1. Последовательный порт 2 — это порт доп. платы последовательного интерфейса, установленной в гнездо для доп. платы 2.

При выполнении команды STUP(237) устанавливается соответствующий флаг изменения параметров последовательного порта (A619.01, A619.02 или A619...A636). Этот флаг остается включенным до тех пор, пока не завершается изменение параметров.

Команду STUP(237) можно использовать с целью изменения параметров связи для некоторого порта непосредственно во время работы ПЛК в соответствии с текущими условиями работы. Например, командой STUP(237) можно воспользоваться для переключения в режим связи по протоколу Host Link для связи с компьютерной станцией (для программирования и мониторинга) при наступлении определенных условий во время выполнения процедуры связи через модемное соединение.

Если после изменения параметров связи, произведенного с помощью команды STUP(237), ПЛК будет выключен и вновь включен, параметры будут возвращены к прежним значениям (т. е. тем, которые действовали до выполнения STUP(237)).

Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	Включен, если значения в C выходят за указанный диапазон. Включен, если команда STUP(237) была выполнена для порта, чей флаг изменения параметров связи был установлен. Включен, если команда STUP(237) была выполнена в задаче обработки прерывания. Выключен во всех остальных случаях.

Меры предосторожности

К параметрам связи относятся: режим связи (протокол), скорость передачи данных, формат данных (способ передачи и макс. объем передаваемых данных в режиме макропрограммы протокола) и другие параметры. Подробную информацию о настройке параметров для используемого последовательного порта см. в руководстве *Серия CJ. Платы и модули последовательного интерфейса — Руководство по эксплуатации (W336)*.

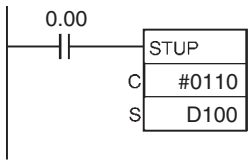
Сопутствующие флаги и слова

В следующей таблице перечислены флаги, которые могут использоваться при выполнении команды STUP(237). Эти флаги находятся во вспомогательной области.

Название	Адрес	Содержание
Флаг изменения параметров последовательного порта 1	A619.01	Включен во время изменения параметров связи для послед. порта 1.
Флаг изменения параметров последовательного порта 2	A619.02	Включен во время изменения параметров связи для послед. порта 2.
Флаги изменения параметров порта для портов 1...4 модулей последовательного интерфейса 1...15.	A620.01... A620.04 A635.01... A635.04	Включен, если в данный момент для порта модуля последовательного интерфейса изменяются параметры связи.

Примеры

При включении бита CIO 0.00 будут изменены параметры связи последовательного порта 1 модуля последовательного интерфейса. Вместо прежних значений будут установлены значения, содержащиеся в десяти словах D100...D109. В данном примере текущий режим связи меняется на режим макропрограммы протокола.



S: D100	0	6	0	0
S+1: D101	0	0	0	0
S+2: D102				
...	to			
S+9: D109				

Настройки порта: по умолч., режим протокола: 6 hex (макрос протокола).
Скорость передачи: по умолч. (9600 бит/с)

Передаются

Слова DM, выделенные для настройки параметров связи для модуля последовательного интерфейса

D30000	0	6	0	0
D30001	0	0	0	0
D30002				
...	to			
D30009				

3-24 Сетевые команды

В данном разделе описаны команды, предназначенные для передачи и приема данных по сети.

Команда	Мнемоническое обозначение	Код функции	Стр.
ПЕРЕДАТЬ ПО СЕТИ	SEND	090	945
ПРИНЯТЬ ПО СЕТИ	RECV	098	952
ДОСТАВИТЬ КОМАНДУ	CMND	490	958
ОТПРАВИТЬ ЯВНОЕ СООБЩЕНИЕ	EXPLT	720	967
ЯВНЫЙ ЗАПРОС АТРИБУТА	EGATR	721	975
ЯВНАЯ УСТАНОВКА АТРИБУТА	ESATR	722	982
ЯВНОЕ ЧТЕНИЕ СЛОВА	ECHRD	723	989
ЯВНАЯ ЗАПИСЬ СЛОВА	ECHWR	724	993

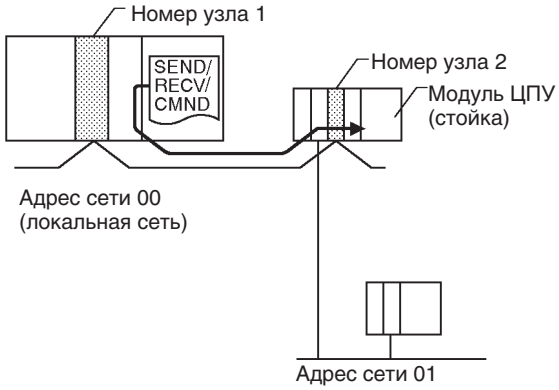
3-24-1 Общие сведения о сетевых командах

Сетевые узлы, оборудованные дополнительными платами последовательного интерфейса и модулями связи серии CJ, используют сетевые команды для приема и передачи данных, а также для осуществления операций управления (переключение режима работы и т. п.). Узлами сети могут быть модули ЦПУ серии CP, модули ЦПУ серии CS/CJ, модули шины ЦПУ серии CS/CJ, а также компьютеры.

Команда	Содержание сообщения	Операция
SEND(090)/ RECV(098)	Команды для передачи и приема данных (команда FINS)	
CMND(490)	Любые команды (команда FINS)	

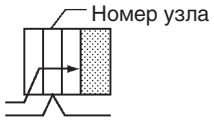
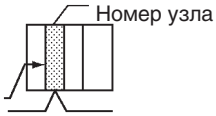
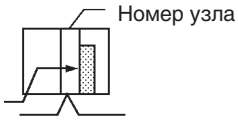
Коммуникационные команды, выполняемые сетевыми командами программирования, называются «командами FINS». Они используются для осуществления связи между различными устройствами системы автоматизации промышленного предприятия. (Более подробно эти команды описаны в руководстве *Серия CS/CJ/CP — Справочное руководство по командам связи*.) С помощью команды FINS можно осуществлять обмен данными (в формате «команда – ответ») с любым устройством (модулем) в любой сети, просто указывая «координаты» адресуемого модуля: адрес сети, номер узла и номер модуля.

В приведенном ниже примере команда FINS передается на модуль ЦПУ через узел с номером 2 в сети с адресом 00.

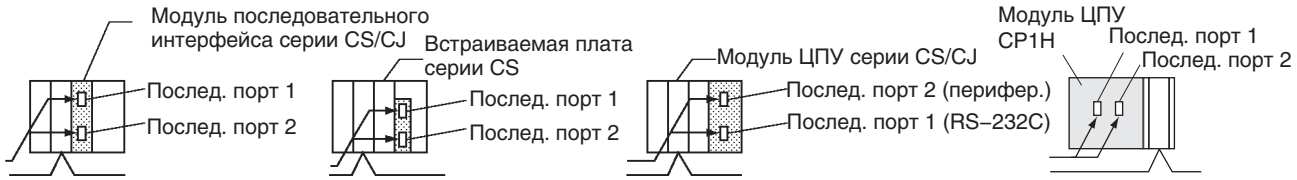


1,2,3...

1. Адрес сети (Network address):
   Адрес сети (локальная сеть = 00).
2. Номер узла (Node number)
   Логический адрес в сети.
3. Номер модуля (Unit number)
   Номер, присвоенный адресуемому модулю.
   a. Модуль ЦПУ серии CP или CS/CJ: 00
   b. Модуль шины ЦПУ серии CS/CJ: номер модуля + 10 hex
   c. Спец. модуль вв./выв. серии CS/CJ: номер модуля + 20 hex
   d. Встраиваемая плата серии CS: E1 hex
   e. Компьютер:01

Номер модуля (hex)	Адресуемое устройство
00	
Номер модуля +10	
E1	
01	

Примечание. Также можно непосредственно указать последовательный порт (адрес модуля), входящий в состав адресуемого устройства.



Адреса последовательных портов:

- Порты модуля последовательного интерфейса серии CS/CJ
  Порт 1: 80 hex + 4 × номер модуля

Номер модуля	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Шестнадц.	80	84	88	8C	90	94	98	9C	A0	A4	A8	AC	B0	B4	B8	BC
Десятичн.	128	132	136	140	144	148	152	156	160	164	168	172	176	180	184	188

  Порт 2: 81 hex + 4 × номер модуля

Номер модуля	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Шестнадц.	81	85	89	8D	91	95	99	9D	A1	A5	A9	AD	B1	B5	B9	BD
Десятичн.	129	133	137	141	145	149	153	157	161	165	169	173	177	181	185	189

- Порты платы последовательного интерфейса серии CS
  Порт 1: E4 hex (228 десят.)
  Порт 2: E5 hex (229 десят.)
- Порты модуля ЦПУ серии CS/CJ
  Периферийный порт: FD hex (253 десят.)
  Порт RS-232C: FC hex (252 десят.)
- Порты доп. платы последовательного интерфейса модуля ЦПУ CP1H и CP1L M (30 или 40 вх./вых.)
  Порт 1: FD hex (253 десят.)
  Порт 2: FC hex (252 десят.)
- Порт доп. платы последовательного интерфейса модуля ЦПУ CP1L L (14 или 20 вх./вых.)
  Порт 1: FC hex (252 десят.)

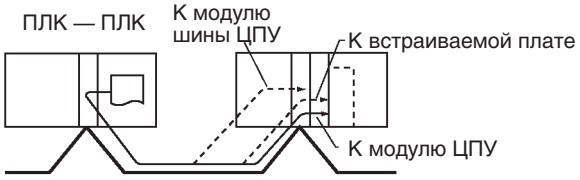
Примечание. Последовательный порт 1 — это порт доп. платы последовательного интерфейса, установленной в гнездо для доп. платы 1. Последовательный порт 2 — это порт доп. платы последовательного интерфейса, установленной в гнездо для доп. платы 2.

Модели сетевых коммуникаций

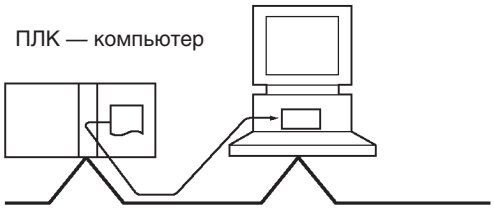
Ниже приведены примеры сетевых коммуникаций трех типов: связь ПЛК с другими устройствами в сети, связь ПЛК с другими устройствами в сети через последовательные порты этих устройств и связь с центральным компьютером по протоколу Host Link.

Связь с другими устройствами в сети

Ниже показан пример реализации связи между ПЛК и устройствами, входящими в состав другого ПЛК (модуль ЦПУ, модуль шины ЦПУ или встраиваемая плата). Дополнительную информацию следует смотреть в руководстве пользователя для сети используемого типа (Controller Link или Ethernet).

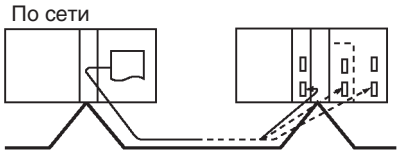


В приведенном ниже примере ПЛК осуществляет обмен данными с персональным компьютером.



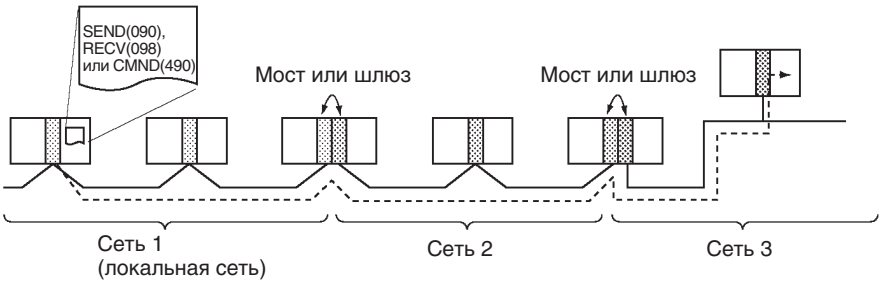
Связь по сети через последовательный порт

Связь между ПЛК и другими устройствами в сети может осуществляться через последовательные порты этих устройств. Ниже первый пример демонстрирует связь через последовательные порты устройств другого ПЛК (модуль ЦПУ, модуль шины ЦПУ или встраиваемая плата), на втором рисунке показан пример связи через последовательный порт в пределах одной стойки ЦПУ.





Примечание. Связь может охватывать до 8 сетевых уровней, включая локальную сеть (локальной называется сеть, в которой находится узел, инициирующий связь).



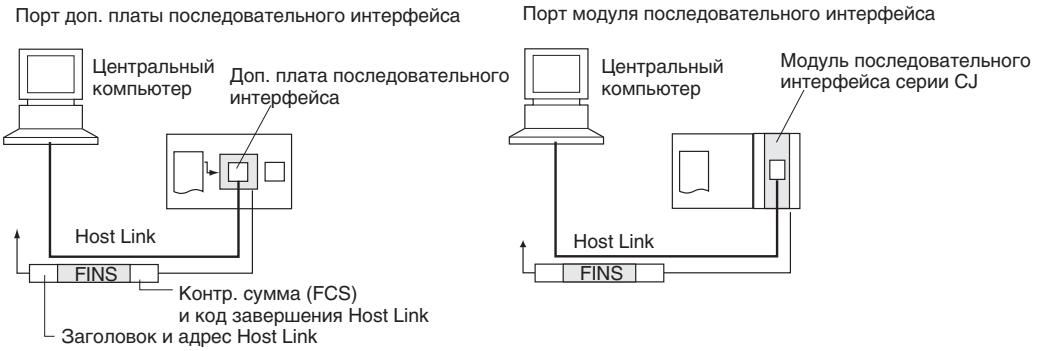
Для осуществления межсетевой связи в каждом модуле ЦПУ каждого ПЛК должна быть зарегистрирована так называемая таблица маршрутизации, описывающая путь доставки данных требуемому узлу. Каждая таблица маршрутизации, в свою очередь, состоит из таблицы локальных сетей и таблицы сетей ретрансляции.

1,2,3...

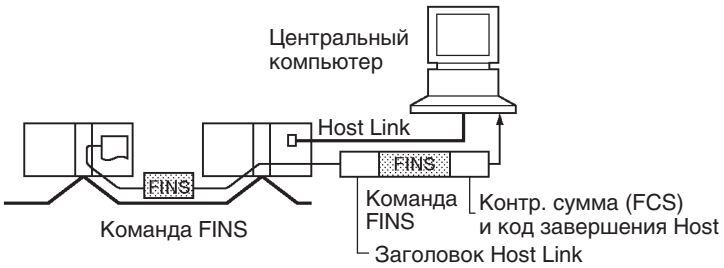
1. Таблица локальных сетей
   Эта таблица содержит номера модулей и адреса сетей для узлов, подключенных к локальному ПЛК.
2. Таблица сетей ретрансляции
   Эта таблица содержит адреса промежуточных сетей и номера узлов в этих сетях, через которые можно получить выход в конечную сеть, к которой локальный ПЛК непосредственно не подключен.

Связь с центральным компьютером (протокол Host Link)

В результате выполнения команды SEND(090), RECV(098) или CMND(490) на центральный компьютер передается команда FINS, дополненная необходимыми кодами заголовка и завершения протокола Host Link.



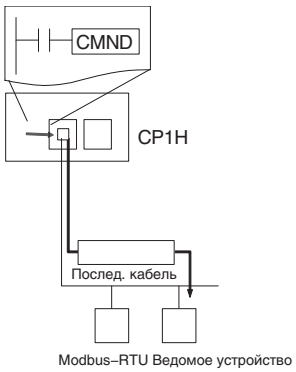
Примечание. Протокол связи Host Link поддерживает передачу данных по сети. В этом случае команда FINS передается по сети обычным образом. Когда команда FINS достигает системы Host Link, она дополняется необходимыми кодами заголовка и завершения протокола Host Link и в таком виде передается на центральный компьютер.



Связь с устройством или ведомым узлом Host Link в режиме шлюза последовательного интерфейса

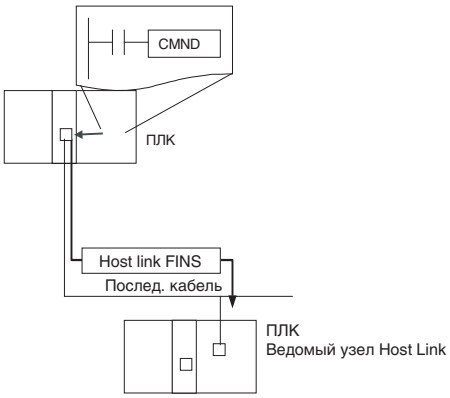
ПЛК может посылать команды FINS (или передавать/принимать данные) другому устройству или ведомому узлу Host Link через свой последовательный порт, работающий в режиме шлюза последовательного интерфейса (Serial Gateway).

- Передача команд FINS другому устройству
  Для последовательного порта, поддерживающего функцию шлюза последовательного интерфейса, выполняется команда CMND(490). Функция шлюза последовательного интерфейса преобразует команду протокола FINS в команду протокола CompoWay/F, Modbus-RTU или Modbus-ASCII.



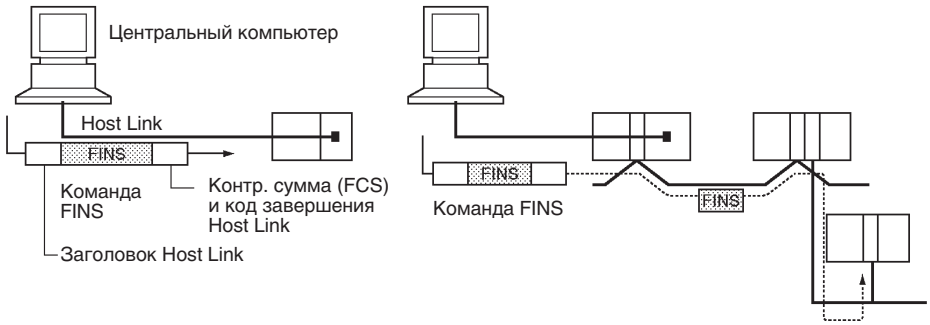
- Передача программируемому контроллеру с функциями ведомого узла Host Link

  Для последовательного порта, поддерживающего функцию шлюза последовательного интерфейса, выполняется команда CMND(490), SEND(090) или RECV(098). Функция шлюза последовательного интерфейса может передать любую команду FINS либо передать/принять данные.



Связь по инициативе центрального компьютера (протокол Host Link)

Центральный компьютер (компьютерная станция и т. п.) может посылать команды FINS программируемому контроллеру, к которому он подключен, а также другим устройствам, присутствующим в сети (модулям ЦПУ, спец. модулям вв./выв., компьютерам и т. п.). В этом случае передаваемая команда FINS должна дополняться необходимыми кодами заголовка и завершения протокола Host Link.

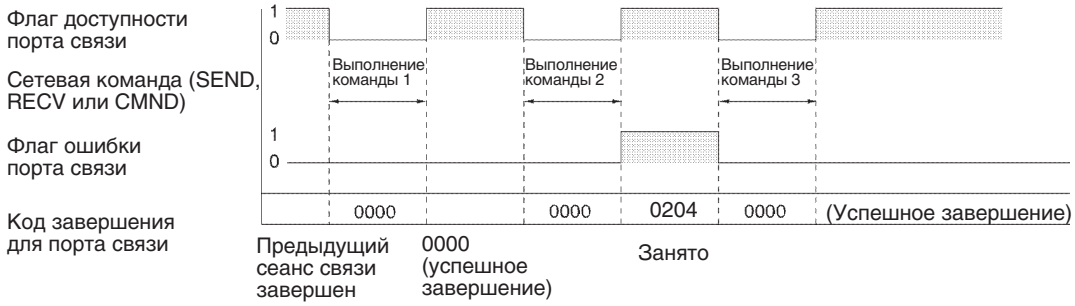


Коммуникационные флаги

Ниже кратко описана работа доступных коммуникационных флагов.

- Флаг доступности порта связи: находится в состоянии «0» во время сеанса связи, переходит в состояние «1», когда связь завершается (успешно или с ошибкой).
- Флаг ошибки порта связи: сохраняет свое состояние вплоть до следующей операции передачи или приема данных.

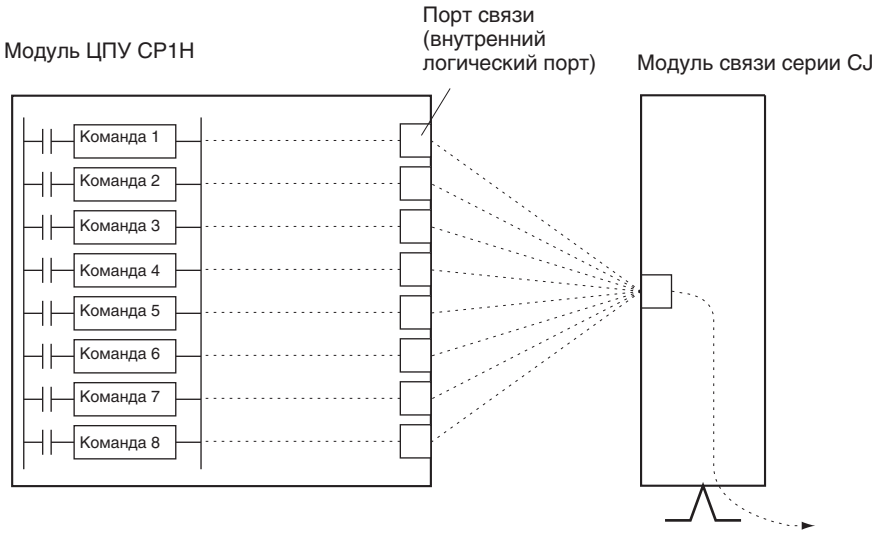
• Флаг ошибки порта связи: сбрасывается в состояние «0» в начале следующей операции передачи или приема данных, даже если предыдущая операция завершилась ошибкой.



О нумерации портов связи

Предусмотрено 8 логических портов связи, что позволяет выполнять одновременно до 8 команд связи. В то же время, для каждого из этих портов связи в любой момент времени может выполняться только одна команда. Если предполагается выполнение более восьми команд, в программе необходимо предусмотреть исключение одновременного выполнения избыточного числа команд.

Номера этих восьми логических портов связи используются совместно сетевыми командами (SEND(090), RECV(098) и CMND(490)), командами последовательного интерфейса (TXDU(256) и RXDU(255)) и командой МАКРОС ПРОТОКОЛА (PMCR(260)). Не следует указывать один и тот же номер порта для двух команд одновременно.

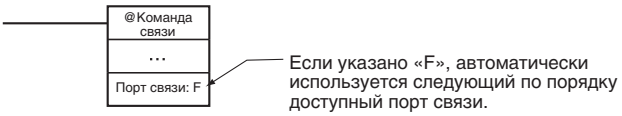


На следующем рисунке показан пример программы, исключающей одновременное выполнение команд связи.



Автоматическое назначение портов связи

Вместо числа 0...7 в качестве номера порта можно указать значение «F». В этом случае для команды автоматически назначается следующий по порядку доступный для использования порт связи.



Благодаря этой функции программисту не требуется помнить номера уже назначенных портов. Но даже при автоматическом распределении портов одновременно может использоваться не более 8 портов. Кроме того, для обращения к флагам и ответам, относящимся к конкретным портам связи, необходимо знать номера портов, для чего требуется использовать слова A218 (Номер используемого порта связи), а также A216 и A217 (Адрес хранения кода завершения связи по сети).

Разница между назначением номеров портов вручную и автоматическим распределением портов поясняется в следующей таблице.

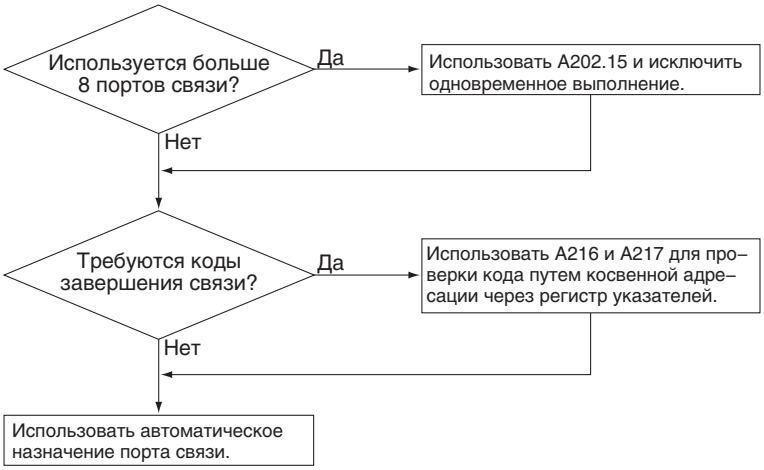
Параметр	Назначение конкретного номера	Автоматическое назначение
Значение номера порта связи, указываемое в управляющих данных.	0...7	F
Исключение одновременного выполнения команд	Требуется	Не требуется, если не требуется одновременно использовать больше 8 портов связи.

Параметр	Назначение конкретного номера	Автоматическое назначение
Использование флагов	Используется команда LD или LD NOT с флагом, который соответствует указанному порту связи.	Используется команда TST(350) или TSTN(351) со словом A218 (Номер используемого порта связи).
Коды завершения связи по сети	Считывается код завершения, соответствующий порту связи, указанному пользователем.	Для чтения кода завершения используется адрес в памяти ввода/вывода, содержащийся в словах A216 и A217 (Адрес хранения кода завершения связи по сети), и косвенная адресация с помощью регистра указателя.

Биты и слова вспомогательной области, используемые при автоматическом назначении портов связи

Адрес	Биты	Название	Описание
A202	15	Флаг наличия доступного для назначения порта связи	Включен, если имеется порт связи, доступный для автоматического назначения. Этот флаг можно использовать перед выполнением команды связи, чтобы проверить, не заняты ли в данный момент все 8 портов связи.
A214	00...07	Флаги первого цикла после завершения связи по сети	Каждый флаг включается на время одного цикла после завершения обмена данными. Биты 00...07 соответствуют портам связи 0...7. Для определения требуемого флага используйте слово A218, содержащее номер используемого порта связи. **Примечание.** Данные флаги не действуют, пока не начинается следующий цикл после выполнения команды связи. Обращайтесь к ним по истечении как минимум одного цикла.
	08...15	Не следует использовать.	
A215	00...07	Флаги первого цикла после ошибки связи по сети	Каждый флаг включается на время одного цикла после возникновения ошибки связи. Биты 00...07 соответствуют портам связи 0...7. Для определения требуемого флага используйте слово A218, содержащее номер используемого порта связи. **Примечание.** Данные флаги не действуют, пока не начинается следующий цикл после выполнения команды связи. Обращайтесь к ним по истечении как минимум одного цикла.
	08...15	Не следует использовать.	
A216 и A217	---	Адрес хранения кода завершения связи по сети	Код завершения команды связи автоматически сохраняется по адресу памяти ввода/вывода, указанному в этих словах. Для обращения к коду завершения связи следует скопировать этот адрес в регистр указателя, а затем использовать косвенную адресацию через регистр указателя.
A218	---	Номер используемого порта связи	Когда в модуле ЦПУ выполняется команда связи, в данное слово записывается номер порта связи, который был использован для этой команды. Портам связи 0...7 соответствуют значения 0000...0007 hex.

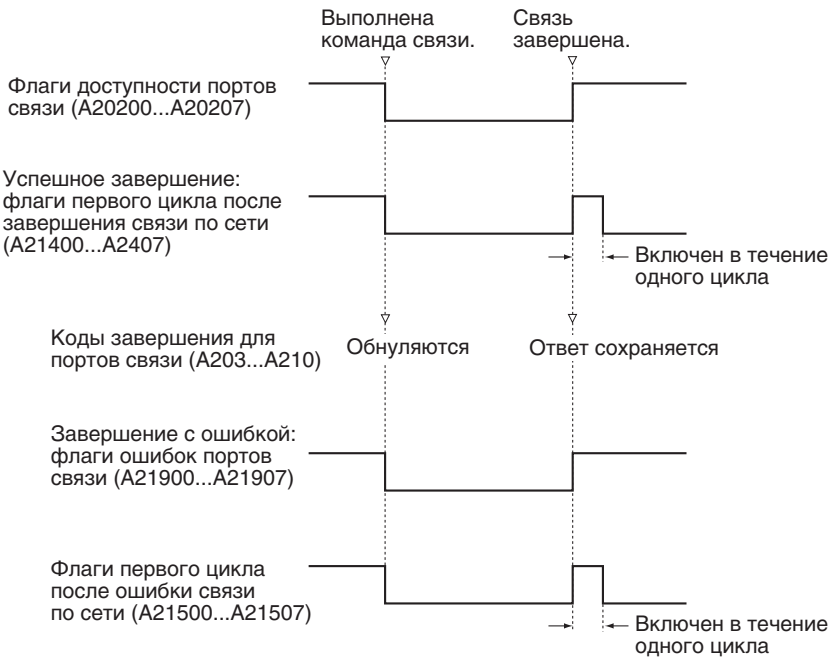
Примечание. (1) Для принятия решения о необходимости использования флага наличия доступного для назначения порта связи (A202.15) и слов адреса хранения кода завершения связи по сети (A216 и A217) можно воспользоваться следующей блок-схемой.



(2) В следующей таблице перечислены биты и слова вспомогательной области, используемые в том случае, когда номера портов связи указываются самим пользователем.

Адрес	Биты	Название	Описание
A202	00...07	Флаги доступности портов связи	Включен, если порт с соответствующим номером доступен для выполнения команды связи. Биты 00...07 соответствуют портам связи 0...7. Включение данного флага свидетельствует о том, что выполнение команды связи завершилось. С началом выполнения следующей команды связи этот флаг вновь сбрасывается.
A203... A210	---	Коды завершения для портов связи	После выполнения команд связи эти слова содержат коды завершения для портов связи с соответствующими номерами. Слова A203...A210 соответствуют портам связи 0...7.
A219	00...07	Флаги ошибок портов связи	Включен, если при выполнении команды связи произошла ошибка. Если этот флаг включен, для выяснения причины ошибки следует прочитать код завершения в словах A203...A210. Выключен, если команда выполнена успешно. Биты 00...07 соответствуют портам связи 0...7.

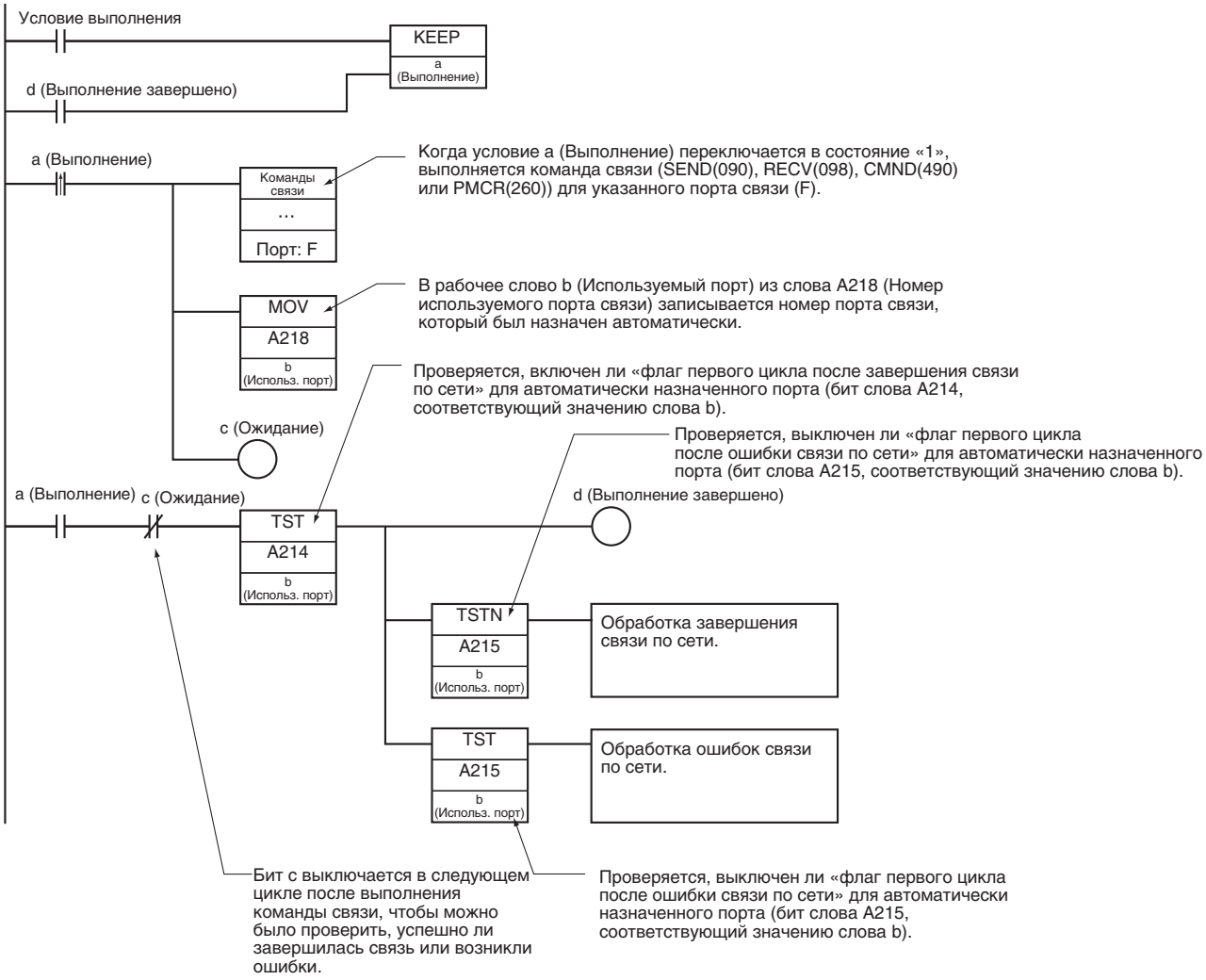
Функционирование флагов и слов



Примеры применения

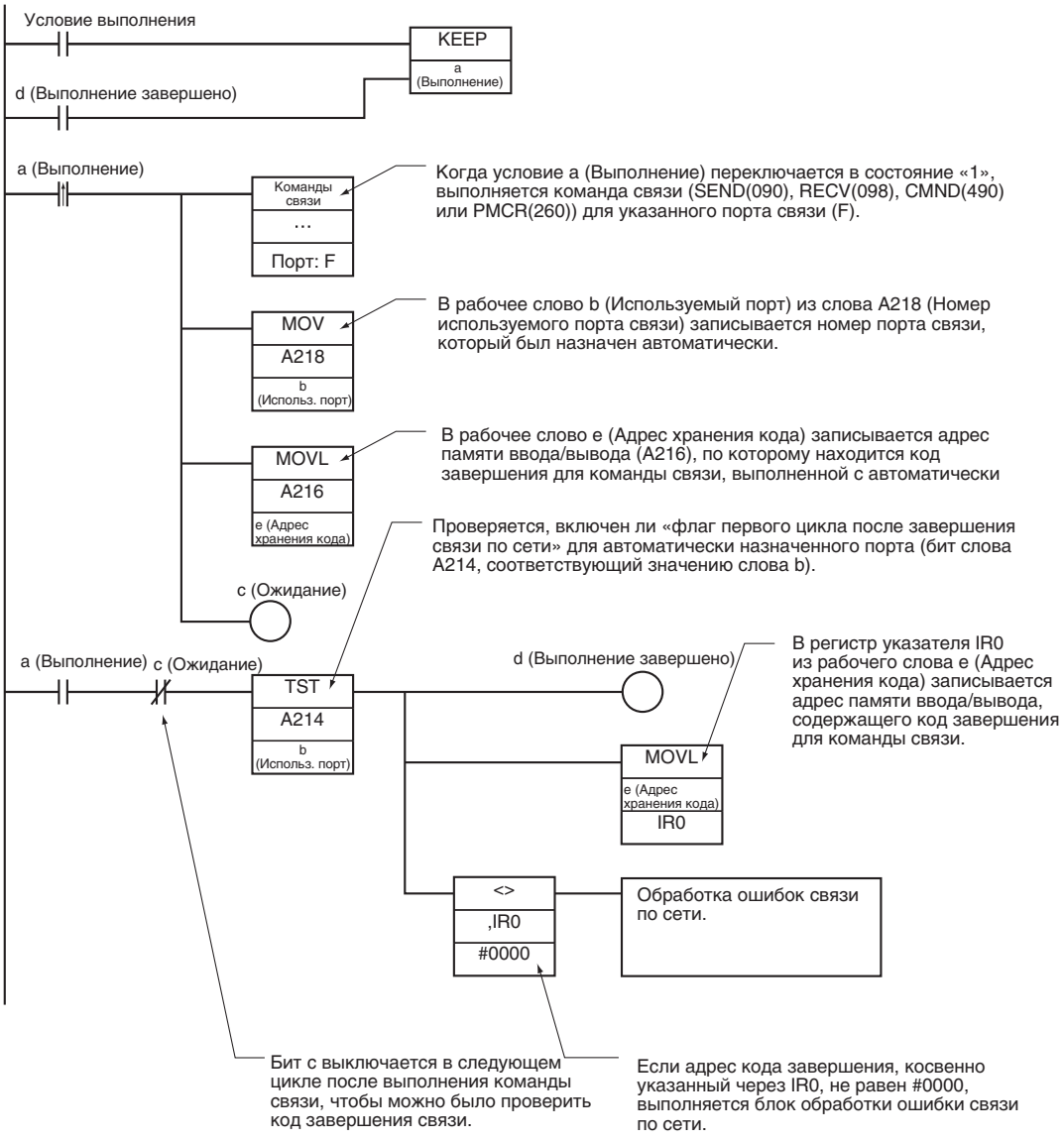
Для использования функции автоматического назначения портов связи в качестве номера порта связи следует указать «F» и составить следующую программу.

Завершение работы и обработка ошибок после выполнения команд связи



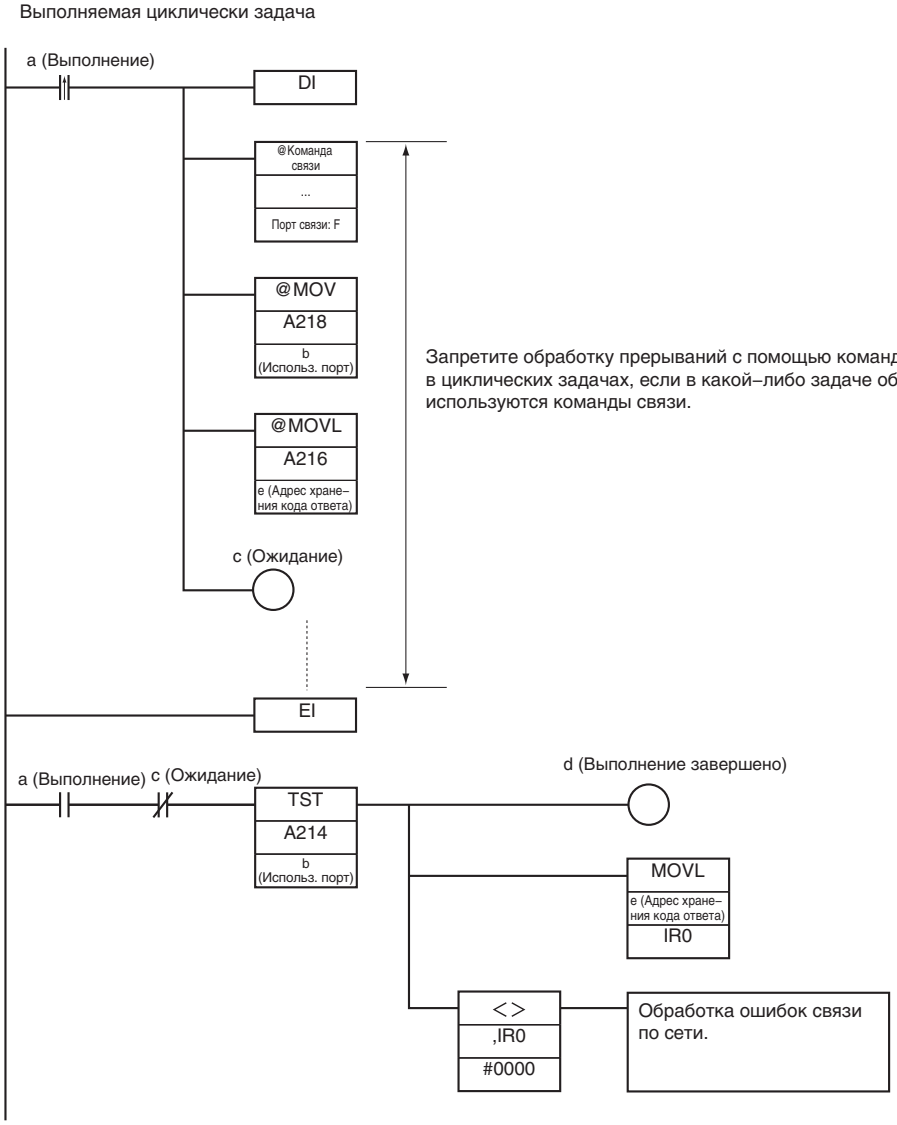
Обращение к коду завершения после выполнения команды связи

Коды завершения, как правило, используются для выяснения причин ошибок в случае возникновения последних. Код завершения 0000 hex, с другой стороны, позволяет убедиться в том, что обмен данными завершился успешно.



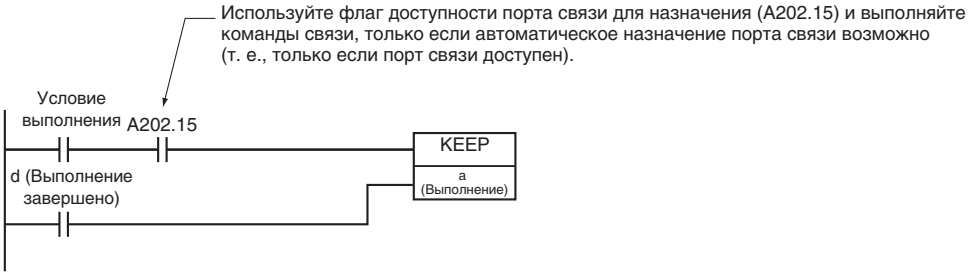
Использование команд связи в задачах обработки прерываний

В тех случаях, когда в задачах обработки прерываний используются команды связи (с автоматическим или ручным назначением номеров портов), в циклических задачах команды связи, использующие автоматическое распределение портов связи, следует располагать между командами DI(693) и EI(694), запрещающими обработку прерываний во время выполнения этих команд связи. Пример программы приведен ниже.



Предотвращение превышения максимального числа портов связи

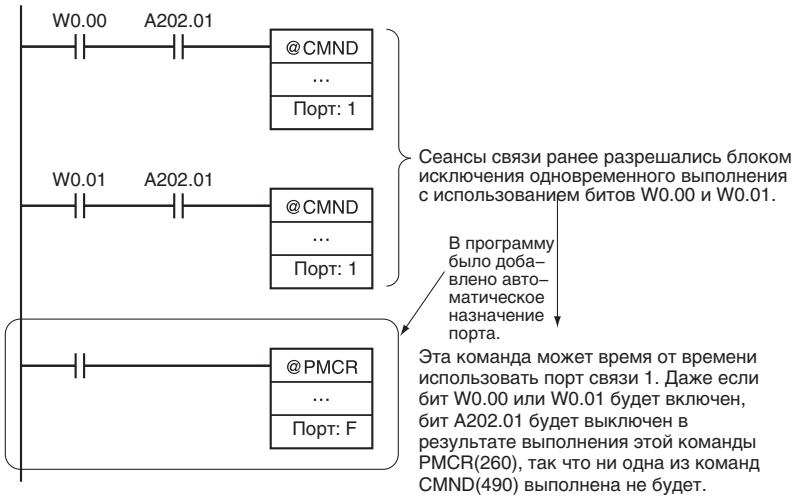
Если существует вероятность одновременного выполнения более восьми команд связи, следует использовать приведенную ниже программу, проверяющую доступность портов связи даже в том случае, когда номера портов связи назначаются модулем ЦПУ автоматически.



Указание номеров портов вручную одновременно с автоматическим назначением портов

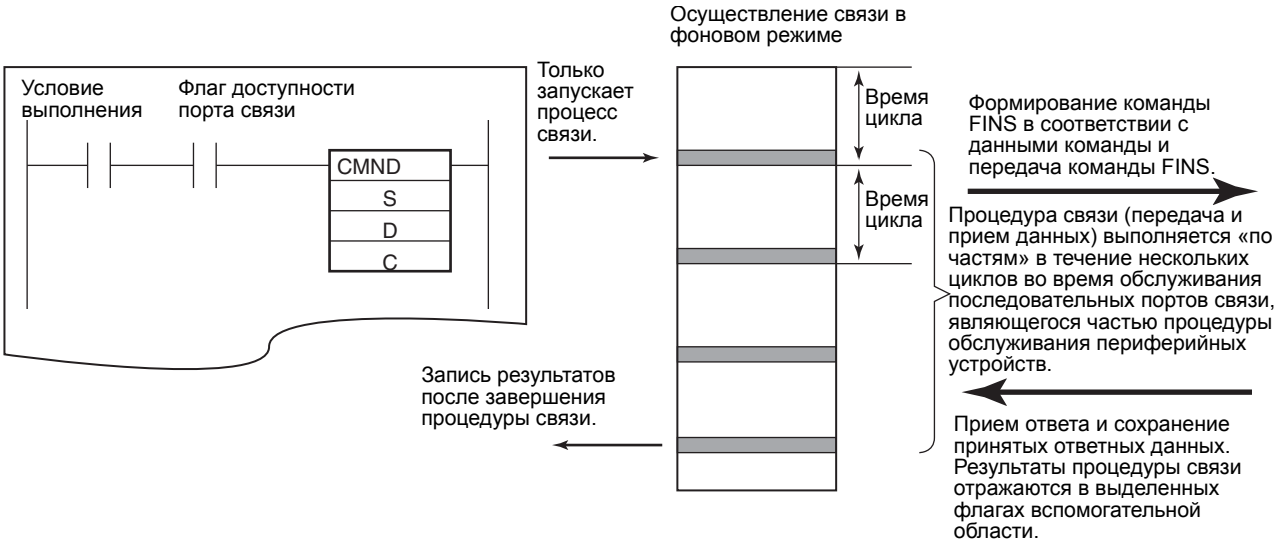
В пределах одной программы можно одновременно назначать номера портов связи вручную и использовать автоматическое назначение. Может, однако, получиться так, что номера портов связи, указанные пользователем, будут использованы функцией автоматического назначения. Поэтому рекомендуется очень внимательно проверить программу, прежде чем добавлять в нее команды связи, использующие автоматическое назначение портов связи.

Пример программы



Время выполнения сетевых команд

Сетевая команда, выполняемая при наступлении условия выполнения, всего лишь запускает процедуру обмена данными. Далее процедура обмена данными продолжает выполняться в фоновом режиме, как часть процедуры обслуживания периферийных устройств (во время обслуживания последовательных портов связи).



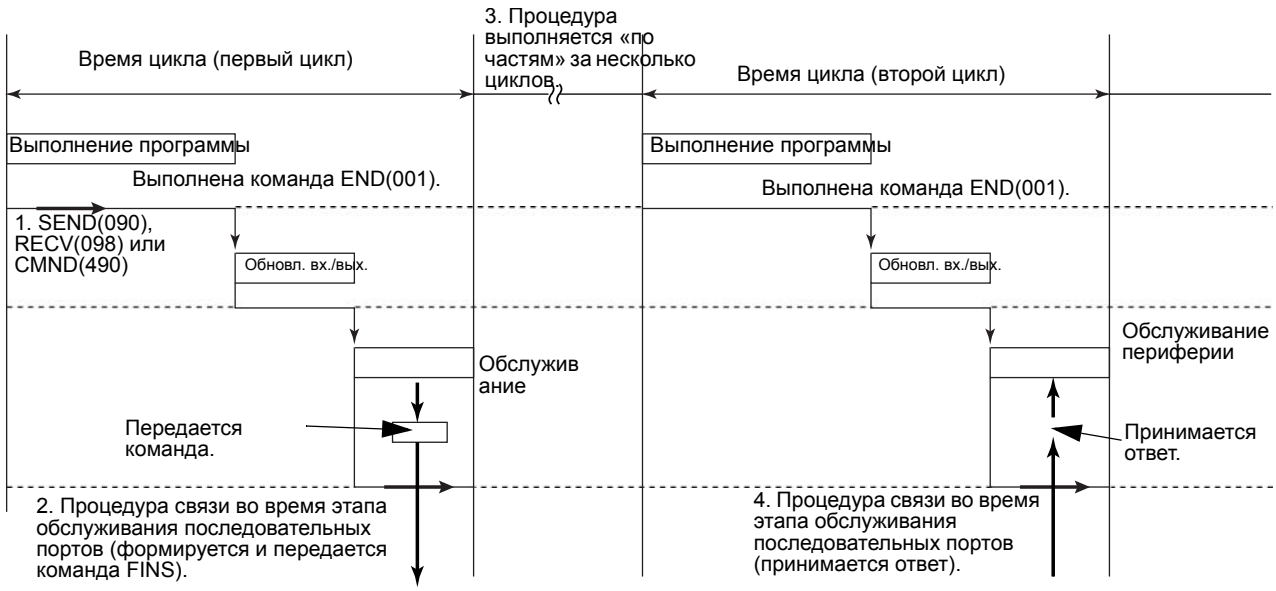
Процедура связи выполняется следующим образом:

1. Если в момент наступления условия выполнения команды связи включен соответствующий флаг доступности порта связи (A202.00...A202.07), система производит следующие действия:
   - Сбрасывает флаг доступности порта связи и флаг ошибки порта связи для используемого порта (A219.00...A219.07).
   - Обнуляет код завершения порта связи (A203...A210) для используемого порта.
   - Считывает содержимое управляющих слов (начиная с C) и запускает процедуру связи (т. е. передает команду FINS или принимает ответ).
2. На этапе обслуживания последовательных портов связи в рамках процедуры обслуживания периферийных устройств в текущем цикле система формирует команду FINS на основании значений операндов (см. примечание) и передает эту команду модулю связи или другому адресуемому узлу.

Примечание. При выполнении команды SEND(090) считывается содержимое операндов S и D, и формируется команда FINS для передачи данных.
При выполнении команды RECV(098) считывается содержимое операнда S, и формируется команда FINS для приема данных.
При выполнении команды CMND(490) считывается содержимое операнда S, и формируется соответствующая команда FINS.

3. Если процедура передачи данных не может быть выполнена целиком за время, выделяемое на обслуживание последовательных портов связи, она продолжает выполняться при следующем обслуживании последовательных портов связи в следующем цикле.
4. Когда адресуемое устройство возвращает ответ, система выполняет следующие действия:

- Записывает актуальные ответные данные в слова назначения, указанные в сетевой команде.
- Устанавливает флаг доступности порта связи для используемого порта.
- Обновляет состояние флага ошибки порта связи (A219.00...A219.07) и содержимое кода завершения порта связи (A203...A210) для используемого порта.



3-24-2 Сведения о командах передачи явных сообщений (только для CP1H)

Способы передачи явных сообщений

Для отправки явных сообщений из программируемого контроллера можно использовать один из следующих способов.

- Использовать команду CMND(490) для отправки команды FINS с кодом 2801 hex (ПЕРЕДАЧА ЯВНОГО СООБЩЕНИЯ).
- Использовать команды для передачи явных сообщений, перечисленные в следующей таблице.

Команды для передачи явных сообщений

В следующей таблице перечислены команды, предназначенные для передачи явных сообщений.

Команда	Название	Принцип работы
EXPLT(720)	ОТПРАВИТЬ ЯВНОЕ СООБЩЕНИЕ	Передает явное сообщение с любым кодом службы. Прим.: действие этой команды эквивалентно использованию CMND(490) для отправки команды FINS с кодом 2801 hex.
EGATR(721)	ЯВНЫЙ ЗАПРОС АТРИБУТА	Передает явное сообщение с кодом службы 0E hex (ЗАПРОС ОДИНОЧНОГО АТРИБУТА).
ESATR(721)	ЯВНАЯ УСТАНОВКА АТРИБУТА	Передает явное сообщение с кодом службы 10 hex (УСТАНОВКА ОДИНОЧНОГО АТРИБУТА).
EGATR(721)	ЯВНОЕ ЧТЕНИЕ СЛОВА	Использует явное сообщение для чтения данных из модуля ЦПУ.
EGATR(721)	ЯВНАЯ ЗАПИСЬ СЛОВА	Использует явное сообщение для записи данных в модуль ЦПУ.

Свойства команд для передачи явных сообщений

- В командах передачи явных сообщений не требуется указывать код команды FINS 2801 hex; их намного проще использовать в программе, чем команду CMND(490).
- Для команд ЯВНЫЙ ЗАПРОС/ЯВНАЯ УСТАНОВКА АТРИБУТА не требуется вводить код службы, требуются только данные, начиная с идентификатора класса.
- Для команд ЯВНОЕ ЧТЕНИЕ/ЗАПИСЬ СЛОВА может быть указан непосредственно адрес памяти ввода/вывода в локальном и удаленном модулях ЦПУ.

  Не требуется указывать коды для типов областей и шестнадцатеричных адресов слов. (Это необходимо для команд CMND(490) с кодом службы 1E (чтение слова) или 1F hex (запись значения слова)).

  Это позволяет легко реализовать обмен данными между модулями ЦПУ путем использования явных сообщений (как команд SEND/RECV для команд FINS).

Действие

Об успешном или неуспешном завершении обмена данными информирует флаг ошибки обмена явными сообщениями.

В случае завершения с ошибкой (т. е. когда флаг включен) соответствующий флаг ошибки порта связи для команды FINS позволяет определить, что именно произошло: явное сообщение не было передано вообще (флаг включен) или сообщение было передано, но содержало ошибку (флаг выключен).

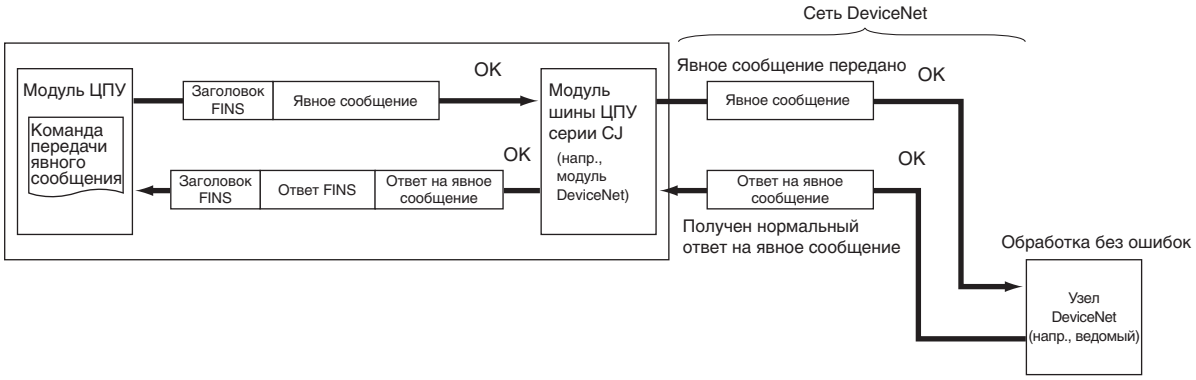
При успешном завершении соответствующий код завершения для порта связи будет иметь значение 0000 hex, а в случае ошибки в нем будет содержаться код ошибки явного сообщения (при ошибке передачи явного сообщения) либо код ошибки FINS (при возникновении ошибки FINS).

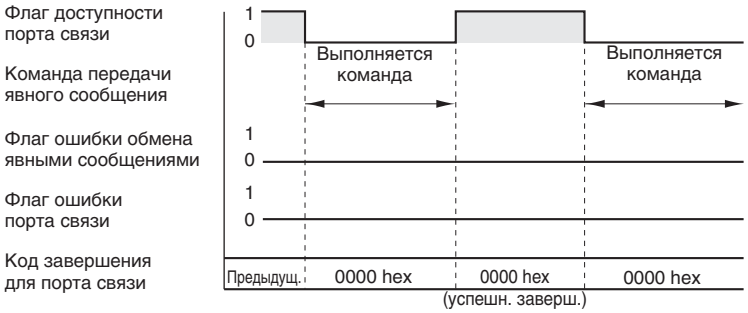
Условие		Флаг ошибки обмена явными сообщениями (A213.00...A213.07: порт связи номер 0...7)	Флаг ошибки порта связи (A219.00...A219.07: порт связи номер 0...7)	Код завершения для порта связи (A203...A210: порт связи номер 0...7)
1) Успешное завершение		ВЫКЛ	ВЫКЛ	0000 hex
2) Завершение с ошибкой	a) Не удалось передать явное сообщение.	ВКЛ	ВКЛ	Код завершения для сообщения FINS
	b) Явное сообщение было передано, но вернулся ответ с кодом ошибки явного сообщения.		ВЫКЛ	Код ошибки явного сообщения

1) Успешное завершение

Явное сообщение передано, возвращен ответ без кода ошибки.

Соответствующий флаг ошибки обмена явными сообщениями (A213.00...A213.07: порт связи 0...7) будет выключен, а код завершения для порта связи (A203...A210: порт связи 0...7) будет содержать код нормального ответа для явного сообщения (0000 hex).





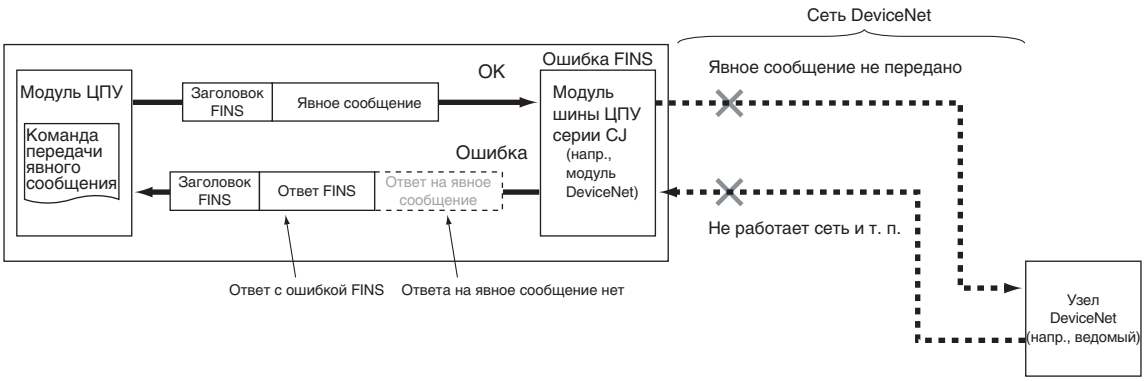
2) Завершение с ошибкой

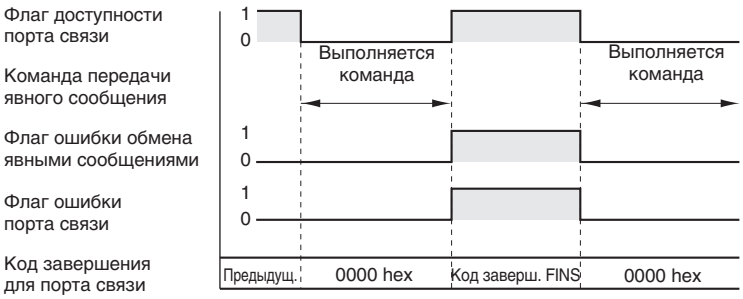
Существуют два возможных варианта завершения передачи явного сообщения с ошибкой, которые рассмотрены ниже.

а) Модулю ЦПУ не удалось передать явное сообщение.

Передача явного сообщения по сети не состоялась (например, из-за неисправности сети). В этом случае будут установлены оба флага: флаг ошибки обмена явными сообщениями (A213.00...A213.07: порт связи 0...7) и флаг ошибки порта связи (A219.00...A219.07: порт связи 0...7).

По завершении процедуры связи код завершения порта связи (A203...A210: порт связи 0...7) будет содержать код ошибки сообщения FINS.

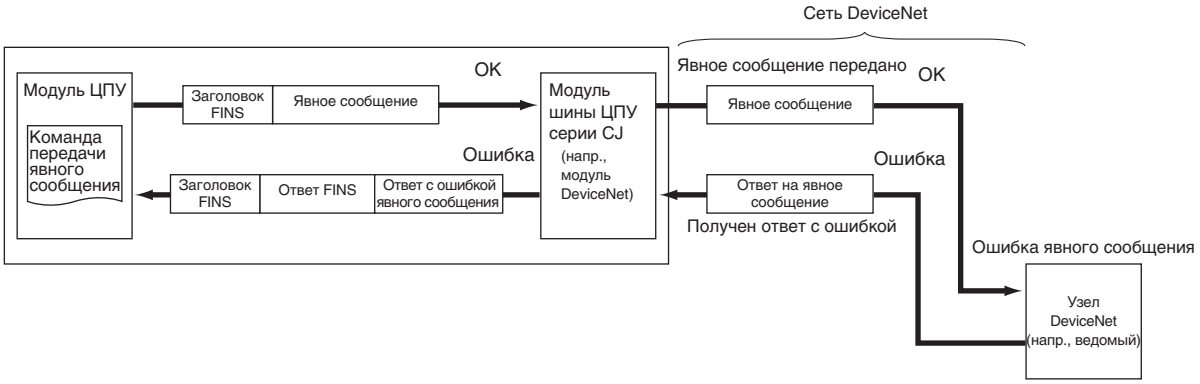


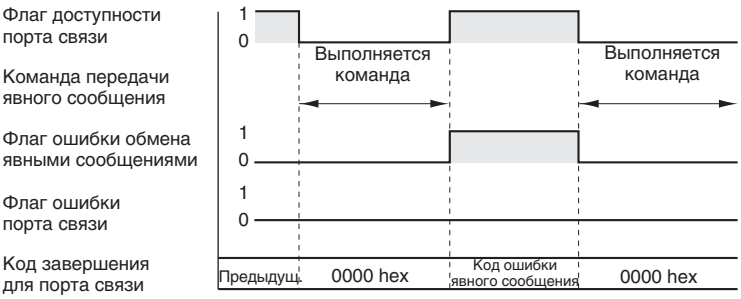


b) Явное сообщение было передано, но был возвращен ответ с кодом ошибки.

Явное сообщение было передано, но кадр команды явного сообщения содержал ошибку (код не поддерживается, недопустимый размер и т. п.). В этом случае флаг ошибки обмена явными сообщениями (A213.00...A213.07: порт связи 0...7) будет включен, а флаг ошибки порта связи (A219.00...A219.07: порт связи 0...7) останется выключенным.

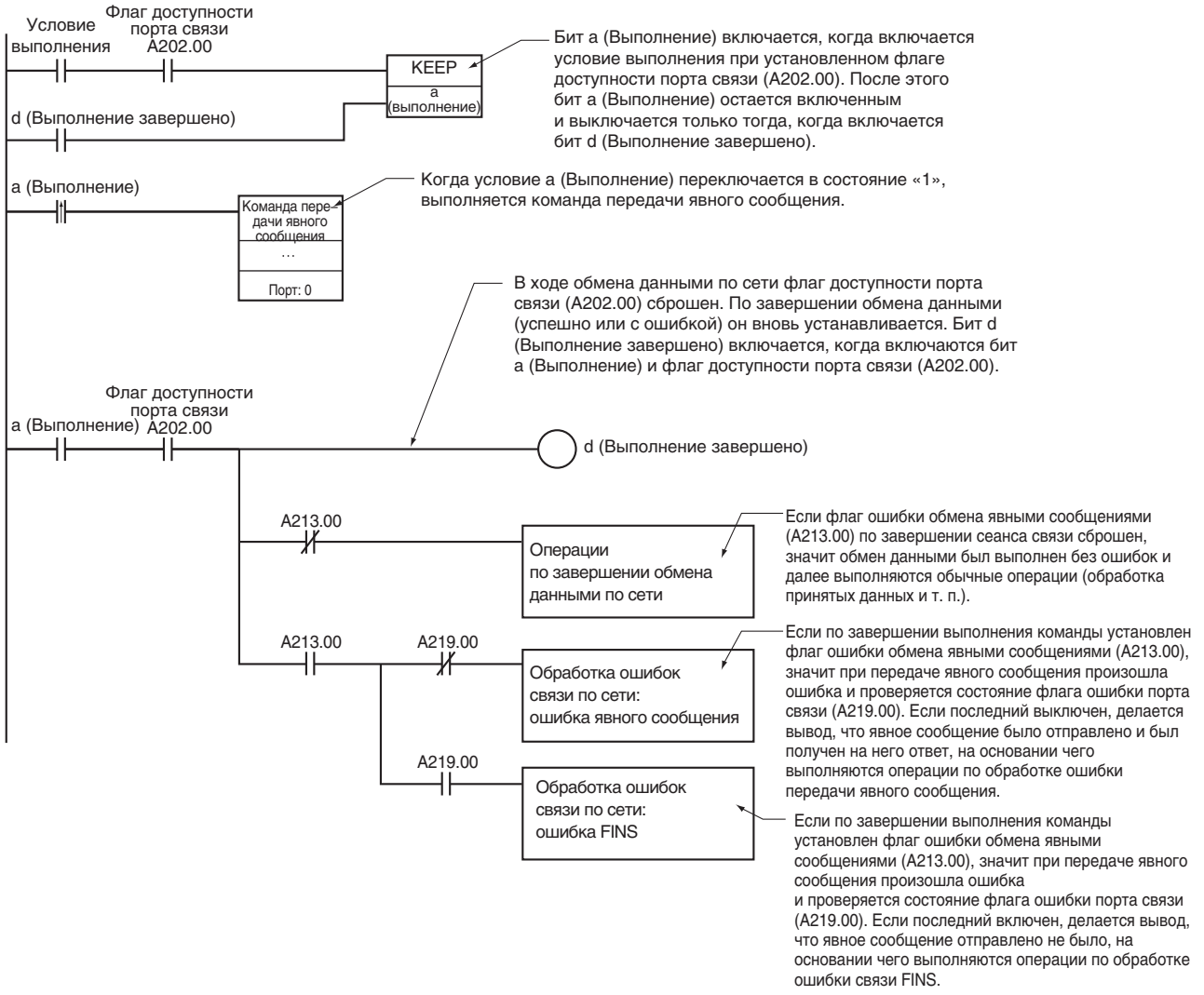
По завершении процедуры связи слово кода ответа порта связи (A203...A210: порт связи 0...7) будет содержать код ошибки явного сообщения.



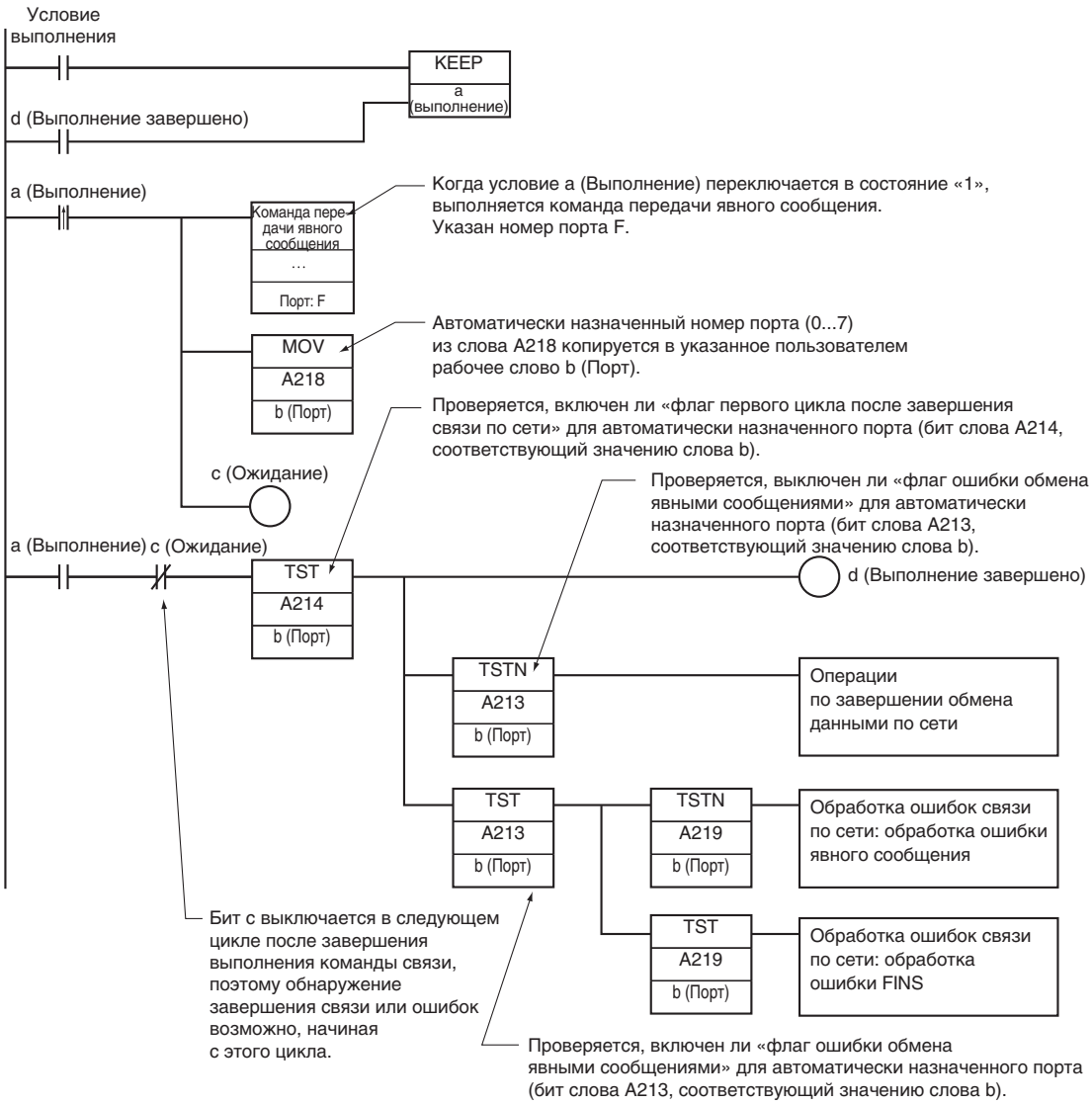


Пример создания программы

Пример 1: указание номера порта связи вручную пользователем.



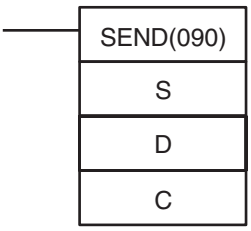
Пример 2: автоматическое назначение номера порта связи.



3-24-3 ПЕРЕДАТЬ ПО СЕТИ: SEND(090)

Назначение Передача данных узлу в сети.

Символ РКС



S: Первое исходное слово (локальный узел)

D: Первое слово назначения (удаленный узел)

C: Первое управляющее слово

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	SEND(090)
	Однократное выполнение по положительному фронту	@SEND(090)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

C: Первое управляющее слово

Пять управляющих слов (C...C+4) содержат сведения о количестве передаваемых слов, о получателе данных и о других параметрах, перечисленных в следующей таблице.

Слово	Биты 00...07	Биты 08...15
C	Количество слов: от 0001 до максимально допустимого[1] (4-разр., hex)	
C+1	Адрес сети назначения: 00...7F (0...127)[2, 4]	Биты 08...11: номер послед. порта[3] (физического порта) 1 hex: порт 1 2 hex: порт 2 (Не задавать 0, 3 или 4.) Биты 12...15: всегда 0
C+2	Адрес модуля назначения: 00...FE[5]	Адрес узла назначения: от 00 до максимально допустимого[6]
C+3	Число повторных попыток: 00...0F (0...15)	Биты 08...11: номер порта связи (внутреннего логического порта): 0...7, автоматическое назначение: F[7] Биты 12...15: требование ответа 0: ответ требуется 8: ответ не требуется[8]
C+4	Время ожидания ответа: 0001...FFFF hex (0,1...6553,5 с) (При значении по умолчанию 0000 ответ ожидается в течение 2 секунд.)	

Примечание.

(1) Максимально допустимое количество слов зависит от типа используемой сети. Допустимый диапазон для Controller Link: 0001...03DE (от 1 до 990 слов).

(2) Для передачи в пределах локальной сети задайте адрес сети назначения равным 00. При наличии двух или большего числа модулей шины ЦПУ адресом сети является наименьший из номеров этих модулей.

(3) Для передачи данных по инициативе ПЛК центральному компьютеру через последовательный порт в режиме Host Link можно использовать два способа, которые описаны ниже.

(a) Указать в качестве адреса модуля назначения (биты 00...07 слова C+2) адрес модуля ЦПУ или модуля последовательного интерфейса серии CJ и задать номер

последовательного порта (биты 08...11 слова C+1) равным 1 (для порта 1) или 2 (для порта 2).

Адрес модуля (C+2, биты 00...07)	Модуль	Номер послед. порта (C+1, биты 08...11)	Последо-вательный порт
00 hex	CP1H с установленной доп. платой послед. интерфейса	1 hex	Порт 1
		2 hex	Порт 2
10 hex + номер модуля	Модуль послед. интерфейса серии CJ (модуль шины ЦПУ)	1 hex	Порт 1
		2 hex	Порт 2

(b) Ввести в биты 00...07 слова C+2 непосредственно адрес модуля назначения. В этом случае в битах 08...11 слова C+1 должен быть указан номер последовательного порта 0.

Порты доп. платы последовательного интерфейса модуля ЦПУ CP1H и CP1L M (30 или 40 вх./вых.)

Порт	Адрес модуля порта
Порт 1	FD hex (253 десятичн.)
Порт 2	FC hex (252 десят.)

Порты доп. платы последовательного интерфейса модуля ЦПУ CP1L L (14 или 20 вх./вых.)

Порт	Адрес модуля порта
Порт 1	FC hex (252 десят.)

Порты модуля последовательного интерфейса серии CJ

Порт	Адрес модуля порта	Пример: номер модуля = 1
Порт 1	80 hex + 4 × номер модуля	80 + 4 × 1 = 84 hex (132 десят.)
Порт 2	81 hex + 4 × номер модуля	81 + 4 × 1 = 85 hex (133 десят.)

(4) Если последовательный порт для функции шлюза последовательного интерфейса (выполняющей преобразование в Host Link FINS) указывается без таблицы маршрутизации, в байте адреса сети назначения следует задать адрес модуля последовательного порта.

(5) Адрес модуля обозначает модуль в соответствии со следующей таблицей.

Модуль	Значение адреса модуля
Модуль ЦПУ CP1H (доп. плата послед. интерфейса) или модуль ЦПУ серии CJ	00 hex
Модуль шины ЦПУ серии CS/CJ	10 hex + номер модуля
Спец. модуль вв./выв. серии CS/CJ	20 hex + номер модуля
Встраиваемая плата серии CS	E1 hex
Компьютер	01 hex

Модуль	Значение адреса модуля
Модуль, подключенный к сети (модуль указывать необязательно)	FE hex
Непосредственное указание адреса модуля последовательного порта	Порты доп. платы последовательного интерфейса модуля ЦПУ CP1H или CP1L M (30 или 40 вх./вых.) Порт 1: FD hex (253 десят.) Порт 2: FC hex (252 десят.) Порт доп. платы последовательного интерфейса модуля ЦПУ CP1L L (14 или 20 вх./вых.) Порт 1: FC hex (252 десят.) Порты модуля последовательного интерфейса серии CS/CJ Порт 1: 80 hex + 4 × номер модуля Порт 2: 81 hex + 4 × номер модуля Порты платы последовательного интерфейса серии CS Порт 1: E4 hex (228 десят.) Порт 2: E5 hex (229 десят.) Порты модуля ЦПУ серии CS/CJ Периферийный порт: FD hex (253 десят.) Порт RS-232C: FC hex (252 десят.)

(6) Максимально допустимый номер узла зависит от типа используемой сети. Допустимый диапазон для Controller Link: 00...20 hex (от 0 до 32 узлов). Для передачи одновременно всем узлам (широковещание) введите для номера узла назначения значение FF; для передачи в пределах локального узла введите значение 00.

(7) Более подробно функция автоматического назначения номера порта связи (логического порта) описана в разделе *Автоматическое назначение портов связи* на стр. 931.

(8) Если в качестве номера узла назначения введено значение FF (широковещание), ответ на передачу не возвращается, даже если в биты 12...15 введено 0.

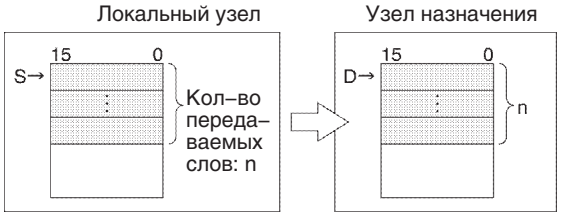
Характеристики операндов

Область	S	D	C
Область CIO	CIO 0...CIO 6143		CIO 0...CIO 6139
Рабочая область	W0...W511		W0...W507
Область битов хранения	H0...H511		H0...H507
Область вспомогательных битов	A0...A959		A0...A955
Область таймеров	T0000...T4095		T0000...T4091
Область счетчиков	C0000...C4095		C0000...C4091
Область DM	D0...D32767		D0...D32763
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767		
Постоянные	---		
Регистры данных	---		

Область	S	D	C
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда SEND(090) передает данные, начиная со слова S, в ячейки памяти адресуемого устройства, начиная с адреса D, по внутренней шине ЦПУ ПЛК или по сети. Количество передаваемых слов указывается операндом C.



Если в качестве номера узла назначения указывается «FF», данные передаются одновременно всем узлам адресуемой сети. Такой режим передачи называют «широковещанием».

Если затребован ответ (биты 12...15 слова C+3 = 0), но ответ не поступает в течение заданного времени ожидания ответа, данные передаются повторно до 15 раз (число повторных попыток задается битами 0...3 слова C+3). В случае широковещательной передачи ответ не возвращается и повторные попытки передачи не предпринимаются.

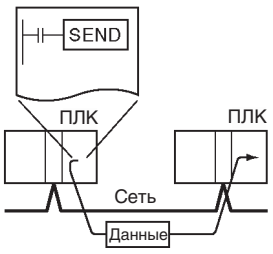
Командой SEND(090) можно воспользоваться для передачи данных конкретному последовательному порту адресуемого устройства, а также самому устройству.

С помощью команды SEND(090) можно передавать данные центральному компьютеру, подключенному к последовательному порту ПЛК (для которого выбран режим Host Link), а также другому ПЛК или компьютеру по сети Controller Link или Ethernet.

Если при выполнении SEND(090) будет включен флаг доступности порта связи для порта связи, указанного в C+3, соответствующий флаг доступности порта связи (порты 00...07: A202.00...A202.07) и соответствующий флаг ошибки порта связи (порты 00...07: A219.00...A219.07) будут сброшены, а в слово кода завершения (порты 00...07: A203...A210) будет записано значение 0000. По завершении передачи данных адресуемому узлу флаг доступности порта связи будет вновь установлен, а состояние флага ошибки порта связи и содержимое слова кода завершения будут отражать результат выполнения команды.

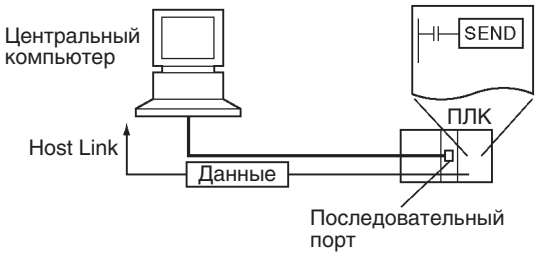
Передача на ПЛК или компьютер по сети (только CP1H)

С помощью команды SEND(090) можно осуществить передачу данных из ПЛК CP1H в указанную область данных другого ПЛК или компьютера, подключенного к сети Controller Link или Ethernet.



Передача на центральный компьютер через Host Link

Если модуль ЦПУ серии CP оборудован дополнительной платой последовательного интерфейса или модулем последовательного интерфейса серии CJ и центральный компьютер подключен непосредственно к последовательному порту этой платы/модуля (последовательный порт должен быть переведен в режим Host Link), команду SEND(090) можно использовать для передачи данных из ПЛК на центральный компьютер (после выполнения SEND(090) данные будут переданы, когда ПЛК в очередной раз получит право на передачу). Также возможна передача данных на другие центральные компьютеры, подключенные к другим ПЛК в сети.



Флаги

Название	Обо-значение	Действие
Флаг ошибки	ER	Включен, если номер последовательного порта, указанный в C+1, выходит за диапазон 00...04. Включен, если для порта связи, номер которого указан в C+3, выключен флаг доступности порта связи. Выключен во всех остальных случаях.

В следующей таблице перечислены сопутствующие биты и флаги вспомогательной области.

Название	Адрес	Действие
Флаг доступности порта связи	A202.00... A202.07	Этот флаг устанавливается в знак того, что для порта связи с соответствующим номером (00...07) возможно выполнение сетевых команд, включая PMCR(260). Во время выполнения сетевой команды с использованием соответствующего порта флаг выключен. После того как команда выполнена, флаг включается вновь.
Флаг ошибки порта связи	A219.00... A219.07	Этот флаг устанавливается в знак того, что при выполнении сетевой команды с использованием порта связи с соответствующим номером (00...07) произошла ошибка. Состояние флага сохраняется вплоть до выполнения следующей сетевой команды. При выполнении следующей сетевой команды флаг сбрасывается, даже если ранее произошла ошибка.
Коды завершения для портов связи	A203... A210	После выполнения команды сетевого обмена данными в этих словах содержатся коды завершения для соответствующих портов (00...07). Непосредственно во время выполнения сетевой команды соответствующее слово содержит значение 0000, по завершении выполнения команды в это слово записывается код завершения. При выполнении следующей команды эти слова вновь обнуляются.

Меры предосторожности

Если для порта, номер которого указан в C+3, выключен флаг доступности порта, команда воспринимается как NOP(000) и не выполняется. В этом случае будет установлен флаг ошибки.

Если для D указан адрес в текущем банке области EM, передаваемые данные записываются в текущий банк EM адресуемого узла.

Если данные будут передаваться за пределы локальной сети, пользователь должен зарегистрировать таблицы маршрутизации в ПЛК (модулях ЦПУ) в каждой сети. (В таблицах маршрутизации детализируются маршруты, по которым данные передаются адресуемым узлам в других сетях.)

Подробные сведения о кодах завершения сетевого обмена данными можно найти в описании ответных кодов команд FINS в руководстве *Серия CS/CJ/CP — Справочное руководство по командам связи* (W342).

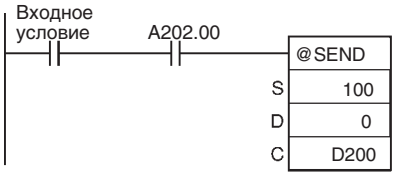
Для порта связи одновременно может выполняться только одна сетевая команда. Чтобы команда SEND(090) не оказалась выполненной, когда порт занят, в программе следует использовать флаг доступности порта связи (A202.00...A202.07) в качестве нормально разомкнутого условия.

Номера портов связи 00...07 используются не только сетевыми командами, но и командой PMCR(260), поэтому команда SEND(090) не должна выполняться одновременно с командой PMCR(260), если для обеих команд указан один и тот же номер порта связи.

Электромагнитные помехи и другие факторы могут приводить к искажению или полной утрате передаваемых или принимаемых пакетов данных, поэтому рекомендуется устанавливать ненулевое число повторных попыток передачи. В этом случае при отсутствии ответа в течение заданного времени ожидания команда SEND(090) будет выполнена еще раз.

Пример 1

Если входное условие и бит A202.00 (флаг доступности порта связи для порта 7) оба одновременно включены, десять слов (CIO 100...CIO 109) передается центральному компьютеру, который подключен к порту 1 модуля последовательного интерфейса серии CJ с адресом модуля 10 (hex) в составе узла номер 3 в сети 0.



C	D200	0 0 0 A	Кол–во передаваемых слов: 10 слов
C+1	D201	0 1 0 0	Передача в сеть 0, в порт 1 модуля послед. интерфейса
C+2	D202	0 0 1 0	Номер узла 0, адрес модуля 10
C+3	D203	0 0 0 0	Требуется ответ, номер порта 0, повторять не надо
C+4	D204	0 0 0 0	Время ожидания ответа: 2 с (0000: (знач. по умолч.)

На центральном компьютере также должна работать программа, принимающая данные и возвращающая ответы.

Пример 2

Если бит CIO 0.00 и бит A202.07 (флаг доступности порта связи для порта 07) оба включены, десять слов (D100...D109) передаются узлу под номером 3 в локальной сети, где они записываются в 10 слов (D200...D209). Если ответ не поступает в течение 10 секунд, предпринимается до 3 повторных попыток передачи данных.



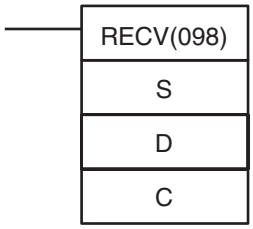
C: D300	0 0 : 0 A	Кол–во передаваемых слов: 10 слов
C+1: D301	0 0 : 0 0	Передача по локальной сети самому устройству
C+2: D302	0 3 : 0 0	Номер узла 3, адрес модуля 00 (модуль ЦПУ)
C+3: D303	0 7 : 0 3	Требуется ответ, номер порта 7, 3 попытки
C+4: D304	0 0 : 6 4	Время ожидания ответа: 0064 hex (10 секунд)

3-24-4 ПРИНЯТЬ ПО СЕТИ: RECV(098)

Назначение

Эта команда запрашивает данные у узла сети и принимает эти данные.

Символ РКС



S: Первое исходное слово (удаленный узел)

D: Первое слово назначения (локальный узел)

C: Первое управляющее слово

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	RECV(098)
	Однократное выполнение по положительному фронту	@RECV(098)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограм-мы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

C: Первое управляющее слово

Пять управляющих слов (C...C+4) содержат сведения о количестве принимаемых слов, об источнике передаваемых данных и о других параметрах, которые перечислены в следующей таблице.

Слово	Биты 00...07	Биты 08...15
C	Количество слов: от 0001 до максимально допустимого[1] (4-разр., hex)	
C+1	Адрес запрашиваемой сети: 00...7F (0...127)[2, 4]	Биты 08...11: номер послед. порта (физического порта) 1 hex: порт 1 2 hex: порт 2 (Не задавать 0, 3 или 4.) Биты 12...15: всегда 0
C+2	Адрес запрашиваемого модуля[5]	Адрес запрашиваемого узла: от 00 до максимально допустимого[6]
C+3	Число повторных попыток: 00...0F (0...15)	Номер порта: 00...07 (F: автоматическое назначение)[7] Требование ответа: требуется (всегда).
C+4	Время ожидания ответа: 0001...FFFF hex (0,1...6553,5 с) (При значении по умолчанию 0000 ответ ожидается в течение 2 секунд.)	

Примечание.

(1) Максимально допустимое количество слов зависит от типа используемой сети. Допустимый диапазон для Controller Link: 0001...03DE (от 1 до 990 слов).

(2) Для приема данных от узла в пределах локальной сети задайте адрес запрашиваемой сети равным 00. При наличии двух или большего числа модулей шины ЦПУ адресом сети является наименьший из номеров этих модулей.

(3) Для приема данных по инициативе ПЛК от центрального компьютера через последовательный порт в режиме Host Link можно использовать два способа, которые описаны ниже.

(a) Указать в качестве адреса запрашиваемого модуля (биты 00...07 слова C+2) адрес модуля ЦПУ серии CP или модуля последовательного интерфейса и задать номер последовательного порта (биты 08...11 слова C+1) равным 1 (для порта 1) или 2 (для порта 2).

Адрес модуля (C+2, биты 00...07)	Модуль	Номер послед. порта (C+1, биты 08...11)	Последовательный порт
00 hex	Модуль ЦПУ CP1H	1 hex	Порт 1
		2 hex	Порт 2
10 hex + номер модуля	Модуль последовательного интерфейса серии CJ (модуль шины ЦПУ)	1 hex	Порт 1
		2 hex	Порт 2

(b) Ввести в биты 00…07 слова C+2 непосредственно адрес запрашиваемого модуля. В этом случае в битах 08…11 слова C+1 должен быть указан номер последовательного порта 0.

Порты доп. платы последовательного интерфейса модуля ЦПУ CP1H и CP1L M (30 или 40 вх./вых.)

Порт	Адрес модуля порта
Порт 1	FD hex (253 десятичн.)
Порт 2	FC hex (252 десят.)

Порт доп. платы последовательного интерфейса модуля ЦПУ CP1L L (14 или 20 вх./вых.)

Порт	Адрес модуля порта
Порт 1	FC hex (252 десят.)

Порты модуля последовательного интерфейса серии CJ

Порт	Адрес модуля порта	Пример: номер модуля = 1
Порт 1	80 hex + 4 × номер модуля	80 + 4 × 1 = 84 hex (132 десят.)
Порт 2	81 hex + 4 × номер модуля	81 + 4 × 1 = 85 hex (133 десят.)

(4) Если последовательный порт для функции шлюза последовательного интерфейса (выполняющей преобразование в Host Link FINS) указывается без таблицы маршрутизации, в байте адреса запрашиваемой сети следует задать адрес модуля последовательного порта.

(5) Адрес модуля обозначает модуль в соответствии со следующей таблицей.

Модуль	Значение адреса модуля
Модуль ЦПУ серии CP (доп. плата послед. интерфейса) или модуль ЦПУ серии CJ	00 hex
Модуль шины ЦПУ серии CS/CJ	10 hex + номер модуля
Спец. модуль вв./выв. серии CS/CJ	20 hex + номер модуля

Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

Модуль	Значение адреса модуля
Встраиваемая плата серии CS	E1 hex
Компьютер	01 hex
Модуль, подключенный к сети (модуль указывать необязательно)	FE hex
Непосредственное указание адреса модуля последовательного порта	Модуль ЦПУ CP1H или CP1L M (30 или 40 вх./вых.) Порт 1: FD hex (253 десят.) Порт 2: FC hex (252 десят.) Модуль ЦПУ CP1L L (14 или 20 вх./вых.) Порт 1: FC hex (252 десят.) Порты модуля последовательного интерфейса серии CS/CJ Порт 1: 80 hex + 4 × номер модуля Порт 2: 81 hex + 4 × номер модуля Порты платы последовательного интерфейса серии CS Порт 1: E4 hex (228 десят.) Порт 2: E5 hex (229 десят.) Порты модуля ЦПУ серии CS/CJ Периферийный порт: FD hex (253 десят.) Порт RS-232C: FC hex (252 десят.)

(6) Максимально допустимый номер узла зависит от типа используемой сети. Допустимый диапазон для Controller Link: 00...20 hex (от 0 до 32 узлов). Для передачи в пределах локального узла задайте номер запрашиваемого узла равным 00.

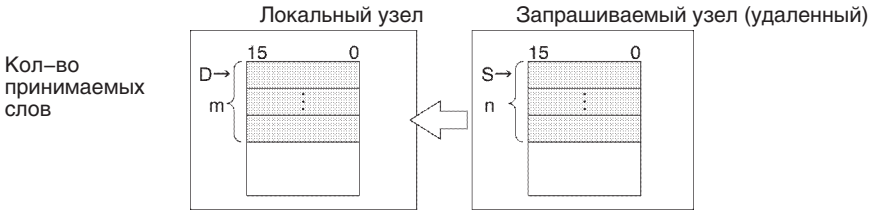
(7) Более подробно функция автоматического назначения номера порта связи (логического порта) описана в разделе *Автоматическое назначение портов связи* на стр. 931.

Характеристики операндов

Область	S	D	C
Область CIO	CIO 0...CIO 6143		CIO 0...CIO 6139
Рабочая область	W0...W511		W0...W507
Область битов хранения	H0...H511		H0...H507
Область вспомогательных битов	A0...A447 A448...A959	A448...A959	A0...A443 A448...A955
Область таймеров	T0000...T4095		T0000...T4091
Область счетчиков	C0000...C4095		C0000...C4091
Область DM	D0...D32767		D0...D32763
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767		
Постоянные	---		
Регистры данных	---		
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда RECV(098) посылает указанному устройству запрос на передачу локальному ПЛК некоторого количества слов данных (указанного в C), начиная со слова S. Данные принимаются по внутренней шине ЦПУ ПЛК или по сети и записываются в область данных ПЛК, начиная с адреса D.



Команда RECV(098) всегда требует возвращения ответа, поскольку именно в ответе содержатся принимаемые данные. Если ответ от запрошенного устройства не поступает в течение времени ожидания ответа, заданного в C+4, запрос на передачу данных отправляется повторно до 15 раз (количество повторных попыток задается битами 0...3 слова C+3).

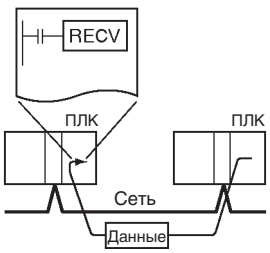
В качестве источника данных для команды RECV(098) может быть указан определенный последовательный порт устройства или само устройство.

Данные могут приниматься от центрального компьютера, подключенного к последовательному порту ПЛК (работающему в режиме Host Link), а также от ПЛК или компьютера, подключенного по сети Controller Link или Ethernet.

Если при выполнении RECV(098) будет включен флаг доступности порта связи для порта связи, указанного в C+3, соответствующий флаг доступности порта связи (порты 00...07: A202.00...A202.07) и соответствующий флаг ошибки порта связи (порты 00...07: A219.00...A219.07) будут сброшены, а в слово кода завершения (порты 00...07: A203...A210) будет записано значение 0000. По завершении приема данных от запрошенного узла флаг доступности порта связи будет вновь установлен, а состояние флага ошибки порта связи и содержимое слова кода завершения будут отражать результат выполнения команды.

Прием данных по сети

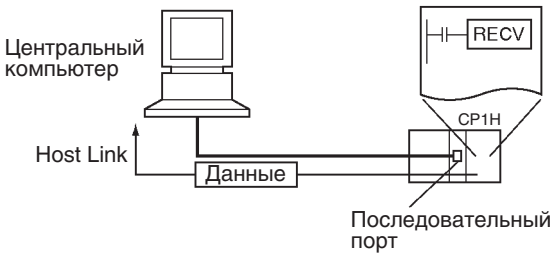
Команда RECV(098) служит для приема данных из указанной области данных удаленного ПЛК или компьютера, подключенного по сети Controller Link или Ethernet, и записи принятых значений в указанную область данных локального ПЛК.



Прием данных от центрального компьютера через Host Link

Если модуль ЦПУ серии CP оборудован дополнительной платой последовательного интерфейса или модулем последовательного интерфейса серии CJ и центральный компьютер подключен непосредственно к последовательному порту этой платы/модуля (последовательный порт должен быть переведен в режим Host Link),

команду RECV(098) можно использовать для приема данных от центрального компьютера (после выполнения RECV(098) данные будут переданы, когда ПЛК в очередной раз получит право на передачу команд). Также возможен прием данных от других центральных компьютеров, подключенных к другим ПЛК в сети.



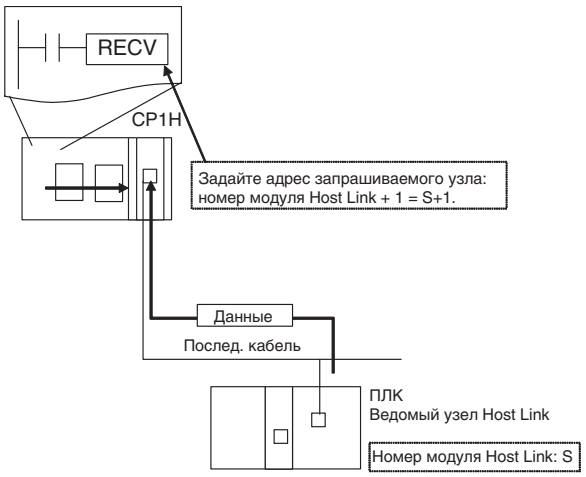
Если команда RECV(098) выполняется для дополнительной платы последовательного интерфейса, установленной в модуль ЦПУ серии CP, или для модуля последовательного интерфейса серии CJ, команда на центральный компьютер передается через дополнительный порт платы/модуля. Под командой понимается сообщение FINS, заключенное между полем заголовка и полем завершения протокола Host Link. Передается команда FINS ПРОЧИТАТЬ ОБЛАСТЬ ПАМЯТИ (код команды 0101) с кодом заголовка Host Link: 0F hex.

На центральном компьютере также должна работать программа, обрабатывающая поступающие запросы на передачу данных (команды FINS, заключенные между кодами заголовка и завершения Host Link).

Если запрашиваемый последовательный порт находится в самом локальном ПЛК, адресные параметры должны иметь следующие значения: адрес сети (C+1) = 00 (локальная сеть), адрес узла (C+2) = 00 (локальный ПЛК), адрес модуля = 00 (доп. плата послед. интерфейса модуля ЦПУ) или номер модуля + 10 hex (модуль послед. интерфейса серии CJ).

Прием данных от ПЛК с функциями ведомого узла Host Link через шлюз послед. интерфейса

Для приема данных от ПЛК, подключенного к модулю последовательного интерфейса в качестве ведомого узла Host Link, можно использовать функцию шлюза последовательного интерфейса. В этом случае в качестве адреса запрашиваемого узла должно быть указано следующее значение: номер модуля Host Link + 1.



Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	Включен, если номер последовательного порта, указанный в C+1, выходит за диапазон 00...04. Включен, если для порта связи, номер которого указан в C+3, выключен флаг доступности порта связи. Выключен во всех остальных случаях.

В следующей таблице перечислены сопутствующие биты и флаги вспомогательной области.

Название	Адрес	Действие
Флаг доступности порта связи	A202.00... A202.07	Этот флаг устанавливается в знак того, что для порта связи с соответствующим номером (00...07) возможно выполнение сетевых команд, включая PMCR(260). Во время выполнения сетевой команды с использованием соответствующего порта флаг выключен. После того как команда выполнена, флаг включается вновь.
Флаг ошибки порта связи	A219.00... A219.07	Этот флаг устанавливается в знак того, что при выполнении сетевой команды с использованием порта связи с соответствующим номером (00...07) произошла ошибка. Состояние флага сохраняется вплоть до выполнения следующей сетевой команды. При выполнении следующей сетевой команды флаг сбрасывается, даже если ранее произошла ошибка.
Коды завершения для портов связи	A203... A210	После выполнения команды сетевого обмена данными в этих словах содержатся коды завершения для соответствующих портов (00...07). Непосредственно во время выполнения сетевой команды соответствующее слово содержит значение 0000, по завершении выполнения команды в это слово записывается код завершения. Содержимое этих слов очищается в начале выполнения программы.

Меры предосторожности

Если для порта, номер которого указан в C+3, выключен флаг доступности порта, команда воспринимается как NOP(000) и не выполняется. В этом случае будет установлен флаг ошибки.

Если для D указан адрес в текущем банке области EM, передаваемые данные записываются в текущий банк EM адресуемого узла.

Если данные будут передаваться за пределы локальной сети, пользователь должен зарегистрировать таблицы маршрутизации в ПЛК (модулях ЦПУ) в каждой сети. (В таблицах маршрутизации детализируются маршруты, по которым данные передаются адресуемым узлам в других сетях.)

Подробные сведения о кодах завершения сетевого обмена данными можно найти в описании ответных кодов команд FINS в руководстве *Серия CS/CJ/CP — Справочное руководство по командам связи* (W342).

Для порта связи одновременно может выполняться только одна сетевая команда. Чтобы команда RECV(098) не оказалась выполненной, когда порт занят, в программе следует использовать флаг доступности порта связи (A202.00...A202.07) в качестве нормально разомкнутого условия.

Номера портов связи 00...07 используются не только сетевыми командами, но и командой PMCR(260), поэтому команда RECV(098) не должна выполняться одновременно с командой PMCR(260), если для обеих команд указан один и тот же номер порта связи.

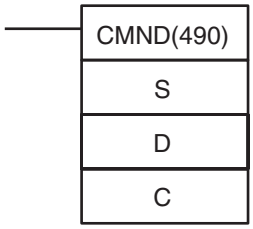
Электромагнитные помехи и другие факторы могут приводить к искажению или полной утрате передаваемых или принимаемых пакетов данных, поэтому рекомендуется устанавливать ненулевое число повторных попыток передачи. В этом случае при отсутствии ответа в течение заданного времени ожидания команда RECV(098) будет выполнена еще раз.

3-24-5 ДОСТАВИТЬ КОМАНДУ: CMND(490)

Назначение

Передача команды FINS и прием возвращаемого ответа. Команды FINS подробно описаны в руководстве *Серия CS/CJ/CP — Справочное руководство по командам связи*.

Символ РКС



S: Первое слово команды

D: Первое слово ответа

C: Первое управляющее слово

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	CMND(490)
	Однократное выполнение по положительному фронту	@CMND(490)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

C: Первое управляющее слово

Шесть управляющих слов (C...C+5) содержат информацию о количестве байтов в данных команды и данных ответа, о получателе команды и о других параметрах, которые перечислены в следующей таблице.

Слово	Биты 00...07	Биты 08...15
C	Кол-во байтов в данных команды: от 0002 до максимально допустимого[1] (4-разр., hex)	
C+1	Кол-во байтов в данных ответа: от 0000 до максимально допустимого[1,2,3] (4-разр., hex)	
C+2	Адрес сети назначения: 00...7F[4, 6]	Биты 08...11: номер послед. порта (физического порта) 1 hex: порт 1 2 hex: порт 2 (Не задавать 0, 3 или 4.) Биты 12...15: всегда 0

Слово	Биты 00...07	Биты 08...15
C+3	Адрес модуля назначения: 00...FE[5, 7, 9]	Номер узла назначения: от 00 до максимально допустимого[8]
C+4	Число повторных попыток: 00...0F (0...15)	Биты 08...11: номер порта (внутреннего логического порта): 0...7 (F: автоматическое назначение)[10] Биты 12...15: требование ответа 0: ответ требуется 8: ответ не требуется[11]
C+5	Время ожидания ответа: 0001...FFFF hex (0,1...6553,5 с) (При значении по умолчанию 0000 ответ ожидается в течение 2 секунд.)	

Примечание.

(1) Количество байтов в данных команды, указываемое в C, может принимать значения от 0002 до максимальной длины пакета данных (в шестнадцатеричном формате). Например, количество байтов для системы Controller Link может варьироваться в пределах от 0002 до 07C6 hex (2...1990 байт). Количество байтов для локального модуля ЦПУ: 07C6 hex (1990 байт). Количество байтов в данных команды зависит от типа используемой сети.

(2) Количество байтов в данных ответа, указываемое в C+1, может принимать значения от 0000 до максимальной длины пакета данных (в шестнадцатеричном формате). Например, количество байтов для системы Controller Link может варьироваться в пределах от 0000 до 07C6 hex (0...1990 байт). Количество байтов для локального модуля ЦПУ: 07C6 hex (1990 байт). Количество байтов в данных ответа зависит от типа используемой сети.

(3) Сведения о максимально возможных длинах пакетов данных команд и ответов можно найти в руководстве по эксплуатации для используемой сети. Если команды FINS в процессе обмена данными передаются через несколько сетей, максимальные размеры пакетов данных команд и ответов определяются сетью, у которой эти показатели имеют наименьшие значения.

(4) Для передачи в пределах локальной сети задайте адрес сети назначения равным 00. При наличии двух или большего числа модулей шины ЦПУ адресом сети является наименьший из номеров этих модулей.

(5) Команда FINS может быть передана по инициативе ПЛК центральному компьютеру через последовательный порт, функционирующий в режиме Host Link или в режиме шлюза последовательного интерфейса (преобразующего команды FINS в команды протокола CompoWay/F, Modbus-RTU или Modbus-ASCII). Это можно осуществить двумя способами.

(a) Указать в качестве адреса модуля назначения (биты 00...07 слова C+3) адрес модуля ЦПУ CP1H или модуля последовательного интерфейса серии CJ и задать номер последовательного порта (биты 08...11 слова C+2) равным 1 (для порта 1) или 2 (для порта 2).

Адрес модуля (C+3, биты 00...07)	Модуль	Номер послед. порта (C+2, биты 08...11)	Последовательный порт
00 hex	Модуль ЦПУ CP1H	1 hex	Порт 1
		2 hex	Порт 2

Адрес модуля (C+3, биты 00...07)	Модуль	Номер послед. порта (C+2, биты 08...11)	Последо-вательный порт
10 hex + номер модуля	Модуль последовательного интерфейса серии CJ (модуль шины ЦПУ)	1 hex	Порт 1
		2 hex	Порт 2

(b) Ввести в биты 00...07 слова C+3 непосредственно адрес модуля назначения. В этом случае в битах 08...11 слова C+2 должен быть указан номер последовательного порта 0.

Порты доп. платы последовательного интерфейса модуля ЦПУ CP1H или CP1L M (30 или 40 вх./вых.)

Порт	Адрес модуля порта
Порт 1	FD hex (253 десятичн.)
Порт 2	FC hex (252 десят.)

Порт доп. платы последовательного интерфейса модуля ЦПУ CP1L L (14 или 20 вх./вых.)

Порт	Адрес модуля порта
Порт 1	FC hex (252 десят.)

Порты модуля последовательного интерфейса серии CJ

Порт	Адрес модуля порта	Пример: номер модуля = 1
Порт 1	80 hex + 4 × номер модуля	80 + 4 × 1 = 84 hex (132 десят.)
Порт 2	81 hex + 4 × номер модуля	81 + 4 × 1 = 85 hex (133 десят.)

(6) Если последовательный порт для функции шлюза последовательного интерфейса (выполняющей преобразование в Host Link FINS) указывается без таблицы маршрутизации, в байте адреса сети назначения следует задать адрес модуля последовательного порта.

(7) Адрес модуля обозначает модуль в соответствии со следующей таблицей.

Модуль	Значение адреса модуля
Модуль ЦПУ CP1H (доп. плата послед. интерфейса) или модуль ЦПУ серии CJ	00 hex
Модуль шины ЦПУ серии CS/CJ	10 hex + номер модуля
Спец. модуль вв./выв. серии CS/CJ	20 hex + номер модуля
Встраиваемая плата серии CS	E1 hex
Компьютер	01 hex

Предварительная версия
Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

Модуль	Значение адреса модуля
Модуль, подключенный к сети (модуль указывать необязательно)	FE hex
Непосредственное указание адреса модуля последовательного порта	Модуль ЦПУ CP1H или CP1L M с установленной доп. платой послед. интерфейса Порт 1: FD hex (253 десят.) Порт 2: FC hex (252 десят.) Модуль ЦПУ CP1L L с установленной доп. платой послед. интерфейса Порт 1: FC hex (252 десят.) Порты модуля последовательного интерфейса серии CS/CJ Порт 1: 80 hex + 4 × номер модуля Порт 2: 81 hex + 4 × номер модуля Порты платы последовательного интерфейса серии CS Порт 1: E4 hex (228 десят.) Порт 2: E5 hex (229 десят.) Порты модуля ЦПУ серии CS/CJ Периферийный порт: FD hex (253 десят.) Порт RS-232C: FC hex (252 десят.)

(8) Максимально допустимый номер узла зависит от типа используемой сети. Допустимый диапазон для Controller Link: 00...20 hex (от 0 до 32 узлов). Для передачи одновременно всем узлам (широковещание) введите для номера узла назначения значение FF; для передачи в пределах локального узла введите значение 00.

(9) Указывая последовательный порт в режиме шлюза последовательного интерфейса (преобразование в Host Link FINS), в качестве адреса модуля назначения задайте следующее значение: номер модуля Host Link адресуемого ПЛК + 1 (диапазон значений: 1...32).

(10) Более подробно функция автоматического назначения номера порта связи (логического порта) описана в разделе *Автоматическое назначение портов связи* на стр. 931.

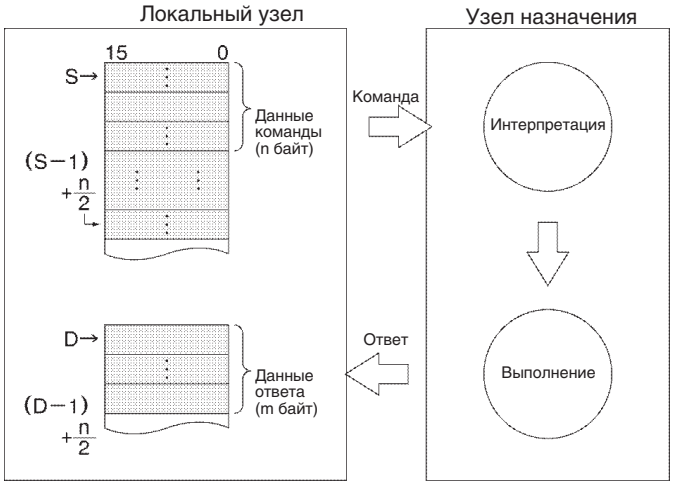
(11) Если в качестве номера узла назначения введено значение FF (широковещание), ответ на передачу не возвращается, даже если в биты 12...15 введено 0.

Область	S	C	D
Область CIO	CIO 0...CIO 6143		CIO 0...CIO 6138
Рабочая область	W0...W511		W0...W506
Область битов хранения	H0...H511		H0...H506
Область вспомогательных битов	A0...A447 A448...A959	A448...A959	A0...A442 A448...A954
Область таймеров	T0000...T4095		T0000...T4090
Область счетчиков	C0000...C4095		C0000...C4090
Область DM	D0...D32767		D0...D32762
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767		
Постоянные	---		
Регистры данных	---		

Область	S	C	D
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда CMND(490) передает указанное число байтов данных команды FINS, начиная со слова S, указанному устройству по внутренней шине ЦПУ ПЛК или по сети. Полученный ответ записывается в слова памяти, начиная со слова D.



В качестве получателя команды FINS может быть указан определенный последовательный порт адресуемого устройства или само устройство. Функционирование команды CMND(490) эквивалентно применению команды SEND(090) с кодом команды FINS 0102 (ЗАПИСАТЬ В ОБЛАСТЬ ПАМЯТИ) и команды RECV(098) с кодом 0101 (ПРОЧИТАТЬ ОБЛАСТЬ ПАМЯТИ).

Если номер узла назначения задан равным FF, данные команды передаются одновременно всем узлам адресуемой сети. Такой режим передачи называют «широковещанием».

Если затребован ответ (биты 12...15 слова C+4 = 0), но ответ не поступает в течение заданного времени ожидания ответа, данные команды передаются повторно до 15 раз (число повторных попыток задается битами 0...3 слова C+3). В случае широковещательной передачи ответ не возвращается и повторные попытки передачи не предпринимаются. Для команд, не требующих ответа, в соответствующих битах соответствующего управляющего слова следует указать, что ответ не требуется.

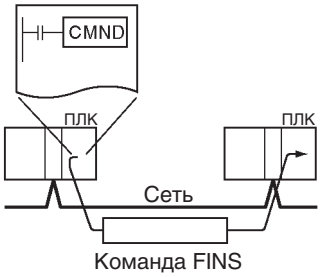
Если объем данных ответа превысит допустимое количество байтов данных ответа, заданное в C+1, произойдет ошибка.

Команда FINS может быть передана центральному компьютеру, подключенному к последовательному порту ПЛК, а также другому ПЛК (модулю ЦПУ CP1H, модулю ЦПУ серии CS/CJ, модулю шины ЦПУ серии CS/CJ или встраиваемой плате серии CS) или компьютеру, подключенному по сети Controller Link или Ethernet.

Если при выполнении CMND(490) будет включен флаг доступности порта связи для порта связи, указанного в C+3, соответствующий флаг доступности порта связи (порты 00...07: A202.00...A202.07) и соответствующий флаг ошибки порта связи (порты 00...07: A219.00...A219.07) будут сброшены, а в слово кода завершения (порты 00...07: A203...A210) будет записано значение 0000. По завершении передачи данных команды адресуемому узлу флаг доступности порта связи будет вновь установлен, а состояние флага ошибки порта связи и содержимое слова кода завершения будут отражать результат выполнения команды.

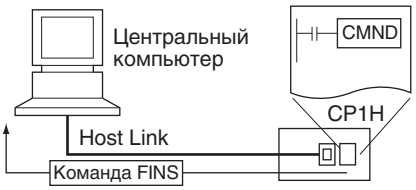
Передача по сети (только для CP1H)

С помощью команды CMND(490) любая команда FINS может быть передана персональному компьютеру или ПЛК (модулю ЦПУ CP1H, модулю ЦПУ серии CS/CJ, модулю шины ЦПУ серии CS/CJ или встраиваемой плате серии CS), подключенному по сети Controller Link или Ethernet.



Передача через Host Link

Если модуль ЦПУ оборудован дополнительной платой последовательного интерфейса или модулем последовательного интерфейса серии CJ и центральный компьютер подключен непосредственно к последовательному порту этой платы/модуля (последовательный порт должен быть переведен в режим Host Link), команду CMND(490) можно использовать для передачи любой команды FINS из ПЛК на центральный компьютер (после выполнения CMND(490) данные будут переданы, когда ПЛК в очередной раз получит право на передачу). Также возможна передача данных на другие центральные компьютеры, подключенные к другим ПЛК в сети.



С целью отправки команды FINS на подключенный центральный компьютер в команде CMND(490) может использоваться любой из портов модуля ЦПУ (доп. платы послед. интерфейса) или модуля послед. интерфейса серии CJ. (Номер последовательного порта (1 hex или 2 hex) указывается в битах 08...11 слова C+2.) Под командой понимается сообщение FINS, заключенное между полем заголовка и полем завершения протокола Host Link. Может быть передана любая команда FINS, код заголовка Host Link всегда имеет значение 0F hex.

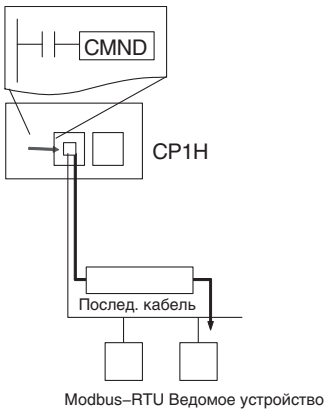
На центральном компьютере также должна работать программа, обрабатывающая принимаемые команды (команды FINS с кодами заголовка и завершения Host Link).

Если адресуемый последовательный порт находится в самом локальном ПЛК, адресные параметры должны иметь следующие значения: адрес сети (C+2) = 00 (локальная сеть), адрес узла (C+3) = 00 (локальный ПЛК), адрес модуля = 00 (доп. плата послед. интерфейса в модуле ЦПУ CP1H) или номер модуля + 10 hex (модуль послед. интерфейса серии CJ).

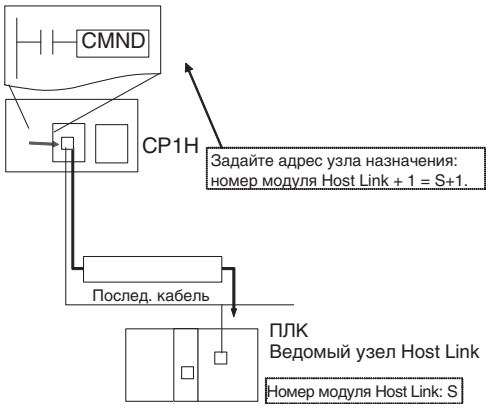
Связь с устройством или ведомым узлом Host Link в режиме шлюза последовательного интерфейса

ПЛК может посылать команды FINS (или передавать/принимать данные) другому устройству или ведомому узлу Host Link через последовательный порт дополнительной платы последовательного интерфейса серии CP1H или модуля последовательного интерфейса серии CJ, работающий в режиме шлюза последовательного интерфейса (Serial Gateway).

- Передача команд FINS другому устройству
  (преобразование в CompoWay/F, Modbus-RTU или Modbus-ASCII)
  В том случае, когда команда FINS передается через последовательный порт платы или модуля последовательного интерфейса или через один из последовательных портов модуля ЦПУ (периферийный порт или порт RS-232C), функция шлюза последовательного интерфейса преобразует перечисленные ниже команды FINS в команды протокола CompoWay/F, Modbus-RTU или Modbus-ASCII.
  Преобразование в команду CompoWay/F: 2803 hex
  Преобразование в команду Modbus-RTU: 2804 hex
  Преобразование в команду Modbus-ASCII: 2805 hex (не поддерживается для последовательных портов дополнительных плат последовательного интерфейса CP1H).



- Передача программируемому контроллеру с функциями ведомого узла Host Link
  Функцией шлюза последовательного интерфейса можно воспользоваться для передачи любой команды FINS программируемому контроллеру, который подключен в качестве ведомого узла Host Link через последовательный порт дополнительной платы последовательного интерфейса CP1H или модуля последовательного интерфейса серии CJ. В этом случае в качестве адреса узла назначения должно быть указано следующее значение: номер модуля Host Link + 1.



Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	Включен, если номер последовательного порта, указанный в C+2, выходит за диапазон 00...04. Включен, если для порта связи, номер которого указан в C+4, выключен флаг доступности порта связи. Выключен во всех остальных случаях.

В следующей таблице перечислены сопутствующие биты и флаги вспомогательной области.

Название	Адрес	Действие
Флаг доступности порта связи	A202.00... A202.07	Этот флаг устанавливается в знак того, что для порта связи с соответствующим номером (00...07) возможно выполнение сетевых команд, включая PMCR(260). Во время выполнения сетевой команды с использованием соответствующего порта флаг выключен. После того как команда выполнена, флаг включается вновь.
Флаг ошибки порта связи	A219.00... A219.07	Этот флаг устанавливается в знак того, что при выполнении сетевой команды с использованием порта связи с соответствующим номером (00...07) произошла ошибка. Состояние флага сохраняется вплоть до выполнения следующей сетевой команды. При выполнении следующей сетевой команды флаг сбрасывается, даже если ранее произошла ошибка.
Коды завершения для портов связи	A203... A210	После выполнения команды сетевого обмена данными в этих словах содержатся коды завершения для соответствующих портов (00...07). Непосредственно во время выполнения сетевой команды соответствующее слово содержит значение 0000, по завершении выполнения команды в это слово записывается код завершения. Содержимое этих слов очищается в начале выполнения программы.

Меры предосторожности

Если для порта, номер которого указан в C+4, выключен флаг доступности порта, команда воспринимается как NOP(000) и не выполняется. В этом случае будет установлен флаг ошибки.

Если данные будут передаваться за пределы локальной сети, пользователь должен зарегистрировать таблицы маршрутизации в ПЛК (модулях ЦПУ) в каждой сети. (В таблицах маршрутизации детализируются маршруты, по которым данные передаются адресуемым узлам в других сетях.)

Подробные сведения о кодах завершения сетевого обмена данными можно найти в описании ответных кодов команд FINS в руководстве *Серия CS/CJ/CP — Справочное руководство по командам связи* (W342).

Номера портов связи (00...07) используются совместно сетевыми командами и командами последовательного интерфейса (SEND(090), RECV(098), CMND(490), PMCR(260), TXDU(256) или RXDU(255)), поэтому для порта связи с определенным номером может одновременно выполняться только одна из этих команд. Чтобы команда CMND(490) не оказалась выполненной, когда порт занят, в программе следует использовать флаг доступности порта связи (A202.00...A202.07) в качестве нормально разомкнутого условия.

Электромагнитные помехи и другие факторы могут приводить к искажению или полной утрате передаваемых или принимаемых пакетов данных, поэтому рекомендуется устанавливать ненулевое число повторных попыток передачи. В этом случае при отсутствии ответа в течение заданного времени ожидания команда CMND(490) будет выполнена еще раз.

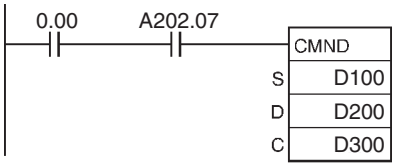
Примеры

Ниже в качестве примера показан сегмент программы, передающий команду FINS другому модулю ЦПУ.

Когда бит CIO 0.00 и бит A202.07 (флаг доступности порта связи для порта 07) оба включены, команда CMND(490) отправляет команду FINS с кодом 0101 (ПРОЧИТАТЬ ОБЛАСТЬ ПАМЯТИ) узлу с номером 3. Возвращаемый ответ записывается в слова D200...D211.

Команда считывает содержимое десяти слов в диапазоне от D10 до D19. Ответ содержит 2-байтовый код команды (0101), 2-байтовый код завершения, а также 10 слов данных — всего 12 слов или 24 байта.

Если ответ не поступает в течение 10 секунд, предпринимается до 3 повторных попыток передачи данных.



	15 8	7 0	
S: D100	0 1	0 1	Код команды: 0101 hex (ПРОЧИТАТЬ ОБЛАСТЬ ПАМЯТИ)
S+1: D101	8 2	0 0	D10 (область данных = 82 hex, адрес = 000A00)
S+2: D102	0 A	0 0	
S+3: D103	0 0	0 A	Кол–во читаемых слов = 0A hex (10 десят.)

	15 8	7 0	
C: D300	0 0	0 8	Кол–во байтов в данных команды: 0008 (8 десят.)
C+1: D301	0 0	1 8	Кол–во байтов в данных ответа: 0018 (24)
C+2: D302	0 0	0 0	Передача по локальной сети самому устройству
C+3: D303	0 3	0 0	Номер узла 3, адрес модуля 00 (модуль ЦПУ)
C+4: D304	0 7	0 3	Требуется ответ, номер порта 7, 3 попытки
C+5: D305	0 0	6 4	Время ожидания ответа: 0064 hex (10 секунд)

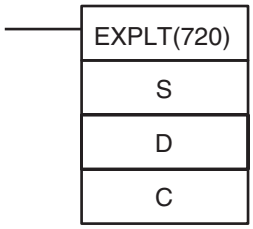
3-24-6 ОТПРАВИТЬ ЯВНОЕ СООБЩЕНИЕ: EXPLT(720)

Назначение

Передача явного сообщения с любым кодом службы.

Эту команду возможно использовать только для модулей ЦПУ CP1H. Для модулей ЦПУ CP1L она недоступна. Если команда EXPLT(720) будет использована для модуля ЦПУ CP1L, будет установлен флаг ошибки (ER).

Символ РКС



S: Первое слово передаваемого сообщения

D: Первое слово принимаемого сообщения

C: Первое управляющее слово

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	EXPLT(720)
	Однократное выполнение по положительному фронту	@EXPLT(720)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

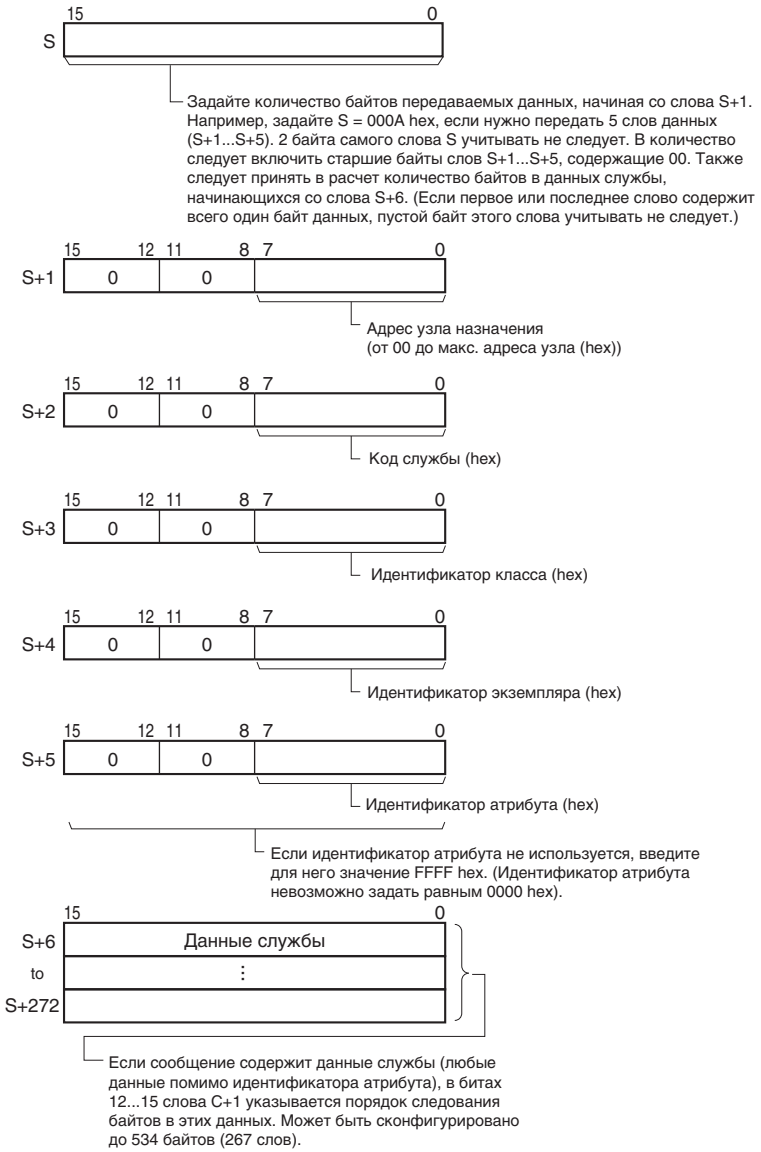
Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

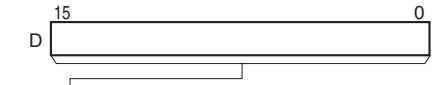
S: Первое слово передаваемого сообщения

Указывает первое слово передаваемого сообщения (S...макс. S+272).

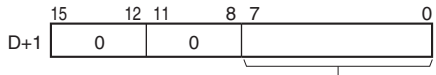


D: Первое слово принимаемого сообщения

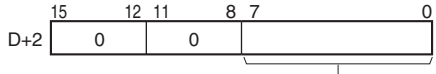
Указывает первое слово принимаемого сообщения (D...макс. D+269).



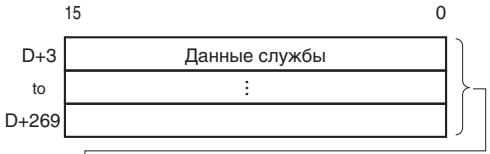
Содержит значение количества байтов данных в слове D+1 и в следующих за ним словах (не включая 2 байта самого слова D). В этом значении учтены старшие байты слов D+1 и D+2, содержащие 00. В нем также учтено количество байтов в данных службы, начинающихся со слова D+3. (Если первое или последнее слово содержит всего один байт данных, пустой байт этого слова не учитывается.)



Содержит адрес запрашиваемого узла (00...3F hex (0...63) для DeviceNet))



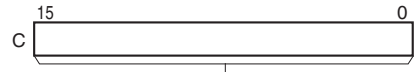
Содержит код службы или код ошибки (hex).
Нормальный ответ: возвращается код службы команды с битом 07 = 1.
Ответ с ошибкой: возвращается 94 hex, независимо от кода службы команды.



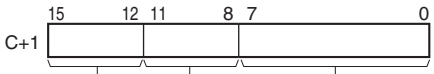
Содержит данные службы ответа (данные, следующие за кодом службы). Порядок следования байтов в этих данных указывается в битах 12...15 слова C+1. Может содержать до 534 байтов (267 слов) данных.

C: Первое управляющее слово

Указывает первое из четырех управляющих слов (C...C+3).



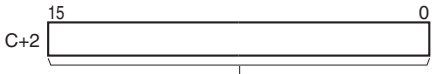
Задайте общее количество слов данных ответа, начиная со слова D.
Диапазон допустимых значений: от 0 до 010E hex (270 слов).
Если количество слов в принятых данных будет больше заданного здесь значения, произойдет ошибка FINS (слишком длинный ответ, код 110B) и никаких данных записано не будет (в область, начинающуюся со слова D+3).
Если количество слов в принятых данных будет меньше заданного здесь значения, содержимое остальных слов (в области, начинающейся со слова D+3) не изменится.



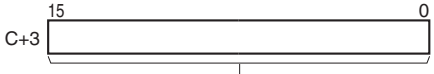
Адрес модуля FINS передающего модуля связи.
Модуль шины ЦПУ: 10...1F hex (номер модуля + 10 hex)
Спец. модуль вв./выв.: 20...7F hex (номер модуля + 20 hex)

Номер порта связи (логического порта) для сетевой команды:
0...7 hex (F hex: автоматическое назначение)

Порядок хранения байтов данных службы (кадра данных) в памяти, начиная со слов S+6 и D+3.
0 hex: от старшего байта к младшему (старший → младший → старший → младший ...)
8 hex: от младшего байта к старшему (младший → старший → младший → старший ...)



Время ожидания ответа
0001...FFFF hex (0,1...6553,5 с)
0000 hex: 2 с (значение по умолчанию)



Формат явного сообщения
0000 hex: DeviceNet (то же, что при использовании команды FINS 2801)

Характеристики операндов

Область	S	D	C
Область CIO	CIO 0...CIO 6143		CIO 0...CIO 6140
Рабочая область	W0...W511		W0...W508
Область битов хранения	H0...H511		H0...H508
Область вспомогательных битов	A0...A959	A448...A959	A0...A956
Область таймеров	T0000...T4095		T0000...T4092
Область счетчиков	C0000...C4095		C0000...C4092
Область DM	D0...D32767		D0...D32764
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767		
Постоянные	---		
Регистры данных	---		
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Данная команда передает явное сообщение (содержащееся в группе слов, начиная с S+2) узлу, адрес которого указан в S+1, через модуль связи, адрес FINS которого указан в битах 00...07 слова C+1. Данные, возвращаемые в ответ на явное сообщение, записываются в группу слов, начиная с D+2.

Пояснения к параметрам количества байтов

В количество байтов передаваемых данных, указываемое входным операндом S, входит 10 байтов слов S+1...S+5, а также количество байтов данных службы, начинающихся со слова S+6. (Например, если данные службы содержат 1 байт, то всего будет 11 байтов данных, поэтому в S следует ввести значение 000B hex.)

В количество байтов принимаемых данных, указываемое выходным операндом D, входят 4 байта слов D+1 и D+2, а также количество байтов данных службы, начинающихся со слова D+3. (Например, если данные службы содержат 1 байт, то всего будет 5 байтов данных и слово D будет содержать значение 0005 hex.)

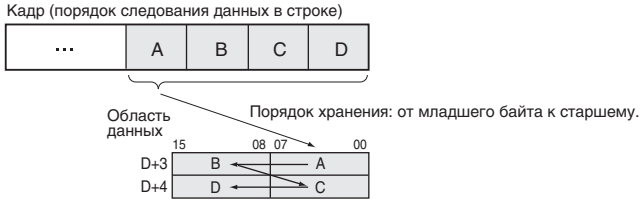
Порядок расположения байтов данных службы в словах S+6 и D+3 определяется содержимым битов 12...15 слова C+1 (0 или 8 hex).

- Хранение данных в порядке от старшего байта к младшему
  В биты 12...15 слова C+1 следует записать 0 hex.



Прим.: A, B, C и D — обозначения байтов данных.

• Хранение данных в порядке от младшего байта к старшему
В биты 12...15 слова C+1 следует записать 8 hex.



Прим.: A, B, C и D — обозначения байтов данных.

Флаги

Название	Обо-значение	Действие
Флаг ошибки	ER	Включен, если для порта связи, номер которого указан в C, выключен флаг доступности порта связи. Выключен во всех остальных случаях.

При успешном выполнении команды соответствующий флаг ошибки обмена явными сообщениями будет выключен, в случае ошибки — включен.

В случае ошибки (т. е. когда включен соответствующий бит в слове A213) соответствующий флаг ошибки порта связи позволяет определить, что именно произошло: явное сообщение не было передано вообще (включен соответствующий бит в A219) или сообщение было передано, но содержало ошибку (выключен соответствующий бит в A219).

При успешном выполнении команды соответствующий код завершения для порта связи (A203...A210) будет иметь значение 0000 hex, а в случае ошибки в нем будет содержаться код ошибки явного сообщения (при ошибке передачи явного сообщения) либо код ошибки FINS (при возникновении ошибки FINS).

Работа команд передачи явных сообщений подробна описана в разделе *3-24-2 Сведения о командах передачи явных сообщений (только для CP1H)*.

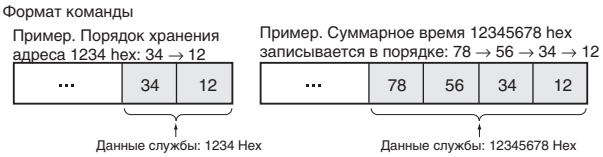
В следующей таблице перечислены сопутствующие биты и флаги вспомогательной области.

Название	Адрес	Действие
Флаг доступности порта связи	A202.00... A202.07	Этот флаг устанавливается в знак того, что для порта связи с соответствующим номером (00...07) возможно выполнение сетевых команд, включая PMCR(260). Во время выполнения сетевой команды с использованием соответствующего порта этот флаг выключен. После того как команда выполнена, этот флаг включается вновь.
Флаг ошибки обмена явными сообщениями	A213.00... A213.07	Этот флаг устанавливается в знак того, что в процессе обмена явными сообщениями с использованием порта связи с соответствующим номером (00...07) произошла ошибка. Флаг устанавливается, если явное сообщение не было отправлено вообще либо было отправлено, но вернулся ответ с кодом ошибки. Состояние этого флага сохраняется вплоть до выполнения следующей команды передачи явного сообщения. При выполнении следующей команды этот флаг сбрасывается, даже если ранее произошла ошибка.

Название	Адрес	Действие
Флаг ошибки порта связи	A219.00... A219.07	Этот флаг устанавливается в знак того, что при выполнении команды передачи явного сообщения с использованием порта связи с соответствующим номером (00...07) явное сообщение передано не было. Состояние этого флага сохраняется вплоть до выполнения следующей сетевой команды. При выполнении следующей команды этот флаг сбрасывается, даже если ранее произошла ошибка.
Коды завершения для портов связи	A203... A210	После выполнения команды сетевого обмена данными в этих словах содержатся коды завершения для соответствующих портов (00...07). Пока флаг ошибки обмена явными сообщениями для некоторого порта выключен, слово кода завершения для этого порта содержит значение 0000. Если для некоторого порта одновременно включены флаг ошибки обмена явными сообщениями и флаг ошибки порта связи, слово кода завершения для этого порта содержит код ошибки FINS. Если флаг ошибки обмена явными сообщениями для некоторого порта включен, тогда как флаг ошибки порта связи выключен, слово кода завершения для этого порта содержит соответствующий код ошибки передачи явного сообщения. Непосредственно во время выполнения сетевой команды соответствующее слово содержит значение 0000, по завершении выполнения команды в это слово записывается код завершения. Содержимое этих слов очищается в начале выполнения программы.

Меры предосторожности

Проследите, чтобы порядок байтов в исходных данных совпадал с порядком следования байтов в кадре явного сообщения («порядок следования данных в строке» на рисунке ниже). Например, когда данные службы состоят из 2-байтовых или 4-байтовых элементов, байты данных (пары из двух разрядов) в кадре сообщения располагаются слева направо, как показано на следующем рисунке.



На приведенных ниже рисунках показан порядок записи байтов данных в ячейки памяти данных, когда данные службы состоят из элементов длиной в 2 или 4 байта.

1. Данные, состоящие из 2-байтовых элементов.

- Хранение данных в порядке от старшего байта к младшему (биты 12...15 в C = 0 hex)
  Пример: хранение значения 1234 hex в D+3.



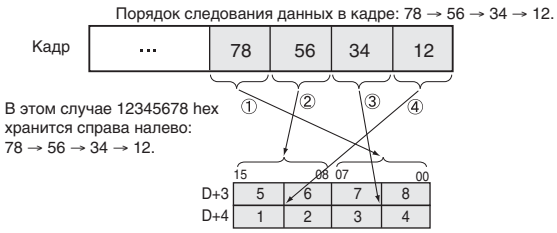
- Хранение данных в порядке от младшего байта к старшему (биты 12...15 в C = 8 hex)
  Пример: хранение значения 1234 hex в D+3.



2. Данные, состоящие из 4-байтовых элементов.
   - Хранение данных в порядке от старшего байта к младшему (биты 12...15 в C = 0 hex)
     Пример: хранение значения 12345678 hex в D+3 и D+4.

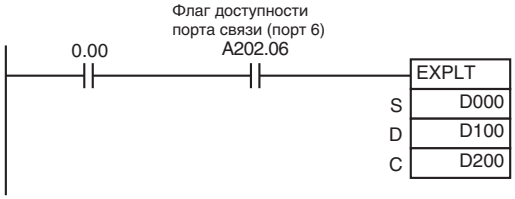


   - Хранение данных в порядке от младшего байта к старшему (биты 12...15 в C = 8 hex)
     Пример: хранение значения 12345678 hex в D+3 и D+4.



Примечание. В приведенных выше примерах показано расположение байтов принятых данных в словах D+3, D+4... Байты передаваемых данных в словах S+6, S+7... располагаются точно так же.

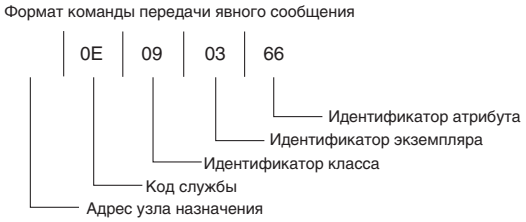
Пример В данном примере с помощью команды EXPLT(720) из ведомого устройства DRT2 (терминал ввода-вывода) считывается общая продолжительность включенного состояния или количество переключений контактов.

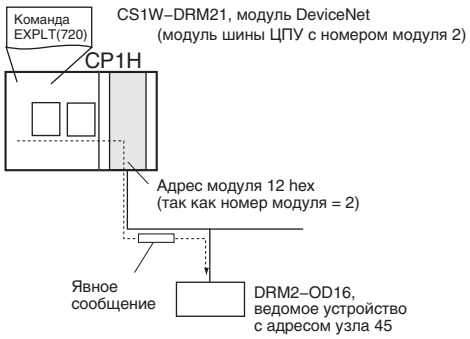


Когда бит CIO 0.00 и бит A202.06 (флаг доступности порта связи для порта 06) оба включены, команда EXPLT(720) считывает из ведомого устройства DRT2 (терминал ввода-вывода) общую продолжительность включенного состояния (сек) или количество переключений контактов. В данном примере считывается общая продолжительность пребывания во включенном состоянии или количество переключений контактов для входа 3.

Код службы = 0E hex, идентификатор класса = 09 hex, идентификатор экземпляра = 03 hex, идентификатор атрибута = 66 hex.

Например, в ответ на запрос общего времени включенного состояния возвращается значение 2 752 039 с.





S:	D0	0	0	0	A	Количество байтов данных: S+1...S+5 = 5 слов = 10 байтов = 0A hex
S+1:	D1	0	0	2	D	Адрес ведомого узла = 45 = 2D hex
S+2:	D2	0	0	0	E	Код службы = 0E hex
S+3:	D3	0	0	0	9	Идентификатор класса = 09 hex
S+4:	D4	0	0	0	3	Идентификатор экземпляра = 03 hex (вход 3)
S+5:	D5	0	0	6	6	Идентификатор атрибута = 66 hex

D:	D100	0	0	0	8	Содержит 08 hex: 8 байтов принятых данных в кадре ответа.
D+1:	D101	0	0	2	D	Возвращает адрес ведомого узла = 45 = 2D hex.
D+2:	D102	0	0	8	E	Код службы = 8E hex (успешное завершение)
D+3:	D103	2	7	F	E	Данные службы = 0029FE27 hex (2 752 039 секунд)
D+4:	D104	2	9	0	0	

C:	D200	0	0	0	5	Содержит 0005 hex, так как D...D+5 = 5 слов.
C+1:	D201	0	6	1	2	Порядок байтов = 0 hex (от старшего к младшему), порт связи = 6 hex (порт 6), адрес модуля DeviceNet = 12 hex
C+2:	D202	0	0	0	0	Время ожидания ответа = 0000 hex (2 с)
C+3:	D203	0	0	0	0	Тип формата явного сообщения = 0000 hex (формат DeviceNet)

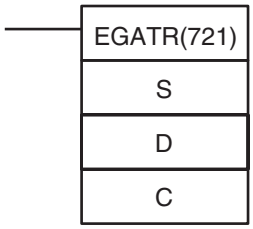
3-24-7 ЯВНЫЙ ЗАПРОС АТРИБУТА: EGATR(721)

Назначение

Передача команды чтения информации/состояния в форме явного сообщения (ЗАПРОС ОДИНОЧНОГО АТРИБУТА, код службы: 0E hex).

Эту команду возможно использовать только для модулей ЦПУ CP1H. Для модулей ЦПУ CP1L она недоступна. Если команда EGATR(721) будет использована для модуля ЦПУ CP1L, будет установлен флаг ошибки (ER).

Символ РКС



S: Первое слово передаваемого сообщения

D: Первое слово принимаемого сообщения

C: Первое управляющее слово

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	EGATR(721)
	Однократное выполнение по положительному фронту	@EGATR(721)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

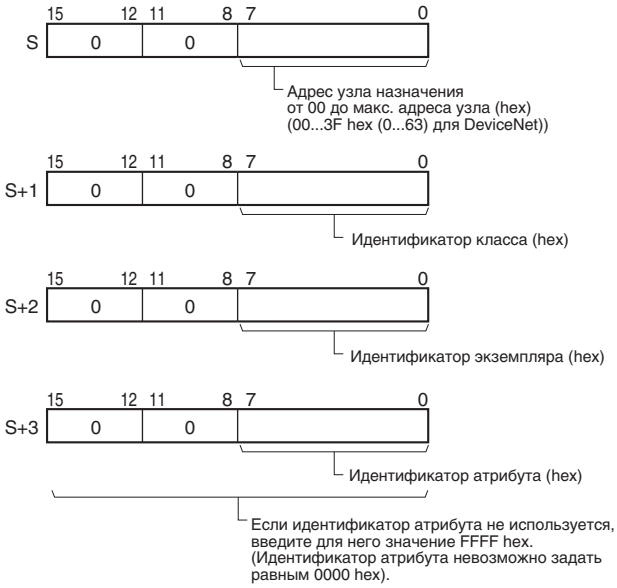
Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

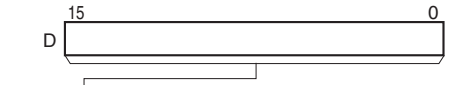
S: Первое слово передаваемого сообщения

Указывает первое слово передаваемого сообщения (S...S+3).



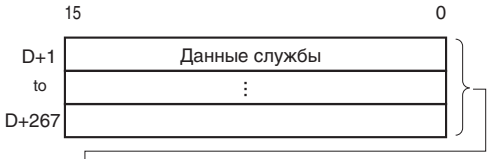
D: Первое слово принимаемого сообщения

Указывает первое слово принимаемого сообщения (D... макс. D+267).



Содержит значение количества байтов принятых данных службы в слове D+1 и в следующих за ним словах (не включая 2 байта самого слова D).

Учитывает только количество байтов данных службы, начинающихся с D+1 (если первое или последнее слово содержит всего один байт данных, пустой байт этого слова не учитывается.)



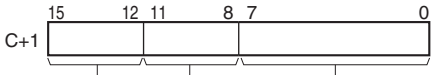
Содержит данные службы ответа (данные, следующие за кодом службы). Порядок следования байтов в этих данных указывается в битах 12...15 слова C+1.
Может содержать до 534 байтов (267 слов) данных.

C: Первое управляющее слово

Указывает первое из четырех управляющих слов (C...C+3).



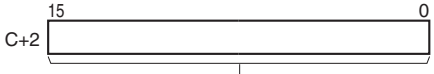
Задайте максимальное количество слов принимаемых данных, начиная со слова D.
Диапазон допустимых значений: от 0 до 010C hex (268 слов).
Если количество слов в принятых данных будет больше заданного здесь значения, произойдет ошибка FINS (слишком длинный ответ, код 110B) и никаких данных записано не будет (в область, начинающуюся со слова D+3).
Если количество слов в принятых данных будет меньше заданного здесь значения, содержимое остальных слов (в области, начинающейся со слова D+3) не изменится.



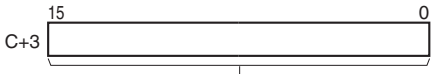
Адрес модуля FINS передающего модуля связи.
Модуль шины ЦПУ: 10...1F hex (номер модуля + 10 hex)
Спец. модуль вв./выв.: 20...7F hex (номер модуля + 20 hex)

Номер порта связи (логического порта) для сетевой команды:
0...7 hex (F hex: автоматическое назначение)

Порядок хранения байтов данных службы (кадра данных) в памяти, начиная со слов S+6 и D+3.
0 hex: от старшего байта к младшему (старший → младший → старший → младший ...)
8 hex: от младшего байта к старшему (младший → старший → младший → старший ...)



Время ожидания ответа
0001...FFFF hex (0,1...6553,5 с)
0000 hex: 2 с (значение по умолчанию)



Формат явного сообщения
0000 hex: DeviceNet (то же, что при использовании команды FINS 2801)

Характеристики операндов

Область	S	D	C
Область CIO	CIO 0...CIO 6140	CIO 0...CIO 6143	CIO 0...CIO 6140
Рабочая область	W0...W508	W0...W511	W0...W508
Область битов хранения	H0...H508	H0...H511	H0...H508
Область вспомогательных битов	A0...A956	A0...A959	A0...A956
Область таймеров	T0000...T4092	T0000...T4095	T0000...T4092
Область счетчиков	C0000...C4092	C0000...C4095	C0000...C4092

Область	S	D	C
Область DM	D0...D32764	D0...D32767	D0...D32764
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767		
Постоянные	---		
Регистры данных	---		
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Данная команда отправляет запрос на передачу информации/состояния в форме явного сообщения (содержащегося в словах S+1...S+3) узлу, адрес которого указан в S, через модуль связи, адрес FINS которого указан в битах 00...07 слова C+1.

После получения ответа на отправленное явное сообщение в группу слов, начиная со слова D+1, записываются данные службы, содержащиеся в ответе (данные, следующие за кодом службы).

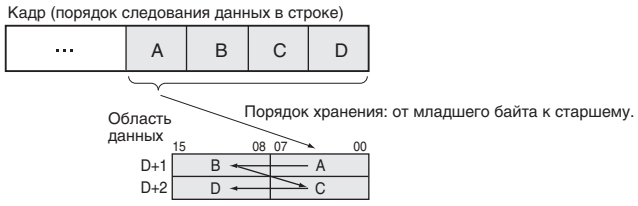
Указываемое в слове D количество байтов принятых данных — это количество байтов данных службы. (Например, если данные службы содержат 1 байт, то D содержит значение 0001 hex. D будет содержать 0001 hex независимо от выбранного порядка расположения байтов (т. е. независимо от того, в каком байте (младшем или старшем) слова D будет храниться байт данных службы)).
Порядок, в котором байты данных службы располагаются в словах S+6... и D+3..., определяется содержимым битов 12...15 слова C+1 (0 или 8 hex).

- Хранение данных в порядке от старшего байта к младшему
  В биты 12...15 слова C+1 следует записать 0 hex.



Прим.: A, B, C и D — обозначения байтов данных.

- Хранение данных в порядке от младшего байта к старшему
  В биты 12...15 слова C+1 следует записать 8 hex.



Прим.: A, B, C и D — обозначения байтов данных.

Флаги

Название	Обо-значение	Действие
Флаг ошибки	ER	Включен, если для порта связи, номер которого указан в C, выключен флаг доступности порта связи. Выключен во всех остальных случаях.

При успешном выполнении команды соответствующий флаг ошибки обмена явными сообщениями будет выключен, в случае ошибки — включен.

В случае ошибки (т. е. когда включен соответствующий бит в слове A213) соответствующий флаг ошибки порта связи позволяет определить, что именно произошло: явное сообщение не было передано вообще (включен соответствующий бит в A219) или сообщение было передано, но содержало ошибку (выключен соответствующий бит в A219).

При успешном выполнении команды соответствующий код завершения для порта связи (A203...A210) будет иметь значение 0000 hex, а в случае ошибки в нем будет содержаться код ошибки явного сообщения (при ошибке передачи явного сообщения) либо код ошибки FINS (при возникновении ошибки FINS).

Работа команд передачи явных сообщений подробна описана в разделе *3-24-2 Сведения о командах передачи явных сообщений (только для CP1H)*.

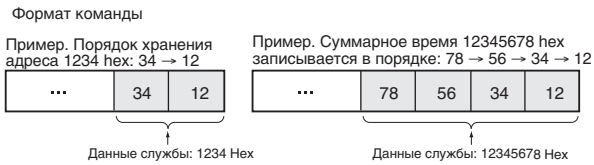
В следующей таблице перечислены сопутствующие биты и флаги вспомогательной области.

Название	Адрес	Действие
Флаг доступности порта связи	A202.00... A202.07	Этот флаг устанавливается в знак того, что для порта связи с соответствующим номером (00...07) возможно выполнение сетевых команд, включая PMCR(260). Во время выполнения сетевой команды с использованием соответствующего порта этот флаг выключен. После того как команда выполнена, этот флаг включается вновь.
Флаг ошибки обмена явными сообщениями	A213.00... A213.07	Этот флаг устанавливается в знак того, что в процессе обмена явными сообщениями с использованием порта связи с соответствующим номером (00...07) произошла ошибка. Флаг устанавливается, если явное сообщение не было отправлено вообще либо было отправлено, но вернулся ответ с кодом ошибки. Состояние этого флага сохраняется вплоть до выполнения следующей команды передачи явного сообщения. При выполнении следующей команды этот флаг сбрасывается, даже если ранее произошла ошибка.

Название	Адрес	Действие
Флаг ошибки порта связи	A219.00... A219.07	Этот флаг устанавливается в знак того, что при выполнении команды передачи явного сообщения с использованием порта связи с соответствующим номером (00...07) явное сообщение передано не было. Состояние этого флага сохраняется вплоть до выполнения следующей сетевой команды. При выполнении следующей команды этот флаг сбрасывается, даже если ранее произошла ошибка.
Коды завершения для портов связи	A203... A210	После выполнения команды сетевого обмена данными в этих словах содержатся коды завершения для соответствующих портов (00...07). Пока флаг ошибки обмена явными сообщениями для некоторого порта выключен, слово кода завершения для этого порта содержит значение 0000. Если для некоторого порта одновременно включены флаг ошибки обмена явными сообщениями и флаг ошибки порта связи, слово кода завершения для этого порта содержит код ошибки FINS. Если флаг ошибки обмена явными сообщениями для некоторого порта включен, тогда как флаг ошибки порта связи выключен, слово кода завершения для этого порта содержит соответствующий код ошибки передачи явного сообщения. Непосредственно во время выполнения сетевой команды соответствующее слово содержит значение 0000, по завершении выполнения команды в это слово записывается код завершения. Содержимое этих слов очищается в начале выполнения программы.

Меры предосторожности

Проследите, чтобы порядок байтов в исходных данных совпадал с порядком следования байтов в кадре явного сообщения («порядок следования данных в строке» на рисунке ниже). Например, когда данные службы состоят из 2-байтовых или 4-байтовых элементов, байты данных (пары из двух разрядов) в кадре сообщения располагаются слева направо, как показано на следующем рисунке.



На приведенных ниже рисунках показан порядок записи байтов данных в ячейки памяти данных, когда данные службы состоят из элементов длиной в 2 или 4 байта.

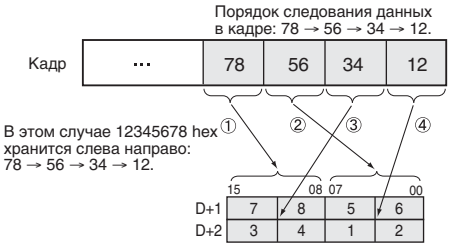
1. Данные, состоящие из 2-байтовых элементов.
   - Хранение данных в порядке от старшего байта к младшему (биты 12...15 в C = 0 hex)
     Пример: хранение значения 1234 hex в D+1.



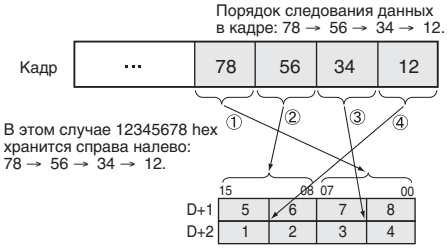
   - Хранение данных в порядке от младшего байта к старшему (биты 12...15 в C = 8 hex)
     Пример: хранение значения 1234 hex в D+1.



2. Данные, состоящие из 4-байтовых элементов.
   - Хранение данных в порядке от старшего байта к младшему (биты 12...15 в C = 0 hex)
     Пример: хранение значения 12345678 hex в D+1 и D+2.

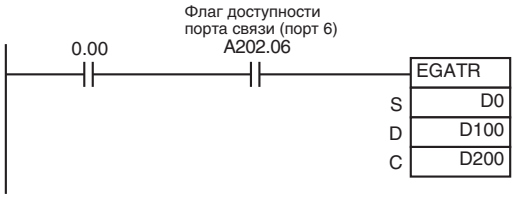


   - Хранение данных в порядке от младшего байта к старшему (биты 12...15 в C = 8 hex)
     Пример: хранение значения 12345678 hex в D+1 и D+2.



Пример

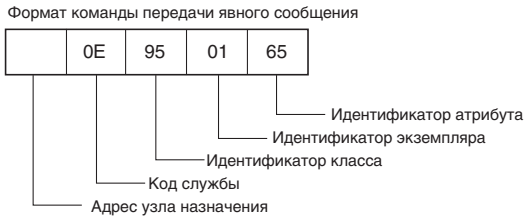
В данном примере с помощью команды EGATR(721) из ведомого устройства DRT2 (терминал ввода-вывода) считываются данные об общем состоянии.

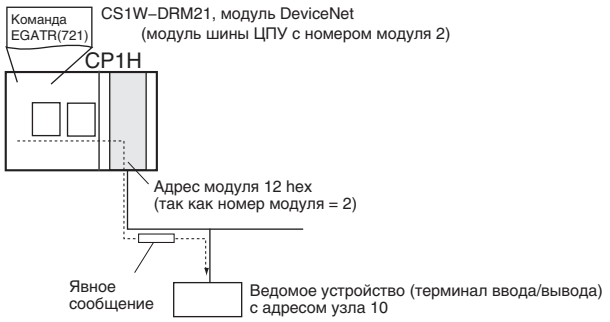


Когда бит CIO 0.00 и бит A202.06 (флаг доступности порта связи для порта 06) оба включены, команда EGATR(721) считывает из ведомого устройства DRT2 (терминал ввода-вывода) данные об общем состоянии. В данном примере считывается общая продолжительность пребывания во включенном состоянии или количество переключений контактов для входа 3.

Код службы = 0E hex, идентификатор класса = 95 hex, идентификатор экземпляра = 01 hex, идентификатор атрибута = 65 hex.

В ответ возвращается 1 байт данных с информацией об общем состоянии.

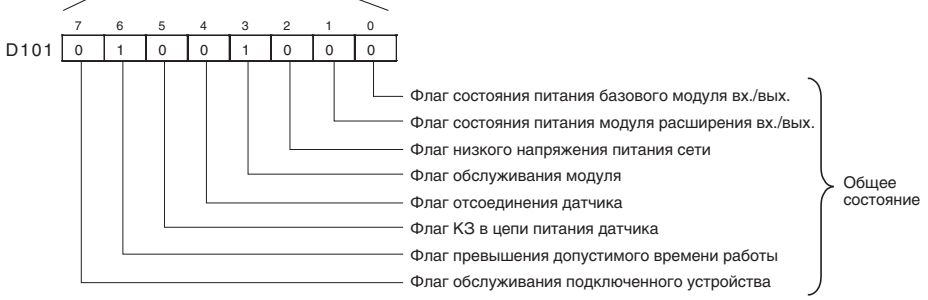




S:	D0	0	0	0	A	Адрес ведомого узла = 10 = 0A hex
S+1:	D1	0	0	9	5	Идентификатор класса = 95 hex
S+2:	D2	0	0	0	1	Идентификатор экземпляра = 01 hex
S+3:	D3	0	0	6	5	Идентификатор атрибута = 65 hex

C:	D200	0	0	0	2	Содержит 0002 hex, так как D...D+1 = 2 слова.
C+1:	D201	8	6	1	2	Порядок байтов = 8 hex (от младшего к старшему), порт связи = 6 hex (порт 6), адрес модуля DeviceNet = 12 hex
C+2:	D202	0	0	0	0	Время ожидания ответа = 0000 hex (2 с)
C+3:	D203	0	0	0	0	Тип формата явного сообщения = 0000 hex (формат DeviceNet)

D:	D100	0	0	0	1	D содержит 0 hex: возвращается 1 байт данных в младший байт слова D+1. В биты 00...07 возвращаются данные об общем состоянии ведомого устройства. (Данные записываются в биты 00...07, так как выбран порядок от младшего к старшему: биты 12...15 слова C+1 = 8 hex).
D+1:	D101	0	0	4	8	



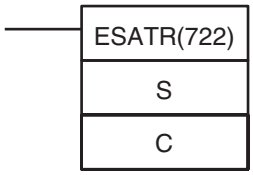
3-24-8 ЯВНАЯ УСТАНОВКА АТРИБУТА: ESATR(722)

Назначение

Передача команды записи информации в форме явного сообщения (УСТАНОВКА ОДИНОЧНОГО АТРИБУТА, код службы: 10 hex).

Эту команду возможно использовать только для модулей ЦПУ CP1H. Для модулей ЦПУ CP1L она недоступна. Если команда ESATR(722) будет использована для модуля ЦПУ CP1L, будет установлен флаг ошибки (ER).

Символ РКС



S: Первое слово передаваемого сообщения

C: Первое управляющее слово

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	ESATR(722)
	Однократное выполнение по положительному фронту	@ESATR(722)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

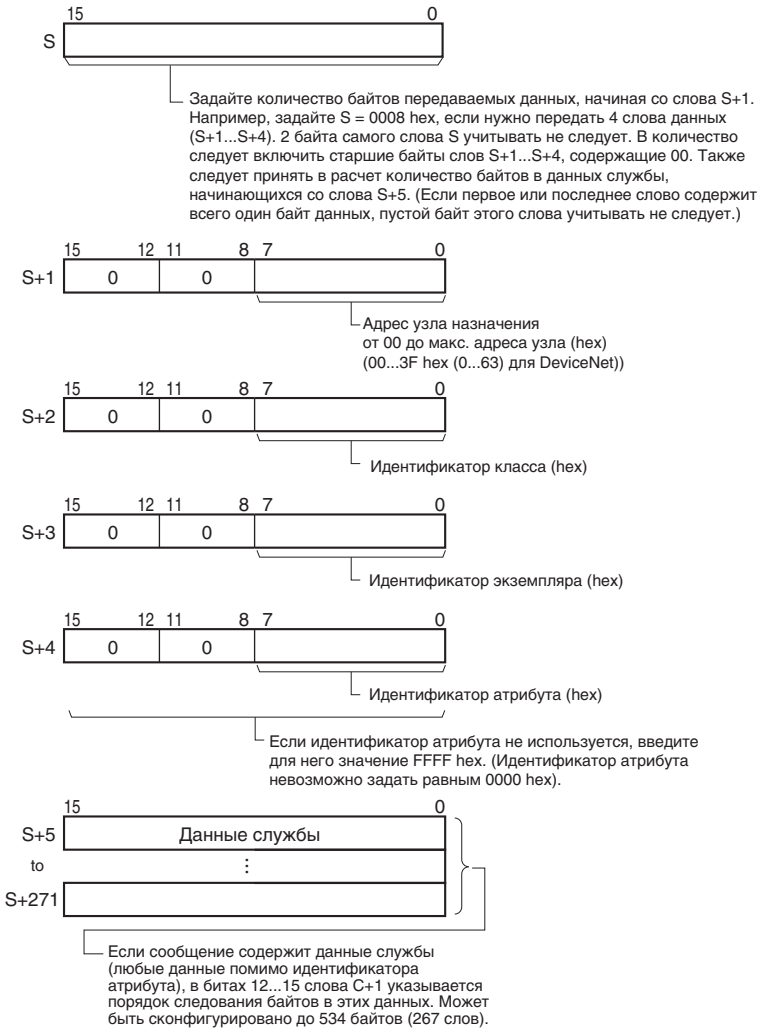
Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

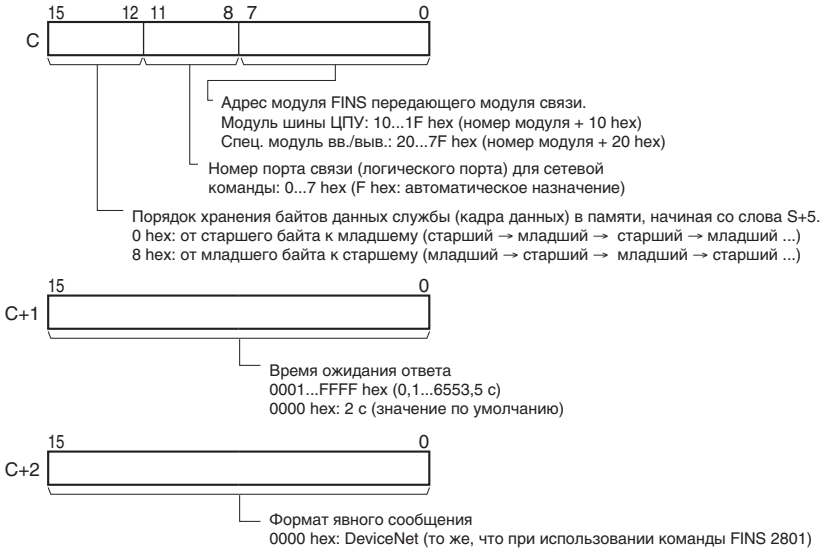
S: Первое слово передаваемого сообщения

Указывает первое слово передаваемого сообщения (S... макс. S+271).



C: Первое управляющее слово

Указывает первое из трех управляющих слов (C...C+2).



Характеристики операндов

Область	S	C
Область CIO	CIO 0...CIO 6143	CIO 0...CIO 6141
Рабочая область	W0...W511	W0...W509
Область битов хранения	H0...H511	H0...H509
Область вспомогательных битов	A0...A959	A0...A957
Область таймеров	T0000...T4095	T0000...T4093
Область счетчиков	C0000...C4095	C0000...C4093
Область DM	D0...D32767	D0...D32765
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767	
Постоянные	---	
Регистры данных	---	
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

Описание

Данная команда передает явное сообщение с кодом службы 10 hex (содержащееся в группе слов, начиная с S+2) узлу, адрес которого указан в S+1, через модуль связи, адрес FINS которого указан в битах 00...07 слова C. Данные, возвращаемые в ответ на отправленное явное сообщение, записываются в группу слов, начиная со слова D+2.

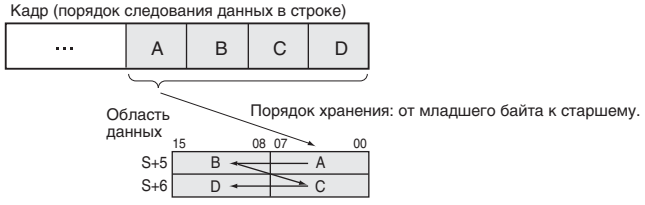
Порядок расположения байтов данных службы в словах S+5 определяется содержимым битов 12...15 слова C (0 или 8 hex).

- Хранение данных в порядке от старшего байта к младшему
  B биты 12...15 слова C следует записать 0 hex.



Прим.: A, B, C и D — обозначения байтов данных.

- Хранение данных в порядке от младшего байта к старшему
  B биты 12...15 слова C следует записать 8 hex.



Прим.: A, B, C и D — обозначения байтов данных.

Флаги

Название	Обо-значение	Действие
Флаг ошибки	ER	Включен, если для порта связи, номер которого указан в C, выключен флаг доступности порта связи. Выключен во всех остальных случаях.

При успешном выполнении команды соответствующий флаг ошибки обмена явными сообщениями будет выключен, в случае ошибки — включен.

В случае ошибки (т. е. когда включен соответствующий бит в слове A213) соответствующий флаг ошибки порта связи позволяет определить, что именно произошло: явное сообщение не было передано вообще (включен соответствующий бит в A219) или сообщение было передано, но содержало ошибку (выключен соответствующий бит в A219).

При успешном выполнении команды соответствующий код завершения для порта связи (A203...A210) будет иметь значение 0000 hex, а в случае ошибки в нем будет содержаться код ошибки явного сообщения (при ошибке передачи явного сообщения) либо код ошибки FINS (при возникновении ошибки FINS).

Работа команд передачи явных сообщений подробна описана в разделе *3-24-2 Сведения о командах передачи явных сообщений (только для CP1H)*.

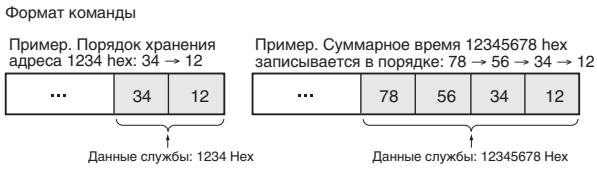
В следующей таблице перечислены сопутствующие биты и флаги вспомогательной области.

Название	Адрес	Действие
Флаг доступности порта связи	A202.00... A202.07	Этот флаг устанавливается в знак того, что для порта связи с соответствующим номером (00...07) возможно выполнение сетевых команд, включая PMCR(260). Во время выполнения сетевой команды с использованием соответствующего порта этот флаг выключен. После того как команда выполнена, этот флаг включается вновь.
Флаг ошибки обмена явными сообщениями	A213.00... A213.07	Этот флаг устанавливается в знак того, что в процессе обмена явными сообщениями с использованием порта связи с соответствующим номером (00...07) произошла ошибка. Флаг устанавливается, если явное сообщение не было отправлено вообще либо было отправлено, но вернулся ответ с кодом ошибки. Состояние этого флага сохраняется вплоть до выполнения следующей команды передачи явного сообщения. При выполнении следующей команды этот флаг сбрасывается, даже если ранее произошла ошибка.

Название	Адрес	Действие
Флаг ошибки порта связи	A219.00... A219.07	Этот флаг устанавливается в знак того, что при выполнении команды передачи явного сообщения с использованием порта связи с соответствующим номером (00...07) явное сообщение передано не было. Состояние этого флага сохраняется вплоть до выполнения следующей сетевой команды. При выполнении следующей команды этот флаг сбрасывается, даже если ранее произошла ошибка.
Коды завершения для портов связи	A203... A210	После выполнения команды сетевого обмена данными в этих словах содержатся коды завершения для соответствующих портов (00...07). Пока флаг ошибки обмена явными сообщениями для некоторого порта выключен, слово кода завершения для этого порта содержит значение 0000. Если для некоторого порта одновременно включены флаг ошибки обмена явными сообщениями и флаг ошибки порта связи, слово кода завершения для этого порта содержит код ошибки FINS. Если флаг ошибки обмена явными сообщениями для некоторого порта включен, тогда как флаг ошибки порта связи выключен, слово кода завершения для этого порта содержит соответствующий код ошибки передачи явного сообщения. Непосредственно во время выполнения сетевой команды соответствующее слово содержит значение 0000, по завершении выполнения команды в это слово записывается код завершения. Содержимое этих слов очищается в начале выполнения программы.

Меры предосторожности

Проследите, чтобы порядок байтов в исходных данных совпадал с порядком следования байтов в кадре явного сообщения («порядок следования данных в строке» на рисунке ниже). Например, когда данные службы состоят из 2-байтовых или 4-байтовых элементов, байты данных (пары из двух разрядов) в кадре сообщения располагаются слева направо, как показано на следующем рисунке.

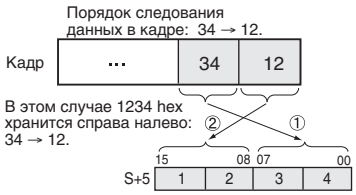


На приведенных ниже рисунках показан порядок записи байтов данных в ячейки памяти данных, когда данные службы состоят из элементов длиной в 2 или 4 байта.

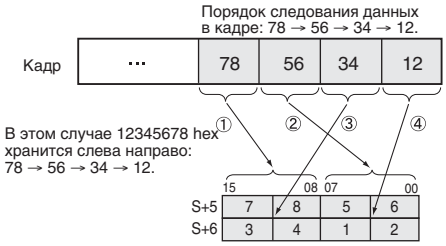
1. Данные, состоящие из 2-байтовых элементов.
   - Хранение данных в порядке от старшего байта к младшему (биты 12...15 в C = 0 hex)
     Пример: хранение значения 1234 hex в S+5.



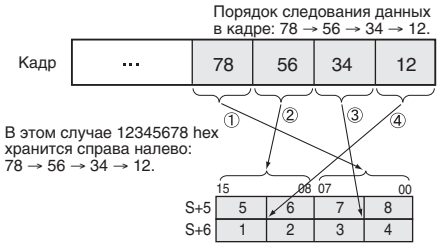
   - Хранение данных в порядке от младшего байта к старшему (биты 12...15 в C = 8 hex)
     Пример: хранение значения 1234 hex в S+5.



2. Данные, состоящие из 4-байтовых элементов.
   - Хранение данных в порядке от старшего байта к младшему (биты 12...15 в C = 0 hex)
     Пример: хранение значения 12345678 hex в S+5 и S+6.

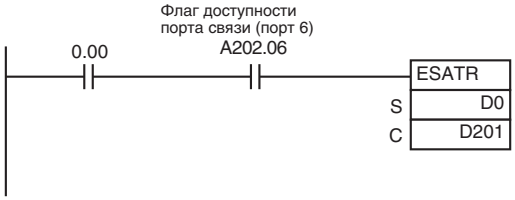


   - Хранение данных в порядке от младшего байта к старшему (биты 12...15 в C = 8 hex)
     Пример: хранение значения 12345678 hex в S+5 и S+6.



Пример

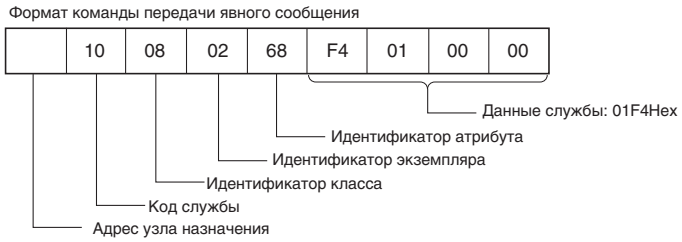
В данном примере с помощью команды ESATR(722) в ведомом устройстве DRT2 (терминал ввода-вывода) устанавливается новое значение количества переключений контактов.

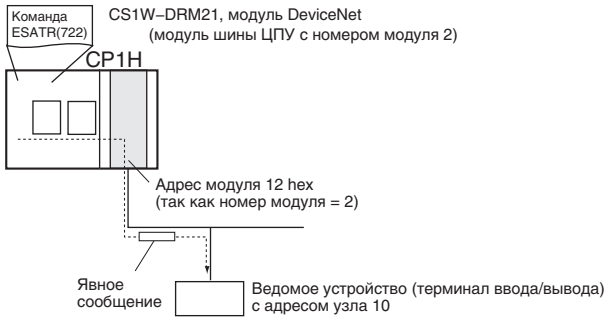


Когда бит CIO 0.00 и бит A202.06 (флаг доступности порта связи для порта 06) оба включены, команда ESATR(722) устанавливает в ведомом устройстве DRT2 (терминал ввода-вывода) новое значение количества переключений контактов для входа 2.

Код службы = 10 hex, идентификатор класса = 08 hex, идентификатор экземпляра = 02 hex, идентификатор атрибута = 68 hex.

В данном случае количество переключений контактов устанавливается равным 500 (1F4 hex), поэтому данные службы должны содержать значение 000001F4.





S	D0	0	0	0	C	Количество байтов данных: S+1...S+6 = 6 слов = 12 байтов = 0C hex
S:+1	D1	0	0	0	A	Адрес ведомого узла = 10 = 0A hex
S+2:	D2	0	0	0	8	Идентификатор класса = 08 hex
S+3:	D3	0	0	0	2	Идентификатор экземпляра = 02 hex
S+4:	D4	0	0	6	8	Идентификатор атрибута = 68 hex
S+5:	D5	0	1	F	4	Данные службы = F401 hex
S+6:	D6	0	0	0	0	

C:	D201	8	6	1	2	Порядок байтов = 8 hex (от младшего к старшему), порт связи = 6 hex (порт 6), адрес модуля DeviceNet = 12 hex
C+1:	D202	0	0	0	0	Время ожидания ответа = 0000 hex (2 с)
C+2:	D203	0	0	0	0	Тип формата явного сообщения = 0000 hex (формат DeviceNet)

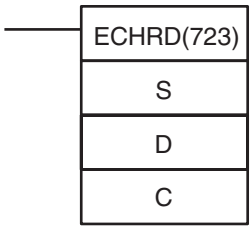
3-24-9 ЯВНОЕ ЧТЕНИЕ СЛОВА: ECHRD(723)

Назначение

Чтение данных в локальный модуль ЦПУ из другого модуля ЦПУ в сети.

Эту команду возможно использовать только для модулей ЦПУ CP1H. Для модулей ЦПУ CP1L она недоступна. Если команда ECHRD(723) будет использована для модуля ЦПУ CP1L, будет установлен флаг ошибки (ER).

Символ РКС



S: Первое исходное слово в удаленном модуле ЦПУ

D: Первое слово назначения в локальном модуле ЦПУ CP1H

C: Первое управляющее слово

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	ECHRD(723)
	Однократное выполнение по положительному фронту	@ECHRD(723)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

S: Первое запрашиваемое слово в удаленном модуле ЦПУ

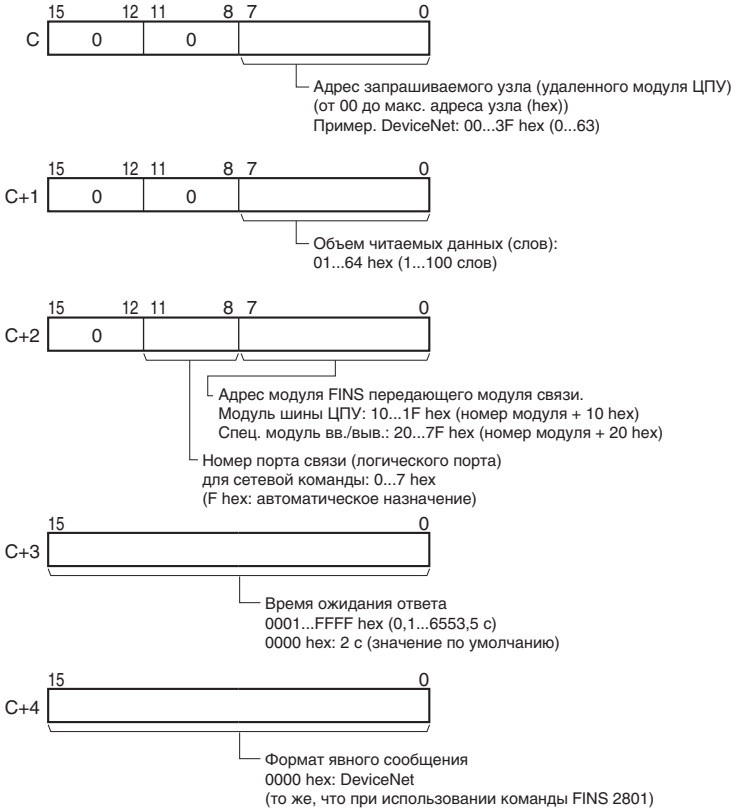
Указывает адрес самого первого слова последовательности слов данных, которые должны быть прочитаны из удаленного модуля ЦПУ.

D: Первое записываемое слово в локальном модуле ЦПУ

Указывает адрес самого первого слова последовательности слов данных в локальном модуле ЦПУ, куда должны быть записаны прочитанные данные.

C: Первое управляющее слово

Указывает первое из пяти управляющих слов (C...C+4).



Характеристики операндов

Область	S	D	C
Область CIO	CIO 0...CIO 6143		CIO 0...CIO 6139
Рабочая область	W0...W511		W0...W507
Область битов хранения	H0...H511		H0...H507
Область вспомогательных битов	A0...A959	A448...A959	A0...A955
Область таймеров	T0000...T4095		T0000...T4091
Область счетчиков	C0000...C4095		C0000...C4091
Область DM	D0...D32767		D0...D32763
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767		
Постоянные	---		
Регистры данных	---		
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Данная команда осуществляет чтение указанного количества слов, начиная со слова, адрес которого указан в S, из удаленного модуля ЦПУ, адрес узла которого указан в C, и записывает полученные данные в слова памяти локального модуля ЦПУ, начиная со слова D.

Примечание. Команда ECHRD(723) передает явное сообщение с кодом службы 1C hex (ЧТЕНИЕ БАЙТОВЫХ ДАННЫХ).

Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	Включен, если для порта связи, номер которого указан в C, выключен флаг доступности порта связи. Выключен во всех остальных случаях.

При успешном выполнении команды соответствующий флаг ошибки обмена явными сообщениями будет выключен, в случае ошибки — включен.

В случае ошибки (т. е. когда включен соответствующий бит в слове A213) соответствующий флаг ошибки порта связи позволяет определить, что именно произошло: явное сообщение не было передано вообще (включен соответствующий бит в A219) или сообщение было передано, но содержало ошибку (выключен соответствующий бит в A219).

При успешном выполнении команды соответствующий код завершения для порта связи (A203...A210) будет иметь значение 0000 hex, а в случае ошибки в нем будет содержаться код ошибки явного сообщения (при ошибке передачи явного сообщения) либо код ошибки FINS (при возникновении ошибки FINS).

Работа сетевых команд подробна описана в разделе *3-24-2 Сведения о командах передачи явных сообщений (только для CP1H)*.

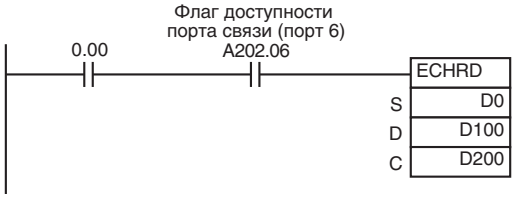
В следующей таблице перечислены сопутствующие биты и флаги вспомогательной области.

Название	Адрес	Действие
Флаг доступности порта связи	A202.00... A202.07	Этот флаг устанавливается в знак того, что для порта связи с соответствующим номером (00...07) возможно выполнение сетевых команд, включая PMCR(260). Во время выполнения сетевой команды с использованием соответствующего порта этот флаг выключен. После того как команда выполнена, этот флаг включается вновь.
Флаг ошибки обмена явными сообщениями	A213.00... A213.07	Этот флаг устанавливается в знак того, что в процессе обмена явными сообщениями с использованием порта связи с соответствующим номером (00...07) произошла ошибка. Флаг устанавливается, если явное сообщение не было отправлено вообще либо было отправлено, но вернулся ответ с кодом ошибки. Состояние этого флага сохраняется вплоть до выполнения следующей команды передачи явного сообщения. При выполнении следующей команды этот флаг сбрасывается, даже если ранее произошла ошибка.

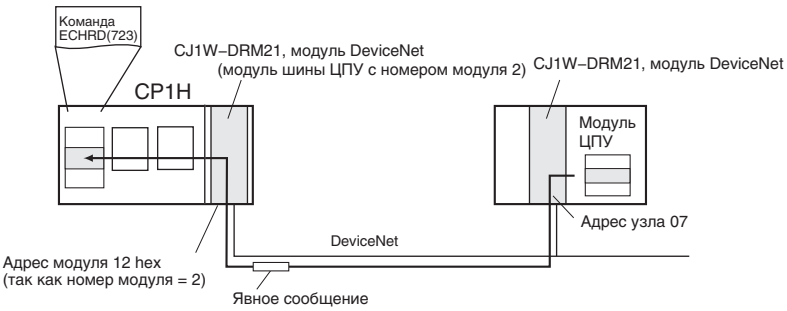
Название	Адрес	Действие
Флаг ошибки порта связи	A219.00... A219.07	Этот флаг устанавливается в знак того, что при выполнении команды передачи явного сообщения с использованием порта связи с соответствующим номером (00...07) явное сообщение передано не было. Состояние этого флага сохраняется вплоть до выполнения следующей сетевой команды. При выполнении следующей команды этот флаг сбрасывается, даже если ранее произошла ошибка.
Коды завершения для портов связи	A203... A210	После выполнения команды сетевого обмена данными в этих словах содержатся коды завершения для соответствующих портов (00...07). Пока флаг ошибки обмена явными сообщениями для некоторого порта выключен, слово кода завершения для этого порта содержит значение 0000. Если для некоторого порта одновременно включены флаг ошибки обмена явными сообщениями и флаг ошибки порта связи, слово кода завершения для этого порта содержит код ошибки FINS. Если флаг ошибки обмена явными сообщениями для некоторого порта включен, тогда как флаг ошибки порта связи выключен, слово кода завершения для этого порта содержит соответствующий код ошибки передачи явного сообщения. Непосредственно во время выполнения сетевой команды соответствующее слово содержит значение 0000, по завершении выполнения команды в это слово записывается код завершения. Содержимое этих слов очищается в начале выполнения программы.

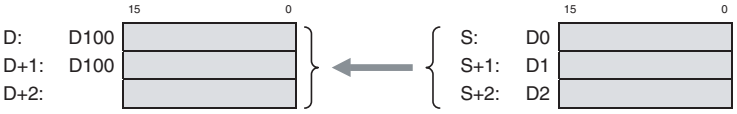
Пример

В данном примере команда ECHRD(723) используется для чтения содержимого слов памяти ввода/вывода модуля ЦПУ серии CJ в сети DeviceNet и записи полученных данных в слова памяти ввода/вывода локального модуля ЦПУ.



Когда бит CIO 0.00 и бит A202.06 (флаг доступности порта связи для порта 06) оба включены, команда ECHRD(723) читает содержимое слов D0...D2 из памяти ввода/вывода модуля ЦПУ серии CJ с адресом узла 7 сети DeviceNet и записывает полученные значения в слова D100...D102 локального модуля ЦПУ.





		15	8	7	0	
C:	D200	0	0	0	7	Адрес узла запрашиваемого удаленного модуля ЦПУ = 07 hex (узел 07)
C+1:	D201	0	0	0	3	Объем читаемых данных (кол–во слов) = 3 hex
C+2:	D202	0	6	1	2	Порт связи = 6 hex (порт 6), адрес модуля DeviceNet = 12 hex
C+3:	D203	0	0	0	0	Время ожидания ответа = 0000 hex (2 с)
C+4:	D204	0	0	0	0	Тип формата явного сообщения = 0000 hex (формат DeviceNet)

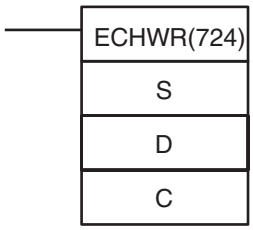
3-24-10 ЯВНАЯ ЗАПИСЬ СЛОВА: ECHWR(724)

Назначение

Запись данных из локального модуля ЦПУ в другой модуль ЦПУ в сети.

Эту команду возможно использовать только для модулей ЦПУ CP1H. Для модулей ЦПУ CP1L она недоступна. Если команда ECHWR(724) будет использована для модуля ЦПУ CP1L, будет установлен флаг ошибки (ER).

Символ РКС



S: Первое исходное слово в локальном модуле ЦПУ

D: Первое слово назначения в удаленном модуле ЦПУ

C: Первое управляющее слово

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	ECHWR(724)
	Однократное выполнение по положительному фронту	@ECHWR(724)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	**Области пошаговых программ**	**Подпрограм-мы**	**Задачи обработки прерываний**
OK	OK	OK	OK

Операнды

S: Первое исходное слово в локальном модуле ЦПУ

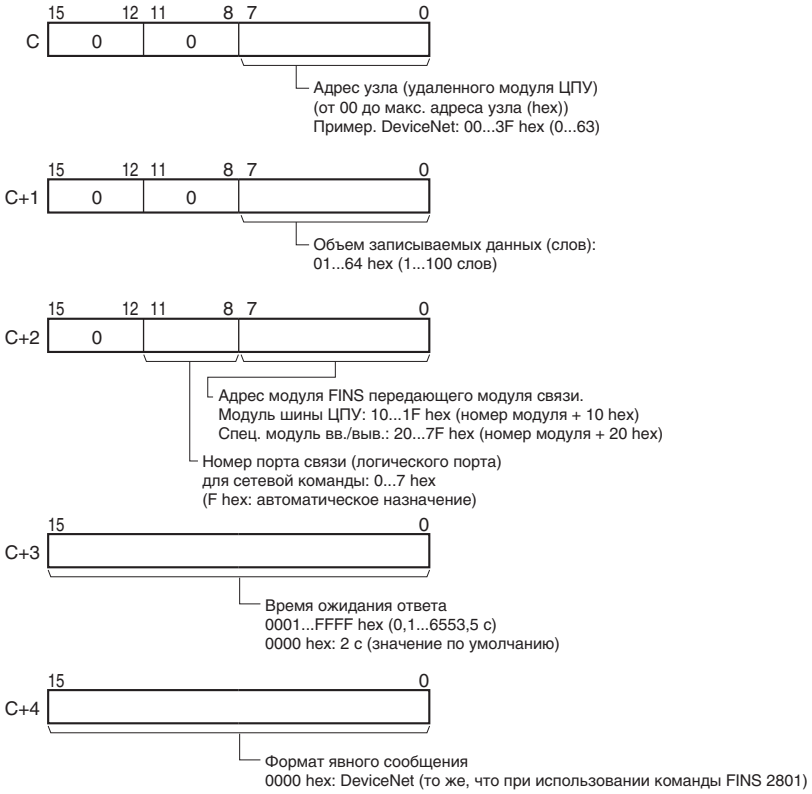
Указывает адрес самого первого слова последовательности слов данных в локальном модуле ЦПУ, содержащих записываемые данные.

D: Первое слово назначения в удаленном модуле ЦПУ

Указывает адрес самого первого слова последовательности слов данных в удаленном модуле ЦПУ, куда должны записываться передаваемые данные.

C: Первое управляющее слово

Указывает первое из пяти управляющих слов (C...C+4).



Характеристики операндов

Область	S	D	C
Область CIO	CIO 0...CIO 6143		CIO 0...CIO 6139
Рабочая область	W0...W511		W0...W507
Область битов хранения	H0...H511		H0...H507
Область вспомогательных битов	A0...A959	A448...A959	A0...A955
Область таймеров	T0000...T4095		T0000...T4091
Область счетчиков	C0000...C4095		C0000...C4091
Область DM	D0...D32767		D0...D32763
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767		
Постоянные	---		
Регистры данных	---		
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Данная команда передает содержимое указанного количества слов локального модуля ЦПУ, начиная со слова S, в предназначенную для записи область памяти, начиная с адреса D, расположенную в удаленном модуле ЦПУ, адрес узла которого указан в C.

Примечание. Команда ECHWR(724) передает явное сообщение с кодом службы 1E hex (ЗАПИСЬ БАЙТОВЫХ ДАННЫХ).

Флаги

Название	Обо-значение	Действие
Флаг ошибки	ER	Включен, если для порта связи, номер которого указан в C, выключен флаг доступности порта связи. Выключен во всех остальных случаях.

При успешном выполнении команды соответствующий флаг ошибки обмена явными сообщениями будет выключен, в случае ошибки — включен.

В случае ошибки (т. е. когда включен соответствующий бит в слове A213) соответствующий флаг ошибки порта связи позволяет определить, что именно произошло: явное сообщение не было передано вообще (включен соответствующий бит в A219) или сообщение было передано, но содержало ошибку (выключен соответствующий бит в A219).

При успешном выполнении команды соответствующий код завершения для порта связи (A203...A210) будет иметь значение 0000 hex, а в случае ошибки в нем будет содержаться код ошибки явного сообщения (при ошибке передачи явного сообщения) либо код ошибки FINS (при возникновении ошибки FINS).

Работа команд передачи явных сообщений подробна описана в разделе *3-24-2 Сведения о командах передачи явных сообщений (только для CP1H)*.

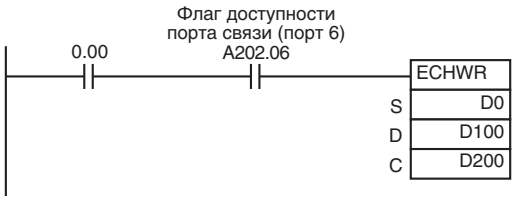
В следующей таблице перечислены сопутствующие биты и флаги вспомогательной области.

Название	Адрес	Действие
Флаг доступности порта связи	A202.00... A202.07	Этот флаг устанавливается в знак того, что для порта связи с соответствующим номером (00...07) возможно выполнение сетевых команд, включая PMCR(260). Во время выполнения сетевой команды с использованием соответствующего порта этот флаг выключен. После того как команда выполнена, этот флаг включается вновь.
Флаг ошибки обмена явными сообщениями	A213.00... A213.07	Этот флаг устанавливается в знак того, что в процессе обмена явными сообщениями с использованием порта связи с соответствующим номером (00...07) произошла ошибка. Флаг устанавливается, если явное сообщение не было отправлено вообще либо было отправлено, но вернулся ответ с кодом ошибки. Состояние этого флага сохраняется вплоть до выполнения следующей команды передачи явного сообщения. При выполнении следующей команды этот флаг сбрасывается, даже если ранее произошла ошибка.

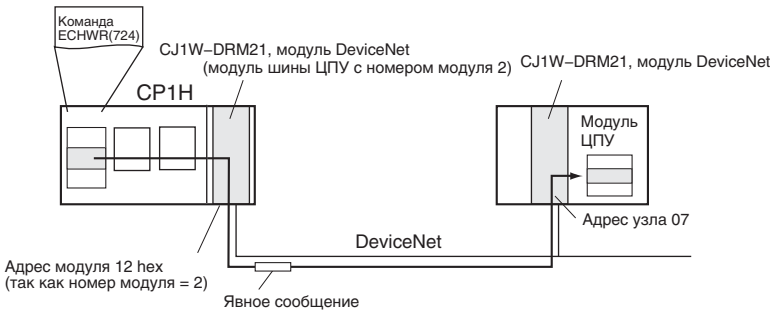
Название	Адрес	Действие
Флаг ошибки порта связи	A219.00... A219.07	Этот флаг устанавливается в знак того, что при выполнении команды передачи явного сообщения с использованием порта связи с соответствующим номером (00...07) явное сообщение передано не было. Состояние этого флага сохраняется вплоть до выполнения следующей сетевой команды. При выполнении следующей команды этот флаг сбрасывается, даже если ранее произошла ошибка.
Коды завершения для портов связи	A203... A210	После выполнения команды сетевого обмена данными в этих словах содержатся коды завершения для соответствующих портов (00...07). Пока флаг ошибки обмена явными сообщениями для некоторого порта выключен, слово кода завершения для этого порта содержит значение 0000. Если для некоторого порта одновременно включены флаг ошибки обмена явными сообщениями и флаг ошибки порта связи, слово кода завершения для этого порта содержит код ошибки FINS. Если флаг ошибки обмена явными сообщениями для некоторого порта включен, тогда как флаг ошибки порта связи выключен, слово кода завершения для этого порта содержит соответствующий код ошибки передачи явного сообщения. Непосредственно во время выполнения сетевой команды соответствующее слово содержит значение 0000, по завершении выполнения команды в это слово записывается код завершения. Содержимое этих слов очищается в начале выполнения программы.

Пример

В данном примере команда ECHWR(724) используется для записи содержимого слов данных памяти ввода/вывода локального модуля ЦПУ в слова памяти ввода/вывода модуля ЦПУ серии CJ в сети DeviceNet.



Когда бит CIO 0.00 и бит A202.06 (флаг доступности порта связи для порта 06) оба включены, команда ECHWR(724) читает содержимое слов D0...D2 в памяти ввода/вывода локального модуля ЦПУ и передает его в слова D100...D102 модуля ЦПУ серии CJ с адресом узла 7 в сети DeviceNet.



		15 8	7 0	
C:	D200	0 0	0 7	Адрес узла принимающего удаленного модуля ЦПУ = 07 hex (узел 07)
C+1:	D201	0 0	0 3	Объем записываемых данных (кол–во слов) = 3 hex
C+2:	D202	0 6	1 2	Порт связи = 6 hex (порт 6), адрес модуля DeviceNet = 12 hex
C+3:	D203	0 0	0 0	Время ожидания ответа = 0000 hex (2 с)
C+4:	D204	0 0	0 0	Тип формата явного сообщения = 0000 hex (формат DeviceNet)

3-25 Команды дисплея

В данном разделе описаны команды, предназначенные для отображения сообщений.

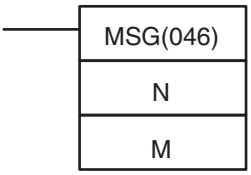
Команда	Мнемоническое обозначение	Код функции	Стр.
ОТОБРАЗИТЬ СООБЩЕНИЕ	MSG	046	997
ОТОБРАЗИТЬ ЗНАЧЕНИЕ СЛОВА НА 7-СЕГМЕНТНОМ ИНДИКАТОРЕ	SCH	047	1000
УПРАВЛЕНИЕ 7-СЕГМЕНТНЫМ ИНДИКАТОРОМ	SCTRL	048	1002

3-25-1 ОТОБРАЗИТЬ СООБЩЕНИЕ: MSG(046)

Назначение

Чтение указанных шестнадцати слов расширенной кодировки ASCII и отображение сообщения на экране устройства программирования.

Символ РКС



N: Номер сообщения

M: Первое слово сообщения

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	MSG(046)
	Однократное выполнение по положительному фронту	@MSG(046)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

N: Номер сообщения

Номер сообщения должен находиться в диапазоне от 0000 до 0007 hex (0...7 десятичн.).

M: Первое слово сообщения

При отображении сообщения операнд M указывает адрес первого из слов, содержащих ASCII-коды сообщения. При удалении сообщения операнд M может иметь любое шестнадцатеричное значение в диапазоне от 0000 до FFFF.

Характеристики операндов

Область	M	N
Область CIO	CIO 0...CIO 6143	
Рабочая область	W0...W511	
Область битов хранения	H0...H511	
Область вспомогательных битов	A0...A959	
Область таймеров	T0000...T4095	
Область счетчиков	C0000...C4095	
Область DM	D0...D32767	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767	
Постоянные	#0000...#0007 (двоичн.) или &0...&7	#0000...#FFFF (двоичн.)
Регистры данных	DR0...DR15	---
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

Описание

При включенном условии выполнения команда MSG(046) считывает 16 слов (M...M+15) с ASCII-кодами (до 32 символов, включая символ «null») и закрепляет их за сообщением, номер которого указан в N. Зарегистрированное таким образом сообщение будет отображаться на экране устройства программирования после любых сообщений об ошибках.

Содержание зарегистрированного сообщения, отображаемое на экране устройства программирования, можно изменить путем изменения содержимого слов в области хранения сообщения.

Для удаления зарегистрированного сообщения следует выполнить команду MSG(046), указав в операнде N номер удаляемого сообщения и записав в операнд M любую константу (от 0000 до FFFF).

Сообщение, зарегистрированное во время выполнения программы, остается на экране устройства программирования даже после прекращения выполнения программы. Однако при следующем запуске программы все сообщения удаляются.

Флаги

Название	Обо-значение	Операция
Флаг ошибки	ER	Включен, если содержимое S не находится в пределах 0000...0007 hex. Выключен во всех остальных случаях.

Меры предосторожности

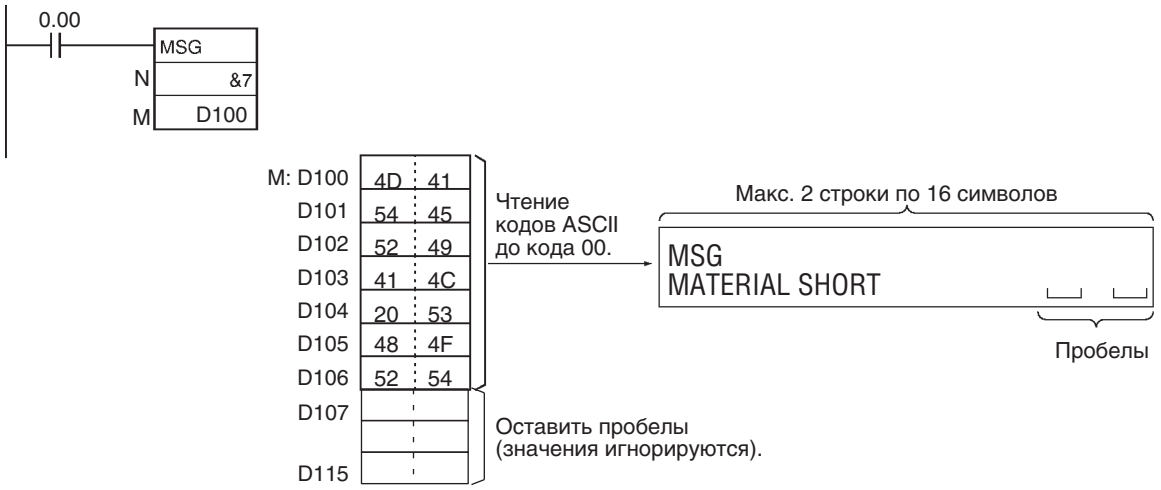
Зарегистрированные сообщения обновляются при каждом выполнении команды MSG(046).

Любые символы сообщения, расположенные после символа «null» (00), на экране устройства программирования представляются «пробелами».

Символ, содержащийся в старшем байте слова, отображается перед символом, содержащимся в младшем байте слова.

Примеры

Если бит CIO 0.00 включен, 32 ASCII-кода, содержащиеся в 16 словах данных (D100...D115), считываются, закрепляются за сообщением с номером 7 и отображаются на экране устройства программирования.



ASCII

Четыре младших бита \ Четыре старших бита	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0			SP	0	@	P	‘	p				ー	タ	ミ		
1			!	1	A	Q	a	q			。	ア	チ	ム		
2			”	2	B	R	b	r			「	イ	ツ	メ		
3			#	3	C	S	c	s			」	ウ	テ	モ		
4			$	4	D	T	d	t			、	エ	ト	ヤ		
5			%	5	E	U	e	u			・	オ	ナ	ユ		
6			&	6	F	V	f	v			ヲ	カ	ニ	ヨ		
7			’	7	G	W	g	w			ァ	キ	ヌ	ラ		
8			(	8	H	X	h	x			ィ	ク	ネ	リ		
9			)	9	I	Y	i	y			ゥ	ケ	ノ	ル		
A			*	:	J	Z	j	z			ェ	コ	ハ	レ		
B			+	;	K	[	k	{			ォ	サ	ヒ	ロ		
C			,	<	L	¥	l	|			ャ	シ	フ	ワ		
D			ー	=	M	]	m	}			ュ	ス	ヘ	ン		
E			.	>	N	^	n	~			ョ	セ	ホ	゛		
F			/	?	O	_	o				ッ	ソ	マ	°		

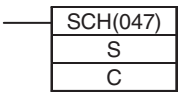
3-25-2 ОТОБРАЗИТЬ ЗНАЧЕНИЕ СЛОВА НА 7-СЕГМЕНТНОМ ИНДИКАТОРЕ: SCH(047)

Назначение

Команда SCH(047) отображает на 7-сегментном индикаторе модуля ЦПУ содержимое двух шестнадцатеричных разрядов, указанных пользователем.

Эту команду возможно использовать только для модулей ЦПУ CP1H. Для модулей ЦПУ CP1L она недоступна. Если команда SCH(047) будет использована для модуля ЦПУ CP1L, будет установлен флаг ошибки (ER).

Символ РКС



S: Слово отображаемых данных
C: Управляющее слово (выбор разряда)

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	SCH(047)
	Однократное выполнение по положительному фронту	@SCH(047)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

S: Слово отображаемых данных

C: Управляющее слово (выбор разряда)

0000 hex: Отобразить два старших разряда
0001 hex: Отобразить два младших разряда

Характеристики операндов

Область	S	C
Область CIO	CIO 0...CIO 6143	---
Рабочая область	W0...W511	---
Область битов хранения	H0...H511	---
Область вспомогательных битов	A0...A959	---
Область таймеров	T0000...T4095	---
Область счетчиков	C0000...C4095	---
Область DM	D0...D32767	---
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	---
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767	---
Постоянные	#0000...#FFFF	#0000, #0001
Регистры данных	DR0...DR15	---
Регистры указателей	---	---
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	---

Описание

При включенном условии выполнения команда SCH(047) отображает на 7-сегментном индикаторе модуля ЦПУ содержимое двух старших или младших шестнадцатеричных разрядов операнда S. Может быть отображено любое шестнадцатеричное значение в диапазоне от 00 до FF. Выбор отображаемой пары разрядов (старшей или младшей) производится с помощью операнда C.

Отображенное значение остается на индикаторе даже после выключения условия выполнения команды SCH(047). Для сброса индикации следует использовать команду SCTRL(048).

Если значение C не будет равно 0000 или 0001 hex, произойдет ошибка, будет установлен флаг ER и на индикаторе ничего не отобразится.

Флаги

Название	Обо-значение	Операция
Флаг ошибки	ER	Операнд C содержит значение, которое не равно 0000 или 0001 hex. Выключен во всех остальных случаях.

Меры предосторожности

Если в программе используется несколько команд SCTRL(048) или SCH(047), наибольшим приоритетом обладает команда, выполняемая самой последней.

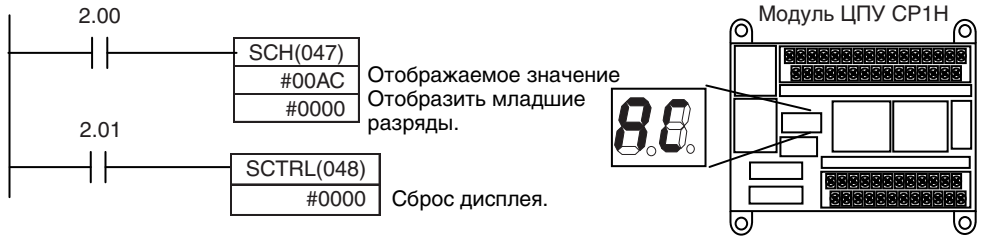
Сообщения на 7-сегментном индикаторе модуля ЦПУ отображаются с соблюдением указанного ниже порядка значимости. Из этого следует, что сообщение о системной ошибке обладает приоритетом по отношению к сообщению пользователя, созданному с помощью команды SCTRL(048) или SCH(047).

1. Сообщения об ошибках.
2. Сообщения SCTRL(048) или SCH(047).
3. Сообщения аналоговой регулировки.
4. Сообщения операций с картой памяти.

Если во время действия команды SCTRL(048) или SCH(047) в работе ПЛК возникнет ошибка, вместо информации команды SCTRL(048) или SCH(047) на индикаторе отобразится сообщение об ошибке. После устранения ошибки индикатор вернется к прежнему состоянию, установленному командой SCTRL(048) или SCH(047).

Пример

Если бит CIO 2.00 включен, на 7-сегментном индикаторе модуля ЦПУ отображается значение «AC». Это значение остается на индикаторе даже после выключения бита CIO 2.00. Индикация сбрасывается командой SCTRL(048), когда включается бит CIO 2.01.



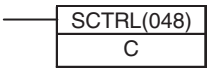
3-25-3 УПРАВЛЕНИЕ 7-СЕГМЕНТНЫМ ИНДИКАТОРОМ: SCTRL(048)

Назначение

Команда SCTRL(048) включает выбранные пользователем сегменты двухразрядного 7-сегментного индикатора модуля ЦПУ.

Эту команду возможно использовать только для модулей ЦПУ CP1H. Для модулей ЦПУ CP1L она недоступна. Если команда SCTRL(048) будет использована для модуля ЦПУ CP1L, будет установлен флаг ошибки выполнения команды (A295.08).

Символ РКС



C: Управляющие данные

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	SCTRL(048)
	Однократное выполнение по положительному фронту	@SCTRL(048)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

C: Управляющие данные

Каждый бит управляющего слова соответствует определенному сегменту одного из двух разрядов индикатора модуля ЦПУ.

ВКЛ: Включить разряд
ВЫКЛ: Не включать разряд

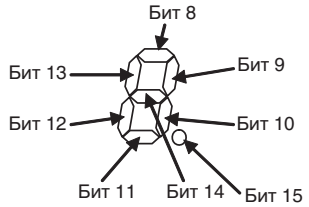
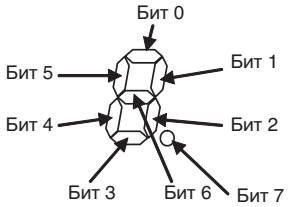
Характеристики операндов

Область	C
Область CIO	CIO 0...CIO 6143
Рабочая область	W0...W511
Область битов хранения	H0...H511
Область вспомогательных битов	A0...A959
Область таймеров	T0000...T4095
Область счетчиков	C0000...C4095
Область DM	D0...D32767
Косвенные адреса DM в двоичном формате	@ D0...@ D32767
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767
Постоянные	#0000...#FFFF
Регистры данных	DR0...DR15
Регистры указателей	---
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15

Описание

При включенном условии выполнения команда SCTRL(048) в соответствии со значением слова C включает или не включает (выключает) определенные сегменты двух разрядов 7-сегментного индикатора модуля ЦПУ. С помощью этой команды можно включить любую комбинацию сегментов.

Индикатор будет оставаться в состоянии, в которое он был переведен командой SCTRL(048), пока не будет выполнена команда SCTRL(048) со значением #0000 в операнде C, означающем гашение всех сегментов. Это также приведет к сбросу индикации, вызванной с помощью команды SCH(047). Системная индикация при этом не сбрасывается.

Левый разряд	Правый разряд
	

Флаги

Команда SCTRL(048) не изменяет состояние какого-либо флага.

Меры предосторожности

Если в программе используется несколько команд SCTRL(048) или SCH(047), наибольшим приоритетом обладает команда, выполняемая самой последней.

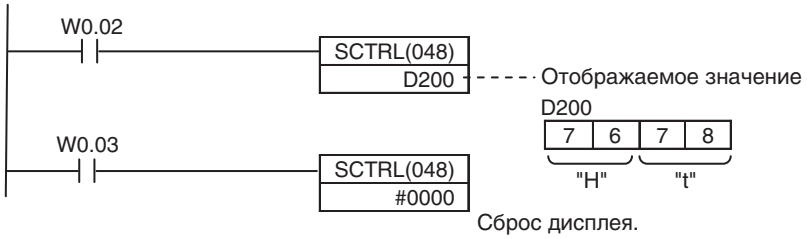
Сообщения на 7-сегментном индикаторе модуля ЦПУ отображаются с соблюдением указанного ниже порядка значимости. Из этого следует, что сообщение о системной ошибке обладает приоритетом по отношению к сообщению пользователя, созданному с помощью команды SCTRL(048) или SCH(047).

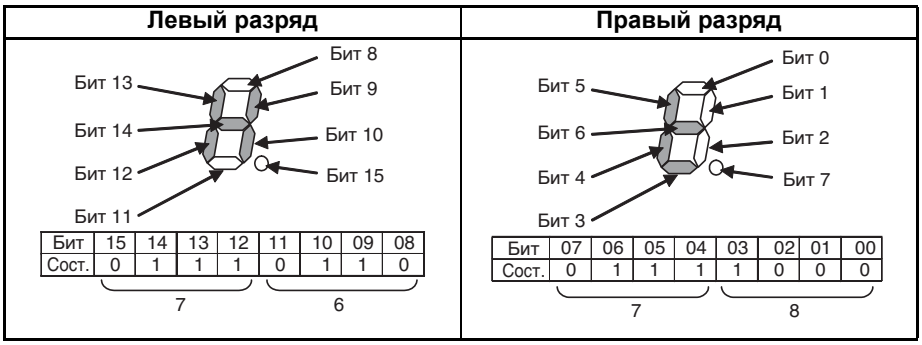
1. Сообщения об ошибках.
2. Сообщения SCTRL(048) или SCH(047).
3. Сообщения аналоговой регулировки.
4. Сообщения операций с картой памяти.

Если во время действия команды SCTRL(048) или SCH(047) в работе ПЛК возникнет ошибка, вместо информации команды SCTRL(048) или SCH(047) на индикаторе отобразится сообщение об ошибке. После устранения ошибки индикатор вернется к прежнему состоянию, установленному командой SCTRL(048) или SCH(047).

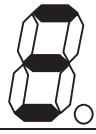
Пример

При включении бита W0.02 отдельные сегменты 7-сегментного индикатора модуля ЦПУ будут включены или выключены в соответствии с содержанием D200 (7678 = «Ht»). Индикатор останется в этом состоянии даже после выключения бита W0.02. Индикация будет сброшена (т. е. будут погашены сегменты) после включения бита W0.03.





Числовые значения и соответствующие им комбинации сегментов

0 (3F)	1 (06)	2 (5B)	3 (4F)	4 (66)	5 (6D)	6 (7D)	7 (27)	8 (7F)	9 (6F)
									
A (77)	B (7C)	C (39)	D (5E)	E (79)	F (71)	G (3D)	H (76)	I (19)	J (0D)
									
K (72)	L (38)	M (55)	N (54)	O (5C)	P (73)	Q (67)	R (50)	S (6D)	T (78)
									
U (3E)	V (1C)	W (6A)	X (1D)	Y (6E)	Z (49)				
									

3-26 Команды для работы с часами

В данном разделе описаны команды, предназначенные для работы с системными часами и данными в формате времени.

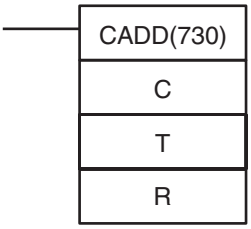
Команда	Мнемоническое обозначение	Код функции	Стр.
ДОБАВИТЬ К КАЛЕНДАРНОЙ ДАТЕ	CADD	730	1006
ВЫЧЕСТЬ ИЗ КАЛЕНДАРНОЙ ДАТЫ	CSUB	731	1009
ЧАСЫ В СЕКУНДЫ	SEC	065	1013
СЕКУНДЫ В ЧАСЫ	HMS	066	1015
КОРРЕКТИРОВКА ЧАСОВ	DATE	735	1017

3-26-1 ДОБАВИТЬ К КАЛЕНДАРНОЙ ДАТЕ: CADD(730)

Назначение

Добавление значения времени к календарной дате в указанных словах.

Символ РКС



C: Первое слово календаря

T: Первое слово времени

R: Первое слово результата

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	CADD(730)
	Однократное выполнение по положительному фронту	@CADD(730)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

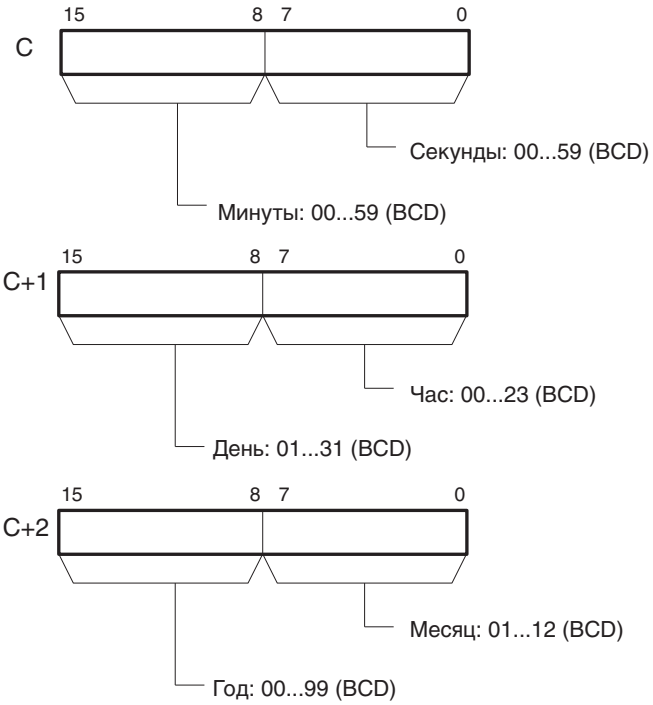
Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

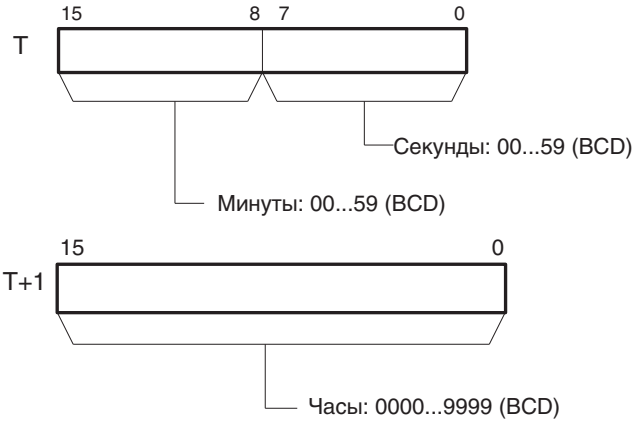
C...C+2: календарная дата и время

Задайте календарную дату и время в словах C...C+2, как показано на следующем рисунке.



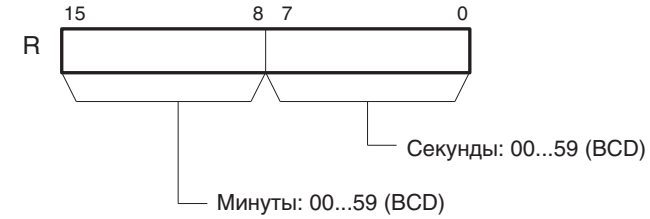
T и T+1: данные времени

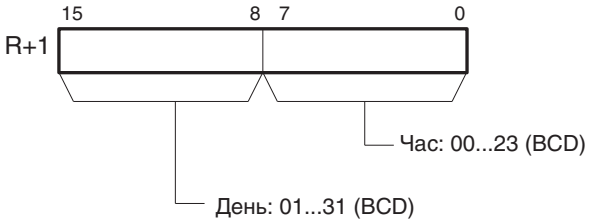
Задайте данные времени в словах T и T+1, как показано на следующем рисунке.

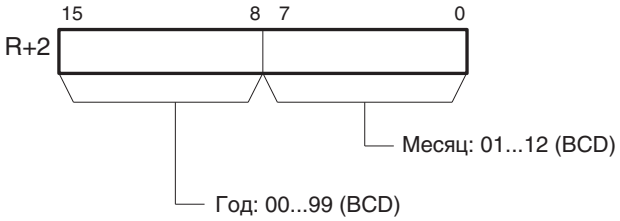


R...R+2: данные результата

Слова R...R+2 содержат результат операции сложения.





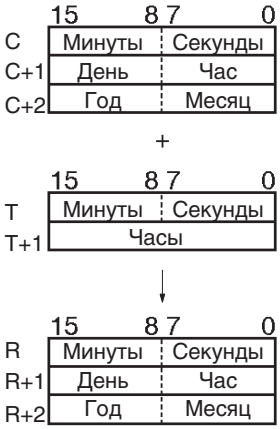


Характеристики операндов

Область	C	T	R
Область CIO	CIO 0...CIO 6141	CIO 0...CIO 6142	CIO 0...CIO 6141
Рабочая область	W0...W509	W0...W510	W0...W509
Область битов хранения	H0...H509	H0...H510	H0...H509
Область вспомогательных битов	A0...A957	A0...A958	A448...A957
Область таймеров	T0000...T4093	T0000...T4094	T0000...T4093
Область счетчиков	C0000...C4093	C0000...C4094	C0000...C4093
Область DM	D0...D32765	D0...D32766	D0...D32765
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767		
Постоянные	---	Только указанные значения	---
Регистры данных	---		
Регистры указателей	–		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR05+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда CADD(730) добавляет значение времени (в словах T и T+1) к календарной дате и времени (в словах C...C+2) и выдает полученную календарную дату и время в слова R...R+2.

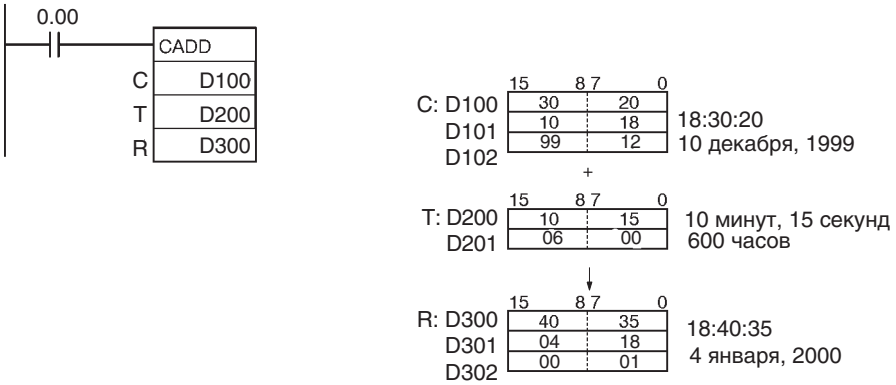


Флаги

Название	Обозначение	Операция
Флаг ошибки	ER	Включен, если календарные данные в словах C...C+2 выходят за указанные диапазоны. Включен, если данные времени в словах T и T+1 выходят за указанные диапазоны. Выключен во всех остальных случаях.

Примеры

Если бит CIO 0.00 включен, значение времени в словах D200 и D201 (часы, минуты, секунды) добавляется к календарной дате и времени в словах D100...D102 (год, месяц, день, часы, минуты, секунды), результат записывается в слова D300...D302.

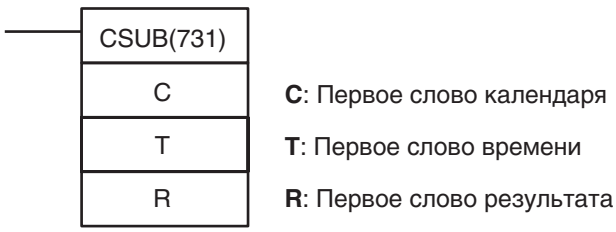


3-26-2 ВЫЧЕСТЬ ИЗ КАЛЕНДАРНОЙ ДАТЫ: CSUB(731)

Назначение

Вычитание значения времени из календарной даты и времени в указанных словах.

Символ РКС



C: Первое слово календаря

T: Первое слово времени

R: Первое слово результата

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	CSUB(731)
	Однократное выполнение по положительному фронту	@CSUB(731)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

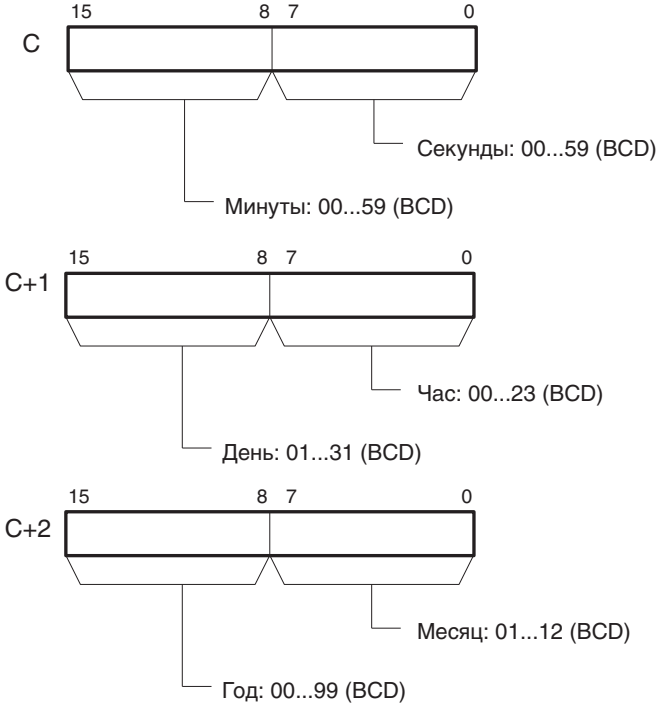
Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

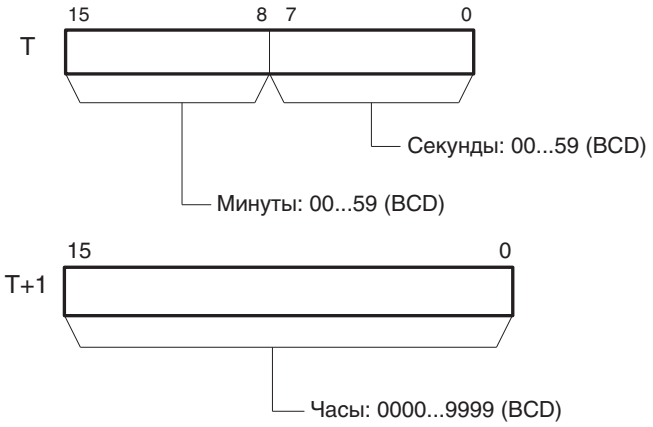
C...C+2: календарная дата и время

Задайте календарную дату и время в словах C...C+2, как показано на следующем рисунке.



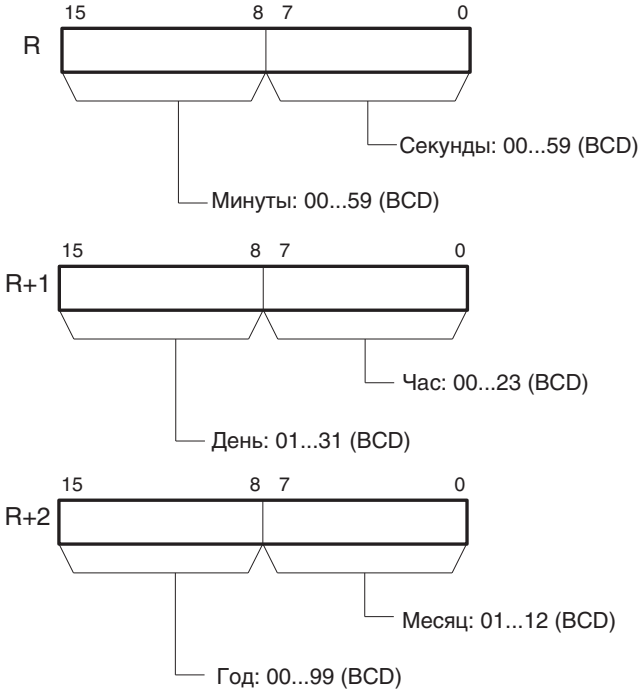
T и T+1: данные времени

Задайте данные времени в словах T и T+1, как показано на следующем рисунке.



R...R+2: данные результата

Слова R...R+2 содержат результат операции сложения.



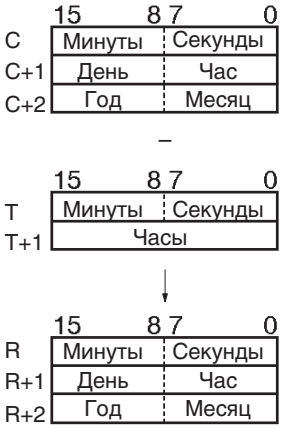
Характеристики операндов

Область	C	T	R
Область CIO	CIO 0...CIO 6141	CIO 0...CIO 6142	CIO 0...CIO 6141
Рабочая область	W0...W509	W0...W510	W0...W509
Область битов хранения	H0...H509	H0...H510	H0...H509
Область вспомогательных битов	A0...A957	A0...A958	A448...A957
Область таймеров	T0000...T4093	T0000...T4094	T0000...T4093
Область счетчиков	C0000...C4093	C0000...C4094	C0000...C4093
Область DM	D0...D32765	D0...D32766	D0...D32765
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767		

Область	C	T	R
Постоянные	---	Только указанные значения	---
Регистры данных	---		
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR05+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда CSUB(731) вычитает значение времени (в словах T и T+1) из календарной даты и времени (в словах C...C+2) и выдает полученную календарную дату и время в слова R...R+2.

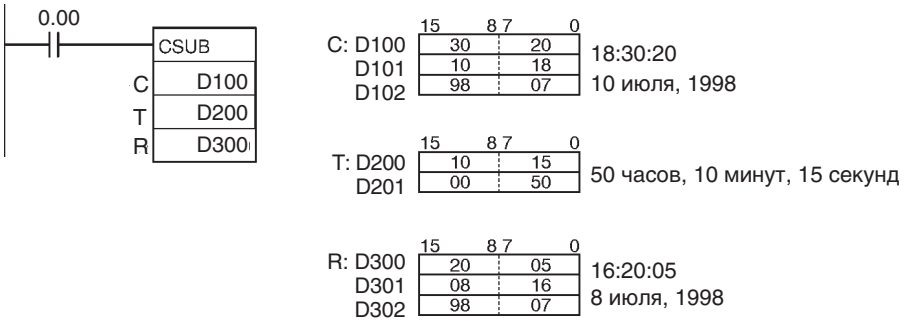


Флаги

Название	Обозначение	Операция
Флаг ошибки	ER	Включен, если календарные данные в словах C...C+2 выходят за указанные диапазоны. Включен, если данные времени в словах T и T+1 выходят за указанные диапазоны. Выключен во всех остальных случаях.

Примеры

Если бит CIO 0.00 включен, значение времени в словах D200 и D201 (часы, минуты, секунды) вычитается из календарной даты и времени в словах D100...D102 (год, месяц, день, часы, минуты, секунды), результат записывается в слова D300...D302.



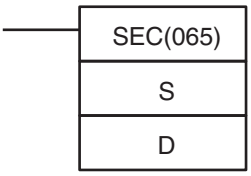
Предварительная версия
Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

3-26-3 ЧАСЫ В СЕКУНДЫ: SEC(065)

Назначение

Преобразование значения времени, представленного в формате «часы/минуты/секунды», в эквивалентное значение в секундах.

Символ РКС



S: Первое исходное слово

D: Первое слово назначения

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	SEC(065)
	Однократное выполнение по положительному фронту	@SEC(065)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

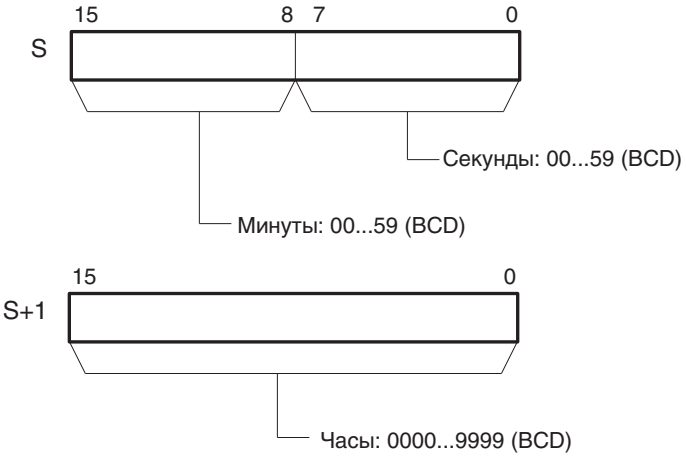
Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

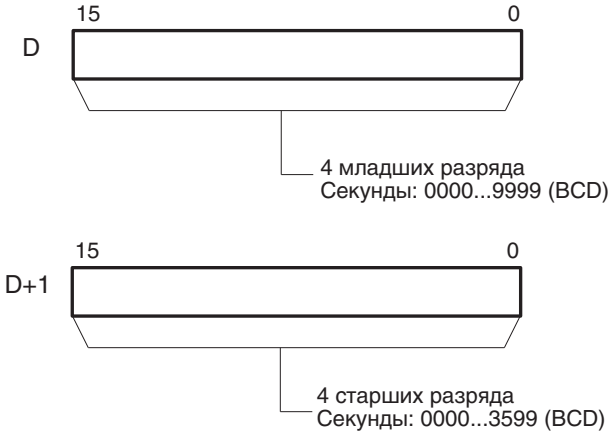
S и S+1: исходное значение

Задайте исходное значение времени в часах, минутах и секундах в словах S и S+1, как показано на следующем рисунке.



D и D+1: результат преобразования

Слова D и D+1 содержат результат выполнения команды: эквивалентное значение времени в секундах.



Характеристики операндов

Область	S	D
Область CIO	CIO 0...CIO 6142	
Рабочая область	W0...W510	
Область битов хранения	H0...H510	
Область вспомогательных битов	A0...A958	A448...A958
Область таймеров	T0000...T4094	
Область счетчиков	C0000...C4094	
Область DM	D0...D32766	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767	
Постоянные	Только указанные значения	---
Регистры данных	---	
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

Описание

Команда SEC(065) преобразует значение времени, выраженное в часах, минутах и секундах в двоично-десятичном коде (слова S и S+1, 8 разрядов), в эквивалентное значение времени в секундах и записывает 8-разрядный результат в двоично-десятичном коде в слова D и D+1.



Флаги

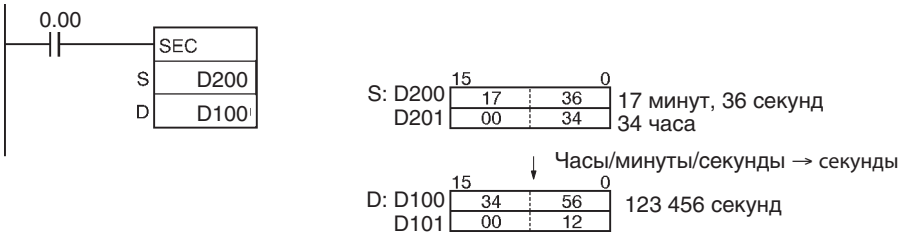
Название	Обозначение	Операция
Флаг ошибки	ER	Включен, если значение минут в S (биты 08...15) не в формате BCD или выходит за диапазон 00...59. Включен, если значение секунд в S (биты 00...07) не в формате BCD или выходит за диапазон 00...59. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если после выполнения команды D = 0000. Выключен во всех остальных случаях.

Меры предосторожности

Максимальное значение времени, которое может быть указано для преобразования: 9999 ч, 59 мин, 59 с (35 999 999 секунд).

Примеры

Если бит CIO 0.00 включен, значение времени в словах D200 и D201 (34 часа, 17 минут и 36 секунд) преобразуется в эквивалентное значение в секундах и выдается в слова D100 и D101.

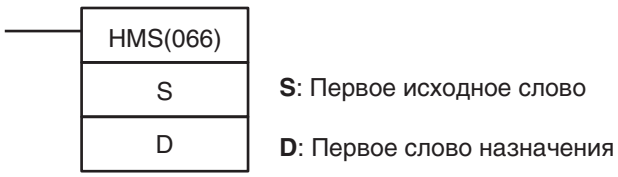


3-26-4 СЕКУНДЫ В ЧАСЫ: HMS(066)

Назначение

Преобразование значения времени, выраженного в секундах, в эквивалентное значение в формате «часы/минуты/секунды».

Символ РКС



S: Первое исходное слово

D: Первое слово назначения

Варианты выполнения

Варианты выполнения		
	Выполнение в каждом цикле при включенном условии выполнения	HMS(066)
	Однократное выполнение по положительному фронту	@HMS(066)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

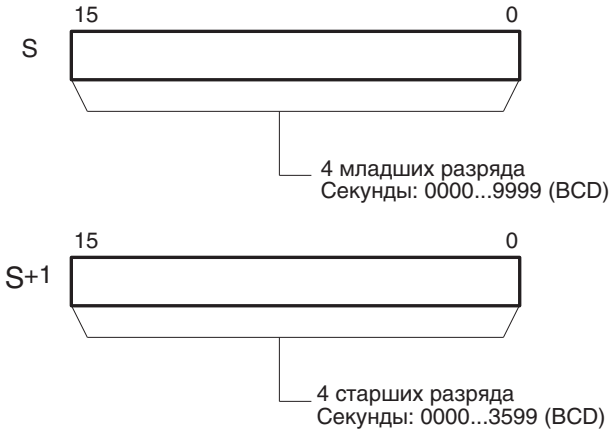
Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

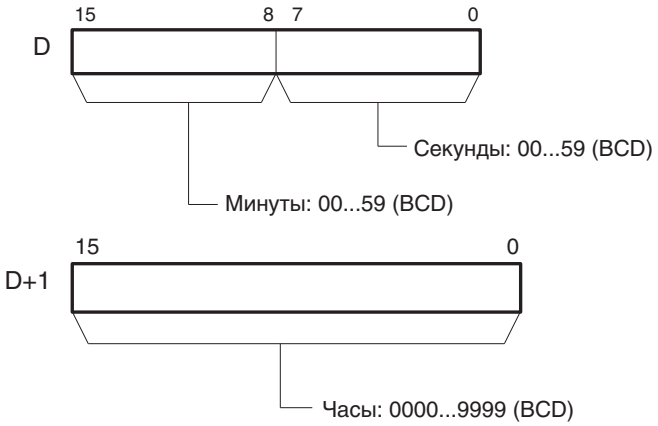
S и S+1: исходное значение

Задайте в словах S и S+1 исходное значение времени в секундах, как показано на следующем рисунке.



D и D+1: результат преобразования

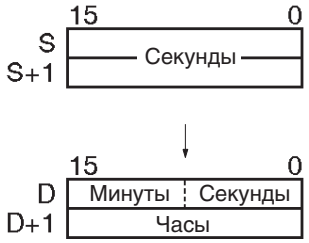
Слова D и D+1 содержат результат выполнения команды: эквивалентное значение времени в часах, минутах и секундах.



Область	S	D
Область CIO	CIO 0...CIO 6142	
Рабочая область	W0...W510	
Область битов хранения	H0...H510	
Область вспомогательных битов	A0...A958	A448...A958
Область таймеров	T0000...T4094	
Область счетчиков	C0000...C4094	
Область DM	D0...D32766	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767	
Постоянные	00000000...35999999 (BCD)	---
Регистры данных	---	
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

Описание

Команда HMS(066) преобразует значение времени, выраженное в секундах в двоично-десятичном коде (слова S и S+1, 8 разрядов), в эквивалентное значение времени в часах, минутах и секундах в двоично-десятичном коде и выдает 8-разрядный результат в слова D и D+1.



Флаги

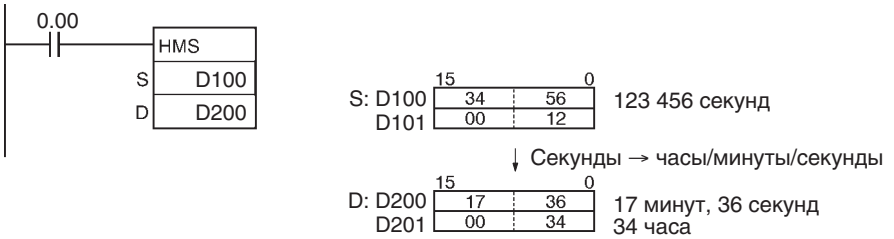
Название	Обо-значение	Операция
Флаг ошибки	ER	Включен, если значение секунд в словах S и S+1 не в формате BCD или выходит за диапазон 0...35 999 999. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если после выполнения команды [D+1, D] = 00000000. Выключен во всех остальных случаях.

Меры предосторожности

Максимальное значение, которое может быть указано для преобразования: 35 999 999 секунд (9999 ч, 59 мин и 59 с).

Примеры

Если бит CIO 0.00 включен, значение времени в словах D100 и D101 (123 456 секунд) преобразуется в эквивалентное значение времени в часах, минутах и секундах, результат выдается в слова D200 и D201.



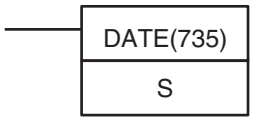
3-26-5 КОРРЕКТИРОВКА ЧАСОВ: DATE(735)

Назначение

Изменение текущих показаний внутренних часов модуля ЦПУ путем записи значений из указанных слов данных.

Примечание. Текущие показания внутренних часов модуля ЦПУ также можно изменить с помощью CX-Programmer или с помощью команды FINS ЗАПИСАТЬ В ЧАСЫ (0702).

Символ РКС



S: Первое исходное слово

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	DATE(735)
	Однократное выполнение по положительному фронту	@DATE(735)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

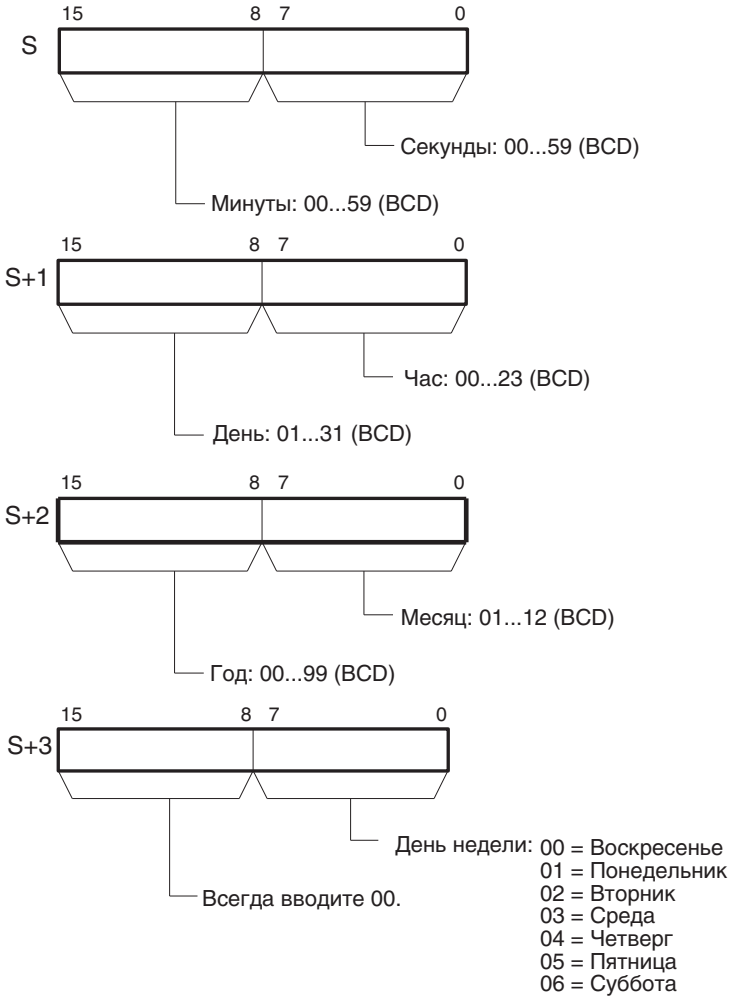
Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

S...S+3: Новые показания часов

Задайте новые показания часов в словах S...S+3, как показано на следующем рисунке.
Слова S...S+3 должны принадлежать одной области данных.



Структура области данных календаря/часов представлена в следующей таблице.

Адреса	Содержание
A351.00...A351.07	Секунды (00...59, BCD)
A351.08...A351.15	Минуты (00...59, BCD)

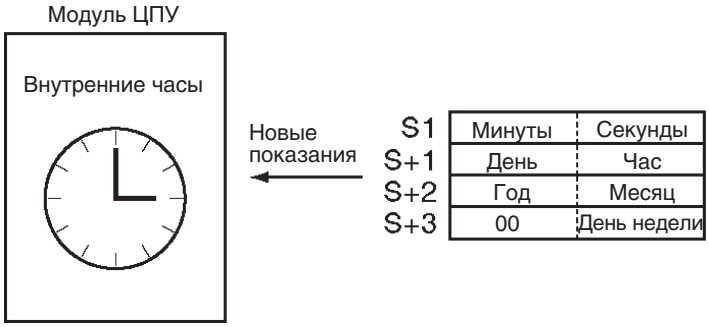
Адреса	Содержание
A352.00...A352.07	Час (00...23, BCD)
A352.08...A352.15	День месяца (01...31, BCD)
A353.00...A353.07	Месяц (01...12, BCD)
A353.08...A353.15	Год (00...99, BCD)
A354.00...A354.07	День недели (00...06 = воскресенье...суббота, hex)
A354.08...A354.15	Всегда вводите 00.

Характеристики операндов

Область	S
Область CIO	CIO 0...CIO 6140
Рабочая область	W0...W508
Область битов хранения	H0...H508
Область вспомогательных битов	A0...A956
Область таймеров	T0000...T4092
Область счетчиков	C0000...C4092
Область DM	D0...D32764
Косвенные адреса DM в двоичном формате	@ D0...@ D32767
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767
Постоянные	---
Регистры данных	---
Регистры указателей	---
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15

Описание

Команда DATE(735) изменяет содержимое внутренних часов модуля ЦПУ, записывая туда значения четырех указанных слов данных. Новые показания внутренних часов сразу же отражаются в словах области данных календаря/часов (A351...A354).



Флаги

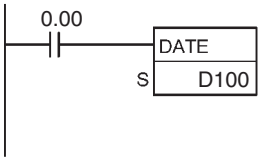
Название	Обозначение	Операция
Флаг ошибки	ER	Включен, если новые показания часов в словах S...S+3 выходят за допустимый диапазон. Выключен во всех остальных случаях.

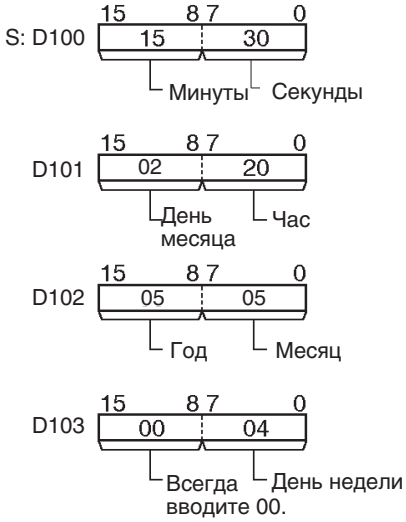
Меры предосторожности

Даже если во внутренние часы будет записана несуществующая дата (например, 31 ноября), ошибка сгенерирована не будет.

Примеры

Если бит CIO 0.00 включен, во внутренних часах модуля ЦПУ устанавливаются новые значения даты и времени: 20:15:30, вторник, 2 мая, 2005.





3-27 Команды для отладки программы

В данном разделе описаны команды, предназначенные для отладки программ.

Команда	Мнемоническое обозначение	Код функции	Стр.
ОТБОР ДАННЫХ ДЛЯ ПАМЯТИ ПРОТОКОЛА ДАННЫХ	TRSM	045	1020

3-27-1 Отбор данных для памяти протокола данных: TRSM(045)

Назначение

При выполнении команды TRSM(045) состояние или значение предварительно выбранного бита или слова считывается и сохраняется в память протокола данных. Команду TRSM(045) можно использовать в любом месте программы и любое количество раз.

Символ РКС



Варианты выполнения

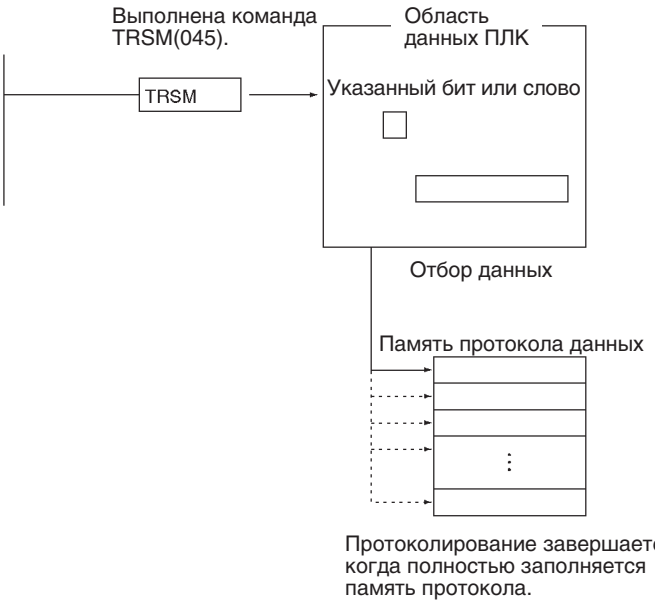
Варианты выполнения	**Выполнение в каждом цикле**	TRSM(045)
	Однократное выполнение по положительному фронту	Не предусмотрено
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Описание

Прежде чем использовать команду TRSM(045) в прикладной программе, необходимо выбрать протоколируемый бит или слово в программе CX-Programmer. Каждый раз, когда выполняется команда TRSM(045), текущее значение указанного бита или слова считывается и записывается (в хронологическом порядке) в память протокола данных. Протоколирование завершается, когда память протокола полностью заполняется. При необходимости, содержимое памяти протокола данных можно просмотреть с помощью программы CX-Programmer.



Команда TRSM(045) лишь определяет момент времени, в который производится отбор данных. Настройка всех остальных параметров, равно как выполнение операций протоколирования данных осуществляются в CX-Programmer. Существуют также два других способа управления моментом отбора данных: отбор данных в конце каждого цикла и отбор данных с указанной периодичностью (которая не зависит от длительности цикла программы).

Для команды TRSM(045) не требуется условие выполнения; она выполняется, как если бы ее условие выполнения всегда было включенным. Команда TRSM(045) подключается непосредственно к левой шине лестничной диаграммы.

Команду TRSM(045) можно использовать для отбора значений указанного бита или слова в некоторой точке программы, когда включено условие выполнения команды. Если условие выполнения команды включено в каждом цикле, значение указанного бита или слова сохраняется в память протокола данных в каждом цикле.

В программе допускается использовать любое количество команд TRSM(045). В пределах каждого цикла в память протокола данных сохраняется столько значений указанного бита или слова, сколько команд TRSM(045) выполняется в данном цикле.



Примечание. Более подробно операция протоколирования данных описана в руководстве пользователя по программе CX-Programmer.

Ниже кратко перечислены действия, которые должны быть выполнены в программе CX-Programmer для создания протокола данных.

1,2,3...

1. Настройте следующие параметры в программе CX-Programmer.
   a. Задайте адрес протоколируемого бита или слова.
   b. Задайте условие запуска протоколирования. Можно задать одно из трех следующих условий, при наступлении которого данные в памяти протокола данных начинают считаться действительными.
      i) Включение (0 > 1) бита запуска протоколирования.
      ii) Включение (0 > 1) указанного бита.
      iii) Совпадение содержимого указанного слова с заданным значением.
   c. Выберите «TRSM» для интервала отбора данных, чтобы отбор данных производился при выполнении команды TRSM(045) в программе.
   d. Задайте величину задержки.
2. После того как бит запуска отбора данных переключается из состояния «0» в состояние «1» с помощью программы CX-Programmer, каждое выполнение команды TRSM(045) в программе приводит к считыванию значения указанного бита или слова и сохранению этого значения в память протокола данных. Вместе с включением бита запуска отбора данных устанавливается флаг активности протоколирования (A508.13).
3. Когда оказывается соблюдено условие запуска протоколирования (т. е. включается бит запуска протоколирования, включается указанный бит или значение указанного слова совпадает с заданным значением), протоколирование вступает в силу и все последующие считываемые значения (со сдвигом вперед или назад на некоторое количество считанных значений, определяемое заданной задержкой) считаются действительными значениями протокола данных. Одновременно с наступлением условия запуска протоколирования устанавливается флаг контроля события протоколирования (A508.11).
4. После того как очередное выполнение команды TRSM(045) приводит к полному заполнению памяти протокола данных, протоколирование завершается. По завершении протоколирования устанавливается флаг завершения протоколирования (A508.12), а флаг активности протоколирования (A508.13) сбрасывается.

5. Прочитайте содержимое памяти протокола данных с помощью CX-Programmer.

В следующей таблице перечислены сопутствующие биты и флаги вспомогательной области. Пользователю требуется управлять только битами A508.14 и A508.15, при этом бит A508.15 следует включать только в программе CX-Programmer, не следует управлять его состоянием из прикладной программы.

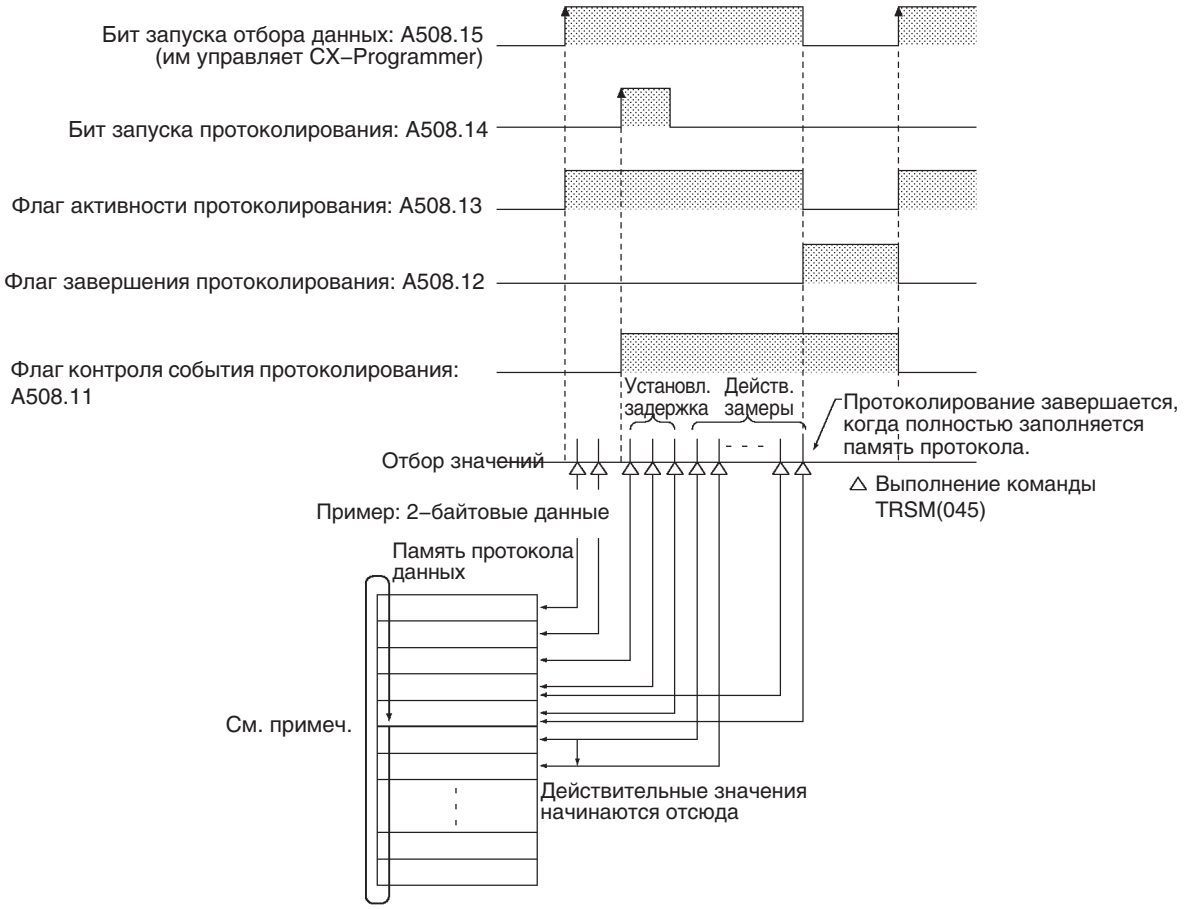
Название	Адрес	Операция
Флаг контроля события протоколирования	A508.11	Данный флаг включается после наступления условия запуска протоколирования (например, при включении бита запуска протоколирования). При следующем запуске отбора данных для другого протокола данных (с помощью бита запуска отбора данных) этот флаг выключается.
Флаг завершения протоколирования	A508.12	Данный флаг включается, когда память протокола данных оказывается полностью заполненной действительными значениями. При следующем переключении бита запуска отбора данных из состояния «0» в состояние «1» этот флаг выключается.
Флаг активности протоколирования	A508.13	Данный флаг включается, когда бит запуска отбора данных переключается из состояния «0» в состояние «1». После завершения протоколирования этот флаг выключается.
Бит запуска протоколирования	A508.14	Переключение данного бита из состояния «0» в состояние «1» является условием запуска протоколирования. После этого все регистрируемые в протоколе данных значения считаются действительными, начиная со значения, сдвинутого относительно момента запуска протоколирования вперед (положительная задержка) или назад (отрицательная задержка) на указанное количество считанных значений.
Бит запуска отбора данных	A508.15	После того как данный бит переключается из состояния «0» в состояние «1» с помощью CX-Programmer, начинается отбор данных и сохранение отобранных значений в память протокола данных с использованием одного из следующих способов отбора данных: 1) Периодический отбор данных (с интервалом в 10...2550 мс) 2) Отбор данных по команде TRSM(045) 3) Отбор данных в конце каждого цикла Данный бит следует устанавливать и сбрасывать только с помощью CX-Programmer.

Меры предосторожности

Если протоколирование данных в текущий момент времени не производится или при настройке параметров в CX-Programmer для интервала отбора данных не было выбрано значение «TRSM», команда TRSM(045) обрабатывается как команда NOP(000).

Не включайте и не выключайте бит запуска отбора данных (A508.15) из прикладной программы. Данный бит следует устанавливать и сбрасывать только с помощью CX-Programmer.

Пример Ниже показан пример полного цикла процедуры протоколирования данных.



Примечание. Память протокола данных имеет кольцевую структуру. Когда оказывается записан самый последний адрес области памяти протокола данных, значения продолжают записываться, начиная с первого адреса области, пока не достигается первое действительное значение.

3-28 Команды для диагностики неисправностей

В данном разделе описаны команды, предназначенные для выявления и устранения ошибок.

Команда	Мнемоническое обозначение	Код функции	Стр.
СИГНАЛИЗАЦИЯ НЕИСПРАВНОСТИ	FAL	006	1025
СИГНАЛИЗАЦИЯ СЕРЬЕЗНОЙ НЕИСПРАВНОСТИ	FALS	007	1033
ОБНАРУЖЕНИЕ НЕИСПРАВНОГО КАНАЛА	FPD	269	1041

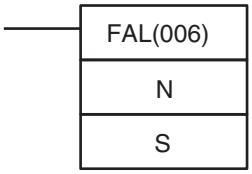
3-28-1 СИГНАЛИЗАЦИЯ НЕИСПРАВНОСТИ: FAL(006)

Назначение

Формирование или сброс некритических ошибок, сконфигурированных пользователем. Некритические ошибки не останавливают работу ПЛК.

Символ РКС

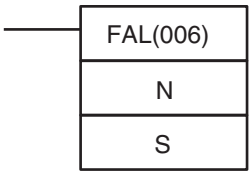
- Формирование или сброс некритических ошибок пользователя.



N: Номер ошибки FAL

S: Первое слово сообщения или

- Формирование некритических системных ошибок.



N: Номер ошибки FAL (значение в A529)

S: Первое слово кода и

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	FAL(006)
	Однократное выполнение по положительному фронту	@FAL(006)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

Функции операндов команды FAL(006) зависят от того, для какой цели применяется команда: для формирования/сброса ошибок пользователя или для формирования системных ошибок.

Формирование или сброс некритических ошибок пользователя.

Функционирование операндов поясняется в следующей таблице.

Примечание. Значение операнда N должно **отличаться** от содержимого слова A529 (номер ошибки FAL/FALS, генерируемой системой).

N	S	Функция
0	#0001...#01FF	Сброс некритической ошибки с соответствующим номером FAL.
	#FFFF	Сброс всех некритических ошибок.
	Другое значение*	Сброс наиболее серьезной некритической ошибки.
1...511 (Эти номера являются общими для ошибок FALS и FAL.)	#0000...#FFFF	Формирование некритической ошибки с соответствующим номером FAL (без сообщения).
	Адрес слова	Формирование некритической ошибки с соответствующим номером FAL. В CX-Programmer отображается сообщение, ASCII-коды которого (16 символов) содержатся в S...S+7.

Примечание. *Другим значением может быть константа от #0200 до #FFFE или адрес слова.

Формирование некритических системных ошибок.

Функционирование операндов поясняется в следующей таблице.

Примечание. Значение операнда N должно **совпадать** с содержимым слова A529 (номер ошибки FAL/FALS, генерируемой системой).

Операнд	Функция
N	1...511 (Эти номера являются общими для ошибок FALS и FAL.)
S	Генерируемый код ошибки (см. *Описание* ниже).
S+1	Генерируемые подробности ошибки (см. *Описание* ниже).

Характеристики операндов

Область	N	S
Область CIO	---	CIO 0...CIO 6143
Рабочая область	---	W0...W511
Область битов хранения	---	H0...H511
Область вспомогательных битов	---	A0...A959
Область таймеров	---	T0000...T4095
Область счетчиков	---	C0000...C4095
Область DM	---	D0...D32767
Косвенные адреса DM в двоичном формате	---	@ D0...@ D32767
Косвенные адреса DM в двоично-десятичном формате (BCD)	---	*D0...*D32767
Постоянные	0...511	#0000...#FFFF (двоичн.)
Регистры данных	---	
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	---	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15

Описание

Действие команды FAL(006) зависит от значения операнда N. Для сброса ошибки в N следует записать 0000, для формирования ошибки в N следует записать значение от 0001 до 01FF. Если значение в N совпадет с содержимым слова A529, будет сгенерирована системная ошибка.

Формирование некритических ошибок пользователя

При выполнении команды FAL(006) с операндом N, содержащим номер FAL (&1...&511), не совпадающий с содержимым слова A529 (номер ошибки FAL/FALS, генерируемой системой), генерируется некритическая ошибка с указанным номером FAL. При этом будут выполнены следующие действия.

1,2,3...

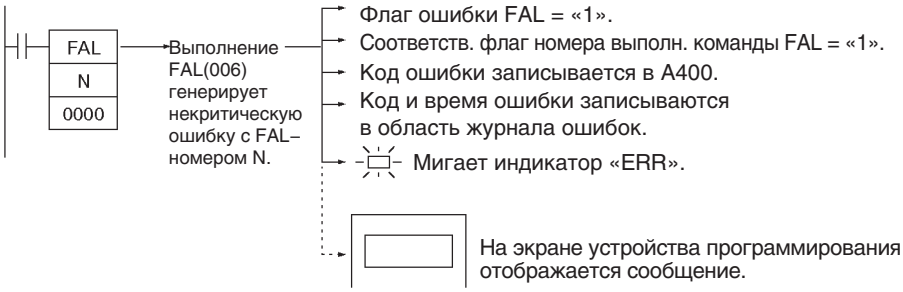
1. Будет установлен флаг ошибки FAL (A402.15) (работа ПЛК остановлена не будет).
2. Будет установлен флаг номера выполненной команды FAL для соответствующего номера FAL. Флаги A360.01...A391.15 соответствуют номерам FAL 0001...01FF (1...511).
3. В слово A400 будет записан код ошибки. Коды ошибок 4101...42FF соответствуют номерам FAL 0001...01FF (1...511).

Примечание. Если одновременно с выполнением команды FAL(006) возникнет критическая ошибка или более серьезная некритическая ошибка, в слово A400 будет записан код наиболее серьезной ошибки.

4. В область журнала ошибок (A100...A199) будут записаны код ошибки и время возникновения ошибки.

Примечание. Если в области настроек ПЛК выбрано не регистрировать ошибки FAL(006) в журнале ошибок, запись об ошибке в области журнала ошибок создана не будет.

5. Будет мигать индикатор «ERR» на модуле ЦПУ.
6. Если в операнде S указан адрес слова, будет зарегистрировано сообщение, содержащееся в S и следующих за ним словах (отобразится на экране устройства программирования).



В следующей таблице перечислены коды ошибок и флаги ошибок FAL для команды FAL(006).

Номер FAL	Коды ошибок FAL	Флаги номера выполненной команды FAL
1...511 десят.	4101...42FF	A360.01...A391.15

Отображение сообщений для некритических ошибок пользователя

Если в операнде S указан адрес слова и в это слово (и следующие за ним слова) записаны ASCII-коды сообщения, при выполнении команды FAL(006) это сообщение будет отображено на экране устройства программирования. (Если сообщение не требуется, введите в S константу.)

Сообщение, текст которого начинается с адреса S, регистрируется при выполнении команды FAL(006). Зарегистрированное сообщение отобразится на экране устройства программирования, после того как последнее будет подключено.

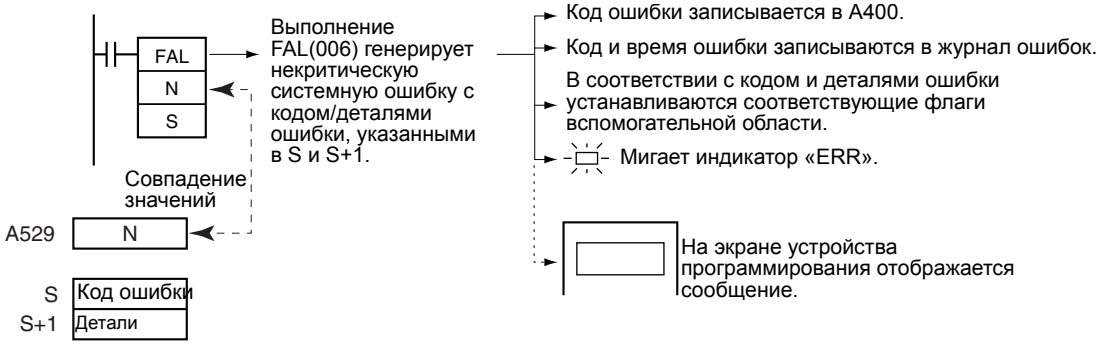
В слова S...S+7 можно записать сообщение длиной до 16 символов (16 кодов ASCII). Первым отображается самый старший (крайний левый) байт каждого слова.

Кодом завершения сообщения служит символ «null» (00 hex). При отсутствии символа «null» будут отображены все 16 символов, записанные в слова S...S+7.

Если содержимое слов сообщения будет изменено уже после выполнения команды FAL(006), отображаемое сообщение изменится соответствующим образом.

Формирование некритических системных ошибок.

При выполнении команды FAL(006) с операндом N, содержащим номер FAL (&1...&511), совпадающий с содержимым слова A529 (номер ошибки FAL/FALS, генерируемой системой), генерируется некритическая ошибка с кодом ошибки и подробной информацией об ошибке, указанными в словах S и S+1. Одновременно с этим выполняются следующие действия.



1,2,3...

1. В слово A400 будет записан указанный код ошибки.
2. В область журнала ошибок (A100...A199) будут записаны код ошибки и время возникновения ошибки.
3. В соответствии с кодом и подробными данными ошибки устанавливаются соответствующие флаги вспомогательной области.
4. Будет мигать индикатор «ERR» на модуле ЦПУ, работа ПЛК при этом остановлена не будет.
5. В CX-Programmer отобразится сообщение о некритической ошибке для указанной системной ошибки.

Примечание.

(1) Команду FAL(006) можно использовать для искусственного вызова некритических системных ошибок при отладке программы. Путем вызова системных ошибок можно, к примеру, проверить, правильно ли отображаются сообщения об ошибках на устройстве операторского интерфейса (такого, как программируемый терминал).

(2) Значение в A529 (номер ошибки FAL/FALS, генерируемой системой) — это номер фиктивной ошибки FAL (ошибки FAL, FALS и FPD имеют общую нумерацию), который используется, когда системой искусственно генерируется некритическая ошибка. Поскольку этот номер FAL является фиктивным, он не изменяет состояние флагов номера выполненной команды FAL (A360.01...A391.15) или кода ошибки.

Если нужно сгенерировать несколько системных ошибок (критических и/или некритических), следует выполнить нужное количество команд FAL/FALS/FPD с одними и теми же значениями в A529 и N, но с разными значениями в S и S+1.

(3) Если одновременно с выполнением команды FAL(006) происходит более серьезная ошибка (в том числе критическая системная ошибка или ошибка FALS(007)), в слово A400 записывается код наиболее серьезной ошибки.

(4) Для сброса системной ошибки, вызванной командой FAL(006), следует выключить и вновь включить питание ПЛК. Можно не выключать ПЛК, но тогда для сброса ошибки необходимо выполнить те же действия, что и в случае действительного возникновения указанной ошибки.

В следующей таблице перечислены коды ошибок и подробные сведения об ошибках, которые указываются в словах S и S+1.

Наименование ошибки	S	S+1
Ошибка флэш-памяти	00F1 hex	--- (не фикс.)
Ошибка задачи обработки прерывания	008B hex	• Бит 15 = 1: конфликт задачи обработки прерывания с обновлением спец. модуля ввода/вывода Биты 00...14: номер спец. модуля ввода/вывода, вызвавшего конфликт обновления
Ошибка настроек ПЛК	009B hex	Место возникновения ошибки настроек ПЛК
Ошибка встроенных аналоговых входов/выходов	008A hex	--- (не фикс.)
Ошибка модуля шины ЦПУ	0200 hex	Номер модуля шины ЦПУ: 0000...000F hex
Ошибка специального модуля ввода/вывода	0300 hex	Номер спец. модуля ввода/вывода: 0000...005F hex или 00FF hex (номер модуля не определен)
Ошибка дополнительной платы	00D0 hex	Номер гнезда доп. платы: 0001 или 0002 hex
Ошибка батареи	00F7 hex	--- (не фикс.)

Отключение регистрации ошибок пользователя в журнале ошибок

В обычном случае, когда команда FAL(006) генерирует ошибку пользователя, код ошибки и время ее возникновения записываются в область журнала ошибок (A100...A199). В области настроек ПЛК, однако, предусмотрен параметр, с помощью которого можно отключить регистрацию ошибок пользователя, генерируемых командой FAL(006), в журнале ошибок.

Даже если ошибка не регистрируется в журнале ошибок, она все равно приводит к установке флага ошибки FAL (A402.15), установке соответствующего флага номера выполненной команды FAL (A360.01...A391.15) и записи кода ошибки в A400.

Если в журнале ошибок должны регистрироваться только ошибки, генерируемые системой, регистрацию ошибок пользователя, вызываемых командой FAL(006), нужно отключить. Этой функцией можно воспользоваться, например, на этапе отладки, когда частое применение команд FAL(006) в нескольких прикладных программах приводит к быстрому заполнению журнала ошибками пользователя. В следующей таблице описана настройка соответствующего параметра в области настроек ПЛК.

Параметр	Значение
Название	Регистрация ошибок FAL в журнале
Значение	0: Регистрировать ошибки FAL в журнале ошибок. 1: Не регистрировать ошибки FAL в журнале ошибок.
Настройка по умолчанию	0: Регистрировать ошибки FAL в журнале ошибок.
Время считывания параметра в области настроек ПЛК	В каждом цикле (когда происходит ошибка FAL)

Перечисленные ниже ошибки регистрируются, даже если бит 15 слова 129 в настройках ПЛК равен «1» (не регистрировать ошибки FAL в журнале ошибок).

- Критические ошибки, генерируемые командой FALS(007).
- Некритические ошибки, генерируемые системой.
- Критические ошибки, генерируемые системой.
- Некритические ошибки, генерируемые системой намеренно с помощью команды FAL(006) или FPD(269).
- Критические ошибки, генерируемые системой намеренно с помощью команды FALS(007).

Сброс некритических ошибок с помощью CX-Programmer

1. Сброс некритических ошибок пользователя
   Выполнение команды FAL(006) с операндом N = 0 приводит к сбросу некритических ошибок. Вид сброса определяется значением операнда S, что отображено в следующей таблице.

S	Действие
&1...&511 (0001...01FF hex)	Будет сброшена ошибка FAL с указанным номером.
FFFF hex	Будут сброшены все некритические ошибки (в том числе системные ошибки).
0200...FFFE hex или адрес слова	Будет сброшена наиболее серьезная некритическая ошибка (даже если это некритическая системная ошибка). При наличии сразу нескольких активных ошибок FAL будет сброшена ошибка с наименьшим номером FAL.

2. Сброс некритических системных ошибок
   Существуют два способа сброса некритических системных ошибок, сгенерированных командой FAL(006).
   - Выключить и вновь включить ПЛК.
   - Не выключать ПЛК, а выполнить сброс системной ошибки, как если бы указанная ошибка произошла на самом деле.

Флаги

Название	Обозначение	Операция
Флаг ошибки	ER	Включен, если значение N выходит за допустимый диапазон (0...511 десят.). Включен, если для сгенерированной некритической системной ошибки указан неверный код ошибки или неверные подробные данные ошибки. Выключен во всех остальных случаях.

В следующих таблицах перечислены сопутствующие слова и флаги вспомогательной области.

- Слова и флаги вспомогательной области только для ошибок пользователя

Название	Адрес	Операция
Флаг ошибки FAL	A402.15	Включен, если команда FAL(006) сгенерировала ошибку.
Флаги номера выполненной команды FAL	A360.01... A391.15	Когда команда FAL(006) генерирует ошибку, включается один из этих флагов. Флаги A360.01... A391.15 соответствуют номерам FAL 0001...01FF.

- Слова и флаги вспомогательной области только для системных ошибок

Название	Адрес	Операция
Номер ошибки FAL/FALS, генерируемой системой	A529	Когда системная ошибка генерируется командой FAL(006), используется номер фиктивной ошибки FAL/FALS. Такое же значение номера фиктивной ошибки FAL/FALS (0001...01FF hex, 1...511 десят.) следует ввести в это слово.

- Слова и флаги вспомогательной области, общие для ошибок пользователя и для системных ошибок

Название	Адрес	Операция
Область журнала ошибок	A100... A199	Область журнала ошибок содержит коды ошибок и время/дату возникновения для 20 самых последних ошибок, включая ошибки, сгенерированные командой FAL(006).
Код ошибки	A400	При возникновении ошибки ее код записывается в слово A400. Номерам ошибок FAL от 0001 до 01FF соответствуют коды ошибок от 4101 до 42FF. При одновременном возникновении двух или более ошибок в A400 записывается код наиболее серьезной ошибки.

Меры предосторожности

Значение N должно находиться в диапазоне от 0000 до 01FF. В противном случае произойдет ошибка и будет установлен флаг ошибки.

Примеры

Формирование некритической ошибки

Если в следующем примере включится бит CIO 0.00, команда FAL(006) сгенерирует некритическую ошибку FAL с номером 31, что приведет к указанным ниже событиям.

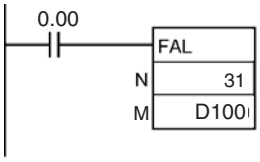
1,2,3...

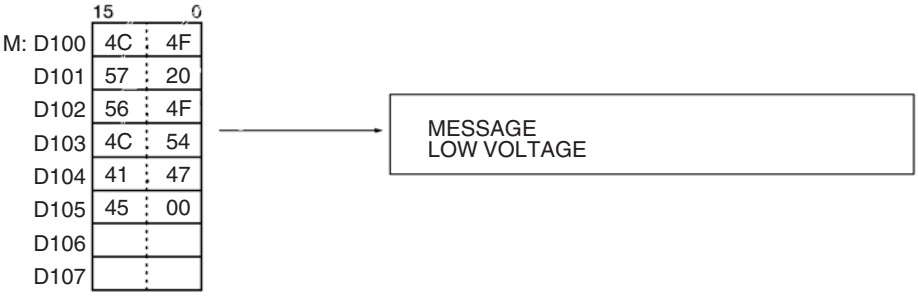
1. Будет установлен флаг ошибки FAL (A402.15).
2. Будет установлен соответствующий флаг номера выполненной команды FAL (A361.14).
3. В слово A400 будет записан соответствующий код ошибки (411F).

Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

Примечание. При одновременном возникновении двух или более ошибок в A400 будет записан код наиболее серьезной ошибки (т. е. ошибки с наибольшим кодом).

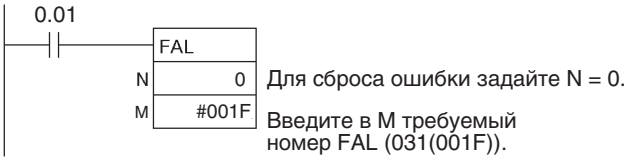
4. В область журнала ошибок (A100...A199) будут записаны код ошибки и время/дата возникновения ошибки.
5. Будет мигать индикатор «ERR» на модуле ЦПУ.
6. На экране устройства программирования отобразится сообщение, ASCII-коды которого содержатся в словах D100...D107. (Если сообщение не требуется, укажите в S константу.)





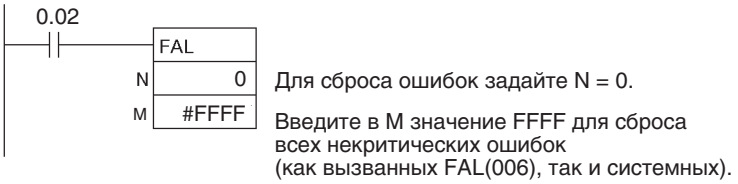
Сброс одной определенной некритической ошибки

При включении бита CIO 0.01 в следующем примере команда FAL(006) сбросит некритическую ошибку FAL с номером 31, выключит соответствующий флаг номера выполненной команды FAL (A361.14) и выключит флаг ошибки FAL (A402.15).



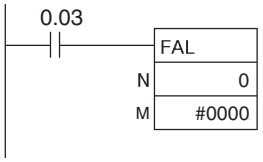
Сброс всех некритических ошибок

При включении бита CIO 0.02 в следующем примере команда FAL(006) сбросит все некритические ошибки, выключит все флаги номера выполненной команды FAL (A360.01...A391.15) и выключит флаг ошибки FAL (A402.15).



Сброс наиболее серьезной некритической ошибки

При включении бита CIO 0.03 в следующем примере команда FAL(006) сбросит наиболее серьезную некритическую ошибку, произошедшую на данный момент, и сбросит код ошибки в слове A400. Если сброшенная ошибка была вызвана командой FAL(006), также будут сброшены соответствующий флаг номера выполненной команды FAL и флаг ошибки FAL (A402.15).



Для сброса ошибок задайте N = 0.

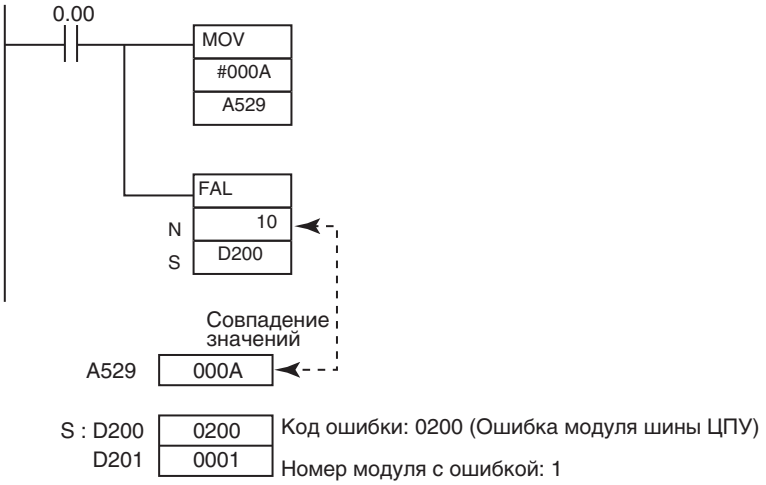
Введите в M значение 0000, либо другую константу от 0200 до FFFE, либо адрес слова для сброса наиболее серьезной некритической ошибки. (В данном примере M = 0000.)

Формирование некритических системных ошибок

При включении бита CIO 0.00 в следующем примере команда FAL(006) сгенерирует ошибку настройки модуля шины ЦПУ для модуля с номером 1. В данном случае используется номер фиктивной ошибки FAL 10, поэтому в A529 записывается соответствующее значение (000A hex).

1,2,3...

1. В слово A400 будет записан указанный код ошибки (0200), если на данный момент это наиболее серьезная ошибка.
2. В область журнала ошибок (A100...A199) будут записаны код ошибки и время/дата возникновения ошибки.
3. Будут установлены флаг ошибки модуля шины ЦПУ (A402.07) и флаг ошибки модуля шины ЦПУ для номера модуля 1 (A417.01).
4. Будет мигать индикатор «ERR» на модуле ЦПУ.
5. На экране устройства программирования отобразится сообщение (CPU BU ST ERR 01), уведомляющее о возникновении ошибки модуля шины ЦПУ с номером 1.



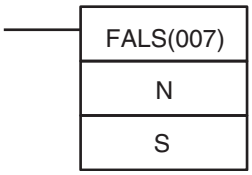
3-28-2 СИГНАЛИЗАЦИЯ СЕРЬЕЗНОЙ НЕИСПРАВНОСТИ: FALS(007)

Назначение

Формирование критических ошибок, сконфигурированных пользователем. Критические ошибки останавливают работу ПЛК.

Символ РКС

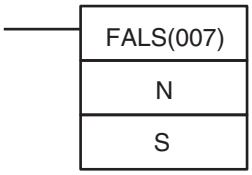
- Формирование критических ошибок пользователя.



N: Номер ошибки FALS

S: Первое слово сообщения или константа (0000...FFFF)

• Формирование критических системных ошибок.



N: Номер ошибки FALS (значение в A529)

S: Первое слово кода и подробностей ошибки

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	FALS(007)
	Однократное выполнение по положительному фронту	Не предусмотрено
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

Формирование критических ошибок пользователя.

Функционирование операндов поясняется в следующей таблице.

Примечание. Значение операнда N должно **отличаться** от содержимого слова A529 (номер ошибки FAL/FALS, генерируемой системой).

Операнд	Функция
N	1...511 (Эти номера являются общими для ошибок FALS и FAL.)
S	Указывает первое из восьми слов, содержащих ASCII-коды сообщения, которое должно отображаться в CX-Programmer. Если сообщение не требуется, введите константу (0000...FFFF).

Формирование критических системных ошибок

Функционирование операндов поясняется в следующей таблице.

Примечание. Значение операнда N должно **совпадать** с содержимым слова A529 (номер ошибки FAL/FALS, генерируемой системой).

Операнд	Функция
N	1...511 (Эти номера являются общими для ошибок FALS и FAL.)
S	Генерируемый код ошибки (см. *Описание* ниже).
S+1	Генерируемые подробные данные ошибки (см. *Описание* ниже).

Характеристики операндов

Область	N	S
Область CIO	---	CIO 0...CIO 6143
Рабочая область	---	W0...W511
Область битов хранения	---	H0...H511
Область вспомогательных битов	---	A0...A959
Область таймеров	---	T0000...T4095
Область счетчиков	---	C0000...C4095
Область DM	---	D0...D32767

Область	N	S
Косвенные адреса DM в двоичном формате	---	@ D0...@ D32767
Косвенные адреса DM в двоично-десятичном формате (BCD)	---	*D0...*D32767
Постоянные	Только указанные значения	#0000...#FFFF (двоичн.)
Регистры данных	---	
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	---	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR+(++)0...,IR15+(++) ,–(– –)IR0..., –(– –)IR15

Описание

Команда FALS(007) генерирует критическую ошибку. С помощью команды FALS(007) можно генерировать как критические ошибки системы, так и критические ошибки, конфигурируемые пользователем. (Системная ошибка генерируется, если значение операнда N совпадает со значением в слове A529.)

Формирование критических ошибок пользователя

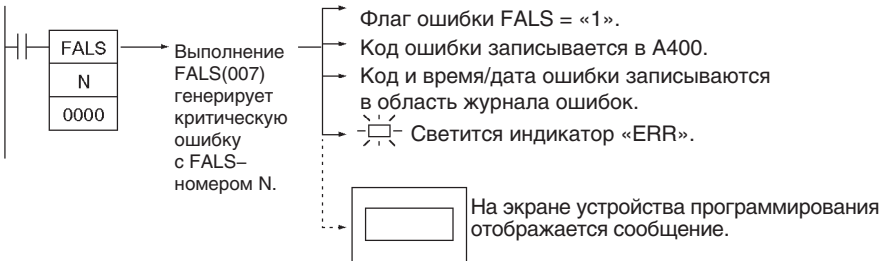
При выполнении команды FALS(007) с операндом N, содержащим номер ошибки FALS (1...511), не совпадающий с содержимым слова A529 (номер ошибки FAL/FALS, генерируемой системой), генерируется критическая ошибка с указанным номером FALS. При этом будут выполнены перечисленные ниже действия.

1,2,3...

1. Будет установлен флаг ошибки FALS (A401.06) (работа ПЛК будет остановлена).
2. В слово A400 будет записан код ошибки. Коды ошибок C101...C2FF соответствуют номерам FALS 0001...01FF (1...511).

Примечание. Если одновременно с выполнением команды FALS(007) произойдет более серьезная ошибка (т. е. ошибка, у которой код ошибки больше), в A400 будет записан код более серьезной ошибки.

3. В область журнала ошибок (A100...A199) будут записаны код ошибки и время/дата возникновения ошибки.
4. Будет непрерывно светиться индикатор «ERR» на модуле ЦПУ.
5. Если в операнде S указан адрес слова, будет зарегистрировано сообщение, ASCII-коды которого содержатся в слове S и следующих за ним словах (сообщение отображается на экране устройства программирования).



Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

В следующей таблице перечислены коды ошибок для команды FALS(007).

Номер FALS	Коды ошибок FALS
1...511	C101...C2FF

Отображение сообщений для критических ошибок пользователя

Если S является адресом слова, при выполнении команды FALS(007) на экране устройства программирования отобразится сообщение, ASCII-коды которого записаны в слово S и следующие за ним слова. (Если сообщение не требуется, введите в S константу.)

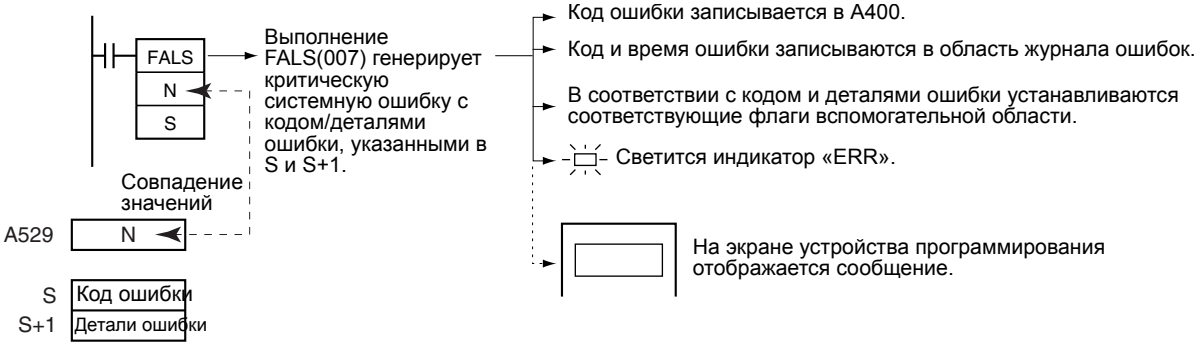
Текст сообщения, начинающийся со слова S, будет зарегистрирован при выполнении команды FALS(007). Зарегистрированное сообщение отобразится на экране устройства программирования, после того как последнее будет подключено.

В слова S...S+7 можно записать сообщение длиной до 16 символов (16 кодов ASCII). Первым отображается самый старший (крайний левый) байт каждого слова.

Кодом завершения сообщения служит символ «null» (00 hex). При отсутствии символа «null» будут отображены все 16 символов, записанные в слова S...S+7.

Если содержимое слов, содержащих текст сообщения, будет изменено уже после выполнения команды FALS(007), содержание отображаемого сообщения изменится соответствующим образом.

Формирование критических системных ошибок



При выполнении команды FALS(007) с операндом N, содержащим номер FALS (1...511), совпадающий с содержимым слова A529 (номер ошибки FAL/FALS, генерируемой системой), будет сгенерирована критическая ошибка с кодом ошибки и подробными данными об ошибке, указанными в словах S и S+1. Одновременно с этим произойдут следующие события.

1,2,3...

1. В слово A400 будет записан указанный код ошибки.
2. В область журнала ошибок (A100...A199) будут записаны код ошибки и время возникновения ошибки.
3. В соответствии с кодом и подробными данными ошибки устанавливаются соответствующие флаги вспомогательной области.
4. Будет непрерывно светиться индикатор «ERR» на модуле ЦПУ, работа ПЛК будет остановлена.
5. На экране устройства программирования отобразится сообщение о критической ошибке для указанной системной ошибки.

Примечание. (1) Значение в A529 (номер ошибки FAL/FALS, генерируемой системой) — это номер фиктивной ошибки FALS (ошибки FAL, FALS и FPD имеют общую нумерацию), который используется, когда

системой искусственно генерируется критическая ошибка. Поскольку этот номер FALS является фиктивным, он не отражается в коде ошибки.

Если нужно сгенерировать несколько разных системных ошибок, следует выполнить нужное количество команд FAL/FALS/FPD с одними и теми же значениями в A529 и N, но с разными значениями в S и S+1.

(2) Если одновременно с выполнением команды FALS(007) происходит более серьезная ошибка (в том числе критическая системная ошибка или другая ошибка FALS(007)), в слово A400 записывается код наиболее серьезной ошибки.

(3) Для сброса системной ошибки, вызванной командой FALS(007), следует выключить и вновь включить питание ПЛК. Можно не выключать ПЛК, но тогда для сброса ошибки необходимо выполнить те же действия, что и в случае действительного возникновения указанной ошибки. См. более подробную информацию в описании поиска и устранения ошибок в руководстве *Серия CP — Руководство по эксплуатации*.

(4) В следующей таблице поясняется влияние бита сохранения памяти ввода/вывода (IOM) на состояние памяти ввода/вывода и состояние выходов модулей выходов после вызова критической системной ошибки с помощью команды FALS(007).

Бит сохранения IOM (A500.12)	Состояние памяти ввода/вывода	Состояние выходов модулей выходов
ВКЛ	Сохраняется	ВЫКЛ
ВЫКЛ	Сбрасывается	ВЫКЛ

Примечание. В отличие от критических ошибок, сконфигурированных пользователем, системные ошибки, генерируемые командой FALS(007), приводят к обнулению содержимого памяти ввода/вывода при выключенном бите сохранения памяти ввода/вывода. Обнуляются следующие области: область CIO, рабочая область, флаги и текущие значения таймеров, регистры указателей и регистры данных.

В следующей таблице перечислены коды ошибок и подробные сведения об ошибках, которые указываются в словах S и S+1.

Наименование ошибки	S	S+1
	Код ошибки	Детали ошибки
Ошибка памяти	80F1 hex	• Биты 00...09: расположение ошибки памяти Бит 00: программа пользователя Бит 04: настройки ПЛК Бит 07: таблица маршрутизации Бит 08: настройки модуля шины ЦПУ Бит 09: ошибка передачи данных карты памяти • Биты 10...15: не действительны
Ошибка шины ввода/вывода	80C0 hex	0A0A hex: ошибка модуля расширения или модуля расширения входов/выходов CPM1A 0000 hex: ошибка модуля серии CJ (модуль 1) 0001 hex: ошибка модуля серии CJ (модуль 2) 0F0F hex: ошибка модуля серии CJ (неизвестный модуль) 0E0E hex: ошибка модуля серии CJ (нет концевой крышки)

Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

Наименование ошибки	S	S+1
	Код ошибки	Детали ошибки
Ошибка дублирования номера модуля	80E9 hex	Дублированный номер модуля шины ЦПУ 0000...000F hex
		Дублированный номер специального модуля вв./вых. 8000...805F hex
Ошибка превышения числа точек ввода/вывода	80E1 hex	Биты 13...15: причина ошибки Биты 00...12: подробная информация • Слишком много слов для модулей расширения и модулей расширения вх./вых. серии CPM1A Биты 13...15: 010 Биты 00...12: все равны «0» • Слишком много модулей расширения или модулей расширения вх./вых. серии CPM1A Биты 13...15: 011 Биты 00...12: все равны «0» • Слишком много слов для модулей серии CJ Биты 13...15: 111 Биты 00...12: все равны «0»
Ошибка настройки таблицы ввода/вывода	80E0 hex	--- (Не фикс.)
Ошибка программы	80F0 hex	• Биты 08...15: причина ошибки Бит 15: ошибка переполнения UM Бит 14: ошибка недопустимой команды Бит 13: ошибка переполнения команд с различением фронтов Бит 12: ошибка задачи Бит 11: ошибка отсутствия «END» Бит 10: ошибка неразрешенного доступа Бит 09: ошибка косвенного BCD-адреса DM Бит 08: ошибка команды • Биты 00...07: не действительны
Ошибка превышения времени цикла	809F hex	--- (Не фикс.)

Сброс критических системных ошибок, вызванных командой FALS(007)

Существуют два способа сброса критических системных ошибок, сгенерированных командой FALS(007).

1. Выключить и вновь включить ПЛК.
2. Не выключать ПЛК, а выполнить сброс системной ошибки, как если бы указанная ошибка произошла на самом деле.

Сброс критических ошибок пользователя, вызванных командой FALS(007)

Для сброса ошибки, сгенерированной с помощью команды FALS(007), сперва необходимо устранить причину ошибки, а затем либо сбросить ошибку из устройства программирования, либо выключить и вновь включить питание ПЛК.

Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	Включен, если значение N выходит за допустимый диапазон 0001...01FF (1...511 десят.). Включен, если для сгенерированной критической системной ошибки указан неверный код ошибки или неверные подробные данные об ошибке. Выключен во всех остальных случаях.

В следующих таблицах перечислены сопутствующие слова и флаги вспомогательной области.

- Слова и флаги вспомогательной области только для ошибок пользователя

Название	Адрес	Операция
Флаг ошибки FALS	A401.06	Включен, если команда FALS(007) сгенерировала ошибку.

- Слова и флаги вспомогательной области только для системных ошибок

Название	Адрес	Действие
Номер ошибки FAL/FALS, генерируемой системой	A529	Когда системная ошибка генерируется командой FALS(007), используется номер фиктивной ошибки FAL/FALS. Такое же значение номера фиктивной ошибки FAL/FALS (0001...01FF hex, 1...511 десят.) следует ввести в это слово.

- Слова и флаги вспомогательной области, общие для ошибок пользователя и для системных ошибок

Название	Адрес	Действие
Область журнала ошибок	A100... A199	Область журнала ошибок содержит коды ошибок и время/дату возникновения для 20 самых последних ошибок, включая ошибки, сгенерированные командой FALS(007).
Код ошибки	A400	При возникновении ошибки ее код записывается в слово A400. Номерам ошибок FALS от 0001 до 01FF (1...511 десят.) соответствуют коды ошибок от C101 до C2FF. При одновременном возникновении двух или более ошибок в A400 записывается код наиболее серьезной ошибки.

Меры предосторожности

Кодом завершения сообщения служит символ «null» (00 hex). При отсутствии символа «null» будут отображены все 16 символов, записанные в слова S...S+7.

N должно находиться в диапазоне от 0001 до 01FF. В противном случае произойдет ошибка и будет установлен флаг ошибки.

Примеры

Формирование ошибки пользователя

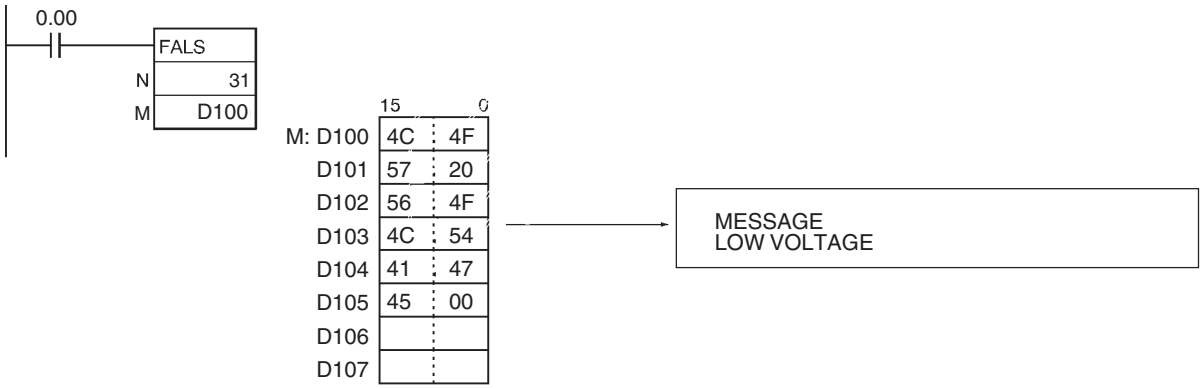
В приведенном ниже примере программы после включения бита CIO 0.00 команда FALS(007) сгенерирует критическую ошибку с номером FALS 31, что приведет к следующим событиям.

1,2,3...

1. Будет установлен флаг ошибки FALS (A401.06).
2. В слово A400 будет записан соответствующий код ошибки (C11F).

Примечание. Слово A400 будет содержать код самой серьезной из возникших на данный момент ошибок, включая некритические и критические системные ошибки, а также ошибки, сгенерированные командами FAL(006) и FALS(007).

3. В область журнала ошибок (A100...A199) будут записаны код ошибки и время/дата возникновения ошибки.
4. Будет непрерывно светиться индикатор «ERR» на модуле ЦПУ.
5. На экране устройства программирования отобразится сообщение, ASCII-коды которого содержатся в словах D100...D107. (Если сообщение не требуется, укажите в S константу.)

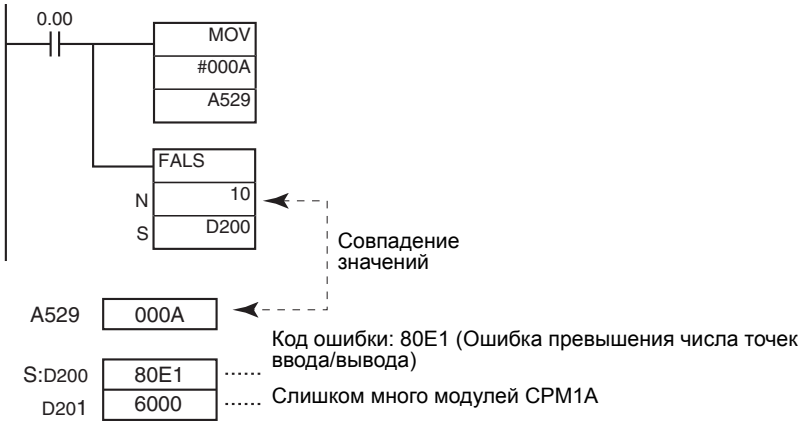


Формирование критической системной ошибки

В приведенном ниже примере после включения бита CIO 0.00 команда FALS(007) сгенерирует ошибку превышения числа точек ввода/вывода (слишком много модулей расширения или модулей расширения вх./вых. серии CPM1A). В данном случае используется номер фиктивной ошибки FALS 10, поэтому в A529 записывается соответствующее значение (000A hex).

1,2,3...

1. В слово A400 будет записан указанный код ошибки (80E1), если это наиболее серьезная ошибка на данный момент.
2. В область журнала ошибок (A100...A199) будут записаны код ошибки и время/дата возникновения ошибки.
3. Будет установлен флаг ошибки превышения числа точек ввода/вывода (A401.11).
4. Будет непрерывно светиться индикатор «ERR» на модуле ЦПУ, работа ПЛК будет остановлена.
5. В CX-Programmer отобразится сообщение (TOO MANY I/O PNT), уведомляющее о возникновении ошибки превышения числа точек ввода/вывода.

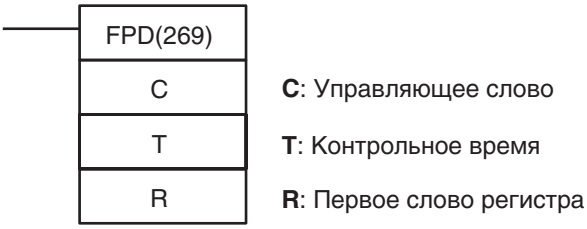


3-28-3 ОБНАРУЖЕНИЕ НЕИСПРАВНОГО КАНАЛА: FPD(269)

Назначение

Выявление причины ошибки в блоке команд путем контроля времени, которое проходит между выполнением команды FPD(269) и срабатыванием диагностического выхода, и путем определения входа, по вине которого не включается выход.

Символ РКС



Варианты выполнения

Варианты выполнения		
	Выполнение в каждом цикле при включенном условии выполнения	FPD(269)
	Однократное выполнение по положительному фронту	Не предусмотрено
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

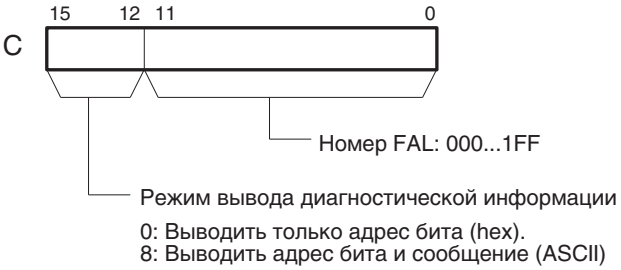
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
Не допускается	OK	OK	Не допускается

Операнды

C: Управляющее слово

Операнд C должен содержать константу от 0000 до 01FF или от 8000 до 81FF.

Назначение каждого из разрядов управляющего слова показано на следующем рисунке.



T: Контрольное время

Операнд T должен иметь значение от 0000 до 270F hex (от 0 до 9999 в десятичном коде). Значение «0» отключает контроль времени; значениям в диапазоне от 1 до 270F соответствуют значения контрольного времени от 0,1 до 999,9 секунд.

R: Первое слово регистра

Функции слов регистра описаны на стр. 1044.

Характеристики операндов

Область	C	T	R
Область CIO	---	CIO 0...CIO 6143	
Рабочая область	---	W0...W511	
Область битов хранения	---	H0...H511	
Область вспомогательных битов	---	A0...A447 A448...A959	A448...A959
Область таймеров	---	T0000...T4095	
Область счетчиков	---	C0000...C4095	
Область DM	---	D0...D32767	
Косвенные адреса DM в двоичном формате	---	@ D0...@ D32767	
Косвенные адреса DM в двоично-десятичном формате (BCD)	---	*D0...*D32767	
Постоянные	Только указанные значения	#0000...#270F (двоичн.)	---
Регистры данных	---		
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	---	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15	

Описание

Команда FPD(269) выполняет две функции: контроль времени и логическую диагностику. Функция контроля времени генерирует некритическую ошибку с указанным номером FAL, если диагностический выход не включается за указанное контрольное время. Функция логической диагностики выявляет вход, препятствующий включению выхода.



Примечание. *Блок логической диагностики начинается с первой команды LD (но не LD TR) или LD NOT после команды FPD(269) и завершается первой командой OUT (но не OUT TR) или другой правосторонней командой.

Функция контроля времени

В момент своего выполнения (после включения условия выполнения A) команда FPD(269) запускает отсчет времени. Если диагностический выход не включится за указанное контрольное время, FPD(269) сгенерирует некритическую ошибку и установит флаг переноса.



Генерируется некритическая ошибка.

Примечание. Диагностический выход должен включиться за указанное контрольное время. Контрольное время может быть установлено автоматически с помощью функции обучения.

При включении флага переноса будут выполнены перечисленные ниже действия (эти действия не будут выполнены, если номер FAL в C = 000).

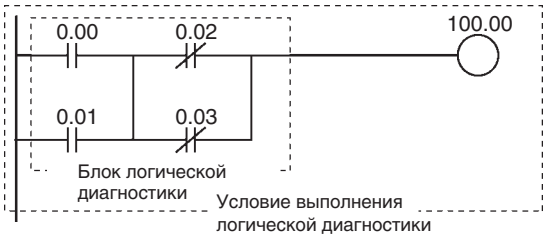
1,2,3...

1. Будет установлен флаг ошибки FAL (A402.15) (работа ПЛК остановлена не будет).
2. Будет установлен флаг номера выполненной команды FAL для указанного номера FAL (флаги A360.01...A391.15 соответствуют номерам FAL 001...1FF).
3. В слово A400 будет записан соответствующий код ошибки. Коды ошибок 4101...42FF соответствуют номерам FAL 001...1FF.
   (При наличии в данный момент более серьезной ошибки (ошибки с более высоким кодом ошибки) в A400 будет записан код более серьезной ошибки.)
4. В область журнала ошибок (A100...A199) будут записаны код ошибки и время/дата возникновения ошибки.
5. Будет мигать индикатор «ERR» на модуле ЦПУ.
6. Если выбран режим выдачи адреса бита и сообщения (старший разряд слова C = 8), отобразится сообщение о некритической ошибке, текст которого (ASCII-коды) записан в слова R+2...R+10.

Функция логической диагностики

В каждом цикле, в котором включено условие выполнения команды FPD(269), команда FPD(269) определяет, по вине какого из битов входов остается выключенным диагностический выход, и записывает адрес бита в область регистра, начиная с R.

Если все три бита входов CIO 0.00...CIO 0.03 в приведенном ниже примере будут одновременно включены, команда FPD(269) установит, что причиной выключенного состояния выхода CIO 100.00 является нормально замкнутое условие CIO 0.02. Команда FPD(269) включит флаг обнаружения адреса бита (бит 15 слова R) и запишет адрес бита в слова регистра R+2...R+4.



Функция логической диагностики выполняется в каждом цикле, пока остается включенным условие выполнения для FPD(269). Работа функции логической диагностики не зависит от работы функции контроля времени.

Если включению диагностического выхода препятствует сразу 2 или больше битов входов, в слова R+2...R+4 записывается адрес самого первого бита входа в условии выполнения (ближайший к левой шине бит в самой верхней строке команд).

Функция логической диагностики проверяет биты входов в командах LD, LD NOT, AND, AND NOT, OR и OR NOT (в том числе выполняемых по фронту и с мгновенным обновлением). Биты входов в других командах и операнды, адресованные косвенно с помощью регистров указателей, не проверяются.

Блок логической диагностики начинается с первой команды LD (но не LD TR) или LD NOT, расположенной после команды FPD(269), и завершается первой командой OUT (но не OUT TR) или другой командой, примыкающей к правой шине.

С помощью старшего разряда операнда C можно выбрать один из двух следующих режимов выдачи диагностической информации.

1,2,3...

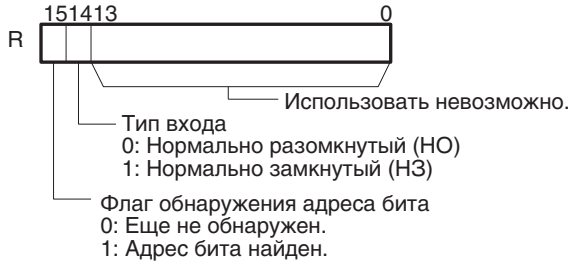
1. Режим выдачи адреса бита (старший разряд C = 0)
   Если адрес бита входа определен, включается бит 15 слова R (флаг обнаружения адреса бита). При этом бит 14 слова R указывает тип входа: нормально включенный или нормально выключенный.
   В слова R+3 и R+2 выдается 8-разрядный шестнадцатеричный адрес памяти ПЛК обнаруженного бита входа.
2. Режим выдачи адреса бита и сообщения (старший разряд C = 8)
   Если адрес бита входа найден, включается бит 15 слова R (флаг обнаружения адреса бита). При этом бит 14 слова R указывает тип входа: нормально включенный или нормально выключенный.
   В слова R+2...R+4 выдается адрес бита входа в виде 6 символов ASCII.

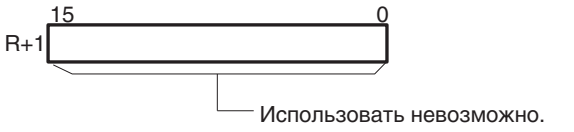
Предназначение слов регистра

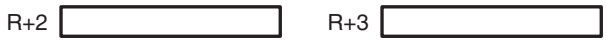
Слова регистра содержат результаты работы функции диагностики и также могут содержать сообщение об ошибке в кодировке ASCII, отображаемое при формировании ошибки функцией контроля времени. Назначение слов регистра зависит от режима выдачи диагностической информации, установленного старшим разрядом операнда C.

Выдача адреса бита (C=0@@@)

Если старший разряд операнда C задан равным «0», в слова R+2 и R+3 выдается 8-разрядный шестнадцатеричный адрес памяти ПЛК бита входа. В слове R также есть два флага, которые указывают, был ли обнаружен бит входа и во входном условии какого типа он используется: нормально разомкнутом или нормально замкнутом.

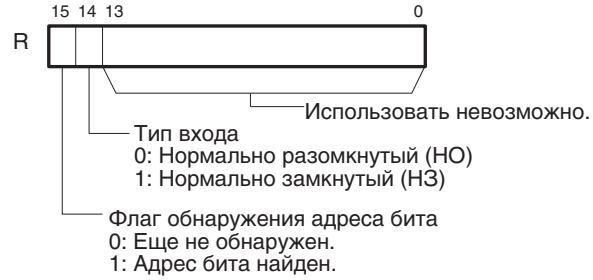






Выдача адреса бита и сообщения (C=8@@@)

Если старший разряд операнда = 8, в слова R+2...R+4 выдается адрес бита входа в кодировке ASCII. В слове R также есть два флага, которые указывают, был ли обнаружен бит входа и во входном условии какого типа он используется: нормально разомкнутом или нормально замкнутом.





Слова регистра R+2...R+4 указывают адрес входа, из-за которого не может включиться диагностический выход. Адрес найденного бита выдается в эти слова в кодах ASCII. В следующей таблице для каждой области показано ее представление в кодах ASCII.

Область	Текст ASCII	Пояснения
Вспомогательная область	A0.00...A959.15	---
Область хранения	H0.00...H511.15	---
Рабочая область	W0.00...W511.15	---
Область CIO	0.00...6143.15	---
Флаги задач	TK0...TK1023	---
Область таймеров	_T0..._T4095	«_» обозначает «пробел». (ASCII-код 20)
Область счетчиков	_C0..._C4095	

15

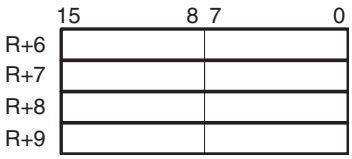
R+2	W	5
R+3	1	1
R+4	1	5

Адрес бита в кодах ASCII

В следующей таблице приведены значения слов регистра R+2...R+5 для бита W511.15:

Слово	Биты 8...15	Биты 0...7
R+2	W	5
R+3	1	1
R+4	1	5
R+5	2D (hex)	Тип входа (hex) 30: нормально разомкнутый 31: нормально замкнутый

Пользователь может записать ASCII-коды сообщения в слова регистра R+6...R+9. Когда функция контроля времени сгенерирует некритическую ошибку, это сообщение отобразится на экране CX-Programmer. Признаком конца сообщения является символ «null» (00 hex).



Отключение регистрации некритических ошибок FPD(269) в журнале ошибок

В обычном случае, когда функция контроля времени FPD(269) генерирует некритическую ошибку, код ошибки и время ее возникновения записываются в область журнала ошибок (A100...A199). В области настроек ПЛК, однако, предусмотрен параметр, с помощью которого можно отключить регистрацию некритических ошибок, генерируемых командой FPD(269), в журнале ошибок.

Даже если ошибка не регистрируется в журнале ошибок, она все равно приводит к установке флага ошибки FAL (A402.15), установке соответствующего флага номера выполненной команды FAL (A360.01...A391.15) и записи кода ошибки в A400.

Если в журнале ошибок должны регистрироваться только ошибки, генерируемые системой, регистрацию ошибок контроля времени, вызываемых командой FPD(269), нужно отключить. Этой функцией можно воспользоваться, например, на этапе отладки, когда частое применение команд FPD(269) и FAL(006) в нескольких прикладных программах приводит к быстрому заполнению журнала ошибками этих команд. В следующей таблице описана настройка соответствующего параметра в области настроек ПЛК.

Параметр	Значение
Название	Регистрация ошибок FAL в журнале
Значение	0: Регистрировать ошибки FAL в журнале ошибок. 1: Не регистрировать ошибки FAL в журнале ошибок.
Настройка по умолчанию	0: Регистрировать ошибки FAL в журнале ошибок.
Время считывания параметра в области настроек ПЛК	В каждом цикле (когда происходит ошибка FAL)

Перечисленные ниже ошибки регистрируются, даже если бит 15 слова 129 в настройках ПЛК равен «1» (не регистрировать ошибки FAL в журнале ошибок).

- Критические ошибки, генерируемые командой FALS(007).
- Некритические ошибки, генерируемые системой.
- Критические ошибки, генерируемые системой.
- Некритические ошибки, генерируемые системой намеренно с помощью команды FAL(006).
- Критические ошибки, генерируемые системой намеренно с помощью команды FALS(007).

Установка контрольного времени с помощью функции обучения

Если в операнде T указан адрес слова, значение контрольного времени может быть установлено автоматически с помощью функции обучения. В случае указания адреса слова в операнде T соблюдайте следующий порядок действий.

1,2,3...

1. Включите бит обучения FPD (A598.00).
2. Команда FPD(269) измерит время, которое пройдет с момента включения условия выполнения FPD(269) до включения диагностического выхода.
3. Если измеренное время окажется больше установленного контрольного времени, в T будет записано значение, в 1,5 раза превышающее измеренное время.

Флаги

Название	Обо-значение	Действие
Флаг ошибки	ER	Включен, если значение C выходит за допустимый диапазон: 0000...01FF или 8000...81FF. Включен, если значение T выходит за допустимый диапазон: 0000...270F. Выключен во всех остальных случаях.
Флаг переноса	CY	Включен, если диагностический выход не включился за контрольное время. Выключен во всех остальных случаях.

В следующей таблице перечислены сопутствующие слова и флаги вспомогательной области.

Название	Адрес	Действие
Флаг ошибки FAL	A402.15	Включен, если функция контроля времени сгенерировала некритическую ошибку (FAL).
Флаги номера выполненной команды FAL	A360.01... A391.15	Если функция контроля времени генерирует некритическую ошибку (FAL), устанавливается соответствующий флаг. Флаги A360.01... A391.15 соответствуют номерам FAL 0001...01FF.
Область журнала ошибок	A100... A199	Область журнала ошибок содержит коды ошибок и время/дату возникновения для 20 самых последних ошибок, включая ошибки, сгенерированные командой FPD(269).
Код ошибки	A400	При возникновении ошибки ее код записывается в слово A400. Номерам ошибок FAL от 0001 до 01FF соответствуют коды ошибок от 4101 до 42FF. При одновременном возникновении двух или более ошибок в A400 записывается код наиболее серьезной ошибки.
Бит обучения FPD	A598.00	Если данный бит будет включен, при выполнении команды FPD(269) контрольное время будет установлено автоматически (функцией обучения).

Меры предосторожности

При использовании функции контроля времени условие выполнения команды FPD(269) должно оставаться включенным в течение всего контрольного времени, заданного в T.

Условие выполнения команды FPD(269) должно представлять из себя комбинацию из нормально разомкнутых и нормально замкнутых входов.

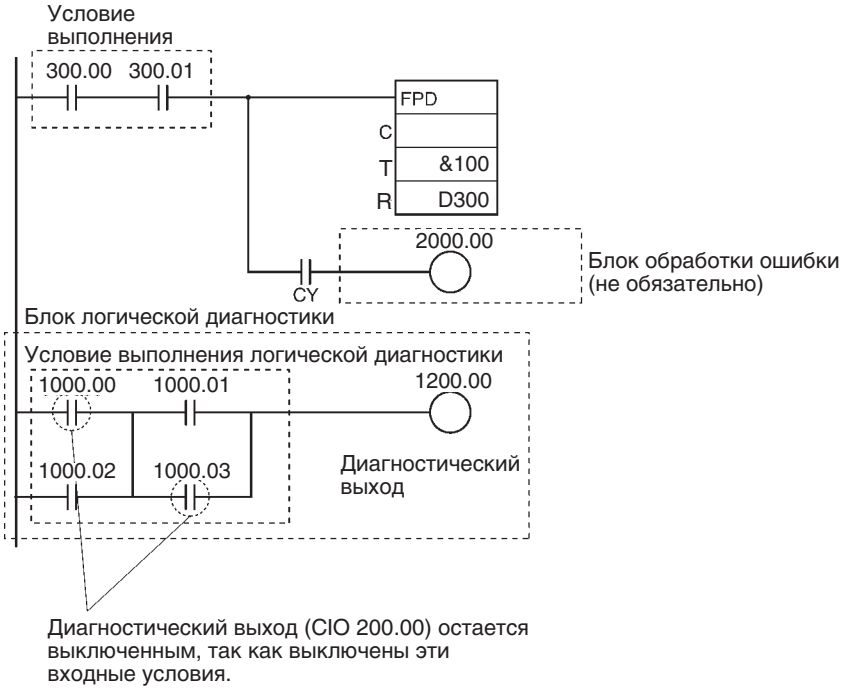
Наличие блока обработки ошибок не является обязательным. Дополняя программу блоком обработки ошибок, обязательно используйте команды вывода или другие правосторонние команды. Команды LD и LD NOT здесь использовать нельзя.

Команду FPD(269) можно использовать в программе несколько раз, но каждая команда должна иметь уникальную настройку слов регистра (R).

Значение контрольного времени обновляется только при выполнении FPD(269). При длительности цикла программы больше 100 мс контрольное время не будет обновляться нормально и команда FPD(269) будет работать неправильно, поскольку отсчет контрольного времени производится с шагом в 100 мс.

Примеры

Приведенный ниже пример программы демонстрирует работу функций контроля времени и логической диагностики. В данном примере диагностический выход (CIO 1200.00) не включается из-за того, что остаются выключенными биты CIO 1000.00 и CIO 1000.03 в условии выполнения блока логической диагностики.



Функция контроля времени

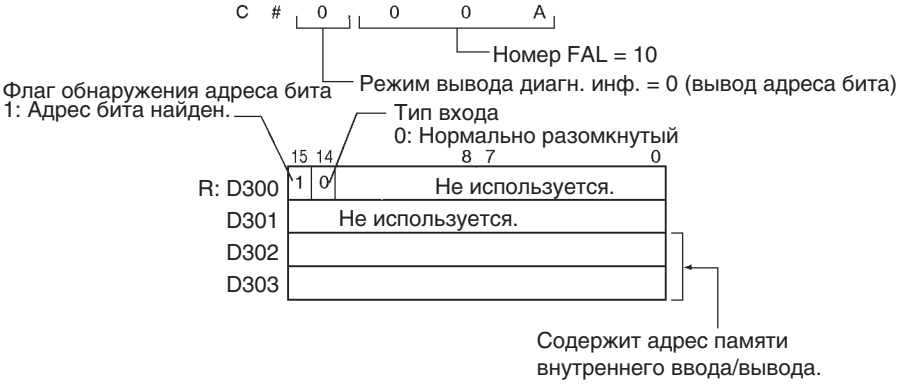
Если диагностический выход (CIO 1200.00) не включается в течение 10 секунд после включения битов CIO 300.00 и CIO 300.01, генерируется некритическая ошибка и происходят следующие события.

1,2,3...

1. Устанавливается флаг переноса.
2. Если в трех младших разрядах операнда C указан номер FAL = 00A hex (10), устанавливается соответствующий флаг номера выполненной команды FAL (A360.10), в слово A400 записывается соответствующий код ошибки (410A) и устанавливается флаг ошибки FAL (A402.15).

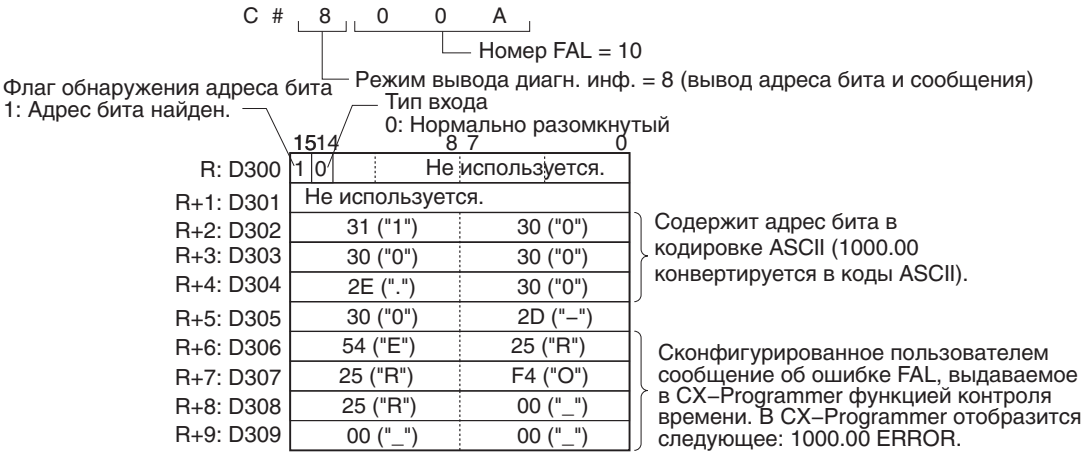
Функция логической диагностики (C=000A)

Поскольку старший разряд C = 0 (режим выдачи адреса бита), в слова D303 и D302 выдается адрес памяти ПЛК бита CIO 1000.00 (так как строка с битом CIO 1000.00 находится выше строки с битом CIO 1000.03).



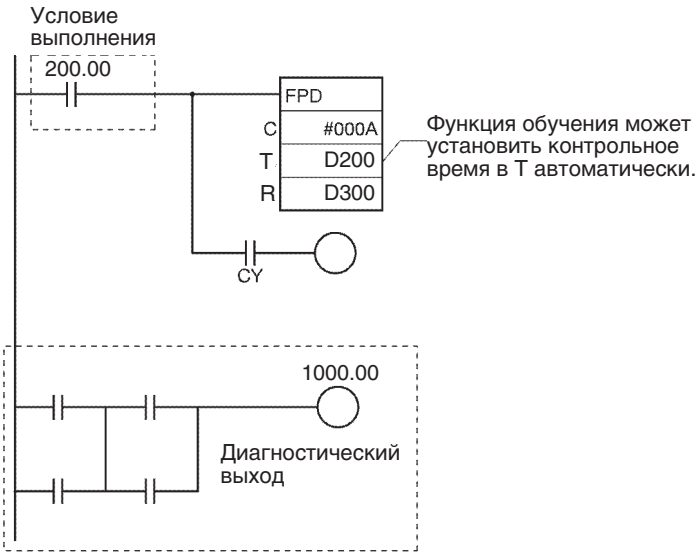
Функция логической диагностики (C=800A)

Поскольку старший разряд C = 8 (режим выдачи адреса бита и сообщения), в слова D302...D304 выдается адрес бита CIO 1000.00 (1000.03) в кодировке ASCII.

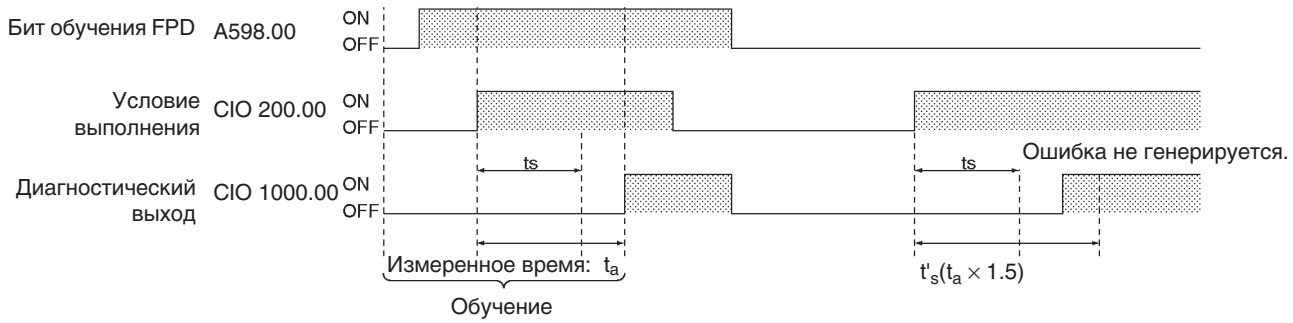


Установка контрольного времени с помощью функции обучения

Значение контрольного времени может быть установлено автоматически. Для этого в операнд T следует записать адрес слова.



Для запуска функции обучения следует включить бит A598.00 (бит обучения FPD). Пока бит A598.00 включен, команда FPD(269) измеряет, сколько времени проходит с момента включения условия выполнения (CIO 200.00) до включения диагностического выхода (CIO 1000.00). Если измеренное время оказывается больше контрольного времени в T, измеренное время умножается на 1,5 и полученное значение записывается в T в качестве нового значения контрольного времени.



t_s: Начальное значение в T

t_a: Измеренное время

t'_s: Новое значение в T после обучения

(Если $t_a > t_s$, $t'_s = t_a \times 1,5$)

3-29 Прочие команды

В данном разделе описаны команды, предназначенные для управления состоянием флага переноса, увеличения максимальной длительности цикла, сохранения и загрузки состояний флагов условий и преобразования адресов памяти.

Команда	Мнемоническое обозначение	Код функции	Стр.
УСТАНОВИТЬ ПЕРЕНОС	STC	040	1051
СБРОСИТЬ ПЕРЕНОС	CLC	041	1052
УВЕЛИЧИТЬ МАКС. ВРЕМЯ ЦИКЛА	WDT	094	1053
СОХРАНИТЬ ФЛАГИ УСЛОВИЙ	CCS	282	1055
ЗАГРУЗИТЬ ФЛАГИ УСЛОВИЙ	CCL	283	1057
ПРЕОБРАЗОВАТЬ АДРЕС СЕРИИ CV	FRMCV	284	1059
ПРЕОБРАЗОВАТЬ В АДРЕС СЕРИИ CV	TOCV	285	1063

3-29-1 УСТАНОВИТЬ ПЕРЕНОС: STC(040)

Назначение

Установка флага переноса (CY).

Символ РКС



Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	STC(040)
	Однократное выполнение по положительному фронту	@STC(040)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Описание

При включенном условии выполнения команда STC(040) устанавливает флаг переноса (CY). Флаг переноса, установленный командой STC(040), может быть вновь сброшен или установлен в результате выполнения последующих команд, влияющих на состояние флага переноса.

Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	---
Флаг равенства	=	---
Флаг переноса	CY	ВКЛ
Флаг отрицательного значения	N	---

Меры предосторожности

Команды ROL(027), ROLL(572), ROR(028) и RORL(573) используют флаг переноса при выполнении операций циклического сдвига. При использовании любой из этих команд производите установку и сброс флага переноса с помощью команд STC(040) и CLC(041).

3-29-2 СБРОСИТЬ ПЕРЕНОС: CLC(041)

Назначение

Сброс флага переноса (CY).

Символ РКС



Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	CLC(041)
	Однократное выполнение по положительному фронту	@CLC(041)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Описание

При включенном условии выполнения команда CLC(040) сбрасывает флаг переноса (CY). Флаг переноса, сброшенный командой CLC(040), может быть вновь установлен или сброшен в результате выполнения последующих команд, влияющих на состояние флага переноса.

Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	---
Флаг равенства	=	---
Флаг переноса	CY	ВЫКЛ
Флаг отрицательного значения	N	---

Меры предосторожности

Команды +C(402), +CL(403), +BC(406) и +BCL(407) используют флаг переноса при выполнении операций сложения. Предусматривайте команду CLC(041) непосредственно перед любой из этих команд, чтобы исключить влияние результатов выполнения любых предыдущих команд.

Команды –C(412), –CL(413), –BC(416) и –BCL(417) используют флаг переноса при выполнении операций вычитания. Предусматривайте команду CLC(041) непосредственно перед любой из этих команд, чтобы исключить влияние результатов выполнения любых предыдущих команд.

Команды ROL(027), ROLL(572), ROR(028) и RORL(573) используют флаг переноса при выполнении операций циклического сдвига. При использовании любой из этих команд производите установку и сброс флага переноса с помощью команд STC(040) и CLC(041).

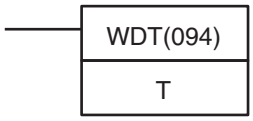
Примечание. Команды +(400), +L(401), +B(404), +BL(405), –(410), –L(411), –B(414) и –BL(415) не используют флаг переноса при выполнении операций сложения и вычитания. В большинстве случаев для сложения или вычитания предпочтительнее использовать эти команды.

3-29-3 УВЕЛИЧИТЬ МАКС. ВРЕМЯ ЦИКЛА: WDT(094)

Назначение

Увеличение максимальной длительности цикла (только для цикла, в котором выполнена команда). Если в некотором цикле необходимо выполнить особую операцию, требующую большого времени, длительность цикла можно временно увеличить с помощью команды WDT(094) во избежание ошибки превышения длительности цикла.

Символ РКС



T: Уставка таймера

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	WDT(094)
	Однократное выполнение по положительному фронту	@WDT(094)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

T: Уставка таймера

Содержит уставку сторожевого таймера в диапазоне от 0000 до 0F9F hex (или от &0000 до &3999 десятичн.).

Характеристики операндов

Область	T
Область CIO	---
Рабочая область	---
Область битов хранения	---
Область вспомогательных битов	---
Область таймеров	---
Область счетчиков	---
Область DM	---
Косвенные адреса DM в двоичном формате	---
Косвенные адреса DM в двоично-десятичном формате (BCD)	---
Постоянные	0000...0F9F (двоичн.)
Регистры данных	---
Регистры указателей	---
Косвенная адресация с помощью регистров указателей	---

Описание

Команда WDT(094) увеличивает максимальную длительность цикла для цикла, в котором она выполняется. Уставка сторожевого таймера, заданная в настройках ПЛК, увеличивается на время T × 10 мс (0...39 990 мс).

В следующей таблице указаны возможные значения уставки сторожевого таймера в настройках ПЛК. По умолчанию максимальная длительность цикла установлена равной 1000 мс, хотя для нее можно установить любое значение от 1 до 40 000 мс с шагом 10 мс.

Название	Функция	Значение
Длительность цикла слежения	Если фактическая длительность цикла превысит установленное здесь максимальное значение, возникнет ошибка превышения допустимой длительности цикла (критическая ошибка).	0: значение по умолчанию (1 000 мс); 1: значение пользователя.
	Задает максимальную длительность цикла. (Этот параметр действует, только если первый параметр задан равным «1».)	0001...0FA0 (10...40 000 мс, с шагом 10 мс)

Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	Включен, если уставка сторожевого таймера превышает 40 секунд. Выключен во всех остальных случаях.

В следующей таблице перечислены сопутствующие слова и флаги вспомогательной области.

Название	Адрес	Действие
Флаг слишком большой длительности цикла	A401.08	Включен, если длительность текущего цикла превысила максимальную длительность цикла (длительность цикла слежения), заданную в настройках ПЛК. Эта критическая ошибка прекращает выполнение программы.
Максимальное время цикла	A262... A263	Эти слова содержат 32-битовое двоичное значение максимальной длительности цикла. Данное значение обновляется в каждом цикле.
Время выполнения текущего цикла	A264 и A265	Эти слова содержат 32-битовое двоичное значение времени выполнения текущего цикла. Данное значение обновляется в каждом цикле.

Меры предосторожности

Команду WDT(094) можно использовать в одном цикле несколько раз. Каждая команда WDT(094) дополнительно увеличивает длительность цикла (т. е. приращения суммируются), но общая длительность цикла не должна превышать 40 000 мс. Если длительность цикла уже увеличена до 40 000 мс, команду WDT(094) больше использовать невозможно.

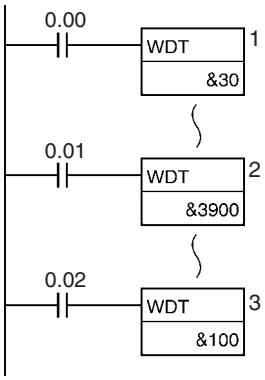
Примеры

В данном примере используется значение максимальной длительности цикла, принимаемое по умолчанию (1 000 мс).

1,2,3...

1. Когда включается бит CIO 0.00, первая команда WDT(094) увеличивает максимальную длительность цикла на 300 мс (30 × 10 мс). Таким образом, на этом этапе макс. длительность цикла составляет 1 300 мс.

2. Когда включается бит CIO 0.01, вторая команда WDT(094) пытается увеличить максимальную длительность цикла еще на 39 000 мс. Поскольку новое значение максимальной длительности цикла (40 300 мс) превышает верхнюю границу (40 000 мс), дополнительные 300 мс игнорируются. В результате, вторая команда WDT(094) увеличивает макс. длительность цикла на 38 700 мс.
3. Когда включается бит CIO 0.02, третья команда WDT(094) пытается увеличить максимальную длительность цикла еще на 1 000 мс. Поскольку к этому времени уже достигнут максимальный предел длительности цикла (40 000 мс), третья команда WDT(094) не выполняется.

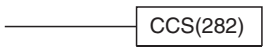


3-29-4 СОХРАНИТЬ ФЛАГИ УСЛОВИЙ: CCS(282)

Назначение

Сохранение текущего состояния флагов условий в отдельную область памяти внутри модуля ЦПУ. Сохраненное текущее состояние флагов может быть затем прочитано с помощью команды CCL(283) и записано («восстановлено») в другом месте программы, в другой задаче или даже в другом цикле.

Символ РКС



Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	CCS(282)
	Однократное выполнение по положительному фронту	@CCS(282)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

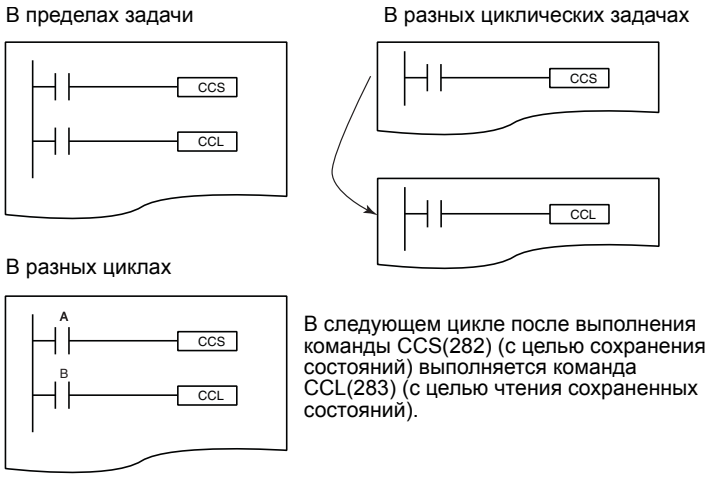
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Описание

При включенном условии выполнения команда CCS(282) сохраняет текущее состояние флагов условий (кроме флагов «Всегда ВКЛ» и «Всегда ВЫКЛ») в отдельную область памяти в пределах модуля ЦПУ. Сохраняются состояния следующих флагов условий: ER, CY, >, =, <, N, OF, UF, >=, <> и <=.

Сохраненные состояния флагов условий могут быть позднее прочитаны («восстановлены») только с помощью команды CCL(283) (ЗАГРУЗИТЬ ФЛАГИ УСЛОВИЙ). Чтение сохраненных состояний возможно:

- в пределах задачи;
- в разных циклических задачах;
- в разных циклах.



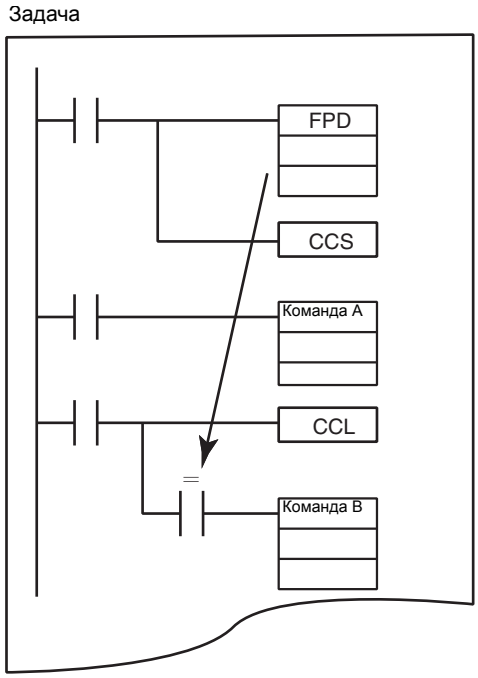
Примечание.

(1) Обмен состояниями флагов условий путем их сохранения/загрузки невозможен между циклической задачей и задачей обработки прерывания.

(2) Выполняемая команда CCS(282) записывает состояние флагов условий вместо ранее сохраненных.

Когда прикладная программа переходит от одной задачи к другой, все флаги условий сбрасываются. Для передачи флагов условий из одной задачи в другую или из одного цикла в другой следует использовать команды CCS(282) и CCL(283).

Например, команды CCS(282) и CCL(283) позволяют сохранить состояние флага CY (ошибка диагностики контроля времени), полученное после выполнения команды FPD(269), и использовать его в программе позднее, а не сразу после выполнения этой команды.



Во флаги условий записываются результаты сравнения. (В данном случае результаты выполнения команды сравнения удастся использовать в команде B, даже если они будут изменены командой A.)

Состояния флагов условий сохраняются в отдельной области памяти модуля ЦПУ.

Восстанавливаются состояния флагов условий.

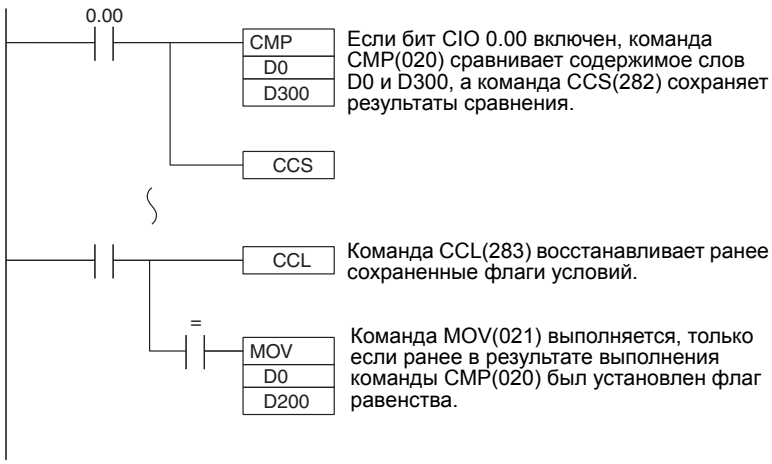
Флаг равенства отражает результат выполнения команды сравнения, а не команды A.

Флаги

Эти команды не изменяют состояние какого-либо флага.

Примеры

В приведенном ниже примере команда CCS(282) сохраняет результаты операции сравнения, благодаря чему эти результаты можно позже использовать в качестве условия выполнения в другом месте программы.



Если бит CIO 0.00 включен, команда CMP(020) сравнивает содержимое слов D0 и D300, а команда CCS(282) сохраняет результаты сравнения.

Команда CCL(283) восстанавливает ранее сохраненные флаги условий.

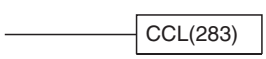
Команда MOV(021) выполняется, только если ранее в результате выполнения команды CMP(020) был установлен флаг равенства.

3-29-5 ЗАГРУЗИТЬ ФЛАГИ УСЛОВИЙ: CCL(283)

Назначение

Восстановление состояний флагов условий, ранее сохраненных в отдельную область памяти внутри модуля ЦПУ командой CCS(282). Команду CCL(283) также можно использовать независимо от команды CCS(282) для очистки состояний флагов условий.

Символ РКС



Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	CCL(283)
	Однократное выполнение по положительному фронту	@CCL(283)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

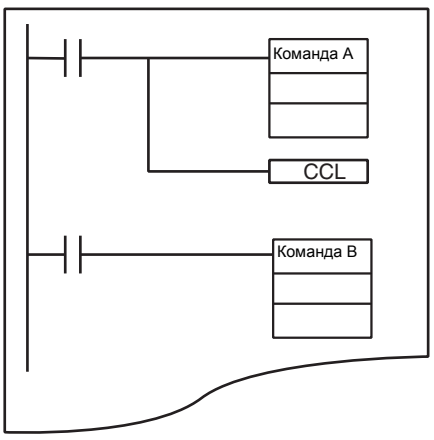
Описание

При включенном условии выполнения команда CCL(283) считывает («восстанавливает») состояние флагов условий (кроме флагов «Всегда ВКЛ» и «Всегда ВЫКЛ»). Восстанавливаются состояния следующих флагов условий: ER, CY, >, =, <, N, OF, UF, >=, <> и <=.

Флаги условий используются всеми командами, поэтому их состояния в пределах цикла ПЛК могут изменяться много раз при выполнении команд. Ранее для того чтобы использовать флаги условий, полученные в результате выполнения команды A, в качестве условий выполнения команды B, требовалось размещать команду B непосредственно после команды A (чтобы промежуточные команды не повлияли на состояние флагов условий). Команды CCS(282) и CCL(283) устраняют это неудобство и позволяют использовать флаги условий команды A в качестве условий выполнения команды B без привязки этих команд друг к другу.

Например, с помощью CCS(282) можно сохранить состояние флага равенства, полученное после выполнения команды сравнения, чтобы использовать результат сравнения не сразу после выполнения команды, а позже.

Задач



Команда CCL(283) используется отдельно для очистки состояний флагов условий после выполнения команды A, чтобы эти результаты не повлияли на выполнение команды B и следующих за ней команд.

Другие примеры использования команд CCS(282) и CCL(283) в программе приведены в разделе *3-29-4 СОХРАНИТЬ ФЛАГИ УСЛОВИЙ: CCS(282)*.

Флаги

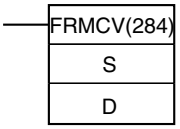
Эти команды не изменяют состояние какого-либо флага.

3-29-6 ПРЕОБРАЗОВАТЬ АДРЕС СЕРИИ CV: FRMCV(284)

Назначение

Преобразование адреса памяти ПЛК серии CV в соответствующий адрес памяти ПЛК серии CS/CJ/CP. Команду FRMCV(284) можно использовать для преобразования программ серии CV, использующих адреса памяти ПЛК, в программы, совместимые с ПЛК серии CP.

Символ РКС



S: Слово, содержащее адрес памяти ПЛК серии CV
D: Адресуемый регистр указателя

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	FRMCV(284)
	Однократное выполнение по положительному фронту	@FRMCV(284)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

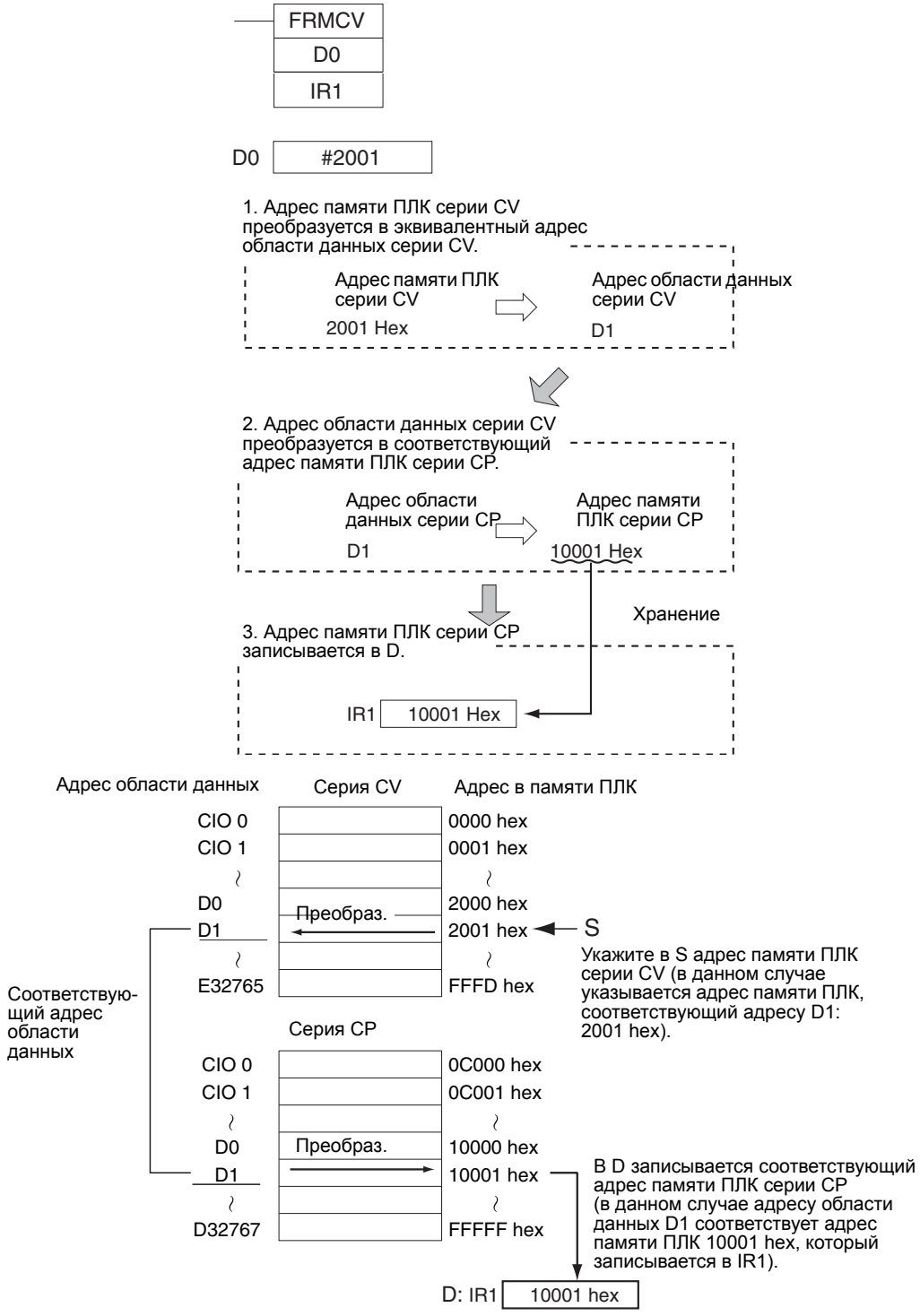
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Описание

При включенном условии выполнения команда FRMCV(284) выполняет следующие действия.

1. Адрес памяти ПЛК серии CV, указанный в S, преобразуется в эквивалентный адрес области данных серии CV.
2. Команда FRMCV(284) определяет адрес памяти ПЛК серии CP, который соответствует такому же адресу области данных в ПЛК серии CV.
3. В операнд D выдается адрес памяти ПЛК серии CP (в D должен быть указан регистр указателя (IR0…IR15)).

Ниже показан пример использования команды FRMCV(284) для преобразования адреса памяти ПЛК серии CV, указанного в D0.



Примечание. Если указанному адресу памяти ПЛК серии CV не соответствует ни один адрес в ПЛК серии CP, произойдет ошибка, будет установлен флаг ошибки и адрес преобразован не будет.

Если регистр указателя используется в операнде с префиксом «,IR», команда будет использовать слово, указанное адресом памяти ПЛК в регистре указателя, а не сам регистр указателя. После того как требуемый адрес памяти ПЛК записан в регистр указателя, сам регистр указателя может использоваться в качестве операнда команды.

Команду FRMCV(284) можно использовать для преобразования программы ПЛК серии CV двух указанных ниже типов в программу, совместимую с ПЛК серии CP. Примеры преобразования см. далее в подразделе *Примеры*.

1. Использование косвенной адресации через область DM в двоичном формате (*DM) (адрес области данных указывается косвенно через адрес DM, по которому записан адрес памяти ПЛК).
2. Использование непосредственно значений адресов памяти ПЛК серии CV (запись адресов памяти ПЛК в регистры указателей, например с помощью команды MOV(021), с использованием прямой адресации).

Характеристики операндов

Область	S	D
Область CIO	CIO 0...CIO 6143	---
Рабочая область	W0...W511	---
Область битов хранения	H0...H511	---
Область вспомогательных битов	A448...A959	---
Область таймеров	T0000...T4095	---
Область счетчиков	C0000...C4095	---
Область DM	D0...D32767	---
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	---
Косвенные адреса DM в двоично-десятичном формате (BCD)	*D0...*D32767	---
Постоянные	Любая константа, кроме 09FF hex, 0A00…0AFF hex или 0D00…0E3F hex	---
Регистры данных	DR0...DR15	---
Регистры указателей	---	IR0...IR15
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15	---

Флаги

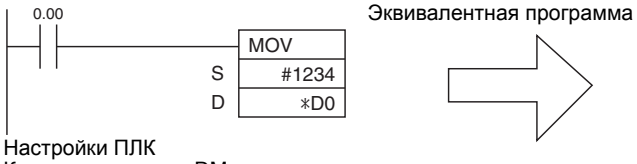
Название	Обо-значение	Действие
Флаг ошибки	ER	Включен, если в S указан один из следующих адресов памяти ПЛК, не существующий в серии CP: область временного хранения (TR) (09FF hex), область связей шины ЦПУ (G) (0A00...0AFF hex), области SFC (0D00...0E3F hex). Выключен во всех остальных случаях.

Примеры

Пример 1. Преобразование программы серии CV, использующей косвенную адресацию посредством слов DM в двоичном формате.

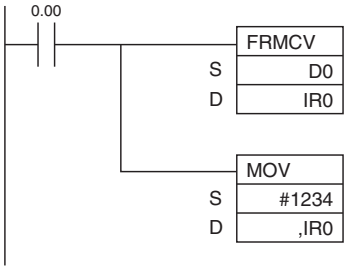
В следующем примере использования команды FRMCV(284) в операнде S указывается слово области DM, содержащийся там адрес памяти ПЛК записывается в регистр указателя, регистр указателя используется для косвенной адресации.

• Программа ПЛК серии CV (используется косвенная адресация посредством DM в двоичном формате)



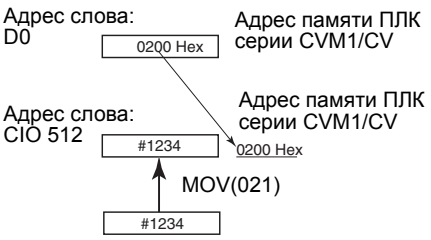
Эквивалентная программа

• Программа серии CP

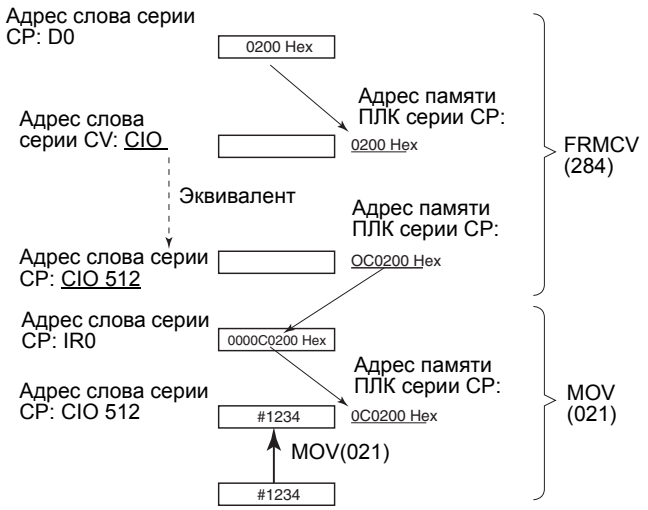


Настройки ПЛК
Косвенные адреса DM:
В случае косвенной адресации в двоичном формате содержимое слова DM воспринимается как адрес памяти ПЛК и указывает соответствующий адрес памяти ввода/вывода.

В данном примере D0 = 200 hex. Этому соответствует адрес области данных CIO 512, поэтому #1234 передается в CIO 512.



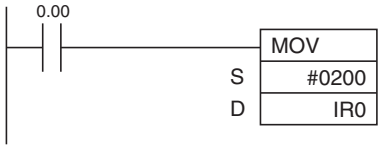
В данном примере D0 = 200 hex. Этому соответствует адрес области данных ПЛК серии CV: CIO 512. Адресу CIO 512 соответствует адрес 0000C200 hex в памяти ПЛК серии CP, поэтому это значение записывается в IR0. Выходной операнд команды MOV(021) адресуется косвенно через содержимое регистра IR0, поэтому значение #1234 передается в слово CIO 512.



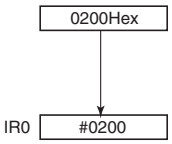
Пример 2. Преобразование программы ПЛК серии CV с прямым указанием адресов памяти ПЛК в регистрах указателей.

В этом примере использования программы FRMCV(284) в операнде S непосредственно указывается адрес памяти ПЛК серии CV.

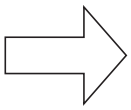
• Программа серии CV
(используются адреса памяти ПЛК, непосредственно указанные в IR)



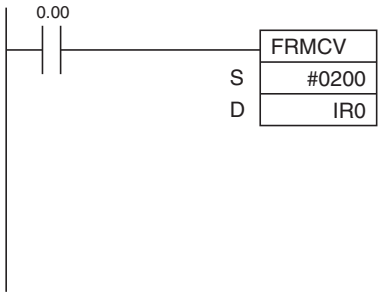
В данном случае в регистр указателя IR0 записывается адрес 0200 hex памяти ПЛК.



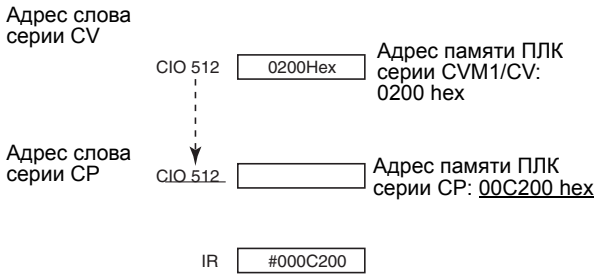
Эквивалентная



• Программа серии CP



В данном случае адресу 0200 hex в памяти ПЛК серии CV соответствует адрес CIO 512. Адресу CIO 512 соответствует адрес 0000C200 hex в памяти ПЛК серии CP, поэтому это значение записывается в регистр IR0.

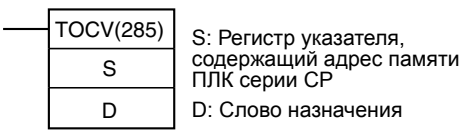


3-29-7 ПРЕОБРАЗОВАТЬ В АДРЕС СЕРИИ CV: TOCV(285)

Назначение

Преобразование адреса памяти ПЛК серии CS/CJ/CP в соответствующий адрес ПЛК серии CV. Команду TOCV(285) можно использовать для преобразования программ серии CP, использующих адреса памяти ПЛК, в программы, совместимые с ПЛК серии CV.

Символ РКС



Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	TOCV(285)
	Однократное выполнение по положительному фронту	@TOCV(285)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

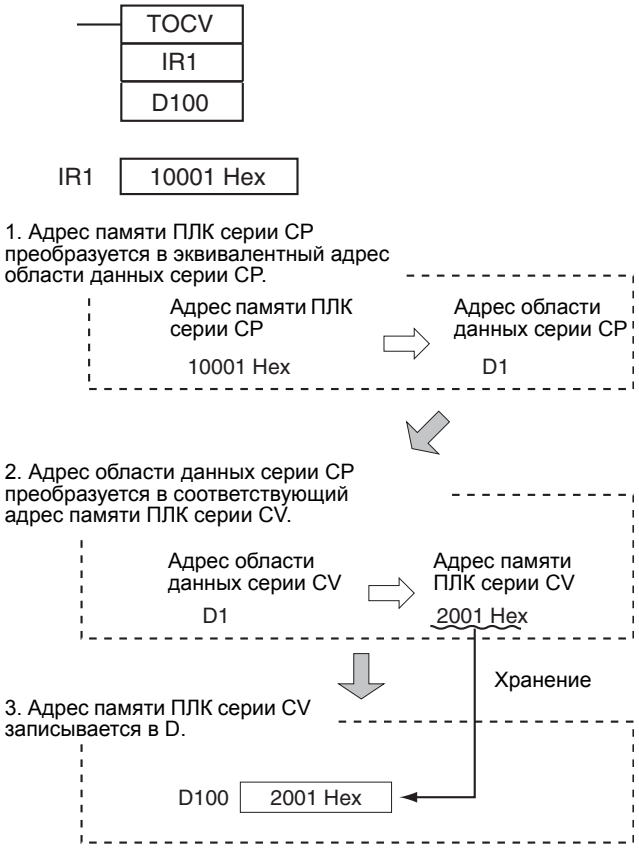
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

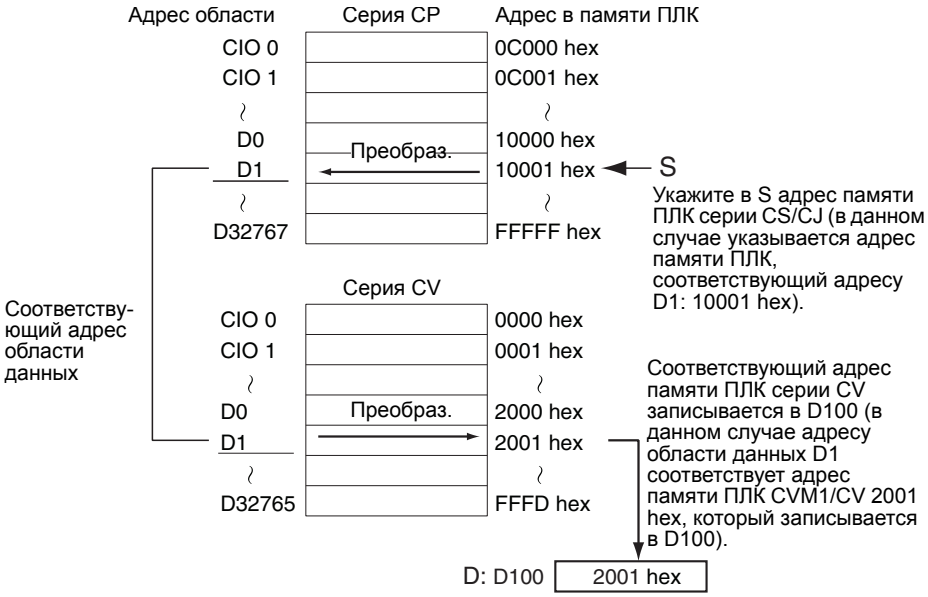
Описание

При включенном условии выполнения команда TOCV(285) выполняет следующие действия.

1. Адрес памяти ПЛК серии CP, указанный в S, преобразуется в эквивалентный адрес области данных серии CP (в S должен быть указан регистр указателя (IR0...IR15)).
2. Команда TOCV(285) определяет адрес памяти ПЛК серии CV, который соответствует такому же адресу области данных в ПЛК серии CP.
3. Адрес памяти ПЛК серии CV записывается в D.

Ниже показан пример использования команды TOCV(285) для преобразования адреса памяти ПЛК, указанного в IR1.





Примечание.

(1) Если указанному адресу памяти ПЛК серии CP не соответствует ни один адрес в ПЛК серии CV, произойдет ошибка, будет установлен флаг ошибки и адрес преобразован не будет.

(2) Значение адреса памяти ПЛК серии CV, сохраненное командой TOCV(285), может быть передано в ПЛК серии CV с помощью CX-Programmer.

(3) Адрес области данных, использованный в программе серии CP, может быть указан в программе серии CV с помощью косвенной адресации через регистр указателя (префикс «,IR») или с помощью косвенной адресации посредством слова DM в двоичном формате (*DM).

Характеристики операндов

Область	S	D
Область CIO	---	CIO 0...CIO 6143
Рабочая область	---	W0...W511
Область битов хранения	---	H0...H511
Область вспомогательных битов	---	A448...A959
Область таймеров	---	T0000...T4095
Область счетчиков	---	C0000...C4095
Область DM	---	D0...D32767
Косвенные адреса DM в двоичном формате	---	@ D0...@ D32767
Косвенные адреса DM в двоично-десятичном формате (BCD)	---	*D0...*D32767
Постоянные	См. примеч. 1	---
Регистры данных	---	DR0...DR15
Регистры указателей	IR0...IR15	---
Косвенная адресация с помощью регистров указателей	---	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15

Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

Примечание. (1) Если в S будет указан один из следующих адресов памяти ПЛК, не существующих в ПЛК серии CV, произойдет ошибка и будет установлен флаг ошибки.

Область или адреса	Адреса памяти ПЛК
Область флагов задач	0000 B800...0000 B801 hex
A512...A959	0000 BA40...0000 BBFF hex
CIO 2556...CIO 6143	0000 C9FC...0000 D7FF hex
T1024...T4095	0000 BE40...0000 BEFF hex и 0000 E400...0000 EFFF hex
C1024...C4095	0000 BF40...0000 BFFF hex и 0000 F400...0000 FFFF hex
Область HR	0000 D800...0000 D9FF hex
Область WR	0000 DE00...0000 DFFF hex
D24576...D32767	0001 6000...0001 7FFF hex

(2) Если в S вместо регистра указателя будет указан адрес любой другой области, произойдет ошибка и будет установлен флаг ошибки.

Флаги

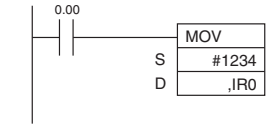
Название	Обозначение	Действие
Флаг ошибки	ER	Включен, если в S указан адрес памяти ПЛК, не существующий в ПЛК серии CV. Включен, если S не содержит константу или регистр указателя. Выключен во всех остальных случаях.

Пример

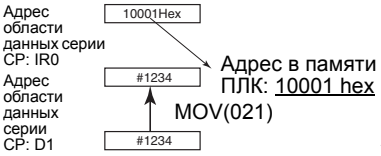
Преобразование программы серии CP, использующей косвенную адресацию через регистры указателей

1. В данном примере использования команды TOCV(285) в операнде S указывается регистр указателя. Адрес памяти ПЛК серии CP, содержащийся в этом регистре, преобразуется в эквивалентный адрес ПЛК серии CV.
2. Полученный адрес ПЛК серии CV передается по указанному адресу области данных.
3. Адрес памяти ПЛК серии CV можно использовать в программе серии CV.

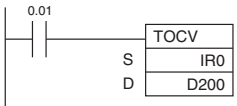
• Программа серии CP (программа использует косвенную адресацию через регистры указателей)



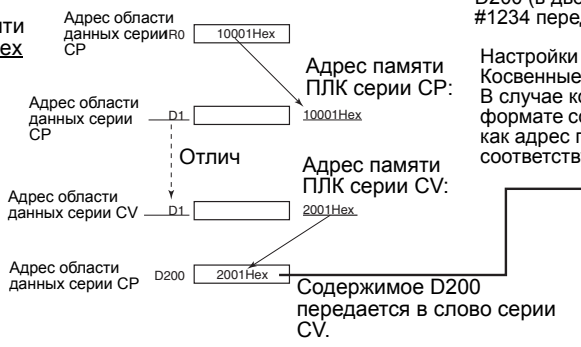
В данном случае IR0 содержит 10001 hex. Адресу 10001 hex памяти ПЛК соответствует адрес области данных D1, поэтому значение #1234 передается в D1.



• Программа серии CP



В данном случае IR0 содержит 10001 hex. Так как адресу 10001 hex памяти ПЛК серии CP соответствует адрес области данных D1, команда TOCV(285) записывает в слово D200 значение 2001 hex (адрес памяти ПЛК серии CV, соответствующий адресу D1).



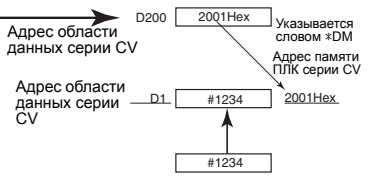
• Программа серии CV



Содержимое D200 передается в слово серии CV.

В ПЛК серии CV выходной операнд команды MOV(021) указан косвенно с помощью слова D200 (в двоичном формате), поэтому значение #1234 передается в D1.

Настройки ПЛК
Косвенные адреса DM:
В случае косвенной адресации в двоичном формате содержимое слова DM воспринимается как адрес памяти ПЛК и указывает соответствующий адрес памяти ввода/вывода.



3-30 Команды для создания программных блоков

В данном разделе описаны программные блоки и команды, предназначенные для их создания.

Команда	Мнемоническое обозначение	Код функции	Стр.
НАЧАЛО ПРОГРАММНОГО БЛОКА	BPRG	096	1072
КОНЕЦ ПРОГРАММНОГО БЛОКА	BEND	801	1072
ПРИОСТАНОВКА ПРОГРАММНОГО БЛОКА	BPPS	811	1075
ПЕРЕЗАПУСК ПРОГРАММНОГО БЛОКА	BPRS	812	1075
ВЫХОД ИЗ БЛОКА ПО УСЛОВИЮ (НЕ)	EXIT (NOT)	806	1081
ЕСЛИ (НЕ)	IF (NOT)	802	1077
ИНАЧЕ	ELSE	803	1077
ЕСЛИ КОНЕЦ	IEND	804	1077
ОДИН ЦИКЛ И ОЖИДАНИЕ (НЕ)	WAIT (NOT)	805	1084
ТАЙМЕР ОЖИДАНИЯ	TIMW (BCD)	813	1088
	TIMWX (двоичн.)	816	
СЧЕТЧИК ОЖИДАНИЯ	CNTW (BCD)	814	1092
	CNTWX (двоичн.)	818	
СКОРОСТНОЙ ТАЙМЕР ОЖИДАНИЯ	TMHW (BCD)	817	1096
	TMHWX (двоичн.)	815	
ЦИКЛ	LOOP	809	1099
КОНЕЦ ЦИКЛА (НЕ)	LEND (NOT)	810	1099

3-30-1 Введение

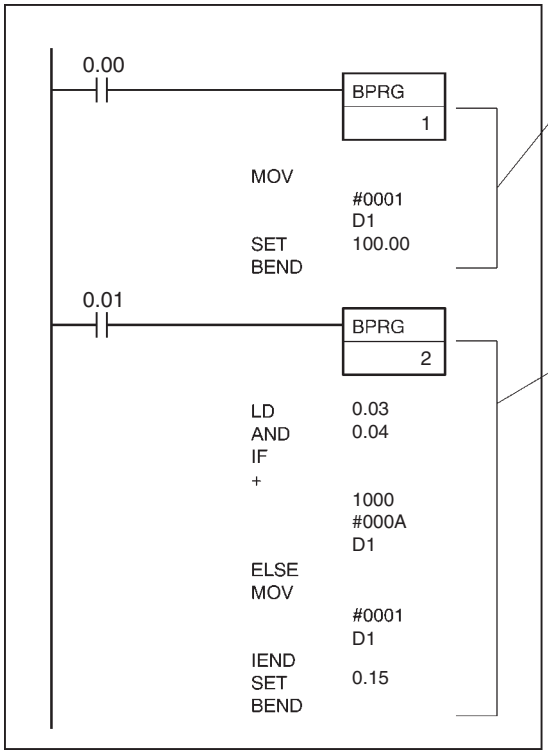
Программные блоки

В программе пользователя в целом (т. е. во всех задачах) может быть использовано до 128 программных блоков. Выполнением каждого программного блока управляет одно единственное условие выполнения. При включенном условии выполнения команды BPRG(096) остальные команды, заключенные между командами BPRG(096) и BEND(801), выполняются без каких-либо условий. Ни одна из команд программного блока, за исключением BPRG(096), не зависит от условия выполнения. Таким образом, все команды, которые должны иметь одно общее условие выполнения, могут быть сгруппированы в один программный блок.

Каждый блок запускается всего одним условием выполнения в лестничной диаграмме, при этом все команды внутри блока записываются в мнемонической форме. Таким образом, программный блок представляет собой комбинацию РКС-символов и мнемонических обозначений команд.

Применение программных блоков упрощает программирование таких операций, как ветвление по условиям или последовательное выполнение шагов, которые довольно сложно программировать на языке релейно-контактных схем.

Ниже представлены примеры двух программных блоков.



Область программного блока № 1

Когда бит CIO 0.00 = 1, выполняется содержимое программного блока 1. Выполняются (без условий) команды MOV(021) и SET, на этом программа блока завершается.

Область программного блока № 2

Когда бит CIO 0.01 = 1, выполняется содержимое программного блока 2. Если биты CIO 0.03 и CIO 0.04 оба включены, выполняется двоичное сложение (CIO 1000 + #000A –> D1). Если хотя бы один или оба этих бита выключены, в D1 записывается #0001. Затем устанавливается (без условий) бит CIO 0.15, на этом программа блока завершается.

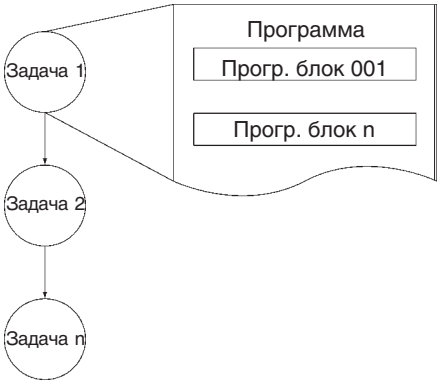
Задачи и программные блоки

Программные блоки могут располагаться внутри задач. Так же как задачи используются для того, чтобы разбить объемную программу пользователя на менее крупные программные модули, так и программные блоки служат для разбиения задач на еще меньшие модули, каждым из которых может управлять всего один элемент лестничной диаграммы (условие выполнения).

Аналогично задачам, программные блоки, которые не выполняются (блоки, у которых выключено условие выполнения), не требуют времени на выполнение, способствуя, тем самым, сокращению длительности цикла (по принципу работы близко к командам перехода). Наконец, так же как и задачи, блоки могут приостанавливаться или перезапускаться из других программных блоков.

Есть между задачами и программными блоками и отличия. Первое отличие состоит в том, что внутри программных блоков не используются входные условия, если только они не вводятся специально с помощью команд IF(802), WAIT(805), EXIT(806), IEND(810) или других команд. Второе отличие в том, что программные блоки не поддерживают некоторые виды команд, например, однократные команды, выполняемые по переключению условия выполнения.

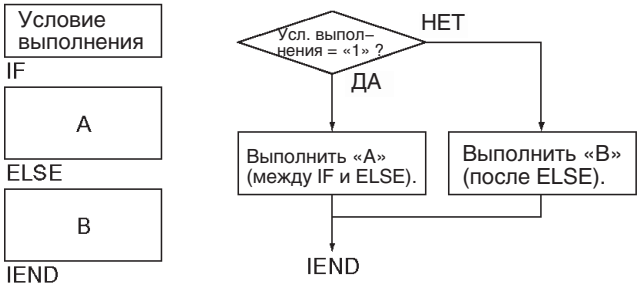
Программные блоки можно использовать как в циклических задачах, так и в задачах, выполняемых по прерыванию. Каждый номер программного блока (от 0 до 127) можно использовать только один раз и нельзя использовать повторно (даже в другой задаче).



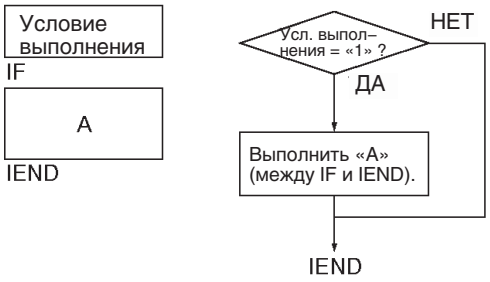
Применение команд программных блоков

Для создания условий выполнения внутри программных блоков используются команды IF(802), ELSE(803) и IEND(810) (вместе с битами).

Например, если, в зависимости от условий, должна выполняться либо команда «A» либо команда «B», используется конструкция IF A ELSE B IEND.



Если, в зависимости от условий, должна выполняться только команда «A», используется конструкция IF A IEND.



Если выполнение требуется задержать до включения условия выполнения или бита (например, при последовательном выполнении шагов), используется команда WAIT(805).

Если требуется задержать выполнение на определенный промежуток времени (например, при последовательном выполнении шагов с выдержкой времени), используется команда TIMW(813), TIMX(816), TMHW(815) или TMHWX(817).

Если выполнение требуется задержать до достижения счетчиком указанного количества (например, при последовательном выполнении шагов с подсчетом количества), используется команда CNTW(814) и CNTWX(818).

Если выполнение некоторой части программного блока должно повторяться, пока не будет соблюдено некоторое условие, используются команды LOOP(809) и LEND(810).

Если выполнение программного блока должно завершаться досрочно (т. е. не доходя до конца блока) при наступлении определенного события, используется команда EXIT(806).

Если некоторый выполняемый программный блок должен быть приостановлен или перезапущен из другого программного блока, используются команды BPPS(811) и BPRS(812).

Команды для создания условий выполнения внутри программных блоков

В следующей таблице перечислены команды, позволяющие использовать внешние условия выполнения внутри программных блоков.

Тип команды	Имя команды	Мнемоническое обозначение
Команды программирования блоков	ЕСЛИ (НЕ)	IF(802) (NOT)
	ОДИН ЦИКЛ И ОЖИДАНИЕ (НЕ)	WAIT(805) (NOT)
	ВЫХОД	EXIT(806) NOT
	КОНЕЦ ЦИКЛА	LEND(810) NOT
Команды лестничной диаграммы	УСЛОВНЫЙ ПЕРЕХОД	CJP(510)
	УСЛОВНЫЙ ПЕРЕХОД НЕ	CJPN(511)

Команды с ограничениями на применение внутри программных блоков

В следующей таблице перечислены команды, которые могут использоваться только для создания условий выполнения для команд IF(802), WAIT(805), EXIT(806), LEND(810), CJP(510 или CJPN(511) и не могут использоваться самостоятельно. Если эти команды используются сами по себе или в сочетании с другими командами, они могут работать непредсказуемым образом.

Мнемоническое обозначение	Название
LD/LD NOT	ЗАГРУЗКА/ЗАГРУЗКА НЕ
AND/AND NOT	И/И НЕ
OR/OR NOT	ИЛИ/ИЛИ НЕ

Мнемоническое обозначение	Название
UP/DOWN	УСЛОВИЕ ВКЛ/ВЫКЛ
>, <,=, >=, <=, <> (S) (L)	Символьные команды сравнения (не правосторонние)
LD TST/TST NOT	ЗАГРУЗКА ПРОВЕРКА БИТА (НЕ)
AND TST/TST NOT	И ПРОВЕРКА БИТА (НЕ)
OR TST/TST NOT	ИЛИ ПРОВЕРКА БИТА (НЕ)
>$, <$,=$, >=$, <=$, <>$	Команды сравнения текстовых строк

Пример правильной программы

```
LD      0.00
AND     0.01
TST     D0      #0010
IF
```

Используется как условие выполнения для IF.

Пример неправильной программы

```
LD      0.00
AND     0.01
TST     D0      #0010
MOV     #0000   0010
```

Не может использоваться как условие выполнения для MOV(021).

Команды, не поддерживаемые в программных блоках

В следующей таблице перечислены команды, которые невозможно использовать внутри программных блоков.

Группа команд	Мнемоническое обозначение	Название	Альтернатива
Выходные битовые команды	OUT	ВЫВОД	Используйте SET и RSET.
	OUT NOT	ВЫВОД НЕ	
	DIFU(013)	ПОЛОЖИТЕЛЬНЫЙ ФРОНТ	Нет
	DIFD(014)	ОТРИЦАТЕЛЬНЫЙ ФРОНТ	Нет
	KEEP(011)	ЗАЩЕЛКА	Нет
Команды управления последовательностью выполнения	FOR(512) и NEXT(513)	ЦИКЛЫ FOR-NEXT	Используйте LOOP(809) и LEND(810) (NOT).
	BREAK(514)	ВЫХОД ИЗ ЦИКЛА	
	IL(002) и ILC(003)	БЛОКИРОВКА и ОТМЕНА БЛОКИРОВКИ	Поделите программу блока на менее крупные блоки.
	JMP(004) 0 и JME(005) 0	МНОЖЕСТВЕННЫЙ ПЕРЕХОД и КОНЕЦ ПЕРЕХОДА	Используйте JMP(004) и JME(005) (но переход будет безусловным).
	END(001)	КОНЕЦ	Используйте BEND(801).
Команды управления таймерами и счетчиками	TIM	ТАЙМЕР	Используйте TIMW(813), TIMWX(816), TMHW(815), TMHWX(817), CNTW(814) и CNTWX(818). Другие команды программного блока не будет выполняться, пока не будет достигнута уставка таймера или счетчика.
	TIMH(015)	СКОРОСТНОЙ ТАЙМЕР	
	TMHH(540)	1 мс ТАЙМЕР	
	TTIM(087)	НАКАПЛИВАЮЩИЙ ТАЙМЕР	
	TIML(542)	ДОЛГОСРОЧНЫЙ ТАЙМЕР	
	MTIM(543)	МНОГОВЫХОДНОЙ ТАЙМЕР	
	CNT	СЧЕТЧИК	
	CNTR(012)	РЕВЕРСИВНЫЙ СЧЕТЧИК	
Команды управления подпрограммами	SBN(092) и RET(093)	ВХОД В ПОДПРОГРАММУ и ВЫХОД ИЗ ПОДПРОГРАММЫ	Нет
Команды сдвига	SFT(010)	РЕГИСТР СДВИГА	Используйте другие команды сдвига.

Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

Группа команд	Мнемоническое обозначение	Название	Альтернатива
Команды для пошагового выполнения	STEP(008) и SNXT(009)	ШАГ и СЛЕДУЮЩИЙ ШАГ	Используйте WAIT(805).
Команды управления данными	PID(190)	ПИД-РЕГУЛЯТОР	Нет
Команды для диагностики	FPD(269)	ОБНАРУЖЕНИЕ НЕИСПРАВНОГО КАНАЛА	Нет
Однократное выполнение команд по переключению условия выполнения	Команды с префиксом @	Команды, выполняемые по положительному фронту	Нет
	Команды с префиксом %	Команды, выполняемые по отрицательному фронту	Нет

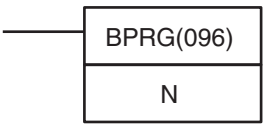
3-30-2 НАЧАЛО/КОНЕЦ ПРОГРАММНОГО БЛОКА: BPRG(096)/BEND(801)

Назначение

Обозначение начала и конца области, содержащей команды программного блока. Для каждой команды BPRG(096) должна быть предусмотрена соответствующая команда BEND(801).

Символы РКС

НАЧАЛО ПРОГРАММНОГО БЛОКА



N: Номер программного блока

КОНЕЦ ПРОГРАММНОГО БЛОКА

BEND(801)

Варианты выполнения

BPRG(096)

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	BPRG(096)
	Однократное выполнение по положительному фронту	Не предусмотрено
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

BEND(801)

Варианты выполнения	**Всегда выполняется в программном блоке**

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
(См. примечание).	OK	OK	OK

Примечание. Допускается только одна команда BPRG(096) в начале каждого программного блока.

Операнды

N: Номер программного блока

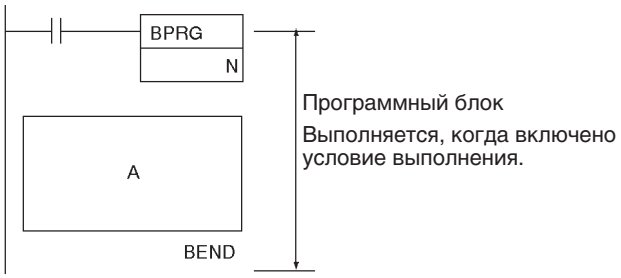
Номер программного блока должен быть указан в диапазоне от 0 до 127 (десятичн).

Характеристики операндов (BPRG(096))

Область	N
Область CIO	---
Рабочая область	---
Область битов хранения	---
Область вспомогательных битов	---
Область таймеров	---
Область счетчиков	---
Область DM	---
Косвенные адреса DM в двоичном формате	---
Косвенные адреса DM в двоично-десятичном формате (BCD)	---
Постоянные	0...127 (десятичн.)
Регистры данных	---
Регистры указателей	---
Косвенная адресация с помощью регистров указателей	---

Описание

Команда BPRG(096) выполняет программный блок, номер которого указан в N. (Фактически, выполняется блок команд, следующий непосредственно за командой BPRG(096) и заканчивающийся командой BEND(801)). Все команды, расположенные между BPRG(096) и BEND(801), выполняются безусловно (должно быть включено лишь условие выполнения блока, общее для всех команд).



Если условие выполнения команды BPRG(096) выключено, программный блок не выполняется и команды программного блока не требуют времени на выполнение.

Даже если условие выполнения команды BPRG(096) включено, выполнение программного блока можно остановить, выполнив команду BPPS(811) из другого программного блока.

Флаги

BPRG(096)

Название	Обозначение	Действие
Флаг ошибки	ER	Включен, если команда BPRG(096) уже выполняется. Включен, если N не находится в диапазоне 0...127. Включен, если какой-либо номер программного блока использован больше одного раза. Выключен во всех остальных случаях.

BEND(801)

Название	Обозначение	Действие
Флаг ошибки	ER	Включен, если программный блок в данный момент не выполняется. Выключен во всех остальных случаях.

Меры предосторожности

Каждый номер программного блока может использоваться только один раз в пределах всей программы пользователя.

Вложение программных блоков не допускается.



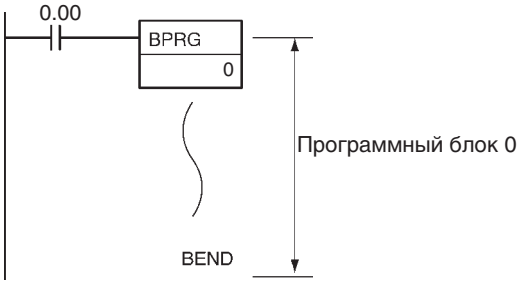
Если программный блок находится в блокируемом сегменте программы и условие выполнения команды IL(002) выключено, программный блок не выполняется.

Соответствующие друг другу команды BPRG(096) и BEND(801) должны находиться в пределах одной задачи.

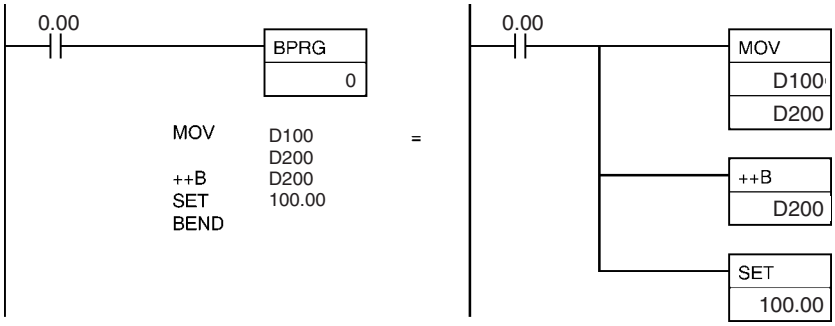
Любое из следующих нарушений приводит к возникновению ошибки и установке флага ошибки: команда BPRG(096) использована внутри программного блока (т. е. не вначале), команда BEND(801) использована вне программного блока, значение N не находится в диапазоне #0000...#007F (двоичн.), программный блок не содержит программу, один и тот же номер программного блока использован несколько раз.

Примеры

Включение бита CIO 0.00 в следующем примере приведет к выполнению программного блока 0. При выключенном бите CIO 0.00 программный блок выполняться не будет.



Оба сегмента программы, показанные ниже, выполняют команды MOV(021), ++B(594) и SET, используя одно и то же условие выполнения (т. е. включение бита CIO 0.00).



3-30-3 ПРИОСТАНОВКА/ПЕРЕЗАПУСК ПРОГРАММНОГО БЛОКА: BPPS(811)/BPRS(812)

Назначение

Приостановка выполнения и возобновление выполнения указанного программного блока из другого программного блока.

Символ РКС

BPPS(811)	N	**N**: Номер программного блока
BPRS(812)	N	

Варианты выполнения

Варианты выполнения	Всегда выполняется в программном блоке

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Примечание. Команды BPRG(096) и BPRS(812) обязательно должны использоваться в пределах программных блоков, даже если последние создаются в подпрограммах и задачах обработки прерываний.

Операнды

N: Номер программного блока

Номер программного блока должен быть указан в диапазоне от 0 до 127 (десятичн).

Характеристики операндов

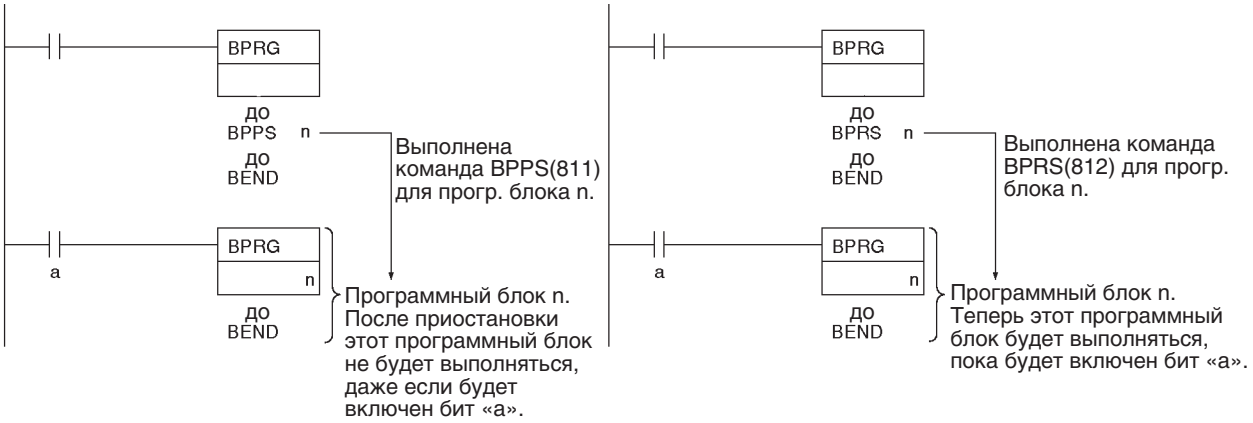
Область	N
Область CIO	---
Рабочая область	---
Область битов хранения	---
Область вспомогательных битов	---
Область таймеров	---
Область счетчиков	---
Область DM	---
Косвенные адреса DM в двоичном формате	---

Область	N
Косвенные адреса DM в двоично-десятичном формате (BCD)	---
Постоянные	0...127 (десятичн.)
Регистры данных	---
Регистры указателей	---
Косвенная адресация с помощью регистров указателей	---

Описание

Команда BPPS(811) используется внутри одного программного блока с целью приостановки выполнения другого программного блока, номер которого указан в N. Команда BPPS(811) приостанавливает выполнение программного блока, даже если его условие выполнения в данный момент включено. Выполнение программного блока не возобновится, пока для него не будет выполнена команда BPRS(812).

Команда BPRS(812) возобновляет выполнение программного блока, номер которого указан в N. Возобновленный программный блок продолжает выполняться, пока остается включенным условие выполнения его команды BPRG(096).



Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	Включен, если команда BPPS(811) или BPRS(812) использована вне программного блока. Включен, если N не находится в диапазоне 0...127. Выключен во всех остальных случаях.

Меры предосторожности

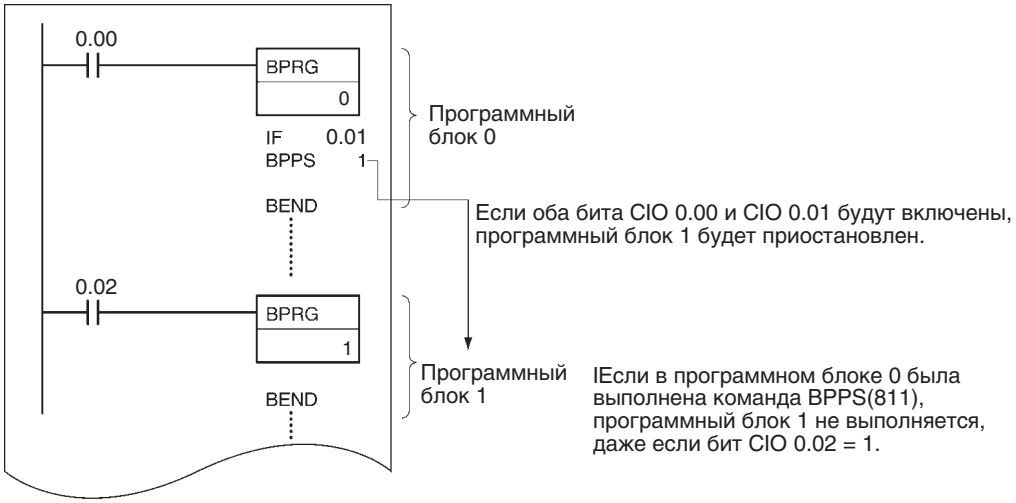
Если команда BPPS(811) или BPRS(812) будет применена вне программного блока или значение операнда N выйдет за диапазон #0000...#007F (двоичн.), произойдет ошибка и будет установлен флаг ошибки.

Командой BPPS(811) можно воспользоваться в том числе для приостановки программного блока, содержащего эту команду. После возобновления работы программного блока с помощью команды BPRS(812) из другого программного блока приостановленный программный блок продолжит выполняться с команды, следующей за командой BPPS(811).

Команды таймера (TIMW(813), TIMWX(816), TMHW(815) или TMHWX(817)) продолжают отсчет времени, даже если выполнение содержащего их программного блока приостановлено.

Примеры

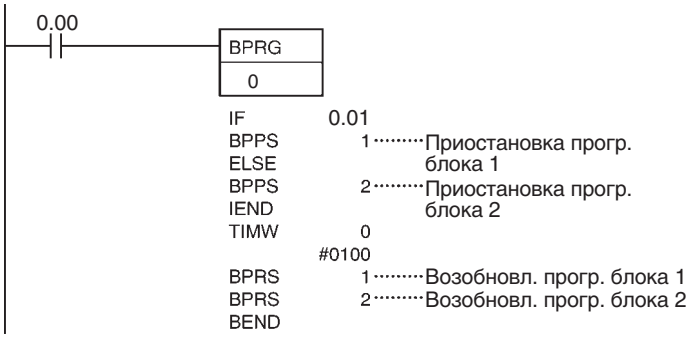
На следующем рисунке показан простой пример приостановки выполнения программного блока.



Примечание.

Если приостанавливаемый программный блок находится после команды BPPS(811), он не будет выполняться уже в текущем цикле. Если программный блок располагается до команды BPPS(811), его выполнение будет приостановлено, начиная со следующего цикла.

При включенном бите CIO 0.00 показанная ниже программа, в зависимости от состояния бита CIO 0.01, приостанавливает выполнение либо программного блока 1, либо программного блока 2. Выполнение приостановленного программного блока возобновляется спустя 10 секунд.



Адрес	Команда	Операнды
000000	LD	0.00
000001	BPRG(096)	00
000002	IF(802)	0.01
000003	BPPS(811)	01
000004	ELSE(803)	
000005	BPPS(811)	02
000006	IEND(804)	
000007	TIMW(803)	0
		# 0100
000008	BPRS(812)	1
000009	BPRS(812)	2
000010	BEND(801)	

3-30-4 Ветвление: IF(802), ELSE(803) и IEND(804)

Назначение

Разветвление программы программного блока в зависимости от условия выполнения или состояния битового операнда.

Символ РКС

IF(802) B

IF(802)

IF(802) NOT B

ELSE(803)

IEND(804)

B: Битовый операнд

Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

Варианты выполнения

Варианты выполнения	Всегда выполняется в программном блоке

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Примечание. Команды IF(802), ELSE(803) и IEND(804) должны находиться непосредственно в программных блоках, даже если последние используются в подпрограммах и задачах обработки прерываний.

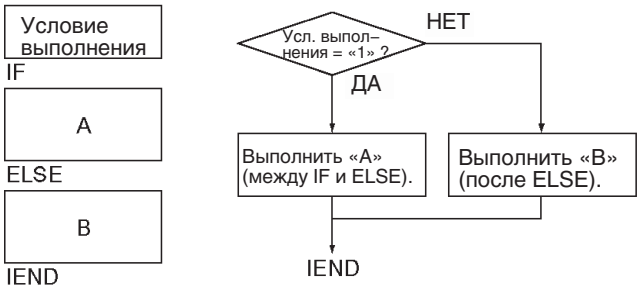
Характеристики операндов

Область	B
Область CIO	CIO 0.00...CIO 6143.15
Рабочая область	W0.00...W511.15
Область битов хранения	H0.00...H511.15
Область вспомогательных битов	A0.00...A447.15 A448.00...A959.15
Область таймеров	T0000...T4095
Область счетчиков	C0000...C4095
Флаги задач	TK00...TK31
Флаги условий	ER, CY, >, =, <, N, OF, UF, >=, <>, <=, ON, OFF, AER
Тактовые импульсы	0,02 с / 0,1 с / 0,2 с / 1 с / 1 мин
Область DM	---
Косвенные адреса DM в двоичном формате	---
Косвенные адреса DM в двоично-десятичном формате (BCD)	---
Постоянные	---
Регистры данных	---
Регистры указателей	---
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15

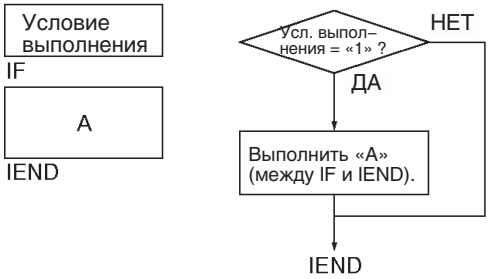
Описание

Применение без операнда в команде IF(802)

Если битовый операнд в команде IF(802) не указывается, для команды IF(802) должно быть предварительно сформировано внешнее условие выполнения, начинающееся с команды LD. Если условие выполнения включено, выполняются команды, расположенные между IF(802) и ELSE(803). Если условие выполнения выключено, выполняются команды, расположенные между ELSE(803) и IEND(804.

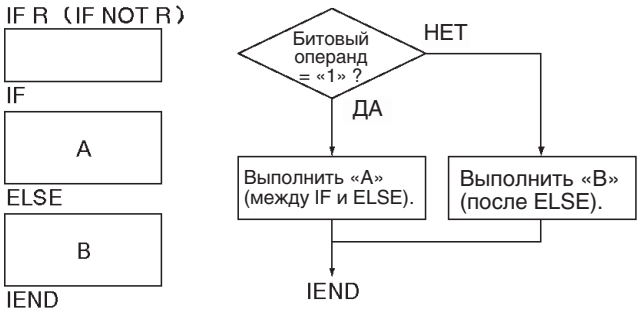


Если команда ELSE(803) отсутствует и условие выполнения включено, выполняются команды, расположенные между IF(802) и IEND(804). Если условие выполнения выключено, выполняются только команды, расположенные после IEND(804).

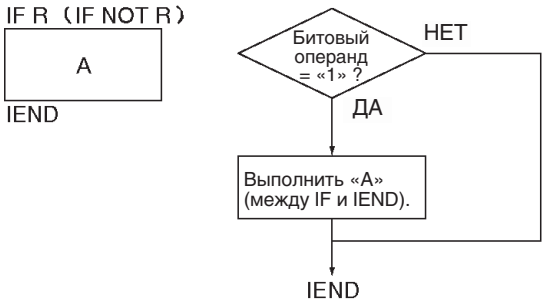


Применение с операндом в команде IF(802) или IF NOT(802)

В команде IF(802) или IF NOT(802) может быть указан битовый операнд (B). Если битовый операнд включен, выполняются команды, расположенные между IF(802) и ELSE(803). Если битовый операнд выключен, выполняются команды, расположенные между ELSE(803) и IEND(804). В случае команды IF NOT(802) соблюдается обратная закономерность: при включенном битовом операнде выполняются команды между IF(802) и ELSE(803), а при выключенном битовом операнде выполняются команды между ELSE(803) и IEND(804).



Если команда ELSE(803) отсутствует и битовый операнд включен, выполняются команды, расположенные между IF(802) и IEND(804). Если битовый операнд выключен, выполняются только команды, расположенные после IEND(804). Команда IF NOT(802) работает точно так же, но с инверсией состояния битового операнда.



Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	Включен, если команды ветвления использованы вне программного блока. Включен в случае вложения более 254 ветвей. Выключен во всех остальных случаях.

Меры предосторожности

В общем случае команды внутри программных блоков выполняются безусловно (т. е. без условия выполнения). Условное выполнение (выполнение в зависимости от условия выполнения или состояния битового операнда) можно реализовать с помощью команд ветвления.

Для выбора между ветвями A и B следует использовать конструкцию IF A ELSE B IEND.

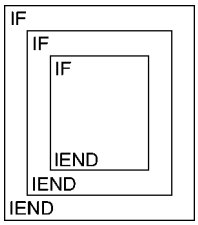
Для выбора между ветвью A и отсутствием действий следует использовать конструкцию IF A IEND.

Допускается вложение ветвей (до 253 уровней).

Если команды ветвления будут использованы вне программного блока или будет вложено более 254 ветвей, произойдет ошибка и будет установлен флаг ошибки.

Вложение ветвей

Исходная ветвь (ветвь верхнего уровня) может содержать до 253 вложенных ветвей.



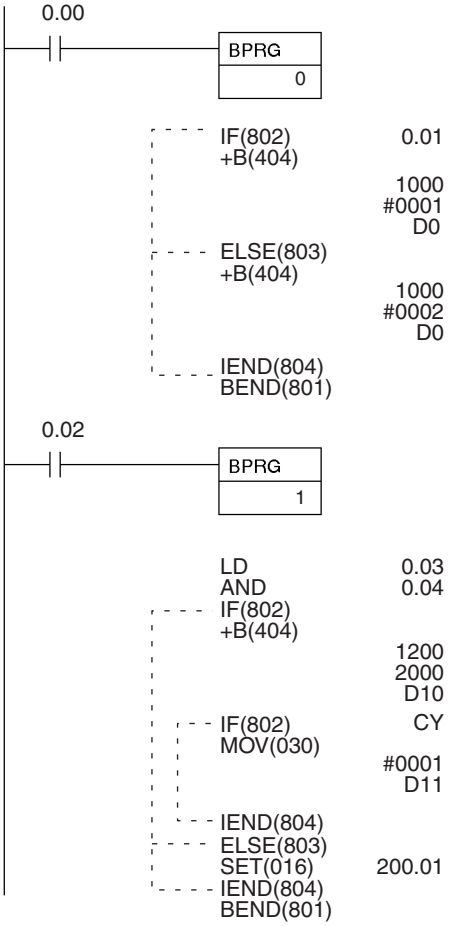
Примеры

Ниже представлены примеры двух разных программных блоков, выполнением которых управляют биты CIO 0.00 и CIO 0.02.

В первом блоке, в зависимости от состояния бита CIO 0.01, выполняется одна из двух операций сложения. Этот блок выполняется, когда включен бит CIO 0.00. Если бит CIO 0.01 включен, к содержимому слова CIO 1000 добавляется 0001. Если бит CIO 0.01 выключен, к содержимому слова CIO 1000 добавляется 0002. И в том и в другом случае результат записывается в D0.

Второй блок выполняется, когда включен бит CIO 0.02. В этом блоке применено двухуровневое вложение. Если биты CIO 0.03 и CIO 0.04 оба включены, складывается содержимое слов CIO 1200 и CIO 2000 и результат помещается в D10, после чего, в зависимости от состояния флага CY, значение 0001 записывается или не записывается в D11. Если

любой из битов CIO 0.03 или CIO 0.04 выключен, операция сложения полностью пропускается и включается бит CIO 200.01.



Адрес	Команда	Операнды
000000	LD	0.00
000001	BPRG(096)	0
000002	IF(802)	0.01
000003	+B(404)	
		1000
		#0001
		D0
000004	ELSE(803)	
000005	+B(404)	
		1000
		#0002
		D0
000006	IEND(804)	
000007	BEND(801)	
000008	LD	0.02
000009	BPRG(096)	1
000010	LD	0.03
000011	AND	0.04
000012	IF(802)	
000013	+B(404)	
		1200
		2000
		D10
000014	IF(802)	CY
000015	MOV(030)	
		#0001
		D11
000016	IEND(804)	
000017	ELSE(803)	
000018	SET(016)	200.01
000019	IEND(804)	
000020	BEND(801)	

3-30-5 ВЫХОД ИЗ БЛОКА ПО УСЛОВИЮ (НЕ): EXIT (NOT)(806)

Назначение

Выход из программного блока (не выполняется ни одна оставшаяся команда программного блока вплоть до BEND(801)) в соответствии с состоянием битового операнда или условием выполнения. Команда EXIT(806) без битового операнда производит выход из программы блока при включенном условии выполнения. Команда EXIT(806) с битовым операндом производит выход из программы блока при включенном бите. Команда EXIT NOT(806) обязательно должна иметь битовый операнд и производит выход из программы, если бит выключен.

Символ РКС

EXIT(806)

EXIT(806) B

EXIT NOT(806) B

B: Битовый операнд

Варианты выполнения

Варианты выполнения	Всегда выполняется в программном блоке	EXIT(806) EXIT(806) B EXIT NOT(806) B

Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограм-мы	Задачи обработки прерываний
OK	OK	OK	OK

Примечание. Команды EXIT(806) и EXIT NOT(806) обязательно должны использоваться в пределах программных блоков, даже если последние создаются в подпрограммах и задачах обработки прерываний.

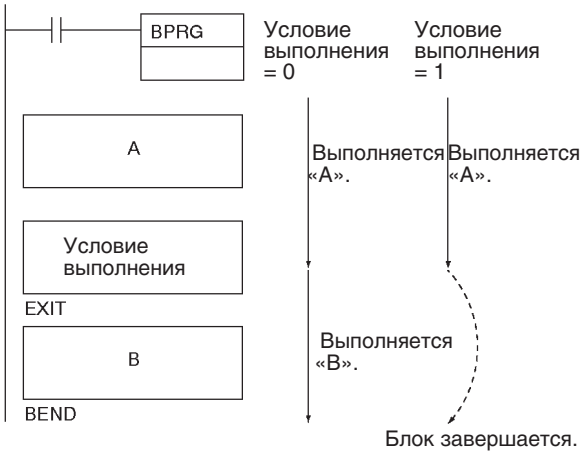
Характеристики операндов

Область	B
Область CIO	CIO 0.00...CIO 6143.15
Рабочая область	W0.00...W511.15
Область битов хранения	H0.00...H511.15
Область вспомогательных битов	A0.00...A959.15
Область таймеров	T0000...T4095
Область счетчиков	C0000...C4095
Флаги задач	TK00...TK31
Флаги условий	ER, CY, >, =, <, N, OF, UF, >=, <>, <=, ON, OFF, AER
Тактовые импульсы	0,02 с / 0,1 с / 0,2 с / 1 с / 1 мин
Область DM	---
Косвенные адреса DM в двоичном формате	---
Косвенные адреса DM в двоично-десятичном формате (BCD)	---
Постоянные	---
Регистры данных	---
Регистры указателей	---
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15

Описание

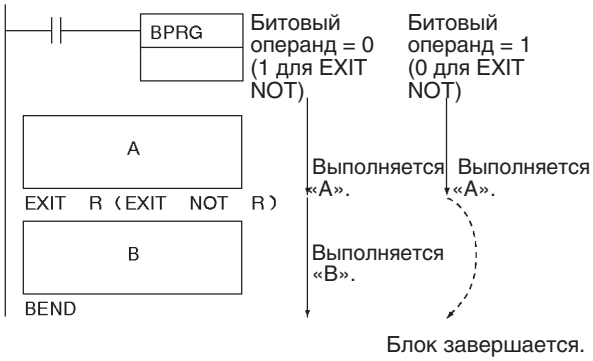
Применение без операнда

Команда EXIT(806) может быть выполнена без операнда. В этом случае должно быть предварительно сформировано условие выполнения, начинающееся с команды LD. Если условие выполнения выключено, оставшиеся команды программного блока (команды после команды EXIT(806)) выполняются в обычном режиме. Если условие включено, оставшиеся команды программного блока (команды между EXIT(806) и BEND(801)) не выполняются.



Применение с операндом

Если битовый операнд B команды EXIT(806) выключен, оставшаяся часть программного блока выполняется в обычном порядке. Если битовый операнд команды EXIT(806) включен, оставшаяся часть команд программного блока (вплоть до команды BEND(801)) не выполняется. В случае команды EXIT NOT(806) оставшаяся часть программного блока выполняется, если битовый операнд включен, и пропускается, если битовый операнд выключен.



Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	Включен, если команды EXIT(806) или EXIT NOT(806) применены вне программного блока. Выключен во всех остальных случаях.

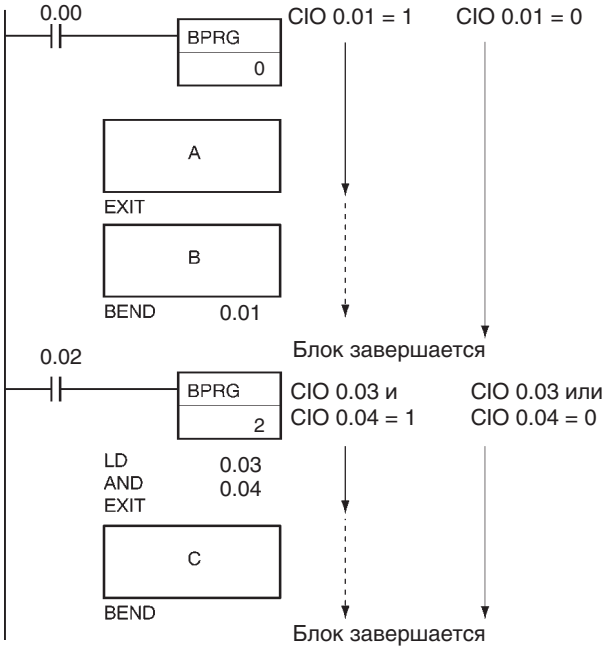
Меры предосторожности

Если команды EXIT(806) или EXIT NOT(806) будут использованы за пределами программного блока, произойдет ошибка и будет установлен флаг ошибки.

Примеры

Если бит CIO 0.00 включен, выполняется программный блок. Если включен бит CIO 0.01, выполняется сегмент A, а сегмент B пропускается и программа переходит непосредственно к команде BEND(801). Сегмент B программы будет пропускаться до тех пор, пока не выключится бит CIO 0.01.

Хотя с точки зрения программирования команды EXIT (NOT)(806) и IF-IEND похожи, время выполнения команды EXIT (NOT)(806) обычно меньше, поскольку команды, расположенные после EXIT (NOT)(806) вплоть до конца программного блока, не выполняются вообще.



Адрес	Команда	Операнды
000200	LD	0.00
000201	BPRG	0
:	A	:
000210	EXIT	0.01
:	B	:
000220	BEND	---
000221	LD	0.02
000222	BPRG	2
000223	LD	0.03
000224	AND	0.04
000225	EXIT	---
:	C	:
000230	BEND	---

3-30-6 ОДИН ЦИКЛ И ОЖИДАНИЕ (НЕ): WAIT(805)/WAIT(805) NOT

Назначение

Приостановка выполнения оставшейся части программного блока до включения условия выполнения или включения либо выключения битового операнда.

Символ РКС

WAIT(805)

WAIT(805) B

WAIT(805) NOT B

B: Битовый операнд

Варианты выполнения

Варианты выполнения	Всегда выполняется в программном блоке

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Примечание. Команды WAIT(805)/WAIT(805) NOT обязательно должны использоваться в пределах программных блоков, даже если последние создаются в подпрограммах и задачах обработки прерываний.

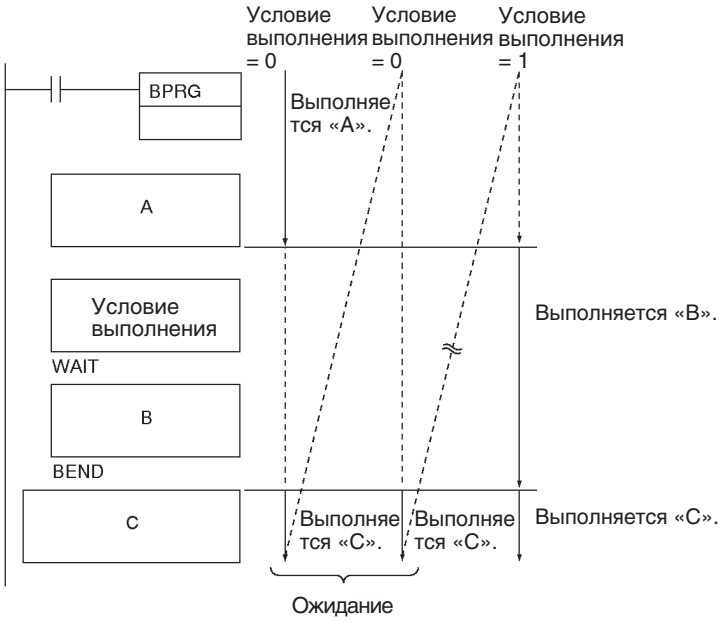
Характеристики операндов

Область	B
Область CIO	CIO 0.00...CIO 6143.15
Рабочая область	W0.00...W511.15
Область битов хранения	H0.00...H511.15
Область вспомогательных битов	A0.00...A447.15 A448.00...A959.15
Область таймеров	T0000...T4095
Область счетчиков	C0000...C4095
Флаги задач	TK00...TK31
Флаги условий	ER, CY, >, =, <, N, OF, UF, >=, <>, <=ON, OFF, AER
Тактовые импульсы	0,02 с / 0,1 с / 0,2 с / 1 с / 1 мин
Область DM	---
Косвенные адреса DM в двоичном формате	---
Косвенные адреса DM в двоично-десятичном формате (BCD)	---
Постоянные	---
Регистры данных	---
Регистры указателей	---
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15

Описание

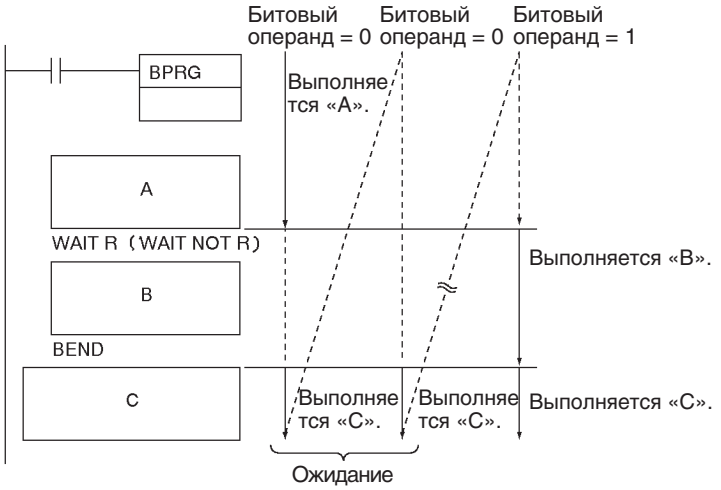
Применение без операнда

Если команда WAIT(805)/WAIT NOT (805) используется без битового операнда, для нее должно быть сформировано внешнее условие выполнения, начинающееся с команды LD. Если условие выполнения команды WAIT(805) выключено, оставшиеся команды программного блока пропускаются. Начиная со следующего цикла, ни одна из команд программного блока не будет выполняться, кроме команд условия выполнения WAIT(805). После включения условия выполнения будут выполнены команды, расположенные после команды WAIT(805), вплоть до конца программы.



Применение с операндом

Для команды WAIT(805) или WAIT NOT(805) может быть указан битовый операнд B. Если битовый операнд выключен (включен для WAIT NOT(805)), остальные команды программного блока пропускаются. Начиная со следующего цикла, ни одна из команд программного блока не будет выполняться, за исключением команд условия выполнения самой команды WAIT(805) или WAIT(805) NOT. После включения условия выполнения (выключения для WAIT NOT(805)) будут выполнены команды, расположенные после WAIT(805) или WAIT NOT(805), вплоть до конца программы.



Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	Включен, если команда WAIT(805) или WAIT(805) NOT применена вне программного блока. Выключен во всех остальных случаях.

Меры предосторожности

Команды WAIT(805) и WAIT(805) NOT могут использоваться для пошагового выполнения операций внутри программных блоков.

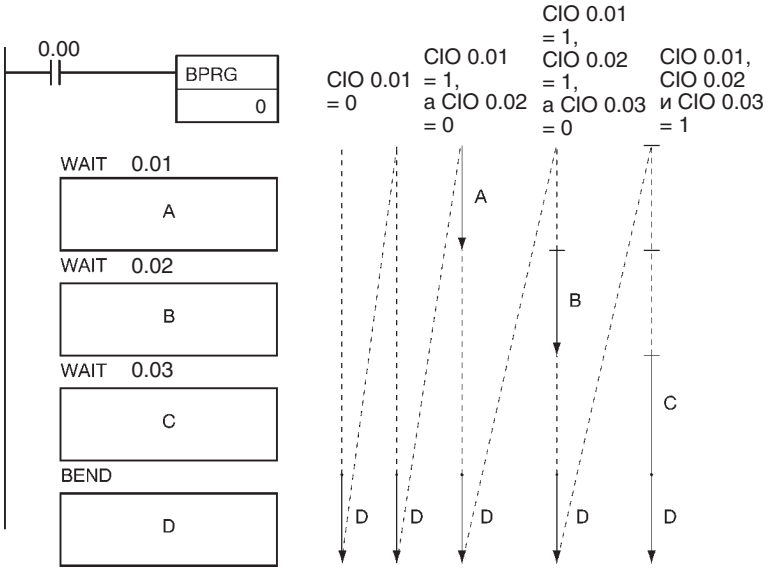
Если команда WAIT(805) или WAIT(805) NOT будет применена вне программного блока, произойдет ошибка и будет установлен флаг ошибки.

Примечание. Адрес программы команды WAIT с указанным операндом и адрес программы первой команды, формирующей условие выполнения команды WAIT без операнда, записываются в память, чтобы выполнение могло продолжиться в соответствии с состоянием условия выполнения или битового операнда. Однако при выполнении онлайн-редактирования из CX-Programmer состояние команды WAIT будет сброшено, и программный блок начнет выполняться с самого начала.

Примеры Если в приведенном ниже примере бит CIO 0.00 включен, выполняется программный блок 00. Алгоритм выполнения формулируется следующим образом.

1,2,3...

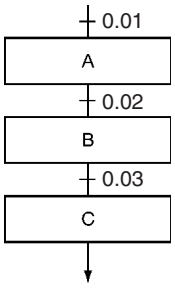
1. Если бит CIO 0.01 выключен, ни одна часть программного блока не будет выполняться, пока не будет включен бит CIO 0.01. После включения бита CIO 0.01 будет выполнен сегмент «A».
2. Если после выполнения сегмента «A» бит CIO 0.02 будет выключен, остальные части программного блока не будут выполняться, пока не будет включен бит CIO 0.02. После включения бита CIO 0.02 будет выполнен сегмент «B».
3. Если после выполнения сегмента «B» бит CIO 0.03 будет выключен, остальные части программного блока не будут выполняться до тех пор, пока не будет включен бит CIO 0.03. После включения бита CIO 0.03 будет выполнен сегмент «C», после чего весь процесс повторится с самого начала.



Следующая таблица отражает взаимосвязь между переключением состояний битовых операндов и выполнением частей программного блока.

Битовые операнды			Выполнение программы		
CIO 0.01	**CIO 0.02**	**CIO 0.03**	**Первый цикл, в котором бит CIO 0.00 = ВКЛ**	**Следующий цикл**	**Последующие циклы**
ВЫКЛ	Любое состояние	Любое состояние	Не выполняется ни одна часть.	Не выполняется ни одна часть, ожидается включение CIO 0.01.	После включения CIO 0.01 выполняется «A» и проверяется состояние CIO 0.02.
ВКЛ	ВЫКЛ	Любое состояние	Выполняется «A».	Ожидание CIO 0.02.	После включения CIO 0.02 выполняется «B» и проверяется состояние CIO 0.03.
ВКЛ	ВКЛ	ВЫКЛ	Выполняются «A» и «B».	Ожидание CIO 0,03.	После включения CIO 0.03 выполняется «C».
ВКЛ	ВКЛ	ВКЛ	Выполняются «A», «B» и «C».	Выполняются «A», «B» и «C».	

Как видно из следующего примера, команды WAIT(805) и WAIT(805) NOT могут использоваться для последовательного выполнения шагов программного блока.



3-30-7 ТАЙМЕР ОЖИДАНИЯ: TIMW(813) и TIMWX(816)

Назначение

Задержка выполнения оставшейся части программного блока до истечения указанного времени. После достижения таймером установленного значения выполнение продолжится со следующей команды после TIMW(813)/TIMWX(816).

Символ РКС

Форма представления текущего значения: двоично-десятичная (BCD)

TIMW(813)	N	**N**: Номер таймера
	SV	**SV**: Уставка

Форма представления текущего значения: двоичная

TIMWX(816)	N	**N**: Номер таймера
	SV	**SV**: Уставка

Варианты выполнения

Варианты выполнения	Всегда выполняется в программном блоке

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограм-мы	Задачи обработки прерываний
OK	OK	OK	Не допустимо.

Примечание. Команды TIMW(813)/TIMWX(816) обязательно должны использоваться в пределах программных блоков, даже если последние создаются в подпрограммах.

Операнды

N: Номер таймера

BCD: 0...4095 (десятичн.)
Двоичн.:0...4095 (десятичн.)

S: Уставка

BCD: #0000...#9999 (BCD)
Двоичн.:&0...&65535 (десятичн.)
#0000...#FFFF (hex)

Характеристики операндов

Область	N	S
Область CIO	---	CIO 0...CIO 6143
Рабочая область	---	W0...W511
Область битов хранения	---	H0...H511
Область вспомогательных битов	---	A0...A447 A448...A959
Область таймеров	0000...4095	T0000...T4095
Область счетчиков	---	C0000...C4095
Область DM	---	D0...D32767
Косвенные адреса DM в двоичном формате	---	@ D0...@ D32767
Косвенные адреса DM в двоично-десятичном формате (BCD)	---	*D0...*D32767
Постоянные	---	BCD: #0000...9999 (BCD) «&» использовать нельзя. Двоичный: &0...&65535 (десятичн.) #0000...#FFFF (hex)
Регистры данных	---	DR0...DR15
Регистры указателей	---	---
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

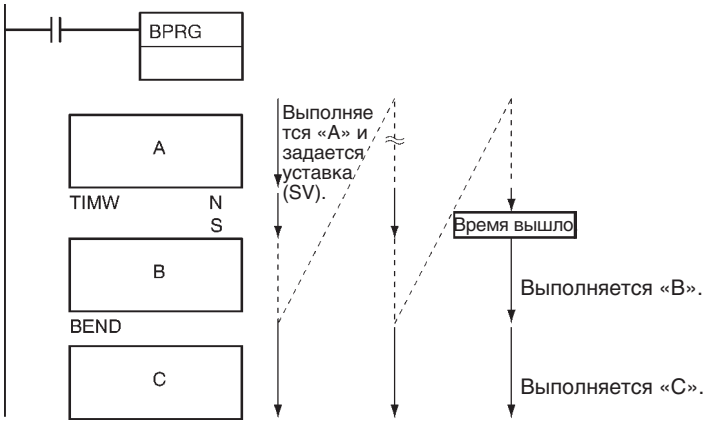
Описание

Команда TIMW(813)/TIMWX(816) задерживает выполнение следующих за ней команд относительно выполнения команды, предшествующей таймеру. Фактически данная команда реализует таймер задержки включения с отсчетом времени в обратном порядке (уставка задается с шагом 100 мс). Команда TIMW(813) может отсчитывать время в диапазоне от 0 до 999,9 с (с точностью от 0 до 0,01 с). Команда TIMWX(816) может отсчитывать время в диапазоне от 0 до 6553,5 с (с точностью от 0 до 0,01 с).

Первая часть программного блока выполняется при первом вхождении в программный блок. После достижения команды TIMW(813)/TIMWX(816) сбрасывается флаг завершения таймера, в уставку таймера записывается требуемое значение и оставшаяся часть программного блока не выполняется до тех пор, пока не истекает установленное время.

Пока таймер отсчитывает время, выполняется только команда TIMW(813)/TIMWX(816), обновляющая состояние таймера. Когда таймер завершает отсчет времени, устанавливается флаг завершения и выполняется оставшаяся часть программного блока. После того как программный блок оказывается выполнен полностью, описанный процесс повторяется.

По своему действию команда TIMW(813)/TIMWX(816) эквивалентна формированию условия выполнения с помощью команды ожидания (WAIT) с таймером, поэтому ее можно использовать для последовательного выполнения шагов с выдержкой времени.



Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	Включен, если команда TIMW(813)/TIMWX(816) применена вне программного блока. Включен, если адрес в N указан косвенно с помощью регистра IR в режиме BCD, но значение по указанному адресу не является текущим значением таймера. Включен, если в режиме BCD уставка не представлена в формате BCD. Выключен во всех остальных случаях.

Меры предосторожности

Оставшаяся часть программного блока, расположенная за таймером, будет также выполнена, если флаг завершения будет установлен принудительно.

Если флаг завершения таймера будет принудительно сброшен, внутри программного блока будет выполняться только команда TIMW(813/ TIMWX(816), пока состояние принудительного сброса не будет отменено.

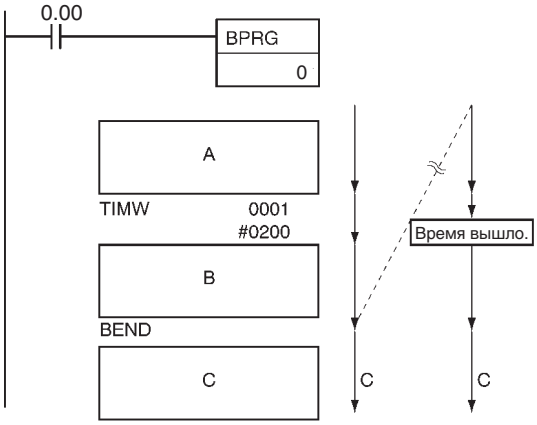
Текущие значения таймеров с номерами от T0 до T15 продолжают обновляться, даже если таймеры находятся в режиме ожидания (standby). Текущие значения таймеров с номерами от T16 до T4095 остаются неизменными, пока таймеры пребывают в режиме ожидания.

Номера таймеров также используются другими командами таймеров. Если сразу несколько команд таймеров используют один и тот же номер таймера, программа работает непредсказуемым образом. Каждый номер таймера должен использоваться только один раз. Использование общего номера таймера несколькими командами допускается лишь в том случае, если эти таймеры не пересекаются во времени. Если в двух или более командах таймеров используется один и тот же номер таймера, при проверке программы будет сгенерирована ошибка.

Если адрес в N указан косвенно с помощью регистра IR в режиме BCD, но значение по указанному адресу не является текущим значением таймера, либо если уставка таймера не представлена в формате BCD, произойдет ошибка и будет установлен флаг ошибки. Ошибка также произойдет и будет установлен флаг ошибки, если адрес в N указан косвенно с помощью регистра IR в двоичном режиме, но значение по указанному адресу не является текущим значением таймера.

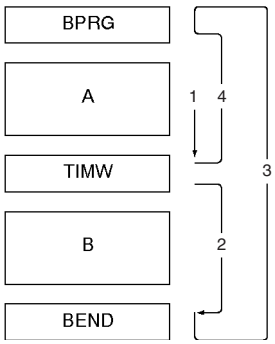
Примеры

В следующем примере сегмент «B» выполняется через 20 секунд после сегмента «A», пока включен бит CIO 0.00.



Адрес	Команда	Операнд
000200	LD	0.00
000201	BPRG	0
. .	A	. .
000210	TIMW	1
		#0200
. .	B	. .
000220	BEND	---

Как видно из приведенного ниже рисунка, пока отсчитывается интервал 20 с до выполнения сегмента «B», сегменты «A» и «B» не выполняются (программа последовательно проходит отрезки 2, 3 и 4 и возвращается к отрезку 2).



3-30-8 СЧЕТЧИК ОЖИДАНИЯ: CNTW(814) и CNTWX(818)

Назначение

Задержка выполнения оставшейся части программного блока до достижения указанного количества единиц счета. После достижения счетчиком заданного значения выполнение продолжится со следующей команды после команды CNTW(814)/CNTWX(818).

Символ РКС

Форма представления текущего значения: двоично-десятичная (BCD)

CNTW(814)	N	**N**: Номер счетчика
	SV	**SV**: Уставка
	I	**I**: Счетный вход

Форма представления текущего значения: двоичная

CNTWX(818)	N	**N**: Номер счетчика
	SV	**SV**: Уставка
	I	**I**: Счетный вход

Варианты выполнения

Варианты выполнения	Всегда выполняется в программном блоке

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Примечание. Команды CNTW(814)/CNTWX(818) обязательно должны использоваться в пределах программных блоков, даже если последние создаются в подпрограммах и задачах обработки прерываний.

Операнды

N: Номер счетчика

BCD: 0...4095 (десятичн.)
Двоичн.: 0...4095 (десятичн.)

S: Уставка

BCD: #0000...#9999 (BCD)
Двоичн.: &0...&65535 (десятичн.)
#0000...#FFFF (hex)

Характеристики операндов

Область	N	S	I
Область CIO	---	CIO 0...CIO 6143	CIO 0.00...CIO 6143.15
Рабочая область	---	W0...W511	W0.00...W511.15
Область битов хранения	---	H0...H511	H0.00...H511.15
Область вспомогательных битов	---	A0...A447 A448...A959	A0.00...A447.15 A448.00...A959.15
Область таймеров	---	T0000...T4095	T0000...T4095
Область счетчиков	0...4095	C0000...C4095	C0000...C4095
Флаги задач	---		TK00...TK31
Флаги условий	---		ER, CY, >, =, <, N, OF, UF, >=, <>, <=, ON,OFF, AER
Тактовые импульсы	---		0,02 с / 0,1 с / 0,2 с / 1 с, 1 мин
Область DM	---	D0...D32767	---
Косвенные адреса DM в двоичном формате	---	@D0...@D3276	---
Косвенные адреса DM в двоично-десятичном формате (BCD)	---	*D0...*D32767	---
Постоянные	---	BCD: #0000...9999 (BCD) «&» использовать нельзя. Двоичный: &0...&65535 (десятичн.) #0000...#FFFF (hex)	---
Регистры данных	---	DR0...DR15	---
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

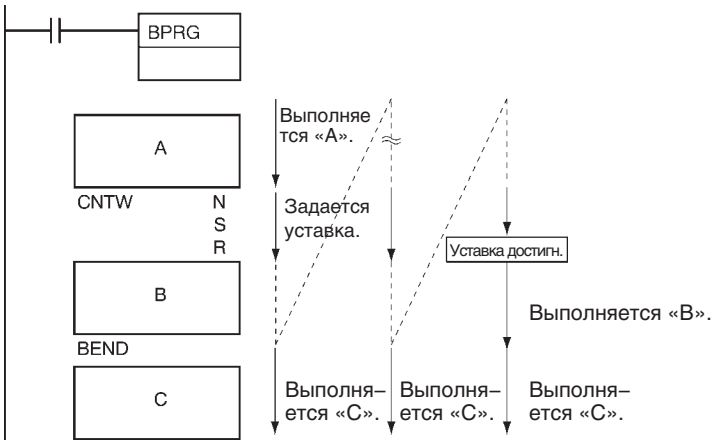
Команда CNTW(814)/CNTWX(818) реализует счетчик обратного счета, задерживающий выполнение команд, расположенных в программном блоке после счетчика, до достижения счетчиком заданного значения (т. е. до обнуления счетчика). Уставка счетчика для команды CNTW(814) указывается в двоично-десятичном формате в диапазоне от 0000 до 9999. Уставка счетчика для команды CNTWX(818) указывается в двоичном формате в диапазоне от 0000 до FFFF hex.

Первая часть программного блока выполняется при первом вхождении в программный блок. После достижения команды CNTW(814)/CNTWX(818) сбрасывается флаг завершения счетчика, в уставку счетчика записывается требуемое значение и оставшаяся часть программного блока не выполняется до тех пор, пока счетчик не обнулится, отсчитав заданное количество единиц счета. Счетчик подсчитывает импульсы (положительные фронты) на входе I.

Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

Пока счетчик производит обратный счет, выполняются только команды CNTW(814)/CNTWX(818), обновляющие состояние счетчика. Когда счетчик завершает счет, устанавливается флаг завершения и выполняется оставшаяся часть программного блока. После того как программный блок оказывается выполнен полностью, описанный процесс повторяется.

Действие команды CNTW(814)/CNTWX(818) эквивалентно формированию условия выполнения с помощью команды ожидания (WAIT) со счетчиком, поэтому ее можно использовать для последовательного выполнения шагов с использованием количества в качестве условия перехода между шагами.



Флаги

Название	Обозначение	Действие
Флаг ошибки	ER	Включен, если команда CNTW(814)/CNTWX(818) применена вне программного блока. Включен, если адрес в N указан косвенно с помощью регистра IR в режиме BCD, но значение по указанному адресу не является текущим значением счетчика. Включен, если в режиме BCD уставка не представлена в формате BCD. Выключен во всех остальных случаях.

Меры предосторожности

Оставшаяся часть программного блока, расположенная за командой CNTW(814)/CNTWX(818), будет также выполнена, если флаг завершения будет установлен принудительно.

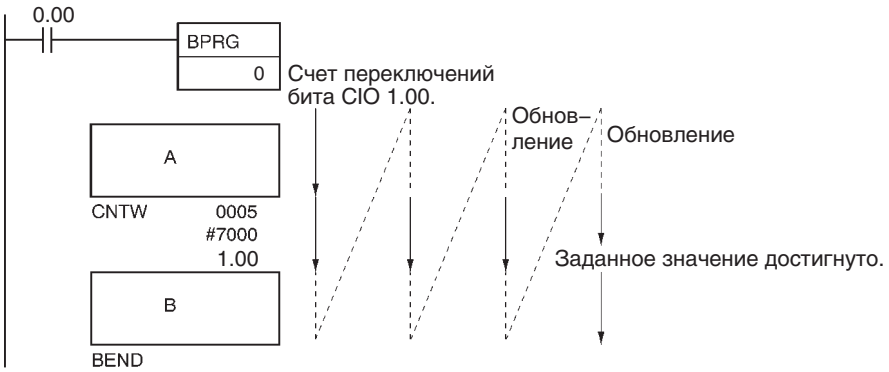
Если флаг завершения счетчика будет принудительно сброшен, внутри программного блока будет выполняться только команда CNTW(814)/ CNTWX(818), пока состояние принудительного сброса не будет отменено.

Номера счетчиков также используются другими командами счетчиков. Если сразу несколько команд счетчиков используют один и тот же номер счетчика, программа работает непредсказуемым образом. Каждый номер счетчика должен использоваться только один раз. Использование общего номера счетчика несколькими командами допускается лишь в том случае, если эти счетчики не пересекаются во времени. Если в двух или более командах счетчиков используется один и тот же номер счетчика, при проверке программы будет сгенерирована ошибка.

Если адрес в N указан косвенно с помощью регистра IR в режиме BCD, но значение по указанному адресу не является текущим значением счетчика, либо если уставка счетчика в режиме BCD не представлена в формате BCD, произойдет ошибка и будет установлен флаг ошибки. Ошибка также произойдет и будет установлен флаг ошибки, если адрес в N указан косвенно с помощью регистра IR в двоичном режиме, но значение по указанному адресу не является текущим значением счетчика.

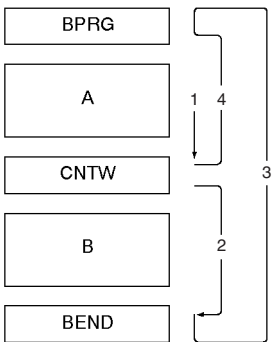
Примеры

Если в приведенном ниже примере бит CIO 0.00 включен, при первом вхождении в блок выполняется сегмент «A», а сегмент «B» будет выполнен лишь после того, как бит CIO 1.00 переключится из состояния «0» в состояние «1» 7000 раз.



Адрес	Команда	Операнд
000200	LD	0.00
000201	BPRG	0
. .	A	. .
000210	CNTW	5
		#7000
		1.00
. .	B	. .
000220	BEND	---

Как видно из приведенного ниже рисунка, пока ведется счет 7000 переключений до выполнения сегмента «B», сегменты «A» и «B» не выполняются (программа последовательно проходит отрезки 2, 3 и 4 и возвращается к отрезку 2).



3-30-9 СКОРОСТНОЙ ТАЙМЕР ОЖИДАНИЯ: TMHW(815) и TMHWX(817)

Назначение

Задержка выполнения оставшейся части программного блока до истечения указанного времени. После достижения таймером установленного значения выполнение продолжится со следующей команды после TMHW(815)/TMHWX(817).

Символ РКС

Форма представления текущего значения: двоично-десятичная (BCD)

TMHW(815)	N	**N**: Номер таймера
	SV	**SV**: Уставка

Форма представления текущего значения: двоичная

TMHWX(817)	N	**N**: Номер таймера
	SV	**SV**: Уставка

Варианты выполнения

Варианты выполнения	Всегда выполняется в программном блоке

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограм-мы	Задачи обработки прерываний
OK	OK	OK	Не допустимо.

Примечание. Команды TMHW(815)/TMHWX(817) обязательно должны использоваться в пределах программных блоков, даже если последние создаются в подпрограммах.

Операнды

N: Номер таймера

BCD: 0...4095 (десятичн.)
Двоичн.: 0...4095 (десятичн.)

S: Уставка

BCD: #0000...#9999 (BCD)
Двоичн.: &0...&65535 (десятичн.)
#0000...#FFFF (hex)

Характеристики операндов

Область	N	Уставка
Область CIO	---	CIO 0...CIO 6143
Рабочая область	---	W0...W511
Область битов хранения	---	H0...H511
Область вспомогательных битов	---	A0...A447 A448...A959
Область таймеров	0...4095	T0000...T4095
Область счетчиков	---	C0000...C4095
Область DM	---	D0...D32767
Косвенные адреса DM в двоичном формате	---	@ D0...@ D32767
Косвенные адреса DM в двоично-десятичном формате (BCD)	---	*D0...*D32767

Область	N	Уставка
Постоянные	---	BCD: #0000...9999 (BCD) «&» использовать нельзя. Двоичный: &0...&65535 (десятичн.) #0000...#FFFF (hex)
Регистры данных	---	DR0...DR15
Регистры указателей	---	---
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

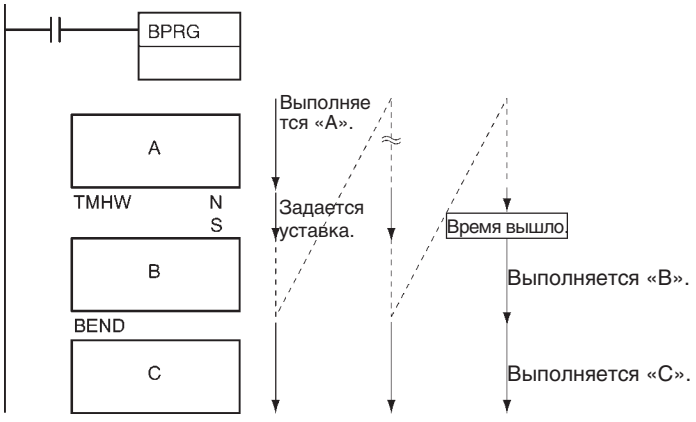
Описание

Команда TMHW(815)/TMHWX(817) задерживает выполнение следующих за ней команд относительно выполнения команды, предшествующей таймеру. Фактически данная команда реализует таймер задержки включения с отсчетом времени в обратном порядке (уставка задается с шагом 10 мс). Команда TMHW(815) может отсчитывать время в диапазоне от 0 до 99,99 с (с точностью от 0 до 0,01 с). Команда TMHWX(817) может отсчитывать время в диапазоне от 0 до 655,35 с (с точностью от 0 до 0,01 с).

Первая часть программного блока выполняется при первом вхождении в программный блок. После достижения команды TMHW(815)/ TMHWX(817) сбрасывается флаг завершения таймера, в уставку таймера записывается требуемое значение и оставшаяся часть программного блока не выполняется до тех пор, пока не истекает установленное время.

Пока таймер отсчитывает время, выполняется только команда TMHW(815)/TMHWX(817), обновляющая состояние таймера. Когда таймер завершает отсчет времени, устанавливается флаг завершения и выполняется оставшаяся часть программного блока. После того как программный блок оказывается выполнен полностью, описанный процесс повторяется.

По своему действию команда TMHW(815)/TMHWX(817) эквивалентна формированию условия выполнения с помощью команды ожидания (WAIT) с таймером, поэтому ее можно использовать для последовательного выполнения шагов с выдержкой времени.



Флаги

Название	Обо-значение	Действие
Флаг ошибки	ER	Включен, если команда TMHW(815)/TMHWX(817) применена вне программного блока. Включен, если адрес в N указан косвенно с помощью регистра IR в режиме BCD, но значение по указанному адресу не является текущим значением таймера. Включен, если в режиме BCD уставка не представлена в формате BCD. Выключен во всех остальных случаях.

Меры предосторожности

Оставшаяся часть программного блока, расположенная за командой TMHW(815)/TMHWX(817), будет также выполнена, если флаг завершения будет установлен принудительно.

Если флаг завершения таймера будет принудительно сброшен, внутри программного блока будет выполняться только команда TMHW(815)/ TMHWX(817), пока состояние принудительного сброса не будет отменено.

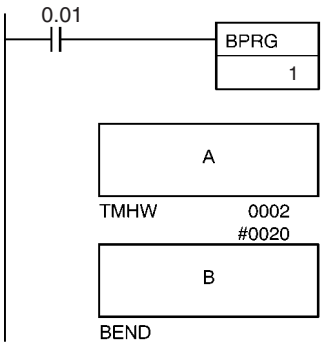
Текущие значения таймеров с номерами от T0 до T15 продолжают обновляться, даже если таймеры находятся в режиме ожидания (standby). Текущие значения таймеров с номерами от T16 до T4095 остаются неизменными, пока таймеры пребывают в режиме ожидания.

Номера таймеров также используются другими командами таймеров. Если сразу несколько команд таймеров используют один и тот же номер таймера, программа работает непредсказуемым образом. Каждый номер таймера должен использоваться только один раз. Использование общего номера таймера несколькими командами допускается лишь в том случае, если эти таймеры не пересекаются во времени. Если в двух или более командах таймеров используется один и тот же номер таймера, при проверке программы будет сгенерирована ошибка.

Если адрес в N указан косвенно с помощью регистра IR в режиме BCD, но значение по указанному адресу не является текущим значением таймера, либо если уставка таймера не представлена в формате BCD, произойдет ошибка и будет установлен флаг ошибки. Ошибка также произойдет и будет установлен флаг ошибки, если адрес в N указан косвенно с помощью регистра IR в двоичном режиме, но значение по указанному адресу не является текущим значением таймера.

Примеры

В следующем примере сегмент «B» выполняется через 20 секунд после сегмента «A», пока включен бит CIO 0.01.



Адрес	Команда	Операнд
000221	LD	0.01
000222	BPRG	1
. .	A	. .

Адрес	Команда	Операнд
000250	TMHW	2
		#0020
. .	B	. .
000281	BEND	---

3-30-10 Создание цикла: LOOP(809)/LEND(810)/LEND(810) NOT

Назначение

Создание цикла. Некоторый сегмент программы циклически повторяется до включения или выключения битового операнда или до включения условия выполнения.

Символ РКС

LOOP(809)

LEND(810)

LEND(810) B **B**: Битовый операнд

LEND(810) NOT B

Варианты выполнения

Варианты выполнения	Всегда выполняется в программном блоке

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Примечание. Команды LOOP(809), LEND(810) и LEND(810) NOT обязательно должны использоваться в пределах программных блоков, даже если последние создаются в подпрограммах и задачах обработки прерываний.

Характеристики операндов

Область	B
Область CIO	CIO 0.00...CIO 6143.15
Рабочая область	W0.00...W511.15
Область битов хранения	H0.00...H511.15
Область вспомогательных битов	A0.00...A447.15 A448.00...A959.15
Область таймеров	T0000...T4095
Область счетчиков	C0000...C4095
Флаги задач	TK00...TK31
Флаги условий	ER, CY, >, =, <, N, OF, UF, >=, <>, <=, ON,OFF, AER
Тактовые импульсы	0,02 с / 0,1 с / 0,2 с / 1 с / 1 мин
Область DM	---
Косвенные адреса DM в двоичном формате	---
Косвенные адреса DM в двоично-десятичном формате (BCD)	---
Постоянные	---
Регистры данных	---

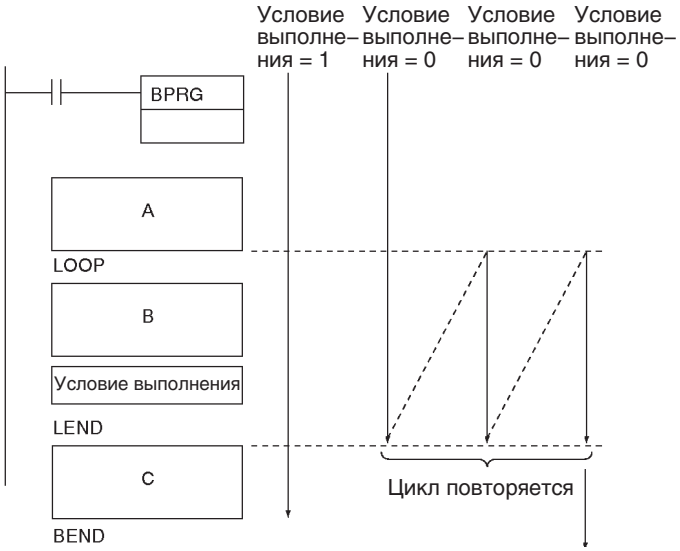
Область	B
Регистры указателей	---
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...-2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15

Описание

Команда LOOP(809) обозначает начало программы цикла. Команда LEND(810) или LEND(810) NOT указывает на конец цикла. Когда программа доходит до команды LEND(810) или LEND(810) NOT, она возвращается к ближайшей предшествующей команде LOOP(809) и выполняется сначала. Так продолжается до тех пор, пока не включается или не выключается битовый операнд команды LEND(810) или LEND(810) NOT (соответственно) или не включается условие выполнения команды LEND(810).

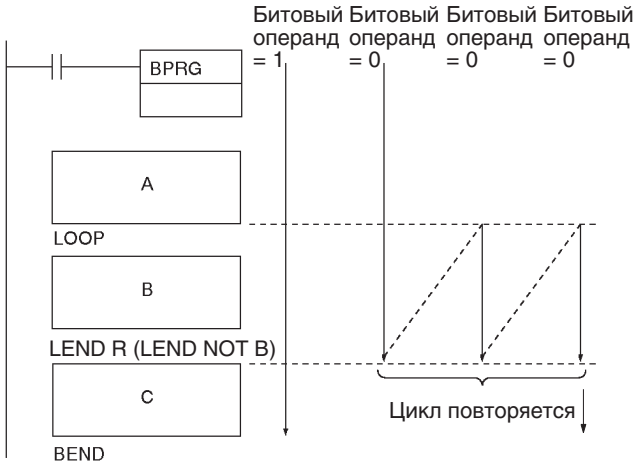
Применение LEND(810) с условие выполнения

Команду LEND(810) можно использовать в программе как с битовым операндом, так и без него. Если битовый операнд в команде LEND(810) не указывается, для этой команды предварительно должно быть сформировано условие выполнения, начинающееся с команды LD. Пока условие выполнения выключено, выполнение программы циклически повторяется, начиная с команды, следующей за командой LOOP(809). При включении условия выполнения цикл завершается и программа переходит к следующей команде, расположенной после команды LEND(810).



Применение команды LEND(810) или LEND(810) NOT с битовым операндом

Обе команды LEND(810) и LEND(810) NOT могут быть использованы в программе с битовым операндом. Если битовый операнд для команды LEND(810) выключен (либо включен для LEND(810) NOT), выполнение программы циклически повторяется, начиная с команды, следующей за командой LOOP(809). После включения битового операнда для команды LEND(810) (или выключения для LEND(810) NOT) цикл завершается и программа переходит к следующей команде, расположенной после команды LEND(810) или LEND(810) NOT.



Прим.: в случае LEND(810) NOT состояние битового операнда будет обратным.

Примечание.

(1) Команды, выполняемые внутри цикла, не обновляют содержимое данных ввода/вывода. Для обновления данных ввода/вывода во время выполнения цикла следует использовать команду IORF(184).

(2) Интенсивное применение длительных циклов может значительно увеличить длительность цикла программы ПЛК. При разработке программы следите за тем, чтобы не оказалась превышена максимальная длительность цикла.

Флаги

Название	Обо-значение	Действие
Флаг ошибки	ER	Включен, если команда создания цикла использована вне программного блока. Выключен во всех остальных случаях.

Меры предосторожности

Вложение циклов (создание цикла внутри другого цикла) не допускается.

Неправильно:

```
LOOP(809)
LOOP(809)
LEND(810)
LEND(810)
```

Не допускается располагать команду LEND перед командой LOOP.

Неправильно:

```
LEND(810)
    :
    :
LOOP(809)
```

В пределах цикла допускается использовать условное ветвление, но ветвление не должно выходить за пределы циклической программы.

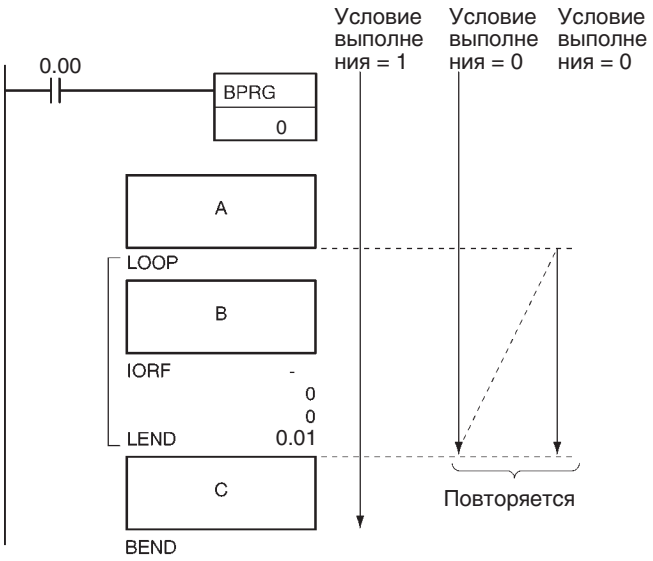
Правильно:	**Неправильно:**
LOOP(809)	LOOP(809)
IF(802)	IF(802)
IF(802)	IF(802)
IEND(804)	IEND(804)
IEND(804)	LEND(810)
LEND(810)	IEND(804)

Если команда LOOP(809) не выполняется, она обрабатывается, как NOP.

Если команда создания цикла будет использована вне программного блока, произойдет ошибка и будет установлен флаг ошибки.

Примеры

Если в следующем примере включен бит CIO 0.00, выполняется программный блок. После того как будет выполнен сегмент «A», сегмент «B» и команда IORF(184) будут выполняться циклически до тех пор, пока не включится бит CIO 0.01, после чего будет выполнен сегмент «C» и программный блок завершится.



Адрес	Команда	Операнд
000220	LD	0.00
000201	BPRG	0
. .	A	. .
000210	LOOP	---
. .	B	. .
000220	IORF	. .
		0000
		0000
000221	LEND	0.01
. .	C	. .
000220	BEND	---

3-31 Команды обработки текстовых строк

В данном разделе описаны команды обработки текстовых строк.

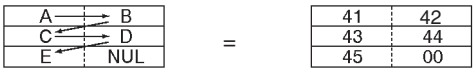
Команда	Мнемоническое обозначение	Код функции	Стр.
КОПИРОВАТЬ СТРОКУ	MOV$	664	1104
ОБЪЕДИНИТЬ СТРОКИ	+$	656	1106
СКОПИРОВАТЬ СИМВОЛЫ СЛЕВА	LEFT$	652	1109
СКОПИРОВАТЬ СИМВОЛЫ СПРАВА	RGHT$	653	1111
СКОПИРОВАТЬ СИМВОЛЫ ВНУТРИ СТРОКИ	MID$	654	1113
НАЙТИ В СТРОКЕ	FIND$	660	1116
ДЛИНА СТРОКИ	LEN$	650	1118
ЗАМЕНИТЬ В СТРОКЕ	RPLC$	661	1120
УДАЛИТЬ СТРОКУ	DEL$	658	1123
ОБМЕН СТРОКАМИ	XCHG$	665	1125
ОЧИСТИТЬ СТРОКУ	CLR$	666	1127
ВСТАВИТЬ В СТРОКУ	INS$	657	1129
Команды сравнения строк	=$, <>$, <$, <=$, >$, >=$	670...675	1132

3-31-1 Обзор операций над текстовыми строками

Данные с начала строки до кода NUL (00 hex) воспринимаются и обрабатываются как единая текстовая строка в ASCII-формате (за исключением специальных 1-байтовых символов). Порядок расположения ASCII-кодов в строке: по возрастанию адресов слов (от младшего слова к старшему), а внутри слов: слева направо (от старшего байта к младшему).

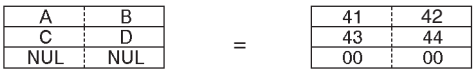
В случае нечетного количества символов младший байт последнего слова содержит значение 00 hex (код NUL).

Пример: текстовая строка ABCDE



В случае четного количества символов в обоих байтах слова, следующего за последним словом строки, хранится значение 0000 hex (два кода NUL).

Пример: текстовая строка ABCD

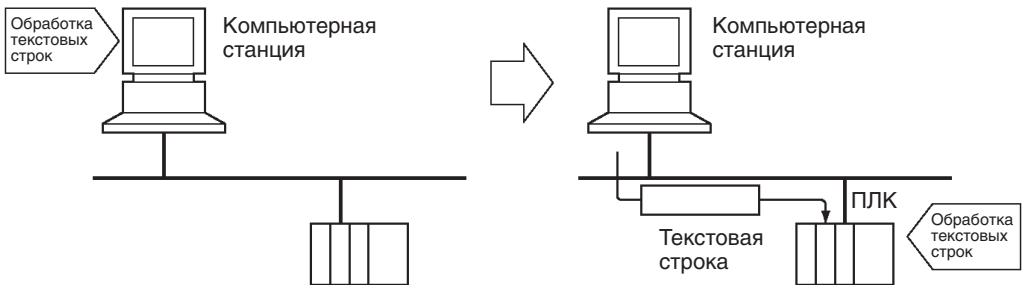


Как показано на следующем рисунке, для указания текстовой строки достаточно указать первое слово этой строки. Данные текстовой строки до следующего кода NUL (00 hex) будут обрабатываться как единый блок данных в формате ASCII.

Пример: MOV$ D0 D100



Команды обработки текстовых строк можно использовать для выполнения на ПЛК различных операций обработки текстовых строк (данных о продукции и т. д.), которые ранее было принято выполнять на компьютерной станции.



Например, можно передать с компьютерной станции в ПЛК данные производственной программы (наименования изделий и т. п.) и силами ПЛК выполнять различные операции над этими данными (вставка, изменение порядка текстовых строк и др.), снижая тем самым вычислительную нагрузку на компьютерную станцию.

Символы ASCII

Символы ASCII, которые можно обработать с помощью команд обработки текстовых строк, представлены в следующей таблице.

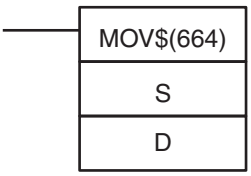
Четыре младших бита \ Четыре старших бита	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0			SP	0	@	P	‘	p				ー	タ	ミ		
1			!	1	A	Q	a	q			。	ア	チ	ム		
2			”	2	B	R	b	r			「	イ	ツ	メ		
3			#	3	C	S	c	s			」	ウ	テ	モ		
4			$	4	D	T	d	t			、	エ	ト	ヤ		
5			%	5	E	U	e	u			・	オ	ナ	ユ		
6			&	6	F	V	f	v			ヲ	カ	ニ	ヨ		
7			’	7	G	W	g	w			ァ	キ	ヌ	ラ		
8			(	8	H	X	h	x			ィ	ク	ネ	リ		
9			)	9	I	Y	i	y			ゥ	ケ	ノ	ル		
A			*	:	J	Z	j	z			ェ	コ	ハ	レ		
B			+	;	K	[	k	{			ォ	サ	ヒ	ロ		
C			,	<	L	¥	l	|			ャ	シ	フ	ワ		
D			―	=	M	]	m	}			ュ	ス	ヘ	ン		
E			.	>	N	^	n	~			ョ	セ	ホ	゛		
F			/	?	O	_	o				ッ	ソ	マ	°		

3-31-2 КОПИРОВАТЬ СТРОКУ: MOV$(664)

Назначение

Передача текстовой строки.

Символ РКС



S: Первое исходное слово

D: Первое слово назначения

Варианты выполнения

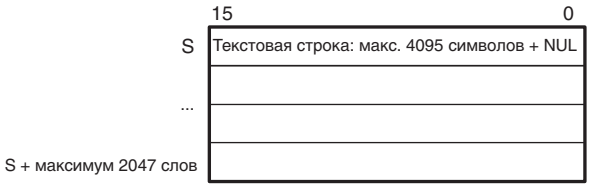
Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	MOV$(664)
	Однократное выполнение по положительному фронту	@MOV$(664)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

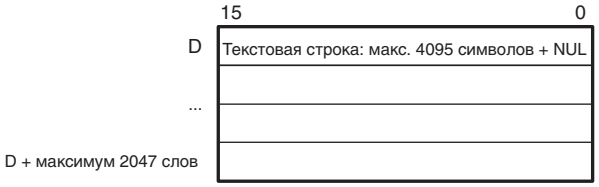
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

S: Первое исходное слово



D: Первое слово назначения



Примечание.

(1) Данные в [S...S + максимум 2047 слов] и [D...D + максимум 2047 слов] должны принадлежать одной области.

(2) Данные в [S...S + максимум 2047 слов] и [D...D + максимум 2047 слов] могут перекрываться.

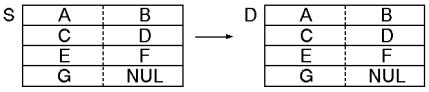
Характеристики операндов

Область	S	D
Область CIO	CIO 0...CIO 6143	
Рабочая область	W0...W511	
Область битов хранения	H0...H511	
Область вспомогательных битов	A0...A447 A448...A959	A448...A959
Область таймеров	T0000...T4095	
Область счетчиков	C0000...C4095	
Область DM	D0...D32767	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Постоянные	---	
Регистры данных	---	

Область	S	D
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...–2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

Описание

Команда MOV$(664) копирует без изменений содержимое текстовой строки S (включая завершающий код NUL) в текстовую строку D. Максимальное количество символов, которое может быть указано операндом S: 4095 (0FFF hex).



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если операндом S определено больше 4095 символов. Включен, если указано выполнение в фоновом режиме и при этом выключен флаг доступности порта связи, соответствующий номеру порта связи, указанному в *Com Port number (Номер COM-порта)* для *Background Execution (Выполнение в фоновом режиме)* (только для CP1H). Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если в D передано значение 0000 (hex). Выключен во всех остальных случаях.

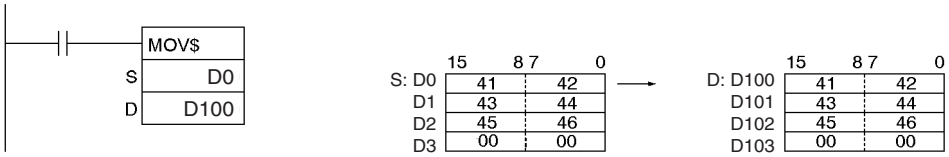
Меры предосторожности

Если операндом S определено больше 4095 символов, будет сгенерирована ошибка и установится флаг ошибки.

Если в D будет передано значение 0000 (hex), будет установлен флаг равенства.

Пример

В данном примере команда MOV$(664) используется для передачи текстовой строки ABCDEF.

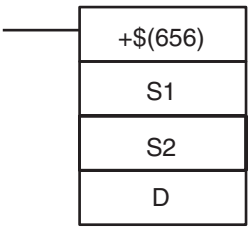


3-31-3 ОБЪЕДИНИТЬ СТРОКИ: +$(656)

Назначение

Объединение двух текстовых строк.

Символ РКС



S1: Текстовая строка 1

S2: Текстовая строка 2

D: Первое слово назначения

Варианты выполнения

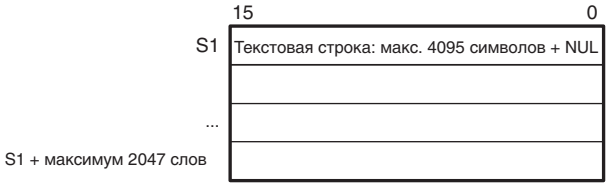
Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	+$(656)
	Однократное выполнение по положительному фронту	@+$(656)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

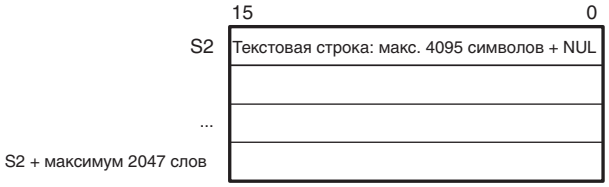
Области программных блоков	Области пошаговых программ	Подпро-граммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

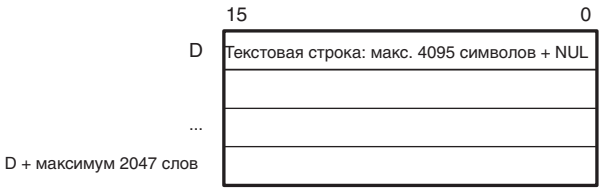
S1: Текстовая строка 1



S2: Текстовая строка 2



D: Первое слово назначения



Примечание.

(1) Данные в [S1...S1 + максимум 2047 слов], [S2...S2 + максимум 2047 слов] и [D...D + максимум 2047 слов] должны принадлежать одной области.

(2) Данные в [S2...S2 + максимум 2047 слов] и [D...D + максимум 2047 слов] не должны перекрываться.

Характеристики операндов

Область	S1	S2	D
Область CIO	CIO 0...CIO 6143		
Рабочая область	W0...W511		
Область битов хранения	H0...H511		
Область вспомогательных битов	A0...A447 A448...A959		A448...A959
Область таймеров	T0000...T4095		
Область счетчиков	C0000...C4095		
Область DM	D0...D32767		

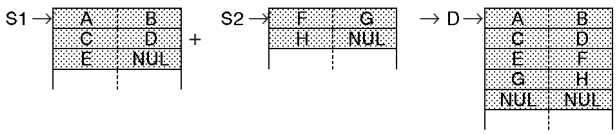
Область	S1	S2	D
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	---		
Регистры данных	---		
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...–2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0V... ,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

+$(664) прикрепляет данные текстовой строки S2 сразу следом за данными текстовой строки S1 и выводит результат в операнд D в формате текстовой строки (включая завершающий код NUL).

Максимальное количество символов, которое может быть указано операндами S1 и S2: 4095 (0FFF hex). Если в пределах 4096 символов отсутствует код NUL, будет сгенерирована ошибка и установится флаг ошибки. Более того, результат объединения может содержать не более 4095 символов (0FFF hex). Если результат объединения содержит больше 4095 символов, в операнд D выводятся только первые 4095 символов (в качестве 4096-го символа добавляется код NUL).

Если код NUL содержится и в S1, и в S2, в операнд D будут выведены два кода NUL (0000 hex).



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если операндами S1 и S2 определено больше 4095 символов. Включен, если указано выполнение в фоновом режиме и при этом выключен флаг доступности порта связи, соответствующий номеру порта связи, указанному в *Com Port number (Номер COM-порта)* для *Background Execution (Выполнение в фоновом режиме)* (только для CP1H). Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если в D передано значение 0000 (hex). Выключен во всех остальных случаях.

Предварительная версия
Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

Меры предосторожности

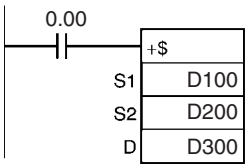
Если операндами S1 и S2 определено больше 4095 символов, будет сгенерирована ошибка и установится флаг ошибки.

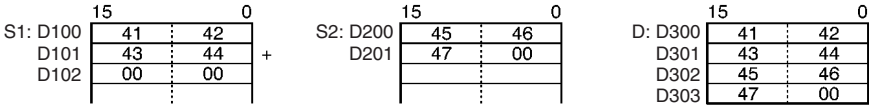
Если в D будет передано значение 0000 (hex), будет установлен флаг равенства.

Начальное слово, определенное операндом D, не должно накладываться на область текстовых данных, определенную операндом S2. При наложении областей корректное выполнение команды будет невозможно.

Пример

В данном примере команда +$(656) используется для объединения текстовых строк ABCD и EFG и вывода результата в операнд D.



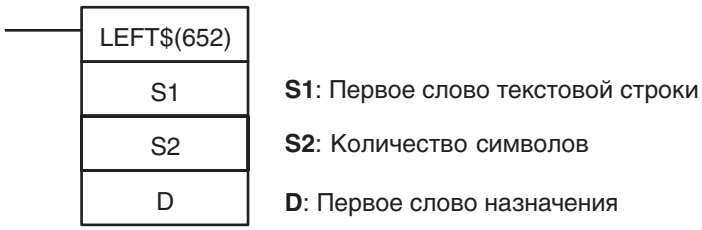


3-31-4 СКОПИРОВАТЬ СИМВОЛЫ СЛЕВА: LEFT$(652)

Назначение

Считывание указанного количества символов, расположенных в начале (слева) текстовой строки, и перенос их в слово назначения.

Символ РКС



S1: Первое слово текстовой строки

S2: Количество символов

D: Первое слово назначения

Варианты выполнения

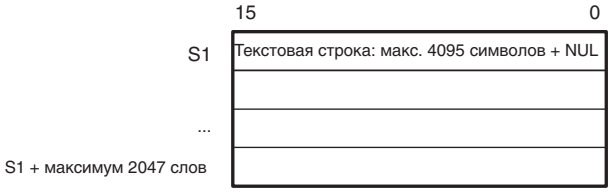
Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	LEFT$(652)
	Однократное выполнение по положительному фронту	@LEFT$(652)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

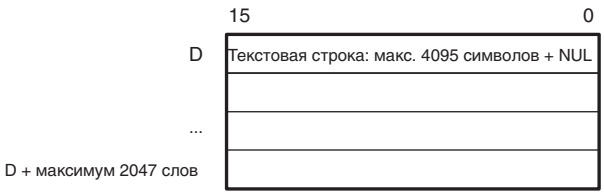
Области программных блоков	Области пошаговых программ	Подпро-граммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

S1: Текстовая строка



S2: Количество символов (0000...0FFF hex или &0...&4095)



Примечание.

(1) Данные в [S1...S1 + максимум 2047 слов] и [D...D + максимум 2047 слов] должны принадлежать одной области.

(2) Данные в [S1...S1 + максимум 2047 слов] и [D...D + максимум 2047 слов] могут перекрываться.

Характеристики операндов

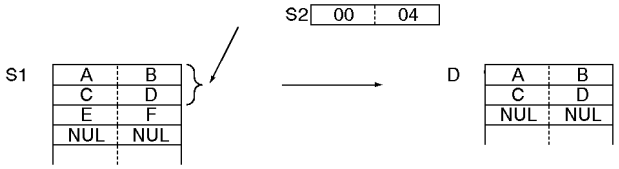
Область	S1	S2	D
Область CIO	CIO 0...CIO 6143		
Рабочая область	W0...W511		
Область битов хранения	H0...H511		
Область вспомогательных битов	A0...A447 A448...A959		A448...A959
Область таймеров	T0000...T4095		
Область счетчиков	C0000...C4095		
Область DM	D0...D32767		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	---	#0000...#0FFF (двоичный) или &0...&4095	---
Регистры данных	---	DR0...DR15	---
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...–2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда LEFT$(652) считывает указанное операндом S2 количество символов, расположенных в начале текстовой строки S1 (начиная со старшего байта первого слова строки), и выводит результат в операнд D в формате текстовой строки (добавляя в конце код NUL).

Если указанное количество читаемых символов S2 превышает число символов, определенное операндом S1, в D будет выведена вся текстовая строка S1.

Если будет указано нулевое количество читаемых символов (0000 hex), в D будут выведены два символа NUL (0000 hex).



Предварительная версия
Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если операндом S1 определено больше 4095 символов. Включен, если операндом S2 определено больше 4095 символов (0FFF hex). Включен, если указано выполнение в фоновом режиме и при этом выключен флаг доступности порта связи, соответствующий номеру порта связи, указанному в *Com Port number (Номер COM-порта)* для *Background Execution (Выполнение в фоновом режиме)* (только для CP1H). Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если в D выдано значение 0000 (hex). Выключен во всех остальных случаях.

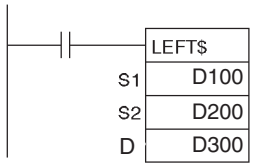
Меры предосторожности

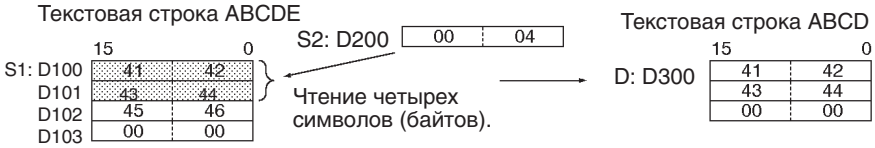
Максимальное количество символов для чтения, которое может быть указано операндом S2: 4095 (0FFF hex). При большем значении будет сгенерирована ошибка и установится флаг ошибки.

Если в D будет выдано значение 0000 (hex), установится флаг равенства.

Пример

В данном примере команда LEFT$(652) используется для считывания четырех символов.



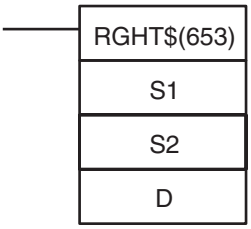


3-31-5 СКОПИРОВАТЬ СИМВОЛЫ СПРАВА: RGHT$(653)

Назначение

Считывание указанного количества символов, расположенных в конце (справа) текстовой строки.

Символ РКС



S1: Первое слово текстовой строки

S2: Количество символов

D: Первое слово назначения

Варианты выполнения

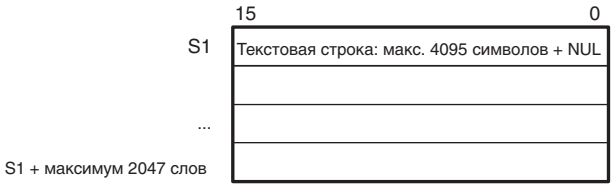
Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	RGHT$(653)
	Однократное выполнение по положительному фронту	@RGHT$(653)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

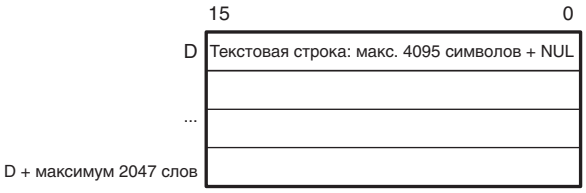
Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

S1: Текстовая строка



S2: Количество символов (0000...0FFF hex или &0...&4095)



Примечание.

(1) Данные в [S1...S1 + максимум 2047 слов] и [D...D + максимум 2047 слов] должны принадлежать одной области.

(2) Данные в [S1...S1 + максимум 2047 слов] и [D...D + максимум 2047 слов] могут перекрываться.

Характеристики операндов

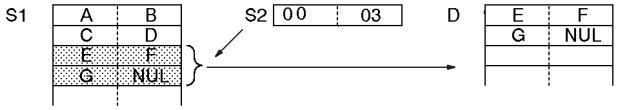
Область	S1	S2	D
Область CIO	CIO 0...CIO 6143		
Рабочая область	W0...W511		
Область битов хранения	H0...H511		
Область вспомогательных битов	A0...A447 A448...A959		A448...A959
Область таймеров	T0000...T4095		
Область счетчиков	C0000...C4095		
Область DM	D0...D32767		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	---	#0000...#0FFF (двоичный) или &0...&4095	---
Регистры данных	---	DR0...DR15	---
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...–2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда RGHT$(653) считывает указанное операндом S2 количество символов, расположенных в конце текстовой строки S1 (примыкающих к коду NUL (00 hex)), и выводит результат в операнд D в формате текстовой строки (добавляя в конце код NUL).

Если указанное количество читаемых символов превышает число символов, определенное операндом S1, в D будет выведена вся текстовая строка S1.

Если будет указано нулевое количество читаемых символов (0000 hex), в D будут выведены два символа NUL (0000 hex).



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если операндом S1 определено больше 4095 символов. Включен, если операндом S2 определено больше 4095 символов (0FFF hex). Включен, если указано выполнение в фоновом режиме и при этом выключен флаг доступности порта связи, соответствующий номеру порта связи, указанному в *Com Port number (Номер COM-порта)* для *Background Execution (Выполнение в фоновом режиме)* (только для CP1H). Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если в D выдано значение 0000 (hex). Выключен во всех остальных случаях.

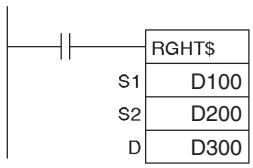
Меры предосторожности

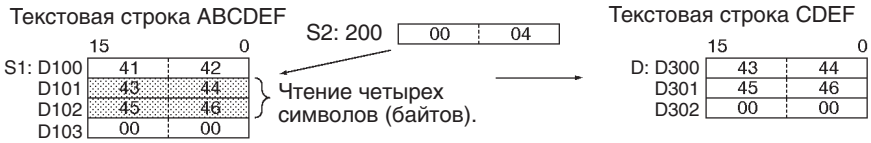
Максимальное количество символов для чтения, которое может быть указано операндом S2: 4095 (0FFF hex). При большем значении будет сгенерирована ошибка и установится флаг ошибки.

Если в D будет выдано значение 0000 (hex), установится флаг равенства.

Пример

В данном примере команда RGHT$(653) используется для считывания четырех символов.



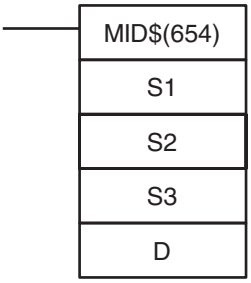


3-31-6 СКОПИРОВАТЬ СИМВОЛЫ ВНУТРИ СТРОКИ: MID$(654)

Назначение

Считывание указанного количества символов, начиная с любой позиции в пределах текстовой строки.

Символ РКС



S1: Первое слово текстовой строки

S2: Количество символов

S3: Начальная позиция

D: Первое слово назначения

Варианты выполнения

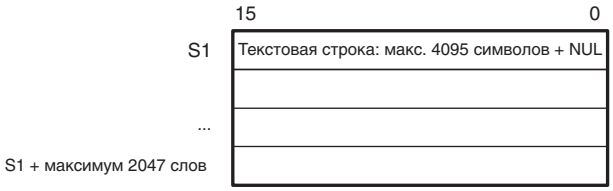
Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	MID$(654)
	Однократное выполнение по положительному фронту	@MID$(654)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

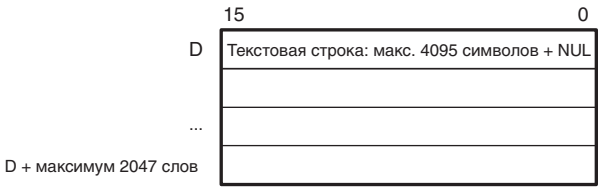
Области программных блоков	Области пошаговых программ	Подпро-граммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

S1: Текстовая строка



S2: Количество символов (0000...0FFF hex или &0...&4095)
S3: Начальная позиция (0001...0FFF hex или &1...&4095)



Примечание.

(1) Данные в [S1...S1 + максимум 2047 слов] и [D...D + максимум 2047 слов] должны принадлежать одной области.

(2) Данные в [S1...S1 + максимум 2047 слов] и [D...D + максимум 2047 слов] могут перекрываться.

Характеристики операндов

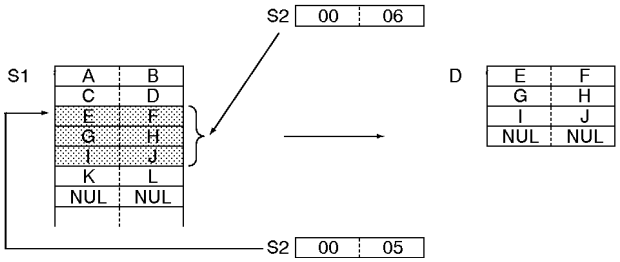
Область	S1	S2	S3	D
Область CIO	CIO 0...CIO 6143			
Рабочая область	W0...W511			
Область битов хранения	H0...H511			
Область вспомогательных битов	A0...A447 A448...A959			A448...A959
Область таймеров	T0000...T4095			
Область счетчиков	C0000...C4095			
Область DM	D0...32767			
Косвенные адреса DM в двоичном формате	@ D0...@ D32767			
Косвенные адреса DM в BCD-формате	*D0...*D32767			
Постоянные	---	#0000...#0FFF (двоичный) или &0...&4095	#0001...#0FFF (двоичный) или &1...&4095	---
Регистры данных	---	DR0...DR15		

Область	S1	S2	S3	D
Регистры указателей	---			
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...–2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15			

Описание

Команда MID$(654) считывает указанное операндом S2 количество символов, расположенных в любом месте текстовой строки S1, начиная со слова, указанного операндом S3, и выводит результат в операнд D в формате текстовой строки (добавляя в конце код NUL).

Если указанное количество читаемых символов S2 превышает фактическое количество символов от позиции S3 до конца строки S1, в D выводятся все символы от позиции S3 до конца строки.



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если операндом S1 определено больше 4095 символов. Включен, если операндом S2 определено больше 4095 символов (0FFF hex). Включен, если значение S3 не находится в диапазоне от 1 до 4095 (0001...0FFF hex). Включен, если S3 лежит за областью фактического количества символов в строке (но тем не менее попадает в диапазон от 1 до 4095). Включен, если указано выполнение в фоновом режиме и при этом выключен флаг доступности порта связи, соответствующий номеру порта связи, указанному в *Com Port number (Номер COM-порта)* для *Background Execution (Выполнение в фоновом режиме)* (только для CP1H). Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если в D выдано значение 0000 (hex). Выключен во всех остальных случаях.

Меры предосторожности

В качестве начального символа операндом S3 может быть указан любой символ строки с 1-го по 4095-й (0001...0FFF hex). При значении вне этого диапазона будет сгенерирована ошибка и установится флаг ошибки.

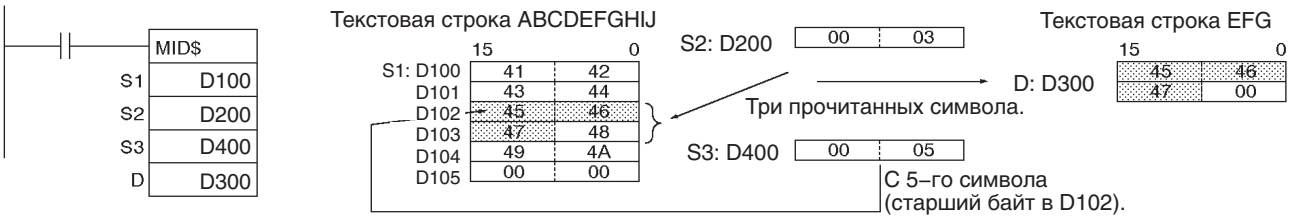
Максимальное количество символов для чтения, которое может быть указано операндом S2: 4095 (0FFF hex). При большем значении будет сгенерирована ошибка и установится флаг ошибки.

Если будет указано нулевое количество читаемых символов (0000 hex), в D будут выведены два символа NUL (0000 hex).

Если в D будет выдано значение 0000 (hex), установится флаг равенства.

Пример

В данном примере команда MID$(654) используется для считывания трех символов.

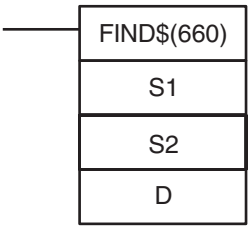


3-31-7 НАЙТИ В СТРОКЕ: FIND$(660)

Назначение

Поиск указанных данных в строке.

Символ РКС



S1: Первое слово входной текстовой строки

S2: Первое слово искомой текстовой строки

D: Первое слово назначения

Варианты выполнения

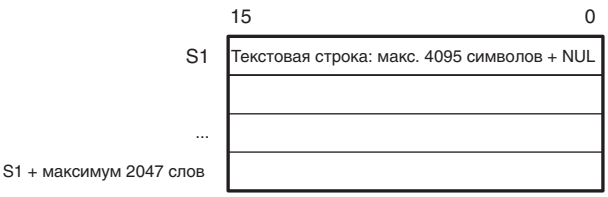
Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	FIND$(660)
	Однократное выполнение по положительному фронту	@FIND$(660)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

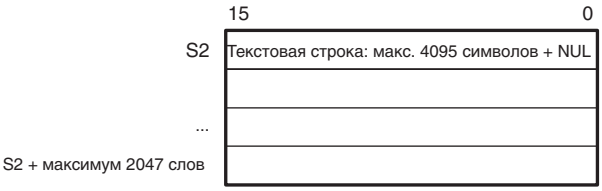
Области программных блоков	Области пошаговых программ	Подпро-граммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

S1: Входная текстовая строка



S2: Искомая текстовая строка



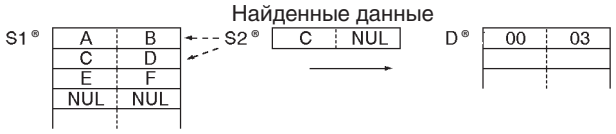
Примечание. Данные в [S1...S1 + максимум 2047 слов] и [S2...S2 + максимум 2047 слов] должны принадлежать одной области.

Характеристики операндов

Область	S1	S2	D
Область CIO	CIO 0...CIO 6143		
Рабочая область	W0...W511		
Область битов хранения	H0...H511		
Область вспомогательных битов	A0...A447 A448...A959		A448...A959
Область таймеров	T0000...T4095		
Область счетчиков	C0000...C4095		
Область DM	D0...D32767		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	---		
Регистры данных	---		
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...–2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15		

Описание

Команда FIND$(660) выполняет поиск текстовой строки, указанной операндом S2, в текстовой строке, указанной операндом S1, и в качестве результата выводит в D номер символа с начала строки S1. Если совпадений не обнаружено, в операнд D выводится значение 0000 hex.



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если операндом S1 или S2 определено больше 4095 символов. Включен, если указано выполнение в фоновом режиме и при этом выключен флаг доступности порта связи, соответствующий номеру порта связи, указанному в *Com Port number (Номер COM-порта)* для *Background Execution (Выполнение в фоновом режиме)* (только для CP1H). Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если в D выдано значение 0000 (hex). Выключен во всех остальных случаях.

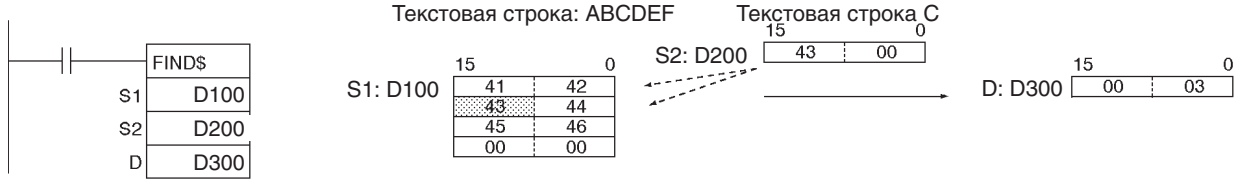
Меры предосторожности

Максимальное количество символов, которое может быть указано операндом S1 или S2: 4095 (0FFF hex). При большем значении будет сгенерирована ошибка и установится флаг ошибки.

Если в D будет выдано значение 0000 (hex), установится флаг равенства.

Пример

В данном примере команда FIND$(660) используется для поиска одного символа в текстовой строке.

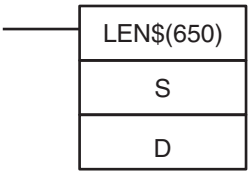


3-31-8 ДЛИНА СТРОКИ: LEN$(650)

Назначение

Вычисление длины текстовой строки.

Символ РКС



S: Первое слово текстовой строки

D: Первое слово назначения

Варианты выполнения

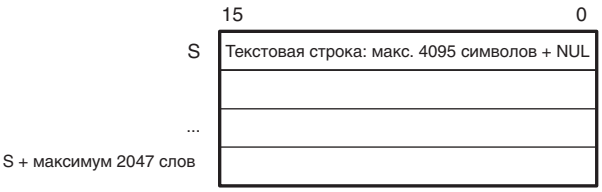
Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	LEN$(650)
	Однократное выполнение по положительному фронту	@LEN$(650)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпро-граммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

S: Текстовая строка



Примечание. Данные в [S...S + максимум 2047 слов] должны принадлежать одной области.

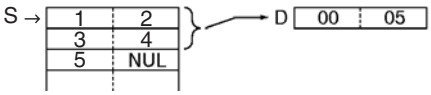
Характеристики операндов

Область	S	D
Область CIO	CIO 0...CIO 6143	
Рабочая область	W0...W511	
Область битов хранения	H0...H511	

Область	S	D
Область вспомогательных битов	A0...A447 A448...A959	A448...A959
Область таймеров	T0000...T4095	
Область счетчиков	C0000...C4095	
Область DM	D0...D32767	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Постоянные	---	
Регистры данных	---	DR0...DR15
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...–2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

Описание

Команда LENS$(650) подсчитывает количество символов в текстовой строке S (не включая код NUL (00 hex)) и выводит результат в операнд D в шестнадцатеричном формате. Если код NUL расположен в начале текстовой строки, результатом будет значение 0000 hex.



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если полученная длина строки превышает 4095 символов. Включен, если указано выполнение в фоновом режиме и при этом выключен флаг доступности порта связи, соответствующий номеру порта связи, указанному в *Com Port number (Номер COM-порта)* для *Background Execution (Выполнение в фоновом режиме)* (только для CP1H). Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если результат вычислений равен 0. Выключен во всех остальных случаях.

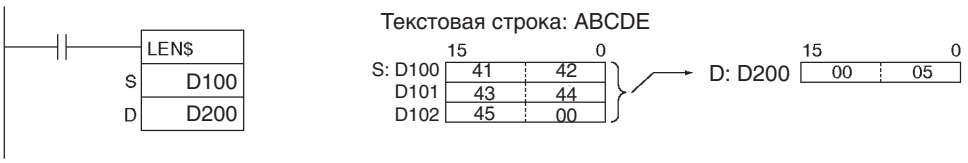
Меры предосторожности

Максимальное количество символов: 4095 (0FFF hex). При большем количестве символов (т. е. при отсутствии символа NUL до 4096-го символа) будет сгенерирована ошибка и установится флаг ошибки.

Если в D будет выдано значение 0000 (hex), установится флаг равенства.

Пример

В данном примере команда LENS$(650) используется для подсчета числа символов и вывода результата.

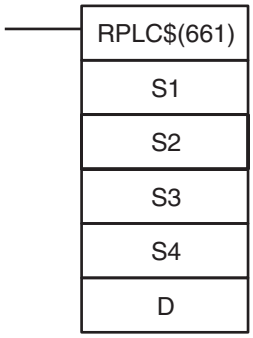


3-31-9 ЗАМЕНИТЬ В СТРОКЕ: RPLC$(661)

Назначение

Замещение текстовой строки указанной текстовой строкой, начиная с указанной позиции.

Символ РКС



S1: Первое слово текстовой строки

S2: Первое слово заменяющей текстовой строки

S3: Количество символов

S4: Начальная позиция

D: Первое слово назначения

Варианты выполнения

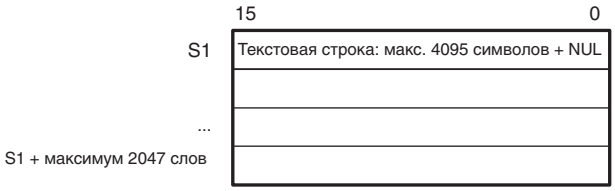
Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	RPLC$(661)
	Однократное выполнение по положительному фронту	@RPLC$(661)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

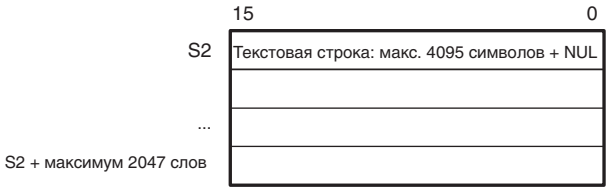
Области программных блоков	Области пошаговых программ	Подпро-граммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

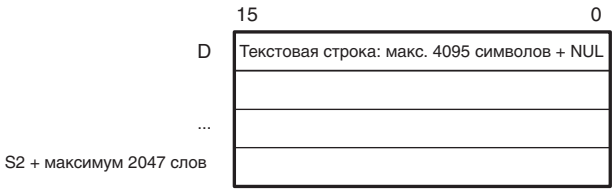
S1: Текстовая строка



S2: Заменяющая текстовая строка



S3: Количество символов (0000...0FFF hex или &0...&4095)
S4: Начальная позиция (0001...0FFF hex или &1...&4095)



Примечание.

(1) Данные в [S1...S1 + максимум 2047 слов], [S2...S2 + максимум 2047 слов] и [D...D + максимум 2047 слов] должны принадлежать одной области.

(2) Данные в [D...D + максимум 2047 слов] и [S1...S1 + максимум 2047 слов] или [S2...S2 + максимум 2047 слов] могут перекрываться.

Характеристики операндов

Область	S1	S2	S3	S4	D
Область CIO	CIO 0...CIO 6143				
Рабочая область	W0...W511				
Область битов хранения	H0...H511				
Область вспомогательных битов	A0...A447 A448...A959				A448... A959
Область таймеров	T0000...T4095				
Область счетчиков	C0000...C4095				
Область DM	D0...D32767				
Косвенные адреса DM в двоичном формате	@ D0...@ D32767				
Косвенные адреса DM в BCD-формате	*D0...*D32767				
Постоянные	---		#0000...#0FFF (двоичный) или &0...&4095	#0001...#0FFF (двоичный) или &1...&4095	---
Регистры данных	---	DR0...DR15			---
Регистры указателей	---				
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...–2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15				

Описание

Команда RPLC$(661) заменяет часть текстовой строки S1, начиная с позиции S4, текстовой строкой S2 и выводит результат в операнд D в формате текстовой строки (добавляя в конце код NUL). Количество заменяемых символов определяется операндом S3.

Максимальное количество символов результата: 4095 (0FFF hex). Если символов будет больше, в результат будет выведено только 4095 символов (в качестве 4096-го символа будет добавлен код NUL).

Можно заменить от 0 до 4095 символов (0000...0FFF hex). Если указано нулевое количество символов, в D выводится без изменений вся строка, определенная операндом S1. Если текстовая строка S2 содержит только NUL, действие команды эквивалентно удалению указанной части текста из строки S1.

Если текстовая строка S1 полностью заменяется кодами NUL, в операнд D выводятся два кода NUL (0000 hex).

Флаги

Название	Обо-значение	Описание работы
Флаг ошибки	ER	Включен, если операндом S1 или S2 определено больше 4095 символов. Включен, если операндом S3 определено больше 4095 символов (0FFF hex). Включен, если значение S4 не находится в диапазоне от 1 до 4095 (0001...0FFF hex). Включен, если указано выполнение в фоновом режиме и при этом выключен флаг доступности порта связи, соответствующий номеру порта связи, указанному в *Com Port number (Номер COM-порта)* для *Background Execution (Выполнение в фоновом режиме)* (только для CP1H). Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если в D выдано значение 0000 (hex). Выключен во всех остальных случаях.

Меры предосторожности

Максимальное количество символов в S1 и S2: 4095 (0FFF hex). При большем количестве символов (т. е. при отсутствии символа NUL до 4096-го символа) будет сгенерирована ошибка и установится флаг ошибки.

В качестве начального символа операндом S4 может быть указан любой символ строки с 1-го по 4095-й (0001...0FFF hex). При значении вне этого диапазона будет сгенерирована ошибка и установится флаг ошибки.

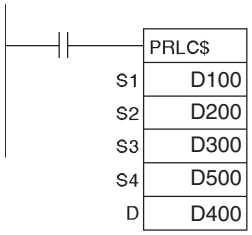
Если начальный символ, указанный операндом S4, находится за пределами текстовой строки, указанной операндом S1, будет сгенерирована ошибка и установится флаг ошибки.

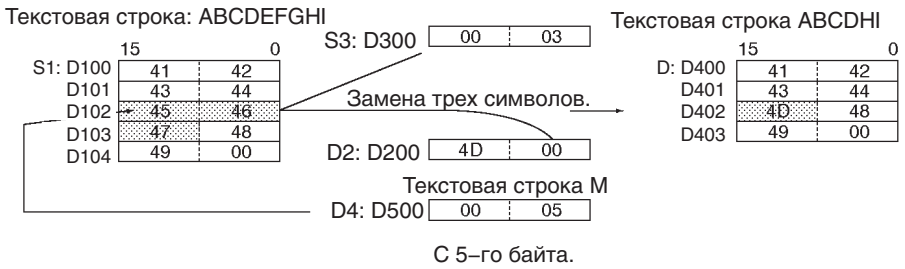
Если в D будет выдано значение 0000 (hex), установится флаг равенства.

Адрес первого слова результата в операнде D должен быть таким, чтобы область результата не перекрывалась с областью, заданной адресом первого слова заменяющей текстовой строки в S2. При наложении этих областей корректная работа команды RPLC$(654) невозможна.

Пример

В данном примере команда RPLC$(654) используется для замены одного (5-го) символа и удаления двух (6 и 7-го) символов, поскольку для этих двух символов команда действует подобно команде Del.



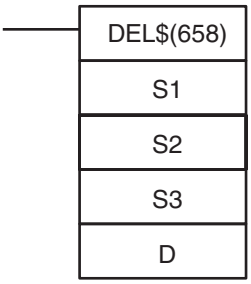


3-31-10 УДАЛИТЬ СТРОКУ: DEL$(658)

Назначение

Удаление указанного количества символов из указанной текстовой строки, начиная с указанной позиции.

Символ РКС



S1: Первое слово текстовой строки

S2: Количество символов

S3: Начальная позиция

D: Первое слово назначения

Варианты выполнения

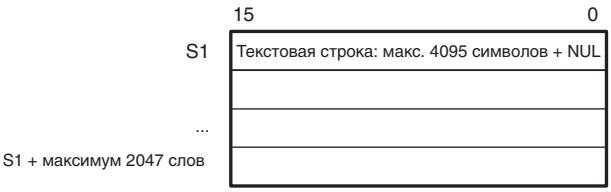
Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	DEL$(658)
	Однократное выполнение по положительному фронту	@DEL$(658)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

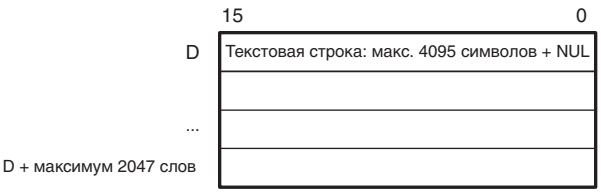
Области программных блоков	Области пошаговых программ	Подпро-граммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

S1: Текстовая строка



S2: Количество символов (0000...0FFF hex или &0...&4095)
S3: Начальная позиция (0001...0FFF hex или &1...&4095)



Примечание.

(1) Данные в [S1...S1 + максимум 2047 слов] и [D...D + максимум 2047 слов] должны принадлежать одной области.

(2) Данные в [S1...S1 + максимум 2047 слов] и [D...D + максимум 2047 слов] могут перекрываться.

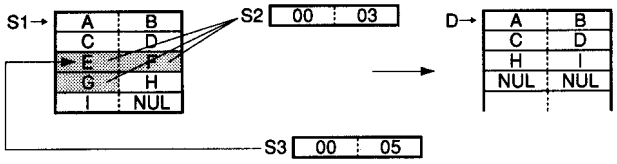
Характеристики операндов

Область	S1	S2	S3	D
Область CIO	CIO 0...CIO 6143			
Рабочая область	W0...W511			

Область	S1	S2	S3	D
Область битов хранения	H0...H511			
Область вспомогательных битов	A0...A447 A448...A959			A448...A959
Область таймеров	T0000...T4095			
Область счетчиков	C0000...C4095			
Область DM	D0...D32767			
Косвенные адреса DM в двоичном формате	@ D0...@ D32767			
Косвенные адреса DM в BCD-формате	*D0...*D32767			
Постоянные	---	#0000...#0FFF (двоичный) или &0...&4095	#0001...#0FFF (двоичный) или &1...&4095	---
Регистры данных	---	DR0...DR15		---
Регистры указателей	---			
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...–2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15			

Описание

Команда DEL$(658) удаляет указанное операндом S2 количество символов из текстовой строки S1, начиная со слова, указанного операндом S3, и выводит результат в операнд D в формате текстовой строки (добавляя в конце код NUL).



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если операндом S1 определено больше 4095 символов. Включен, если операндом S2 определено больше 4095 символов (0FFF hex). Включен, если значение S3 не находится в диапазоне от 1 до 4095 (0001...0FFF hex). Включен, если S3 выходит за пределы S1. Включен, если указано выполнение в фоновом режиме и при этом выключен флаг доступности порта связи, соответствующий номеру порта связи, указанному в *Com Port number (Номер COM-порта)* для *Background Execution (Выполнение в фоновом режиме)* (только для CP1H). Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если в D выдано значение 0000 hex. Выключен во всех остальных случаях.

Меры предосторожности

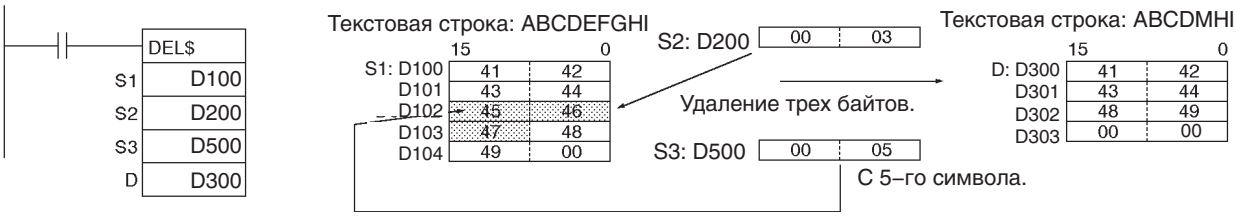
Максимальное количество символов в S1: 4095 (0FFF hex). При большем количестве символов (т. е. при отсутствии символа NUL до 4096-го символа) будет сгенерирована ошибка и установится флаг ошибки.

В качестве начального символа операндом S3 может быть указан любой символ строки с 1-го по 4095-й (0001...0FFF hex). При значении вне этого диапазона будет сгенерирована ошибка и установится флаг ошибки.

Если указанное количество удаляемых символов S2 превышает фактическое количество символов от позиции S3 до конца строки S1, будут удалены все символы от позиции S3 до конца строки. Если задано удаление всех символов от начала до конца строки S1, в операнд D будет выведено значение 000 hex.

Пример

В данном примере команда DEL$(658) используется для удаления трех символов.

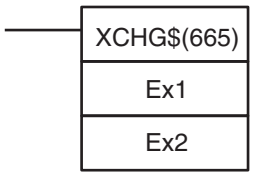


3-31-11 ОБМЕН СТРОКАМИ: XCHG$(665)

Назначение

Обмен значениями указанных строк.

Символ РКС



Ex1: Первое слово строки 1

Ex2: Первое слово строки 2

Варианты выполнения

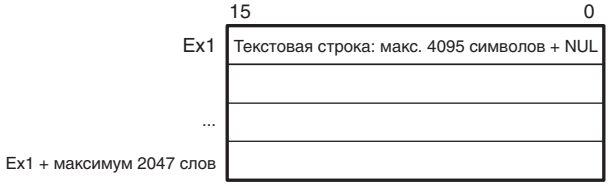
Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	XCHG$(665)
	Однократное выполнение по положительному фронту	@XCHG$(665)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

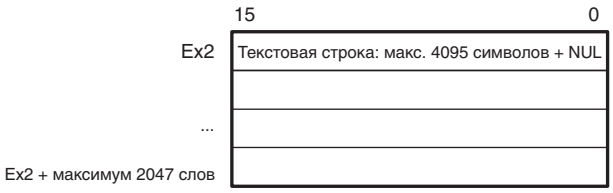
Области программных блоков	Области пошаговых программ	Подпро-граммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

Ex1: Первое слово строки 1



Ex2: Первое слово строки 2



Примечание.

(1) Данные в [Ex1...Ex1 + максимум 2047 слов] и [Ex2...Ex2 + максимум 2047 слов] должны принадлежать одной области.

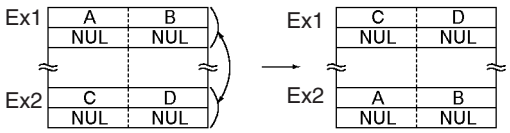
(2) Данные в [Ex1...Ex1 + максимум 2047 слов] и [Ex2...Ex2 + максимум 2047 слов] не должны перекрываться.

Характеристики операндов

Область	Ex1	Ex2
Область CIO	CIO 0...CIO 6143	
Рабочая область	W0...W511	
Область битов хранения	H0...H511	
Область вспомогательных битов	A448...A959	
Область таймеров	T0000...T4095	
Область счетчиков	C0000...C4095	
Область DM	D0...D32767	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Постоянные	---	
Регистры данных	---	
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...–2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

Описание

Команда XCHG$(665) записывает содержимое текстовой строки Ex1 в текстовую строку Ex2, а содержимое текстовой строки Ex2 — в текстовую строку Ex1. Если какая-либо из строк, Ex1 или Ex2, содержит только NUL, во вторую строку будут выведены два кода NUL (0000 hex).



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если операндом Ex1 или Ex2 определено больше 4095 символов. Включен, если данные Ex1 и Ex2 накладываются. Включен, если указано выполнение в фоновом режиме и при этом выключен флаг доступности порта связи, соответствующий номеру порта связи, указанному в *Com Port number (Номер COM-порта)* для *Background Execution (Выполнение в фоновом режиме)* (только для CP1H). Выключен во всех остальных случаях.

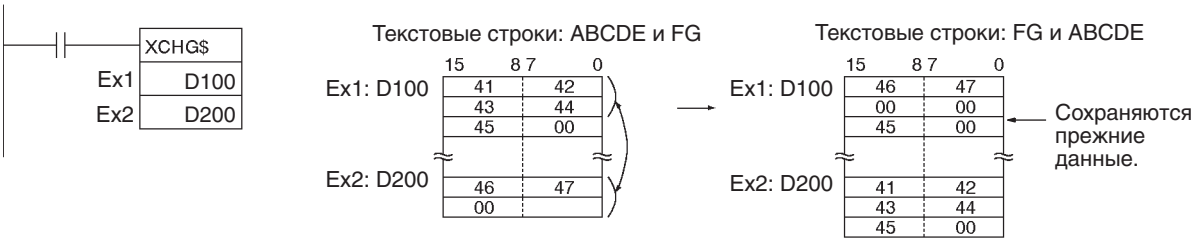
Меры предосторожности

Максимальное количество символов, которое может быть указано операндами Ex1 и Ex2: 4095 (0FFF hex). При большем значении будет сгенерирована ошибка и установится флаг ошибки.

Если текстовые строки, определенные операндами Ex1 и Ex2, накладываются друг на друга, будет сгенерирована ошибка и установится флаг ошибки.

Пример

В данном примере команда XCHG$(665) используется для обмена значениями текстовых строк.

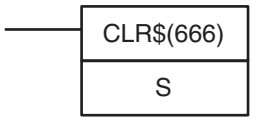


3-31-12 ОЧИСТИТЬ СТРОКУ: CLR$(666)

Назначение

Обнуление полностью всей текстовой строки.

Символ РКС



S: Первое слово текстовой строки

Варианты выполнения

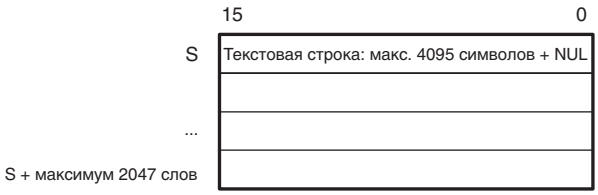
Варианты выполнения		
	Выполнение в каждом цикле при включенном условии выполнения	CLR$(666)
	Однократное выполнение по положительному фронту	@CLR$(666)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

S: Первое слово текстовой строки



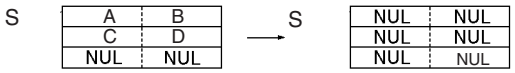
Примечание. Данные в [S...S + максимум 2047 слов] должны принадлежать одной области.

Характеристики операндов

Область	S
Область CIO	CIO 0...CIO 6143
Рабочая область	W0...W511
Область битов хранения	H0...H511
Область вспомогательных битов	A448...A959
Область таймеров	T0000...T4095
Область счетчиков	C0000...C4095
Область DM	D0...D32767
Косвенные адреса DM в двоичном формате	@ D0...@ D32767
Косвенные адреса DM в BCD-формате	*D0...*D32767
Постоянные	---
Регистры данных	---
Регистры указателей	---
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...–2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15

Описание

Команда CLR$(666) обнуляет (заменяет значениями NUL (00 hex)) всю текстовую строку, начиная с первого слова, указанного операндом S, до кода NUL (00 hex). Максимальное количество символов, которое может быть очищено: 4096. Если в пределах 4096 символов отсутствует код NUL, обнуляются только 4096 символов.

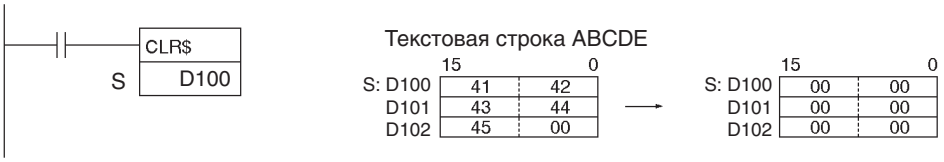


Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если указано выполнение в фоновом режиме и при этом выключен флаг доступности порта связи, соответствующий номеру порта связи, указанному в *Com Port number (Номер COM-порта)* для *Background Execution (Выполнение в фоновом режиме)* (только для CP1H). Выключен во всех остальных случаях.

Пример

В данном примере команда CLR$(666) используется для обнуления текстовой строки ABCDE.

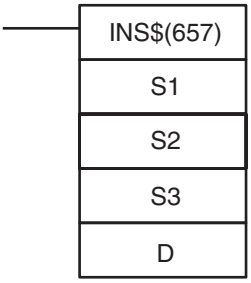


3-31-13 ВСТАВИТЬ В СТРОКУ: INS$(657)

Назначение

Вставка одной текстовой строки в указанную позицию другой текстовой строки.

Символ РКС



S1: Первое слово основной текстовой строки

S2: Первое слово вставляемой текстовой строки

S3: Начальная позиция

D: Первое слово назначения

Варианты выполнения

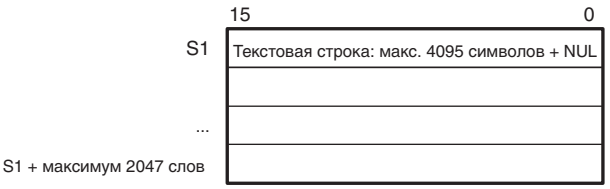
Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	INS$(657)
	Однократное выполнение по положительному фронту	@INS$(657)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

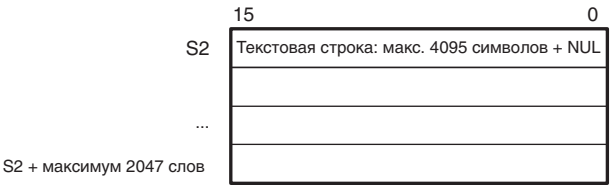
Области программных блоков	**Области пошаговых программ**	**Подпро-граммы**	**Задачи обработки прерываний**
OK	OK	OK	OK

Операнды

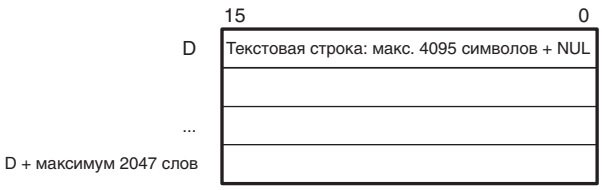
S1: Основная текстовая строка



S2: Вставляемая текстовая строка



S3: Начальная позиция (0001...0FFF hex или &1...&4095)



Примечание.

(1) Данные в [S1...S1 + максимум 2047 слов], [S2...S2 + максимум 2047 слов] и [D...D + максимум 2047 слов] должны принадлежать одной области.

(2) Данные в [S2...S2 + максимум 2047 слов] и [D...D + максимум 2047 слов] не должны перекрываться. Данные в [S1...S1 + максимум 2047 слов] и [D...D + максимум 2047 слов] могут перекрываться. Данные в [S1...S1 + максимум 2047 слов] и [S2...S2 + максимум 2047 слов] также могут перекрываться.

Характеристики операндов

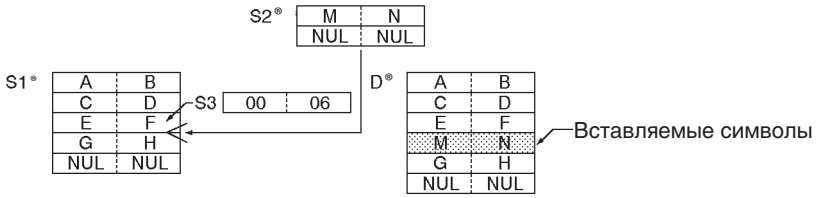
Область	S1	S2	S3	D
Область CIO	CIO 0...CIO 6143			
Рабочая область	W0...W511			
Область битов хранения	H0...H511			
Область вспомогательных битов	A0...A447 A448...A959			A448...A959
Область таймеров	T0000...T4095			
Область счетчиков	C0000...C4095			
Область DM	D0...D32767			
Косвенные адреса DM в двоичном формате	@ D0...@ D32767			
Косвенные адреса DM в BCD-формате	*D0...*D32767			
Постоянные	---		#0000...#0FFF (двоичный) или &0...&4095	---
Регистры данных	---		DR0...DR15	---
Регистры указателей	---			
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...–2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15			

Описание

Команда INS$(657) вставляет текстовую строку, указанную операндом S2, в текстовую строку, указанную операндом S1, после начального слова, указанного операндом S3, и выводит результат в операнд D в формате текстовой строки (в конце добавляется код NUL).

Максимальное количество символов, которое может быть вставлено: 4095 (0FFF hex). Если число символов больше, в результат будет выведено только 4095 символов (в качестве 4096-го символа будет добавлен код NUL).

Если одна из строк, S1 или S2, содержит только NUL, в D выводится вторая (ненулевая) текстовая строка без изменений. Если обе строки, S1 и S2, содержат только NUL, в D выводятся два символа NUL (0000 hex).



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если операндом S1 или S2 определено больше 4095 символов. Включен, если S3 больше 4095 (0FFF hex). Включен, если указано выполнение в фоновом режиме и при этом выключен флаг доступности порта связи, соответствующий номеру порта связи, указанному в *Com Port number (Номер COM-порта)* для *Background Execution (Выполнение в фоновом режиме)* (только для CP1H). Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если в D выдано значение 0000 (hex). Выключен во всех остальных случаях.

Меры предосторожности

Максимальное количество символов в S1 и S2: 4095 (0FFF hex). При большем количестве символов (т. е. при отсутствии символа NUL до 4096-го символа) будет сгенерирована ошибка и установится флаг ошибки.

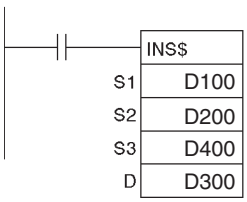
В качестве начального символа операндом S3 может быть указан любой символ строки с 1-го по 4095-й. Если указанное значение не попадает в этот диапазон, будет сгенерирована ошибка и установится флаг ошибки.

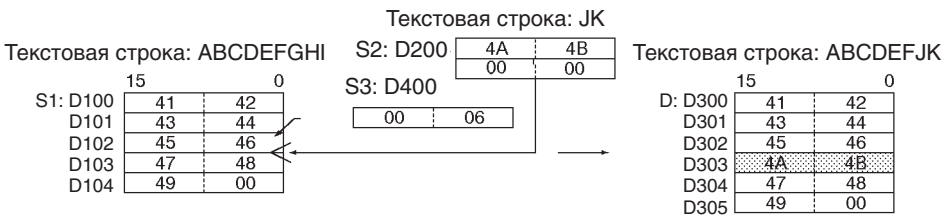
Если в D будет выдано значение 0000 (hex), установится флаг равенства.

Адресуемые слова, определенные операндом D, не должны накладываться на данные текстовой строки, указанные операндом S2. При наложении корректное выполнение команды невозможно.

Пример

В данном примере команда INS$(657) используется для вставки двух символов.





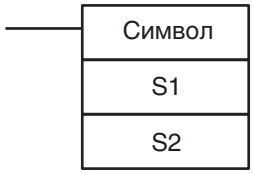
3-31-14 Команды сравнения строк (670..675)

Назначение

Команды сравнения строк (=$, <>$, <$, <=$, >$, >=$) служат для посимвольного сравнения ASCII-кодов двух текстовых строк, начиная с первого символа. При положительном результате сравнения включается условие выполнения (для LOAD, AND или OR).

Символ РКС

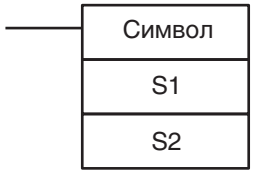
LD (загрузка)



S1: Текстовая строка 1

S2: Текстовая строка 2

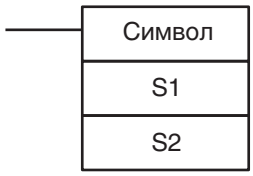
AND (последовательное соединение)



S1: Текстовая строка 1

S2: Текстовая строка 2

OR (параллельное соединение)



S1: Текстовая строка 1

S2: Текстовая строка 2

Варианты выполнения

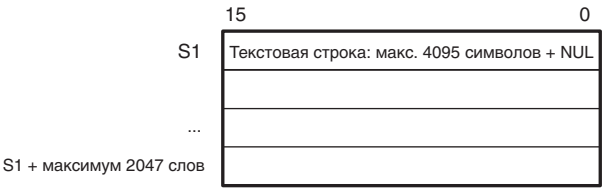
Варианты выполнения	**Включение условия выполнения в каждом цикле, в котором результат сравнения = ИСТИНА.**	Команды сравнения строк
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

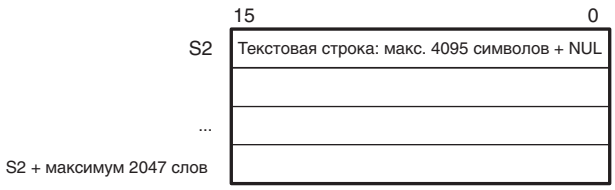
Области программных блоков	Области пошаговых программ	Подпро-граммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

S1: Текстовая строка 1



S2: Текстовая строка 2



Примечание.

(1) Данные в [S1...S1 + максимум 2047 слов] и [S2...S2 + максимум 2047 слов] должны принадлежать одной области.

(2) Данные в [S1...S1 + максимум 2047 слов] и [S2...S2 + максимум 2047 слов] не должны перекрываться.

Характеристики операндов

Область	S1	S2
Область CIO	CIO 0...CIO 6143	
Рабочая область	W0...W511	
Область битов хранения	H0...H511	
Область вспомогательных битов	A0...A447 A448...A959	
Область таймеров	T0000...T4095	
Область счетчиков	C0000...C4095	
Область DM	D0...D32767	
Косвенные адреса DM в двоичном формате	@ D0...@ D32767	
Косвенные адреса DM в BCD-формате	*D0...*D32767	
Постоянные	---	
Регистры данных	---	
Регистры указателей	---	
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048...+2047 ,IR0...–2048...+2047 ,IR15 DR0...DR15, IR0...IR15 ,IR0+(++)...,IR15+(++) ,–(– –)IR0..., –(– –)IR15	

Описание

Команды сравнения текстовых строк сравнивают текстовые строки, указанные операндами S1 и S2. В случае положительного результата сравнения в лестничной диаграмме включается условие выполнения. Максимальное количество символов в S1 и S2: 4095 (0FFF hex).

Ниже приведены 18 разных мнемонических кодов команд сравнения (LD, AND и OR в лестничной диаграмме не отображаются).

LD=$, AND=$, OR=$
LD<>$, AND<>$, OR<>$
LD<$, AND<$, OR<$
LD<=$, AND<=$, OR<=$
LD>$, AND>$, OR>$
LD>=$, AND>=$, OR>=$

Описание этих команд представлено в следующей таблице.

Мнемонический код (включая код функции)	Название	Функция
LD=$(670)	ЗАГРУЗКА СТРОКА РАВНА	Истина, если текстовая строка S1 равна текстовой строке S2.
AND=$(670)	И СТРОКА РАВНА	
OR=$(670)	ИЛИ СТРОКА РАВНА	

Мнемонический код (включая код функции)	Название	Функция
LD<>$(671)	ЗАГРУЗКА СТРОКА НЕ РАВНА	Истина, если текстовая строка S1 не равна текстовой строке S2.
AND<>$(671)	И СТРОКА НЕ РАВНА	
OR<>$(671)	ИЛИ СТРОКА НЕ РАВНА	
LD<$(672)	ЗАГРУЗКА СТРОКА МЕНЬШЕ	Истина, если текстовая строка S1 меньше текстовой строки S2.
AND<$(672)	И СТРОКА МЕНЬШЕ	
OR<$(672)	ИЛИ СТРОКА МЕНЬШЕ	
LD<=$(673)	ЗАГРУЗКА СТРОКА МЕНЬШЕ ИЛИ РАВНА	Истина, если текстовая строка S1 меньше или равна текстовой строке S2.
AND<=$(673)	И СТРОКА МЕНЬШЕ ИЛИ РАВНА	
OR<=$(673)	ИЛИ СТРОКА МЕНЬШЕ ИЛИ РАВНА	
LD>$(674)	ЗАГРУЗКА СТРОКА БОЛЬШЕ	Истина, если текстовая строка S1 больше текстовой строки S2.
AND>$(674)	И СТРОКА БОЛЬШЕ	
OR>$(674)	ИЛИ СТРОКА БОЛЬШЕ	
LD>=$(675)	ЗАГРУЗКА СТРОКА БОЛЬШЕ ИЛИ РАВНА	Истина, если текстовая строка S1 больше или равна текстовой строке S2.
AND>=$(675)	И СТРОКА БОЛЬШЕ ИЛИ РАВНА	
OR>=$(675)	ИЛИ СТРОКА БОЛЬШЕ ИЛИ РАВНА	

Методика сравнения

Сравнение выполняется описанным ниже образом.

Текстовые строки сравниваются посимвольно. Сначала сравниваются между собой ASCII-коды первых символов (байтов) каждой текстовой строки. Если они не равны, их соотношение (больше/меньше) принимается за соотношение текстовых строк. Если два первых ASCII-кода равны, сравниваются следующие символы. Если они не равны, их соотношение (больше/меньше) принимается за соотношение текстовых строк.

Таким образом, все символы сравниваются поочередно, пока не обнаруживается расхождение, на основании которого и принимается решение о соотношении двух строк. В том случае если все символы в двух строках, включая коды NUL, совпадают, две текстовые строки считаются равными.

Если две текстовые строки имеют разную длину, в целях сравнения более короткая строка дополняется символами NUL (00 hex).

Примеры сравнения

AD (414400 hex) и BC (424300 hex):

AD < BC, так как первый символ левой строки 41 (hex) меньше первого символа правой строки 42 (hex).

ADC (41444300 hex) и B (4200 hex):

ADC < B, так как первый символ левой строки 41 (hex) меньше первого символа правой строки 42 (hex).

ABC (41424300 hex) и ABD (41424400 hex):

ABC < ABD, так как строки имеют одинаковое начало (4142), но третий символ левой строки 43 (hex) меньше третьего символа правой строки 44 (hex).

ABC (41424300 hex) и AB (414200 hex):
ABC > AB, так как начало текстовых строк (4142) совпадает, но третий символ левой строки 43 (hex) больше третьего символа правой строки 00 (hex).
AB (414200 hex) и AB (414200 hex):
AB = AB, так как все ASCII-коды совпадают.

Попробуйте использовать эти команды в программе в вариантах LD, AND и OR. Команды LD и OR можно напрямую подключать к левой шине, команды AND к левой шине не подключаются.

Флаги

Название	Обо-значение	Описание работы
Флаг ошибки	ER	Включен, если операндом S1 или S2 определено больше 4095 символов. Выключен во всех остальных случаях.
Флаг большего значения	>	Включен, если результат сравнения: S1 больше S2. Выключен во всех остальных случаях.
Флаг большего значения или равенства	>=	Включен, если результат сравнения: S1 больше или равно S2. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если результат сравнения: S1 равно S2. Выключен во всех остальных случаях.
Флаг неравенства	<>	Включен, если результат сравнения: S1 не равно S2. Выключен во всех остальных случаях.
Флаг меньшего значения	<	Включен, если результат сравнения: S1 меньше S2. Выключен во всех остальных случаях.
Флаг меньшего значения или равенства	<=	Включен, если результат сравнения: S1 меньше или равно S2. Выключен во всех остальных случаях.

Примечание. Команды сравнения строк служат для изменения порядка текстовых строк и расположения их по убыванию или возрастанию ASCII-кодов. Например, расположение по возрастанию ASCII-кодов соответствует сортировке в алфавитном порядке от A до Z, что позволяет упорядочить текстовые строки в алфавитном порядке.

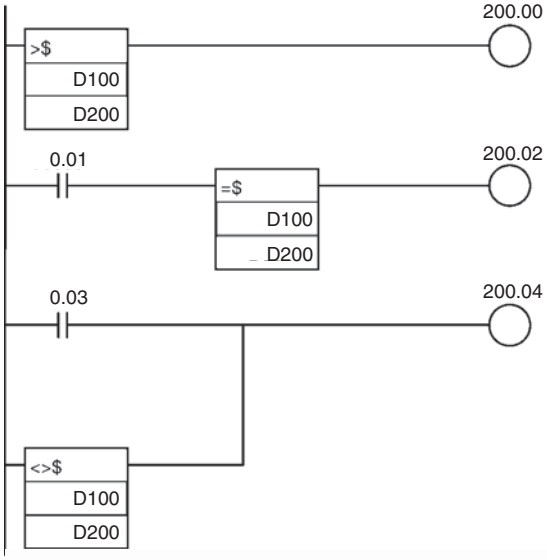
Меры предосторожности Используйте «правостороннюю» команду после этих команд. Команды сравнения строк не могут подключаться непосредственно к правой шине лестничной диаграммы.

Эти команды невозможно использовать в последней цепи логического блока.

Максимальное количество сравниваемых символов: 4095 (0FFF hex). При большем количестве символов (т. е. при отсутствии символа NUL до 4096-го символа) будет сгенерирована ошибка и установится флаг ошибки. В таком случае условие выполнения следующей команды будет выключено.

Пример

В данном примере команды сравнения строк используются для сравнения данных.



Адрес	Мнемонич. обозначение	Операнд
000000	LD > $	--- D100 D200
000001	OUT	200.00
000002	LD	0.01
000003	AND=$	--- D100 D200
000004	OUT	200.02
000005	LD	0.03
000006	OR <> $	--- D100 D200
000007	OUT	200.04

> $	= $	<>$
D100	D100	D100
D200	D200	D200

Текстовая строка ABCD

D100	41	42
D101	44	43
D102	00	00

Текстовая строка ABC

D200	41	42
D201	43	00

> $	= $	<>$
ВКЛ	ВЫКЛ	ВКЛ

Текстовая строка ABC

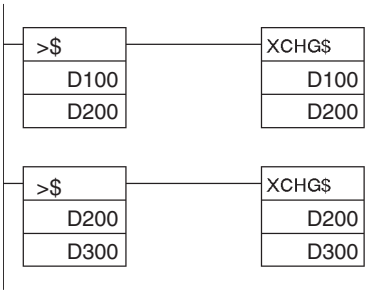
D100	41	42
D101	43	00

Текстовая строка ABC

D200	41	42
D201	43	00

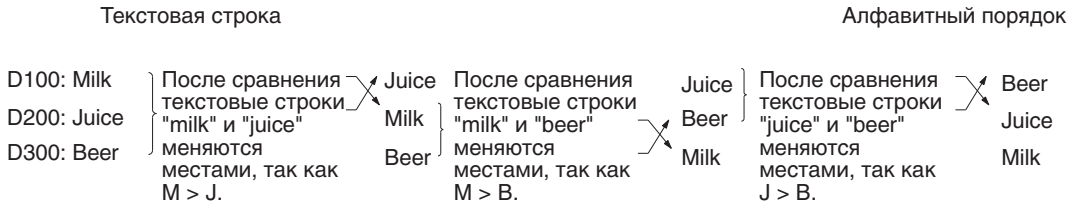
> $	= $	<>$
ВЫКЛ	ВКЛ	ВЫКЛ

В данном примере три текстовые строки упорядочиваются в алфавитном порядке. Исходный порядок таков:
D100: Milk
D200: Juice
D300: Beer
После упорядочивания порядок строк будет следующим: Beer, Juice, Milk.



Две текстовые строки с начальными адресами D100 и D200 сравниваются посимвольно до обнаружения первого несовпадающего символа. Если ASCII–код такого символа в строке с начальным адресом D100 оказывается больше ASCII–кода символа в строке с начальным адресом D200, строки меняются местами.

Две текстовые строки с начальными адресами D200 и D300 сравниваются посимвольно до обнаружения первого несовпадающего символа. Если ASCII–код такого символа в строке с начальным адресом D200 оказывается больше ASCII–кода символа в строке с начальным адресом D300, строки меняются местами.



Таким образом, три текстовые строки могут быть упорядочены в алфавитном порядке.

3-32 Команды управления задачами

В данном разделе описаны команды, используемые для управления задачами.

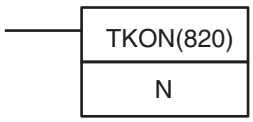
Команда	Мнемоническое обозначение	Код функции	Стр.
ВКЛЮЧИТЬ ЗАДАЧУ	TKON	820	1137
ВЫКЛЮЧИТЬ ЗАДАЧУ	TKOF	821	1141

3-32-1 ВКЛЮЧИТЬ ЗАДАЧУ: TKON(820)

Назначение

Переводит указанную задачу в состояние «выполнимая задача». Также инициирует выполнение задачи обработки прерывания как дополнительной циклической задачи.

Символ РКС



N: Номер задачи

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	TKON(820)
	Однократное выполнение по положительному фронту	@TKON(820)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпро-граммы	Задачи обработки прерываний
OK	OK	OK	Не допускается

Операнды

N: Номер задачи

Диапазон допустимых значений N зависит от указанного типа задачи.

- Циклические задачи:
  N должно быть десятичной константой от 0 до 31 (значения от 0 до 31 соответствуют циклическим задачам от 0 до 31).
- Дополнительные циклические задачи:
  N должно быть десятичной константой от 8000 до 8255 (значения от 8000 до 8255 соответствуют дополнительным циклическим задачам от 0 до 255).

Характеристики операндов

Область	N
Область CIO	---
Рабочая область	---
Область битов хранения	---
Область вспомогательных битов	---
Область таймеров	---
Область счетчиков	---
Область DM	---
Косвенные адреса DM в двоичном формате	---
Косвенные адреса DM в BCD-формате	---
Постоянные	00…31 или 8000…8255 (десятичн.)
Регистры данных	---
Регистры указателей	---
Косвенная адресация с помощью регистров указателей	---

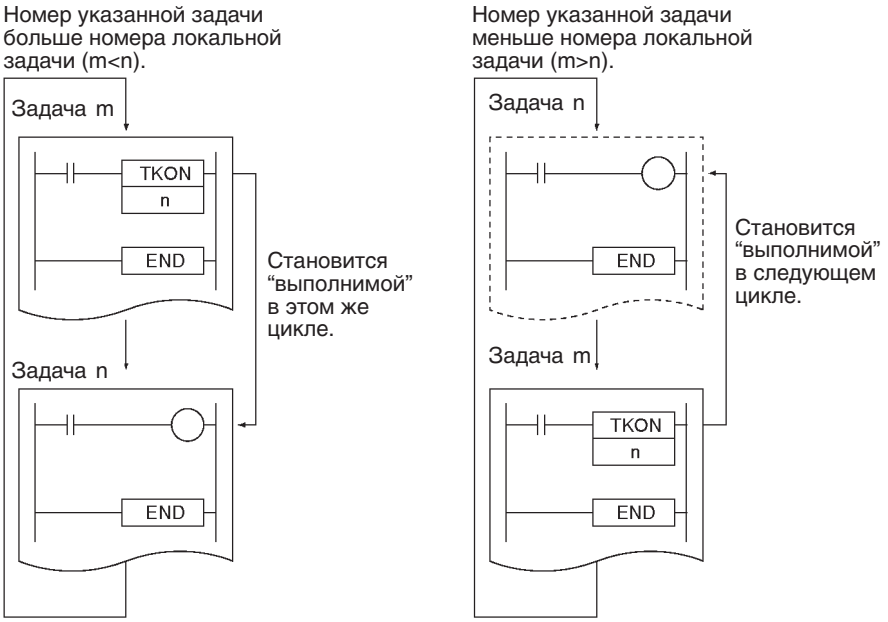
Описание

TKON(820) переводит указанную циклическую задачу или дополнительную циклическую задачу в состояние «выполнимая задача». Если N содержит значение от 0 до 31 (указывает циклическую задачу), одновременно включается флаг соответствующей задачи (TK00...TK31).

Эта команда может быть выполнена только в обычной циклической задаче или дополнительной циклической задаче. При попытке выполнить ее в задаче обработки прерывания возникнет ошибка.

Циклическая задача или дополнительная циклическая задача, указанная в TKON(820), также будет иметь статус «выполнимой» в следующих циклах, пока не будет переведена в состояние ожидания командой TKOF(821).

В любой циклической задаче можно перевести любую задачу в состояние «выполнимая», хотя указанная задача будет выполняться только в следующем цикле, если ее номер меньше номера текущей задачи. Но если ее номер больше номера текущей задачи, она будет выполнена в этом же цикле.



Команда TKON(820) будет восприниматься как команда NOP(000), если указанная задача уже находится в состоянии «выполнимая» или указана локальная задача.

С помощью команды TKOF(821), программы CX-Programmer или команды FINS задачу из состояния «выполнимая» можно перевести в состояние «ожидание».

Состояния «выполнимая задача» и «выполняемая задача» — это не одно и то же. Задачи, имеющие статус «выполнимая задача», выполняются по очереди (в порядке возрастания своих номеров) в ходе циклического выполнения программы контроллера. Если «выполнимая» задача будет переведена в состояние ожидания раньше, чем до нее дойдет очередь, она выполнена не будет.

Примечание.

(1) В программе CX-Programmer на вкладке *General Properties (Общие свойства)* для каждой задачи предусмотрен параметр *Operation start (Начало работы)*, указывающий, является ли циклическая задача «выполнимой» с самого начала, то есть при запуске ПЛК. Циклические задачи, для которых флажок *Operation start (Начало работы)* установлен, автоматически получат статус «выполнимая задача» в начале работы ПЛК. (Щелкнуть правой кнопки мыши на выбранной задачи и выбрать вкладку «Properties». В ней можно добавить, либо снять флажок «Operation start».) Все остальные циклические задачи будут иметь статус «невыполнимая задача».

(2) «Невыполнимую» задачу можно сделать «выполнимой» с помощью команды TKON(820). И наоборот, с помощью команды TKOF(821) циклическую задачу из состояния «выполнимая» можно перевести в состояние «невыполнимая».

(3) Циклические задачи или дополнительные циклические задачи, сделанные «выполнимыми», переходят в это состояние в текущем цикле в порядке возрастания своих номеров. Следовательно, если задача будет переведена в состояния ожидания раньше, чем до нее

дойдет очередь, такая задача в данном цикле выполнена не будет (так как задачи выполняются в порядке возрастания номеров).

Флаги

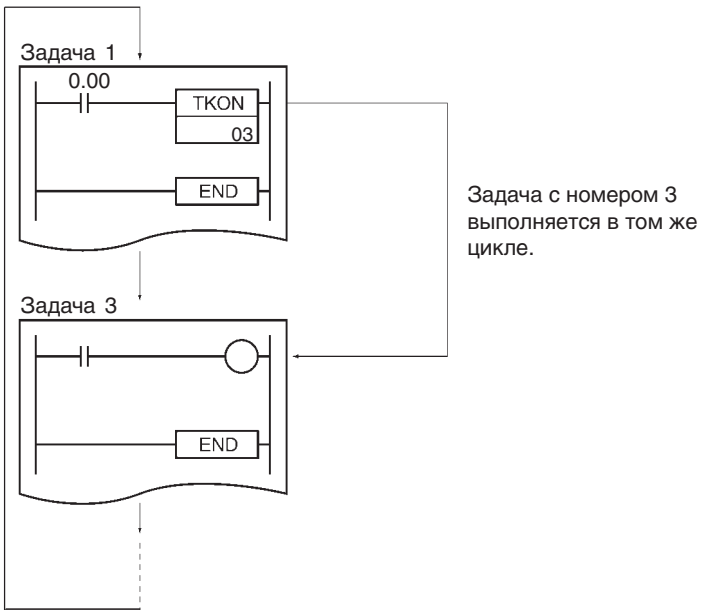
Название	Обо-значение	Описание работы
Флаг ошибки	ER	Включен, если N не является десятичной константой от 00 до 31 или от 8000 до 8255. Включен, если задачи с указанным номером N не существует. Включен, если команда TKON(820) выполняется в задаче обработки прерывания. Выключен во всех остальных случаях.

Название	Адреса	Описание работы
Флаги задач	TK00...TK31	Флаг включен, если соответствующая ему циклическая задача является «выполнимой». Флаг выключен, если задача не является «выполнимой» (то есть является «невыполнимой») или находится в состоянии ожидания. TK00...TK31 соответствуют номерам циклических задач 00...31.

Примеры

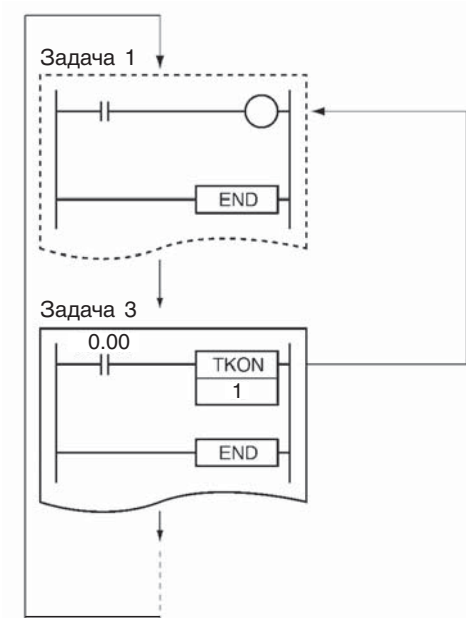
Указание последующей задачи

Если в следующем примере бит CIO 0.00 включен, задача под номером 3 переводится в состояние «выполнимая» задачей под номером 1. Задача 3 будет выполнена в том же цикле, когда до нее дойдет очередь при обработке программы.



Указание предшествующей задачи

Если в следующем примере бит CIO 0.00 включен, задача под номером 1 переводится в состояние «выполнимая» задачей под номером 3. Задача 1 будет выполнена в следующем цикле, когда до нее дойдет очередь при обработке программы.

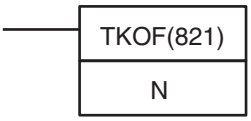


3-32-2 ВЫКЛЮЧИТЬ ЗАДАЧУ: TKOF(821)

Назначение

Переводит указанную циклическую задачу или дополнительную циклическую задачу в состояние ожидания, т. е. отключает выполнение задачи.

Символ РКС



N: Номер задачи

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	TKOF(821)
	Однократное выполнение по положительному фронту	@TKOF(821)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпро-граммы	Задачи обработки прерываний
OK	OK	OK	Не допускается

Операнды

N: Номер задачи

Диапазон допустимых значений N зависит от указанного типа задачи.

- Циклические задачи:
  N должно быть десятичной константой от 0 до 31 (значения от 0 до 31 соответствуют циклическим задачам от 0 до 31).

- Дополнительные циклические задачи:
  N должно быть десятичной константой от 8000 до 8255 (значения от 8000 до 8255 соответствуют дополнительным циклическим задачам от 0 до 255).

Характеристики операндов

Область	N
Область CIO	---
Рабочая область	---
Область битов хранения	---
Область вспомогательных битов	---
Область таймеров	---
Область счетчиков	---
Область DM	---
Косвенные адреса DM в двоичном формате	---
Косвенные адреса DM в BCD-формате	---
Постоянные	00…31 или 8000…8255 (десятичн.)
Регистры данных	---
Регистры указателей	---
Косвенная адресация с помощью регистров указателей	---

Описание

Команда TKOF(821) переводит указанную циклическую задачу или дополнительную циклическую задачу в состояние ожидания и выключает соответствующий флаг задачи (TK00...TK31).

Задача, указанная в TKOF(821), будет оставаться в состоянии ожидания в следующих циклах до переключения в состояние «выполнимая» командой TKON(820), из CX-Programmer или командой FINS.

Задача может быть переведена в режим ожидания из любой другой обычной задачи, но переключение в режим ожидания указанной задачи произойдет только в следующем цикле, если номер этой задачи будет меньше номера локальной задачи (на тот момент она уже будет выполнена в текущем цикле). Но если ее номер больше номера локальной задачи, она будет переключена в состояние ожидания в этом же цикле.

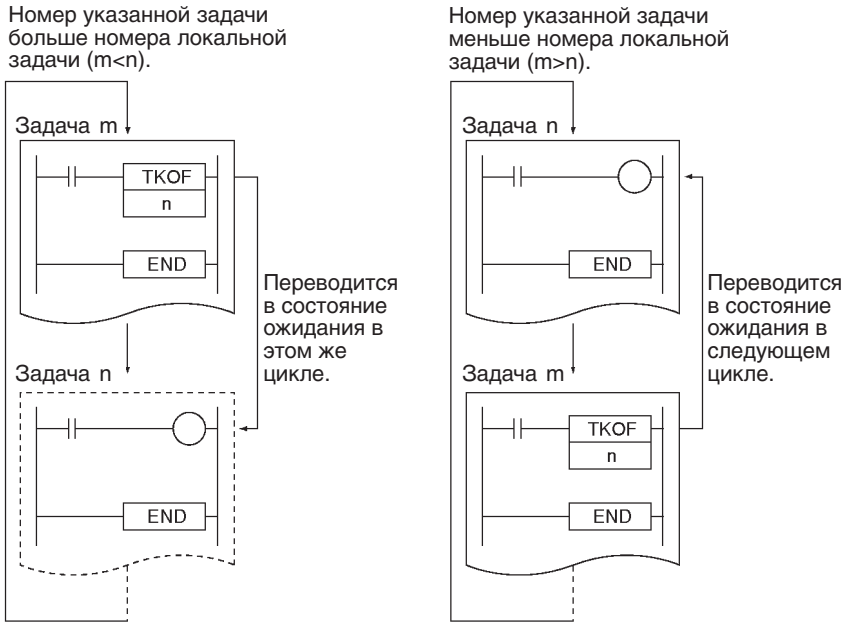
Если в команде TKOF(821) будет указана локальная задача, она будет переведена в состояние ожидания немедленно и следующие команды этой задачи выполняться не будут.

Примечание.

(1) В программе CX-Programmer на вкладке *General Properties (Общие свойства)* для каждой задачи предусмотрен параметр *Operation start (Начало работы)*, указывающий, является ли циклическая задача «выполнимой» с самого начала, то есть при запуске ПЛК. Циклические задачи, для которых флажок *Operation start (Начало работы)* установлен, автоматически получат статус «выполнимая задача» в начале работы ПЛК. Все остальные циклические задачи будут иметь статус «невыполнимая задача».

(2) «Невыполнимую» задачу можно сделать «выполнимой» с помощью команды TKON(820). И наоборот, с помощью команды TKOF(821) циклическую задачу из состояния «выполнимая» можно перевести в состояние «невыполнимая».

(3) Циклические задачи и дополнительные циклические задачи в состоянии «выполнимая» могут быть переключены в состояние ожидания командой TKOF(821).



Обычная задача, для которой настроено выполнение при запуске, переводится в состояние «выполнимая» автоматически в начале работы ПЛК. Все остальные обычные задачи будут иметь статус «невыполнимая задача».

С помощью команды TKOF(821), программы CX-Programmer или команды FINS задачу из состояния «выполнимая» можно перевести в состояние «ожидание».

Состояния «выполнимая задача» и «выполняемая задача» — это не одно и то же. Задачи, имеющие статус «выполнимая задача», выполняются по очереди (в порядке возрастания своих номеров) в ходе циклического выполнения программы контроллера. Если «выполнимая» задача будет переведена в состояние ожидания раньше, чем до нее дойдет очередь, она выполнена не будет.

В отличие от команды TKON(820), эту команду можно использовать не только в циклических задачах, но также и в задачах обработки прерываний.

Флаги

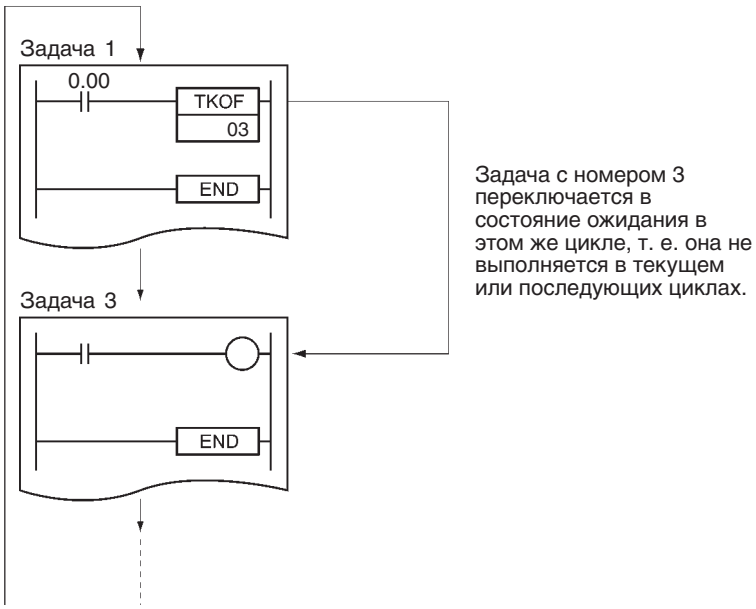
Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если N не является десятичной константой от 00 до 31 или от 8000 до 8255. Включен, если задачи с указанным номером N не существует. Включен, если команда TKOF(821) выполняется в задаче обработки прерывания. Выключен во всех остальных случаях.

Название	Адреса	Описание работы
Флаги задач	TK00...TK31	Флаг включен, если соответствующая ему циклическая задача является «выполнимой». Флаг выключен, если задача не является «выполнимой» (то есть является «невыполнимой») или находится в состоянии ожидания. TK00...TK31 соответствуют номерам циклических задач 00...31.

Примеры

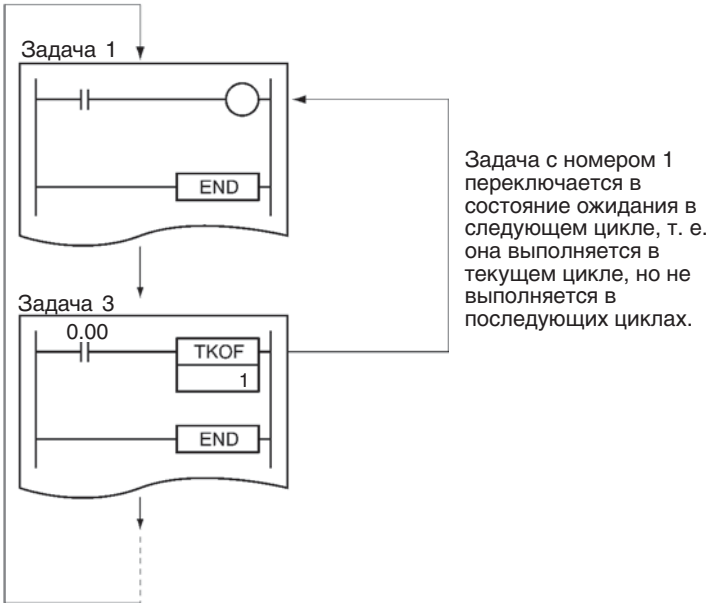
Указание последующей задачи

Если в следующем примере бит CIO 0.00 включен, задача под номером 3 переводится в состояние ожидания задачей под номером 1. Задача 3 не будет выполняться в этом цикле, когда до нее дойдет очередь при обработке программы.



Указание предшествующей задачи

Если в следующем примере бит CIO 0.00 включен, задача под номером 1 переводится в состояние ожидания задачей под номером 3. Задача 1 не будет выполняться в следующем цикле, когда до нее дойдет очередь при обработке программы.



3-33 Команды преобразования моделей

В данном разделе описаны команды, используемые при смене моделей ПЛК.

Команда	Мнемоническое обозначение	Код функции	Стр.
ПЕРЕДАЧА БЛОКА	XFERC	565	1148
РАЗМЕЩЕНИЕ ОДИНОЧНОГО СЛОВА	DISTC	566	1150
СБОР ДАННЫХ	COLLC	567	1154
ПЕРЕМЕЩЕНИЕ БИТА	MOVBC	568	1158
ПОДСЧЕТ БИТОВ	BCNTC	621	1160

Команды преобразования моделей предоставляют такую же функциональность, как и другие команды, но используют в качестве операндов данные в двоично-десятичном формате (BCD) подобно командам серии C. (В серии CP используются операнды в двоичном формате.) Предусмотрено пять команд преобразования моделей (перечисленных в таблице выше). Мнемонический код каждой из них состоит из мнемонического кода аналогичной функции, использующей операнды в двоичном формате, и буквы C в конце кода.

С помощью команд преобразования моделей программы, написанные для ПЛК серии C, могут быть конвертированы в программы для ПЛК серии CP без изменения операндов в этих командах.

При преобразовании программ серии C в программы серии CP в CX-Programmer (см. прим.) происходит автоматическая замена команд (например, команда XFER будет заменена командой XFERC), что избавляет пользователя от необходимости менять операнды вручную.

Примечание. Чтобы выполнить преобразование, нужно выбрать в поле «тип устройства» диалогового окна Change PLC (Смена ПЛК) устройство серии CP.

Отличия от команд серии C

К устройствам серии C относятся модули C200H, C1000H, C2000H, C200HS, C2000HX/HG/HE(-Z), CQM1, CQM1H, CPM1/CPM1A, CPM2C и SRM1.

Название	Команда преобразования моделей	Соответствующая команда серии C	Отличия от команд серии C		При преобразовании в устройство серии CP с помощью CX-Programmer
	Мнемоническое обозначение (код функции)	Мнемоническое обозначение (код функции)	C200H, C1000H или C2000H	C200HS, C2000HX/HG/HE(-Z), CQM1, CQM1H, CPM1/CPM1A, CPM2C или SRM1	
ПЕРЕДАЧА БЛОКА	XFERC(565)	XFER(70)	Отличий нет	Отличий нет	XFER меняется на XFERC. Операнды изменять не требуется.
РАЗМЕЩЕНИЕ ОДИНОЧНОГО СЛОВА	DISTC(566)	DIST(80)	Помимо операции размещения значений выполняет операцию сдвига стека, которая ранее не поддерживалась.	Отличий нет (операция размещения значений и операция сдвига стека)	DIST меняется на DISTC. Операнды изменять не требуется.

Название	Команда преобразования моделей	Соответствующая команда серии C	Отличия от команд серии C		При преобразовании в устройство серии CP с помощью CX-Programmer
	Мнемоническое обозначение (код функции)	Мнемоническое обозначение (код функции)	C200H, C1000H или C2000H	C200HS, C2000HX/HG/HE(-Z), CQM1, CQM1H, CPM1/CPM1A, CPM2C или SRM1	
СБОР ДАННЫХ	COLLC(567)	COLL(81)	Помимо операции сбора данных выполняет операцию чтения из стека, которая ранее не поддерживалась.	Отличий нет (операция сбора данных и операция чтения из стека)	COLL меняется на COLLC. Операнды изменять не требуется.
ПЕРЕМЕЩЕНИЕ БИТА	MOVBC(568)	MOVB(82)	Отличий нет	Отличий нет	MOVB меняется на MOVBC. Операнды изменять не требуется.
ПОДСЧЕТ БИТОВ	BCNTC(621)	BCNT(67)	Отличий нет	Отличий нет	BCNT меняется на BCNTC. Операнды изменять не требуется.

Примечание. В работе флагов условий имеются указанные ниже отличия. Подробные сведения смотрите в описании флагов условий для каждой команды.

- Работа флагов условий отличается для всех команд в том случае, если содержимое слов области DM, используемых для косвенной адресации, представлено не в двоично-десятичном (*BCD), а в двоичном формате, или оказался превышен предельный адрес области DM.
- Для команды DISTC(566) отличие в работе флагов условий по сравнению с моделями C200H, C1000H и C2000H заключается в наличии операции сдвига стека.
- Для команды COLLC(567) отличие в работе флагов условий по сравнению с моделями C200H, C1000H и C2000H заключается в наличии операции чтения из стека.

Отличия от стандартных команд серии CP

Название	Команда преобразования моделей	Соответствующая стандартная команда серии CP	Отличия от стандартных команд серии CP (см. прим.)
	Мнемоническое обозначение (код функции)	Мнемоническое обозначение (код функции)	
ПЕРЕДАЧА БЛОКА	XFERC(565)	XFER(70)	Тип данных первого операнда (число передаваемых слов) двоично-десятичный (0000...9999), а не двоичный (0000...FFFF hex).
РАЗМЕЩЕНИЕ ОДИНОЧНОГО СЛОВА	DISTC(566)	DIST(80)	Помимо операции размещения значений поддерживается операция сдвига стека. Тип данных третьего операнда (величина смещения) двоично-десятичный (размещение значений: 0000...7999, сдвиг стека: 0000... 9999), а не двоичный (0000...FFFF hex).

Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

Название	Команда преобразования моделей	Соответствующая стандартная команда серии CP	Отличия от стандартных команд серии CP (см. прим.)
	Мнемоническое обозначение (код функции)	**Мнемоническое обозначение (код функции)**	
СБОР ДАННЫХ	COLLC(567)	COLL(81)	Помимо операции размещения значений поддерживается операция чтения из стека. Тип данных второго операнда (величина смещения) двоично-десятичный (размещение значений: 0000...7999, чтение из стека по принципу FIFO: 9000...9999, чтение из стека по принципу LIFO: 8000...8999), а не двоичный (0000...FFFF hex).
ПЕРЕМЕЩЕНИЕ БИТА	MOVBC(568)	MOVB(82)	Тип данных для указания исходного и адресуемого битов во втором операнде (управляющие данные) двоично-десятичный (00...15), а не двоичный (00...0F hex).
ПОДСЧЕТ БИТОВ	BCNTC(621)	BCNT(67)	Тип данных первого операнда (число слов для подсчета битов) двоично-десятичный (0000...9999), а не двоичный (0000...FFFF hex). Тип данных третьего операнда (результат подсчета) двоично-десятичный (0000...9999), а не двоичный (0000...FFFF hex).

Примечание. В работе флагов условий имеются указанные ниже отличия. Подробные сведения смотрите в описании флагов условий для каждой команды.

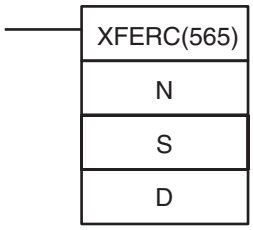
- Флаг ошибки включается, если данные указанных выше операндов представлены не в двоично-десятичном формате.
- Для команды DISTC(566) работа флагов условий расширена на операцию сдвига стека.
- Для команды COLLC(567) работа флагов условий расширена на операцию чтения из стека.

3-33-1 ПЕРЕДАЧА БЛОКА: XFERC(565)

Назначение

Передает указанное количество последовательно расположенных слов.

Символ РКС



N: Количество слов

S: Первое исходное слово

D: Первое слово назначения

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	XFERC(565)
	Однократное выполнение по положительному фронту	@XFERC(565)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

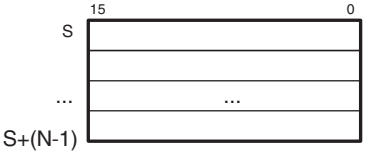
Операнды

N: Количество слов

Указывает количество слов, которое должно быть передано. Допустимый диапазон значений N: 0000...9999 BCD.

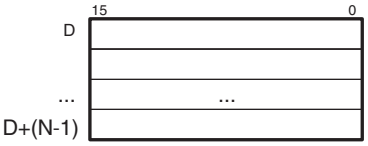
S: Первое исходное слово

Указывает первое исходное слово.



D: Первое слово назначения

Указывает первое слово назначения.

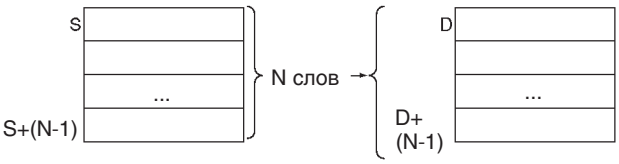


Характеристики операндов

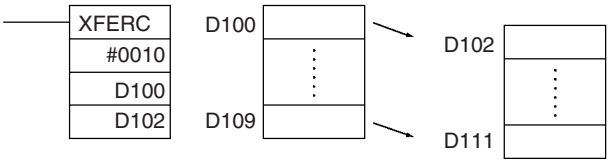
Область	N	S	D
Область CIO	CIO 0...CIO 6143		
Рабочая область	W0...W511		
Область битов хранения	H0...H511		
Область вспомогательных битов	A0...A959		A448...A959
Область таймеров	T0000...T4095		
Область счетчиков	C0000...C4095		
Область DM	D0...D32767		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	#0000...#9999 (BCD)	---	---
Регистры данных	DR0...DR15	---	
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…–2048…+2047 ,IR15 DR0…DR15, IR0…IR15 ,IR0+(++)…,IR15+(++) ,–(– –)IR0…, –(– –)IR15		

Описание

Команда XFERC(565) копирует N слов, начиная со слова S (S...S+(N-1)), в N слов, начиная со слова D (D...D+(N-1)).



Исходные слова и слова назначения могут накладываться, так что с помощью команды XFERC(565) можно выполнять пословный сдвиг данных.



Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если операнд N (количество слов) представлен не в формате BCD.

Примечание. В ПЛК серии C использование команды передачи блока (XFER) вызывает включение флага ошибки, если содержимое косвенно адресованного слова области DM (*DM) представлено не в двоично-десятичном формате или превышен предельный адрес области DM. Выполнение команды XFERC(565) в этих случаях к включению флага ошибки не приводит.

Меры предосторожности

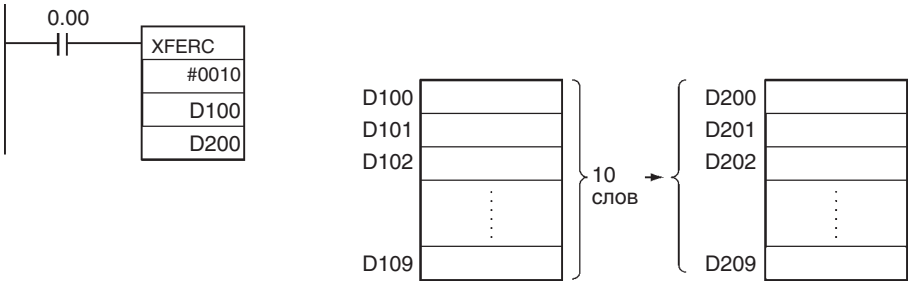
Проследите, чтобы исходные слова (S...S+N-1) и слова назначения (D...D+N-1) не вышли за границу области данных.

Передача большого количества слов командой XFERC(565) занимает некоторое время. Если во время выполнения команды XFERC(565) будет прервано питание, передача может быть не завершена.

Содержимое N должно быть в двоично-десятичном формате. Если значение N указано не в двоично-десятичном формате, возникнет ошибка и установится флаг ошибки.

Пример

Если в следующем примере бит CIO 0.00 включен, значения десяти слов, расположенных по адресам от D100 до D109, копируются в соответствующие слова, расположенные по адресам от D200 до D209.

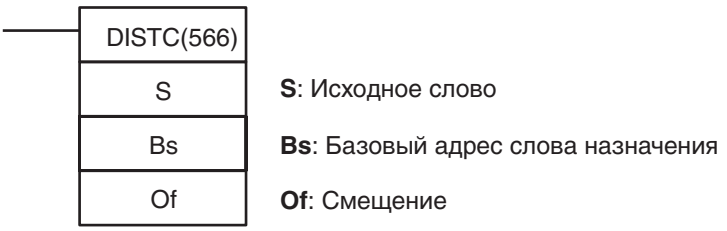


3-33-2 РАЗМЕЩЕНИЕ ОДИНОЧНОГО СЛОВА: DISTC(566)

Назначение

Передает содержимое исходного слова в слово назначения, адрес которого определяется путем смещения базового адреса на указанную величину.

Символ РКС



Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	DISTC(566)
	Однократное выполнение по положительному фронту	@DISTC(566)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

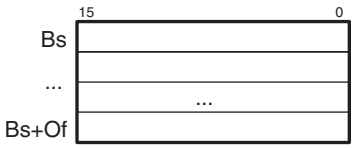
Bs: Базовый адрес слова назначения

Указывает базовый адрес слова назначения. Фактический адрес слова назначения определяется путем добавления величины смещения к данному адресу.

Of: Смещение

- Операция размещения значений (0000...7999 BCD)

  Это значение добавляется к базовому адресу для определения фактического адреса слова назначения. Смещение может иметь любое значение от 0000 до 7999 в двоично-десятичном формате (BCD), но Bs и Bs+Of должны находиться в одной области данных.



- Операция сдвига стека (9000...9999 BCD)

  Если старший разряд Of = 9, три младших разряда Of указывают количество слов в стеке. Смещение может иметь любое значение от 9000 до 9999 BCD.

Характеристики операндов

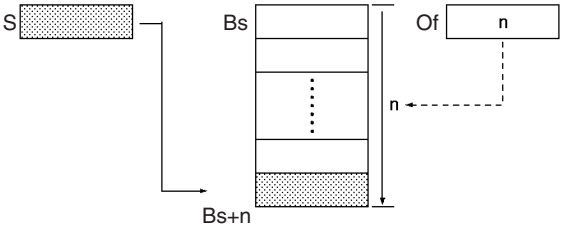
Область	S	Bs	Of
Область CIO	CIO 0...CIO 6143		
Рабочая область	W0...W511		
Область битов хранения	H0...H511		
Область вспомогательных битов	A0...A959	A448...A959	A0...A959
Область таймеров	T0000...T4095		
Область счетчиков	C0000...C4095		
Область DM	D0...D32767		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	#0000...#FFFF (hex)	---	#0000... #7999 для размещения значений #9000...#9999 для функции стека

Область	S	Bs	Of
Регистры данных	DR0...DR15	---	DR0...DR15
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…–2048…+2047 ,IR15 DR0…DR15, IR0…IR15 ,IR0+(++)…,IR15+(++) ,–(– –)IR0…, –(– –)IR15		

Описание

Операция размещения значений

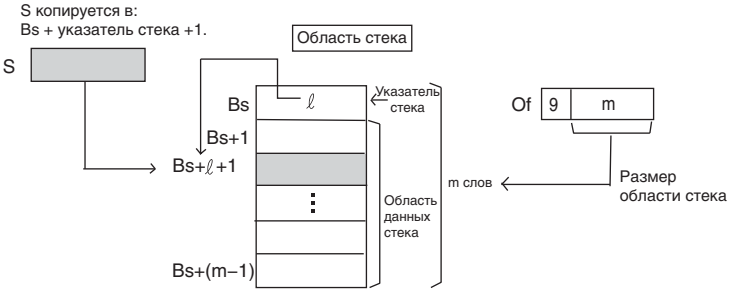
Команда DISTC(566) копирует слово S в слово назначения, адрес которого определяется путем добавления величины смещения (Of) к базовому адресу (Bs). Используя одну и ту же команду DISTC(566) и изменяя смещение адреса (Of), содержимое исходного слова можно разместить в несколько различных слов некоторой области данных.



Операция сдвига стека

Если старший разряд (биты 12...15) операнда Of = 9 (BCD), команда DISTC(566) реализует стек в диапазоне адресов Bs...Bs+Of-9000. По базовому адресу слова назначения (Bs) хранится указатель стека, а остальные слова стека содержат данные стека.

Команда DISTC(566) копирует слово S в слово назначения, адрес которого определяется путем добавления указателя стека (содержимое Bs) + 1 к адресу Bs. Используя одну и ту же команду DISTC(566) и изменяя смещение адреса (Of), содержимое исходного слова можно разместить в несколько различных слов некоторой области данных.



Каждый раз, когда содержимое слова S копируется в слово стека, указатель стека, хранящийся по адресу Bs, автоматически увеличивается на 1.

Примечание. Для чтения данных из стека используйте команду COLLC(567).

Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если задана операция сдвига стека, но значение указателя стека в Bs не в формате BCD. Включен, если задана операция сдвига стека, а указатель стека указывает на слово за пределами области данных стека.
Флаг равенства	=	Включен, если исходное слово = 0000. Выключен во всех остальных случаях.

Примечание. В ПЛК серии C использование команды размещения одиночного слова (DIST) вызывает включение флага ошибки, если содержимое косвенно адресованного слова области DM (*DM) представлено не в двоично-десятичном формате или превышен предельный адрес области DM. Выполнение команды DISTC(566) в этих случаях к включению флага ошибки не приводит.

Меры предосторожности

Выполнив команду DISTC(566) с объявлением области стека для реализации операции сдвига стека, всегда указывайте область стека такой же длины в последующих командах DISTC(566). Если в последующих командах DISTC(566) будет указываться другой размер области стека, операции со стеком могут выполняться некорректно.

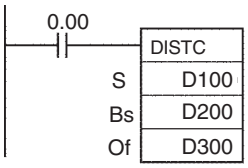
Проследите, чтобы величина смещения или размер стека, указанные в Of, при добавлении к Bs не приводили к выходу за границу области данных.

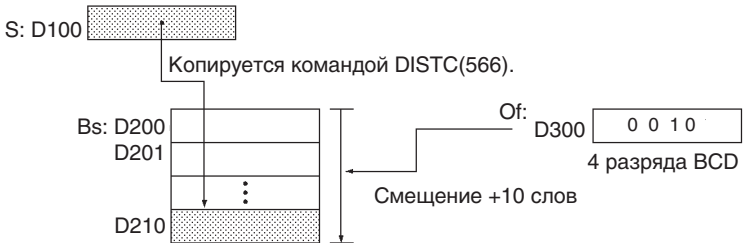
Примеры

Операция размещения значений

Старший байт D300 = 0, поэтому команда DISTC(566) выполняет операцию размещения значений.

Если в следующем примере бит CIO 0.00 включен, содержимое слова D100 копируется в слово D210 (D200 + 10), если содержимое D300 = 0010 BCD. Изменяя величину смещения в D300, содержимое D100 можно скопировать в другие слова.



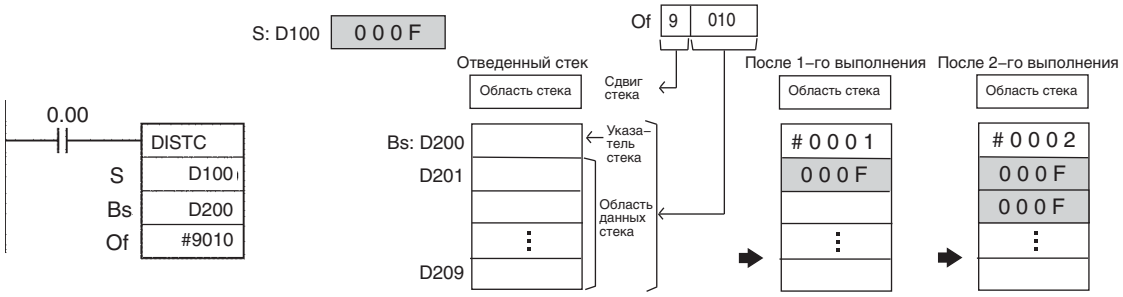


Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

Операция сдвига стека

Старший байт операнда Of = 9, поэтому команда DISTC(566) выполняет операцию сдвига стека.

Если в приведенном ниже примере бит CIO 0.00 включен, команда DISTC(566) объявляет слова в диапазоне от D200 до D209 как область стека размером в 10 слов (поскольку 3 младших разряда значения Of содержат #010). Одновременно содержимое слова D100 копируется в слово, адрес которого определяется как: D200 + указатель стека + 1. Затем указатель стека увеличивается на 1.

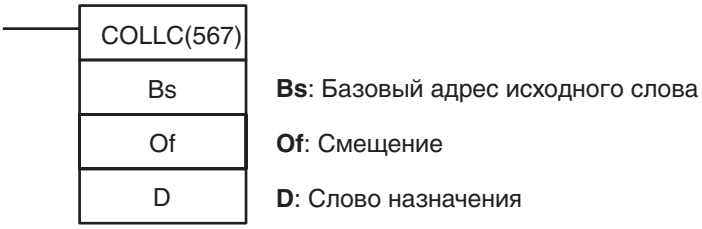


3-33-3 СБОР ДАННЫХ: COLLC(567)

Назначение

Передает в слово назначения содержимое исходного слова, адрес которого определяется путем смещения базового адреса на указанную величину.

Символ РКС



Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	COLLC(567)
	Однократное выполнение по положительному фронту	@COLLC(567)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпро-граммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

Bs: Базовый адрес исходного слова

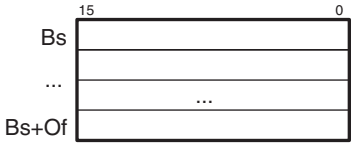
Указывает базовый адрес исходного слова. Фактический адрес исходного слова определяется путем добавления величины смещения к данному адресу.

Of: Смещение

Работа команды COLLC(567) зависит от значения Of.

- Операция сбора данных (Of = 0000...7999 BCD)

Значение Of добавляется к базовому адресу для определения фактического адреса исходного слова. Смещение может иметь любое значение от 0000 до 7999 BCD, но Bs и Bs+Of должны находиться в одной области данных.



- Операция чтения из стека LIFO (Of = 8000...8999 BCD)
  Если старший разряд Of = 8, команда COLLC(567) реализует функцию стека LIFO («последним вошел — первым вышел»). Началом стека является Bs, длина стека определяется тремя младшими разрядами Of.
- Операция чтения из стека FIFO (Of = 9000...9999 BCD)
  Если старший разряд Of = 9, команда COLLC(567) реализует функцию стека FIFO («первым вошел — первым вышел»). Началом стека является Bs, длина стека определяется тремя младшими разрядами Of.

Характеристики операндов

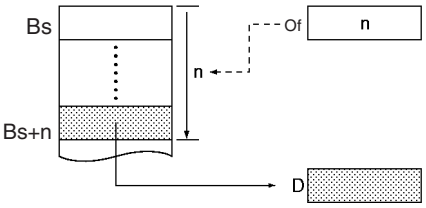
Область	Bs	Of	D
Область CIO	CIO 0...CIO 6143		
Рабочая область	W0...W511		
Область битов хранения	H0...H511		
Область вспомогательных битов	A0...A959		A448...A959
Область таймеров	T0000...T4095		
Область счетчиков	C0000...C4095		
Область DM	D0...D32767		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	---	#0000... #7999 для сбора данных #8000... #8999 для чтения из стека в обратном порядке (LIFO) #9000...#9999 для чтения из стека в порядке поступления (FIFO)	---
Регистры данных	---	DR0...DR15	
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…–2048…+2047 ,IR15 DR0…DR15, IR0…IR15 ,IR0+(++)…,IR15+(++) ,–(– –)IR0…, –(– –)IR15		

Описание

В зависимости от значения Of команда COLLC(567) работает как команда сбора данных, как команда стека FIFO или как команда стека LIFO.

Операция сбора данных (Of = 0000...7999 BCD)

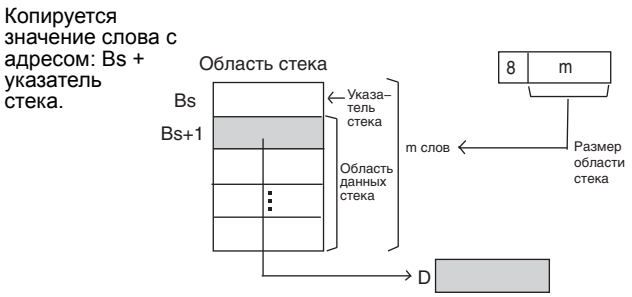
Команда COLLC(567) копирует исходное слово (адрес которого определяется путем добавления Of к Bs) в слово назначения. Используя одну и ту же команду COLLC(567) и изменяя смещение адреса (Of), можно собирать данные из различных исходных слов области данных.



Операция чтения из стека LIFO (Of = 8000...8999 BCD)

Если старший разряд Of = 8, команда COLLC(567) выполняет функцию стека LIFO («последним вошел — первым вышел»). В этом случае три младших разряда Of указывают размер стека.

Команда COLLC(567) копирует в D значение, записанное в стек самым последним. Адрес исходного слова определяется как: Bs + указатель стека (содержимое Bs). После того как значение скопировано, указатель стека уменьшается на 1.

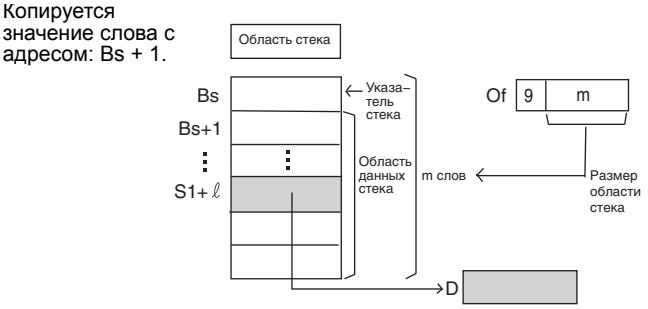


Примечание. Для записи данных в область стека используйте команду DISTC(566).

Операция чтения из стека FIFO (Of = 9000...9999 BCD)

Если старший разряд Of = 9, команда COLLC(567) выполняет функцию стека FIFO («первым вошел — первым вышел»). В этом случае три младших разряда Of указывают размер стека.

Команда COLLC(567) копирует в D самое «старое» значение (записанное в стек самым первым). Адрес исходного слова определяется как Bs + 1. После того как значение скопировано, указатель стека уменьшается на 1.



Примечание. Для записи данных в область стека используйте команду DISTC(566).

Флаги

Название	Об-означение	Описание работы
Флаг ошибки	ER	Включен, если смещение в операнде Of задано не в BCD-формате. Включен, если задана функция стека LIFO или FIFO, но указатель стека в Bs представлен не в BCD-формате. Включен, если задана функция стека LIFO или FIFO, а указатель стека указывает на слово за пределами области данных стека. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если исходное слово = 0000. Выключен во всех остальных случаях.

Примечание. В ПЛК серии C использование команды сбора данных (COLL) вызывает включение флага ошибки, если содержимое косвенно адресованного слова области DM (*DM) представлено не в двоично-десятичном формате или превышен предельный адрес области DM. Выполнение команды COLLC(567) в этих случаях к включению флага ошибки не приводит.

Меры предосторожности

Выполнив команду DISTC(566) с объявлением области стека для реализации операции сдвига стека, всегда указывайте область стека такой же длины в последующих командах COLLC(567). Если в последующих командах COLLC(567) будет указываться другой размер области стека, операции со стеком могут выполняться некорректно.

Проследите, чтобы величина смещения или размер стека, указанные в Of, при добавлении к Bs не приводили к выходу за границу области данных.

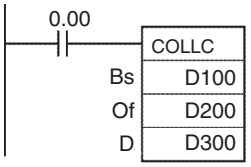
Значение смещения Of должно быть указано в двоично-десятичном формате (BCD).

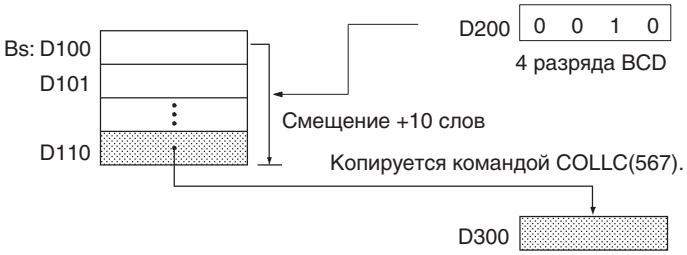
Примеры

Операция сбора данных

Старший байт D200 = 0, поэтому команда COLLC(567) выполняет операцию сбора данных.

Если в следующем примере бит CIO 0.00 включен, содержимое слова D110 (D100 + 10) копируется в слово D300, если содержимое D200 = 10 (0010 BCD). Изменяя величину смещения в D200, в D300 можно скопировать содержимое других слов.

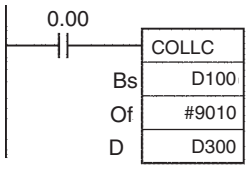


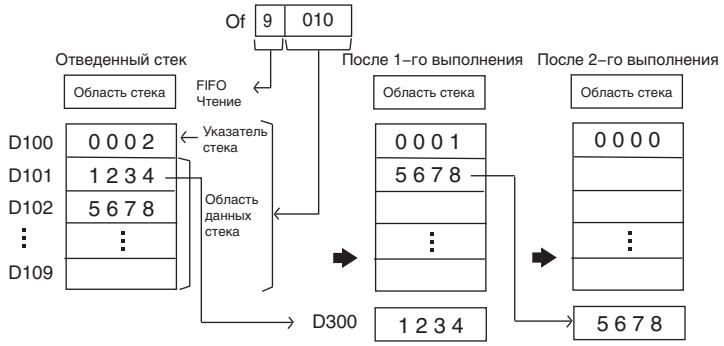


Функция стека FIFO

Старший байт Of = 9, поэтому команда COLLC(567) выполняет функцию стека FIFO.

Если в приведенном ниже примере бит CIO 0.00 включен, команда COLLC(567) объявляет слова в диапазоне от D100 до D109 как область стека размером в 10 слов (поскольку 3 младших разряда значения Of содержат #010). Одновременно содержимое слова D101 (Bs +1) копируется в слово D300. Затем указатель стека уменьшается на 1.

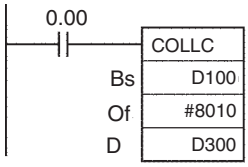


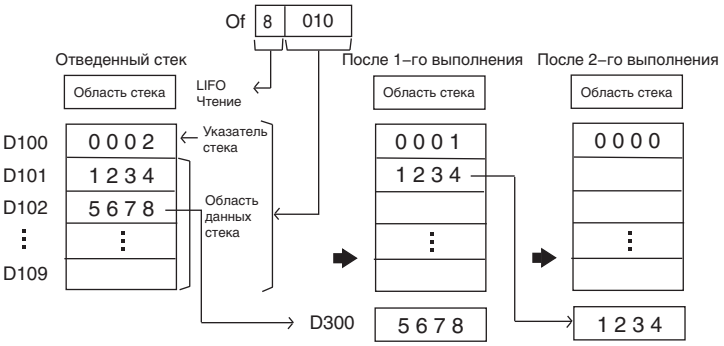


Функция стека LIFO

Старший байт Of = 8, поэтому команда COLLC(567) выполняет функцию стека LIFO.

Если в приведенном ниже примере бит CIO 0.00 включен, команда COLLC(567) объявляет слова в диапазоне от D100 до D109 как область стека размером в 10 слов (поскольку 3 младших разряда значения Of содержат #010). Одновременно содержимое исходного слова (D100 + указатель стека) копируется в слово D300. Затем указатель стека уменьшается на 1.



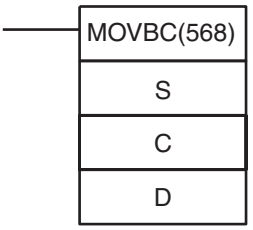


3-33-4 ПЕРЕМЕЩЕНИЕ БИТА: MOVBC(568)

Назначение

Передает состояние указанного бита.

Символ РКС



S: Исходное слово или данные

C: Управляющее слово

D: Слово назначения

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	MOVBC(568)
	Однократное выполнение по положительному фронту	@MOVBC(568)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

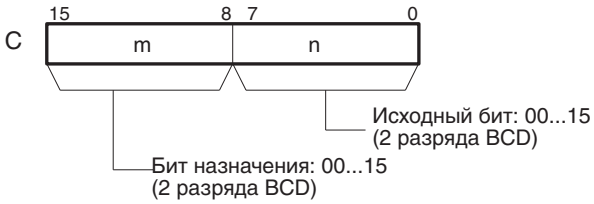
Доступные области программы

Области программных блоков	Области пошаговых программ	Подпро-граммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

C: Управляющее слово

Два младших разряда слова C указывают требуемый бит в исходном объекте S, а два старших разряда слова C указывают требуемый бит в конечном объекте D.

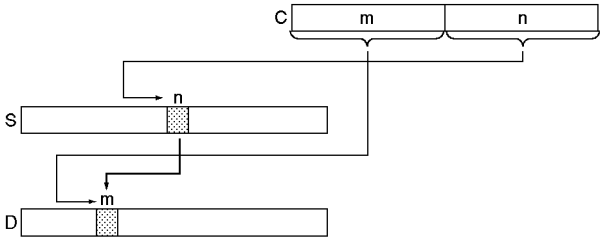


Характеристики операндов

Область	S	C	D
Область CIO	CIO 0...CIO 6143		
Рабочая область	W0...W511		
Область битов хранения	H0...H511		
Область вспомогательных битов	A0...A959		A448...A959
Область таймеров	T0000...T4095		
Область счетчиков	C0000...C4095		
Область DM	D0...D32767		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	#0000...#FFFF (двоичн.)	Только указанные значения	---
Регистры данных	DR0...DR15		
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…–2048…+2047 ,IR15 DR0…DR15, IR0…IR15 ,IR0+(++)…,IR15+(++) ,–(– –)IR0…, –(– –)IR15		

Описание

Команда MOVBC(568) копирует указанный бит (n) слова S в указанный бит (m) слова D. Прочие биты слова адресата остаются неизменными.



Примечание. В операндах S и D может быть указано одно и то же слово. В этом случае бит будет скопирован в другой бит в пределах данного слова.

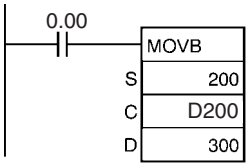
Флаги

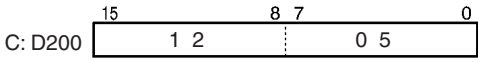
Название	Обо-значение	Описание работы
Флаг ошибки	ER	Включен, если значения двух младших и двух старших разрядов C указаны не в двоично-десятичном формате или выходят за пределы указанного диапазона 00...15. Выключен во всех остальных случаях.

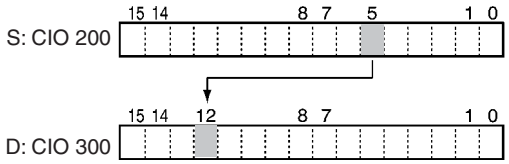
Примечание. В ПЛК серии C использование команды перемещения бита (MOVB) вызывает включение флага ошибки, если содержимое косвенно адресованного слова области DM (*DM) представлено не в двоично-десятичном формате или оказался превышен предельный адрес области DM. Выполнение команды MOVBC(568) к включению флага ошибки в этих случаях не приводит.

Примеры

Если в следующем примере бит CIO 0.00 включен, 5-й бит исходного слова (CIO 200) копируется в 12-й бит слова адресата (CIO 300) в соответствии со значением управляющего слова (1205).





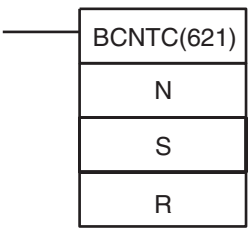


3-33-5 ПОДСЧЕТ БИТОВ: BCNTC(621)

Назначение

Определяет общее количество включенных битов в указанном слове или словах.

Символ РКС



N: Количество слов

S: Первое исходное слово

R: Слово результата

Варианты выполнения

Варианты выполнения	**Выполнение в каждом цикле при включенном условии выполнения**	BCNTC(621)
	Однократное выполнение по положительному фронту	@BCNTC(621)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпро-граммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

N: Количество слов

Количество слов должно иметь значение от 0001 до 9999 (BCD).

S: Первое исходное слово

S и S+(N-1) должны принадлежать одной области.

Характеристики операндов

Область	N	S	R
Область CIO	CIO 0...CIO 6143		
Рабочая область	W0...W511		
Область битов хранения	H0...H511		
Область вспомогательных битов	A0...A959		A448...A959
Область таймеров	T0000...T4095		
Область счетчиков	C0000...C4095		
Область DM	D0...D32767		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	#0001... #9999 (BCD)	---	
Регистры данных	DR0...DR15	---	DR0...DR15
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…–2048…+2047 ,IR15 DR0…DR15, IR0…IR15 ,IR0+(++)…,IR15+(++) ,–(– –)IR0…, –(– –)IR15		

Описание

Команда BCNTC(621) подсчитывает суммарное количество включенных битов во всех словах диапазона S...S+(N-1) и записывает результат в операнд R в двоично-десятичном формате (BCD).



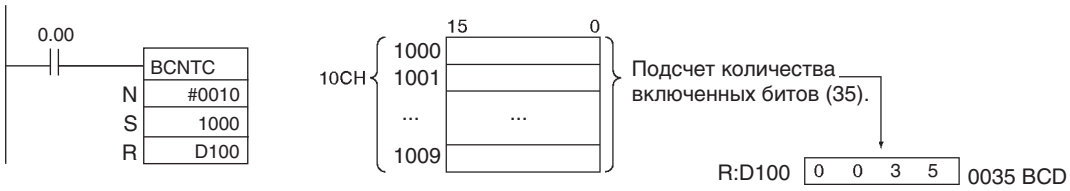
Флаги

Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если значение N выходит за диапазон 0001...9999 BCD. Включен, если результат превышает 9999 BCD. Выключен во всех остальных случаях.
Флаг равенства	=	Включен, если результат = 0000. Выключен во всех остальных случаях.

Примечание. В ПЛК серии C использование команды подсчета битов (BCNTC) вызывает включение флага ошибки, если содержимое косвенно адресованного слова области DM (*DM) представлено не в двоично-десятичном формате или превышен предельный адрес области DM. Выполнение команды BCNTC(621) к включению флага ошибки в этих случаях не приводит.

Меры предосторожности Если N не является двоично-десятичным значением от 0001 до 9999 или результат превышает 9999, будет сгенерирована ошибка.

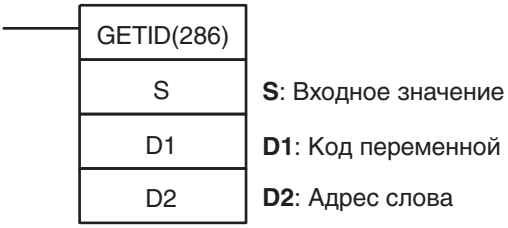
Пример Если в следующем примере бит CIO 0.00 включен, команда BCNTC(621) подсчитывает суммарное количество включенных битов в десяти словах диапазона CIO 1000...CIO 1009 и записывает результат в слово D100.



3-33-6 СЧИТАТЬ ИДЕНТИФИКАТОР ПЕРЕМЕННОЙ: GETID(286)

Назначение Возвращает код типа переменной (области данных) команды FINS и адрес слова для указанной переменной или адреса. Эта команда обычно используется для считывания адреса, присвоенного переменной в функциональном блоке.

Символ РКС



Варианты выполнения

Варианты выполнения		GETID(286)
	Выполнение в каждом цикле при включенном условии выполнения	
	Однократное выполнение по положительному фронту	@GETID(286)
	Однократное выполнение по отрицательному фронту	Не предусмотрено
Модификатор мгновенного обновления		Не предусмотрено

Доступные области программы

Области программных блоков	Области пошаговых программ	Подпрограммы	Задачи обработки прерываний
OK	OK	OK	OK

Операнды

S: Входное значение

Указывает переменную или адрес, для которого будут считаны тип переменной и адрес слова.

D1: Код переменной

Содержит код типа переменной (код области данных) FINS, соответствующий входному значению.

D2: Адрес слова

Содержит 4-разрядный шестнадцатеричный адрес слова, соответствующий входному значению.

Характеристики операндов

Область	S	D1	D2
Область CIO	CIO 0...CIO 6143		
Рабочая область	W0...W511		
Область битов хранения	H0...H511		
Область вспомогательных битов	A0...A959		
Область таймеров	T0000...T4095		
Область счетчиков	C0000...C4095		
Область DM	D0...D32767		
Косвенные адреса DM в двоичном формате	@ D0...@ D32767		
Косвенные адреса DM в BCD-формате	*D0...*D32767		
Постоянные	---		
Регистры данных	DR0...DR15		
Регистры указателей	---		
Косвенная адресация с помощью регистров указателей	,IR0...,IR15 -2048…+2047 ,IR0…–2048…+2047 ,IR15 DR0…DR15, IR0…IR15 ,IR0+(++)…,IR15+(++) ,–(– –)IR0…, –(– –)IR15		

Описание

Команда GETID(286) считывает адрес области данных, соответствующий указанной входной переменной или адресу, и выдает 4-разрядное шестнадцатеричное значение кода области данных в слово D1 и 4-разрядное шестнадцатеричное значение номера адреса слова в D2.

В следующей таблице представлены коды типов переменных (областей данных) и соответствующие им диапазоны адресов областей данных ПЛК.

Область данных		Разрядность	Код области данных (выдается в D1)	Адрес (выдается в D2)
Область CIO	CIO	Слово	00B0 hex	0000...17FF hex (0000...6143)
Рабочая область	W		00B1 hex	0000...01FF hex (000...511)
Область битов хранения	H		00B2 hex	0000...01FF hex (000...511)
Область вспомогательных битов			00B3 hex	0000...03BF hex (000...959)
Область DM			0082 hex	0000...7FFF hex (00000...32767)

CX-Programmer версии 5.0 и выше автоматически назначает адреса переменным в функциональных блоках (за исключением переменных с AT). Например, если нужно косвенно указать расширенные настройки специального модуля (такого как модуль управления движением) и при этом в начале области расширенных настроек используется переменная, требуется указать адрес этой переменной. В этом случае можно использовать команду GETID(286), для того чтобы считать адрес области данных этой переменной.

Флаги

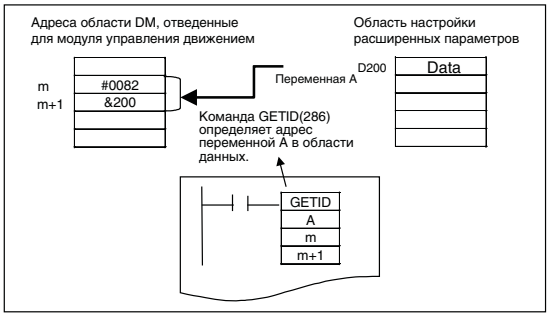
Название	Обозначение	Описание работы
Флаг ошибки	ER	Включен, если S выходит за пределы допустимого диапазона.

Пример

Обычная программа



Использование функциональных блоков



РАЗДЕЛ 4
Время выполнения и размер команд

В данном разделе приводятся значения времени выполнения и количества шагов для каждой команды, которая поддерживается программируемыми контроллерами серии CP1H. Также приводится методика расчета времени выполнения и количества шагов для экземпляра функционального блока.

4-1 Время выполнения и размер команд 1166
4-2 Время выполнения экземпляра функционального блока 1191

Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

4-1 Время выполнения и размер команд

Приведенная ниже группа таблиц содержит значения времени выполнения для всех команд программирования, доступных для ПЛК серии CP.

Общее время выполнения всех команд, из которых состоит программа пользователя, одновременно является и временем выполнения программы, которое входит в качестве одного из слагаемых в формулу вычисления длительности цикла программы (см. примечание).

Примечание. Любая программа пользователя состоит из задач, которые выполняются в каждом цикле, и задач, которые выполняются только по наступлению условия прерывания.

Время выполнения большинства команд зависит от условий, при которых происходит выполнение команды. Для выполнения команды требуется время, даже если условие выполнения этой команды выключено.

В каждой из приведенных ниже таблиц также имеется столбец *Размер (кол-во шагов)*, предоставляющий информацию о размере каждой команды. Команды программируемого контроллера серии CP могут занимать в области программ пользователя от 1 до 7 шагов. Размер команды зависит как от самой команды, так и от используемых ею операндов. Количество шагов в программе и количество команд — это не одно и то же.

Примечание.

(1) Объем программы определяется как количество шагов в программе. По существу, 1 шаг эквивалентен 1 слову (в ПЛК OMRON прежних выпусков объем программы измерялся в словах).

(2) Большинство команд поддерживают однократное выполнение по переключению условия выполнения (обозначается символами ↑, ↓, @, и %). В следующей таблице указано увеличение времени выполнения для модифицированных команд.

Символ	Модуль ЦПУ CP1H	Модуль ЦПУ CP1L
↑ или ↓	+0,24 мкс	+4,82 мкс
@ или %	+0,24 мкс	+3,04 мкс

(3) Для команд, которые не выполняются, следует использовать ориентировочные значения времени выполнения, приведенные ниже.

Модуль ЦПУ CP1H	Модуль ЦПУ CP1L
Приблиз. 0,1 мкс	Приблиз. 0,1 мкс

(4) В тех случаях, когда программа ПЛК предыдущей модели (серии C или серии CV/CVM1) преобразуется в программу для ПЛК серии CP, для пересчета размера программы и преобразования количества слов в количество шагов можно воспользоваться следующей методикой. Чтобы вычислить размер программы ПЛК серии CP в шагах, к прежнему значению следует добавить значения (n) для каждой команды, приведенные в следующей таблице. Пересчет делать не требуется, если исходной программой является

программа ПЛК серии CS/CJ (так как размер при этом не поменяется).

Прежний размер: «a» слов — размер в CP1H: «a+n» шагов			
Тип команды	**Варианты выполнения команды**	**Значение «n» при переходе от ПЛК серии C к ПЛК серии CP**	**Значение «n» при переходе от ПЛК серии CV/ CVM1 к ПЛК серии CP**
Базовые команды	Нет	-1 для OUT, SET, RESET и KEEP 0 для всех остальных команд	0
	Выполнение по положительному фронту	0	+1
	Мгновенное обновление	---	0
	Выполнение по положительному фронту и мгновенное обновление	---	+2
Специальные команды	Нет	0	-1
	Выполнение по положительному фронту	+1	0
	Мгновенное обновление	---	+3
	Выполнение по положительному фронту и мгновенное обновление	---	+4

Примеры:

В ПЛК серии C для одной команды ВЫВОД требуется 2 слова, поэтому в ПЛК серии CP эта команда будет занимать 1 шаг (2 - 1).

В ПЛК серии CV/CVM1 для каждой команды ПЕРЕДАТЬ с модификатором мгновенного обновления (!MOV) требуется 4 слова, поэтому в ПЛК серии CP эта команда будет занимать 7 шагов (4 + 3).

Входные битовые команды

Команда	**Мнемони-ческое обозначение**	**Код**	**Размер (кол-во шагов) (см. примеч.)**	**Время выполнения (ВКЛ) (мкс)**		**Условия**
				CP1H	**CP1L**	
ЗАГРУЗКА	LD	---	1	0,10	0,55	---
	!LD	---	2	+24,10	+5,60	Увеличение при мгновенном обновлении
ЗАГРУЗКА НЕ	LD NOT	---	1	0,10	0,55	---
	!LD NOT	---	2	+24,10	+5,62	Увеличение при мгновенном обновлении
И	AND	---	1	0,10	0,61	---
	!AND	---	2	+24,10	+5,60	Увеличение при мгновенном обновлении
И НЕ	AND NOT	---	1	0,10	0,65	---
	!AND NOT	---	2	+24,10	+5,62	Увеличение при мгновенном обновлении

Команда	Мнемони-ческое обозначение	Код	Размер (кол-во шагов) (см. примеч.)	Время выполнения (ВКЛ) (мкс)		Условия
				CP1H	CP1L	
ИЛИ	OR	---	1	0,10	0,68	---
	!OR	---	2	+24,10	+5,65	Увеличение при мгновенном обновлении
ИЛИ НЕ	OR NOT	---	1	0,10	0,65	---
	!OR NOT	---	2	+24,10	+5,62	Увеличение при мгновенном обновлении
И ЗАГРУЗКА	AND LD	---	1	0,05	0,42	---
ИЛИ ЗАГРУЗКА	OR LD	---	1	0,05	0,39	---
НЕ	NOT	520	1	0,05	0,42	---
УСЛОВИЕ ВКЛ	UP	521	3	0,50	2,37	---
УСЛОВИЕ ВЫКЛ	DOWN	522	4	0,50	2,89	---
ЗАГРУЗКА ПРОВЕРКА БИТА	LD TST	350	4	0,35	4,22	---
ЗАГРУЗКА ПРОВЕРКА БИТА НЕ	LD TSTN	351	4	0,35	4,22	---
И ПРОВЕРКА БИТА	AND TST	350	4	0,35	6,50	---
И ПРОВЕРКА БИТА НЕ	AND TSTN	351	4	0,35	4,29	---
ИЛИ ПРОВЕРКА БИТА	OR TST	350	4	0,35	4,31	---
ИЛИ ПРОВЕРКА БИТА НЕ	OR TSTN	351	4	0,35	4,29	---

Примечание. Если используется операнд длиной в два слова, к значению размера, указанному в соответствующем столбце следующей таблицы, следует добавить 1.

Выходные битовые команды

Команда	Мнемони-ческое обозначение	Код	Размер (кол-во шагов) (см. примеч.)	Время выполнения (ВКЛ) (мкс)		Условия
				CP1H	CP1L	
ВЫВОД	OUT	---	1	0,35	1,10	---
	!OUT	---	2	+23,07	+6,05	Увеличение при мгновенном обновлении
ВЫВОД НЕ	OUT NOT	---	1	0,35	1,07	---
	!OUT NOT	---	2	+23,07	+5,94	Увеличение при мгновенном обновлении
ЗАЩЕЛКА	KEEP	11	1	0,40	5,55	---
ПОЛОЖИТЕЛЬНЫЙ ФРОНТ	DIFU	13	2	0,50	2,37	---
ОТРИЦАТЕЛЬНЫЙ ФРОНТ	DIFD	14	2	0,50	2,34	---
УСТАНОВКА	SET	---	1	0,30	1,40	---
	!SET	---	2	+23,17	+7,85	Увеличение при мгновенном обновлении
СБРОС	RSET	---	1	0,30	1,33	Указано слово
	!RSET	---	2	+23,17	+7,78	Увеличение при мгновенном обновлении
УСТАНОВИТЬ НЕСКОЛЬКО БИТОВ	SETA	530	4	11,77	16,10	Установка 1 бита
				67,03	107,50	Установка 1000 битов
СБРОСИТЬ НЕСКОЛЬКО БИТОВ	RSTA	531	4	11,8	16,11	Сброс 1 бита
				69,63	110,70	Сброс 1000 битов

Команда	Мнемони-ческое обозначение	Код	Размер (кол-во шагов) (см. примеч.)	Время выполнения (ВКЛ) (мкс)		Условия
				CP1H	CP1L	
УСТАНОВИТЬ ОДИН БИТ	SETB	532	2	0,5	25,13	---
	!SETB		3	+23,31	+30,88	---
СБРОСИТЬ ОДИН БИТ	RSTB	533	2	0,5	25,36	---
	!RSTB		3	+23,31	+31,11	---
ВЫВОД В ОДИН БИТ	OUTB	534	2	0,45	27,03	---
	!OUTB		3	+23,22	+32,68	---

Примечание. Если используется операнд длиной в два слова, к значению размера, указанному в соответствующем столбце следующей таблицы, следует добавить 1.

Команды управления последовательностью выполнения

Команда	Мнемони-ческое обозначение	Код	Размер (кол-во шагов) (см. примеч.)	Время выполнения (ВКЛ) (мкс)		Условия
				CP1H	CP1L	
КОНЕЦ	END	1	1	9,18	6,2	---
ПУСТАЯ ОПЕРАЦИЯ	NOP	0	1	0,05	0,6	---
БЛОКИРОВКА	IL	2	1	0,15	3,4	---
ОТМЕНА БЛОКИРОВКИ	ILC	3	1	0,15	3,4	---
ВЛОЖЕННАЯ БЛОКИРОВКА С ЗАПОМИНАНИЕМ ФРОНТА	MILH	517	3	10,3	11,9	Во время блокировки
				13,3	11,9	Не во время блокировки и блокировка не установлена
				16,6	10,5	Не во время блокировки и блокировка установлена
ВЛОЖЕННАЯ БЛОКИРОВКА БЕЗ ЗАПОМИНАНИЯ ФРОНТА	MILR	518	3	10,3	11,9	Во время блокировки
				13,3	11,9	Не во время блокировки и блокировка не установлена
				16,6	10,5	Не во время блокировки и блокировка установлена
ОТМЕНА ВЛОЖЕННОЙ БЛОКИРОВКИ	MILC	519	2	8,3	6,4	Блокировка не отменена
				9,6	6,4	Блокировка отменена
ПЕРЕХОД	JMP	4	2	0,95	4,2	---
КОНЕЦ ПЕРЕХОДА	JME	5	2	---	---	---
УСЛОВНЫЙ ПЕРЕХОД	CJP	510	2	0,95	6,9	Когда удовлетворяется условие JMP
УСЛОВНЫЙ ПЕРЕХОД НЕ	CJPN	511	2	0,95	4,3	Когда удовлетворяется условие JMP
МНОЖЕСТВЕН-НЫЙ ПЕРЕХОД	JMP0	515	1	0,15	1,0	---
КОНЕЦ МНОЖЕСТВЕН-НОГО ПЕРЕХОДА	JME0	516	1	0,15	3,4	---
ЦИКЛ FOR	FOR	512	2	1,00	6,2	Указана константа
ВЫХОД ИЗ ЦИКЛА	BREAK	514	1	0,15	3,4	---
ЦИКЛ NEXT	NEXT	513	1	0,45	4,8	Если цикл продолжается
				0,55	4,8	Если цикл завершается

Примечание. Если используется операнд длиной в два слова, к значению размера, указанному в соответствующем столбце следующей таблицы, следует добавить 1.

Команды управления таймерами и счетчиками

Команда	Мнемоническое обозначение	Код	Размер (кол-во шагов) (см. примеч.)	Время выполнения (ВКЛ) (мкс)		Условия
				CP1H	CP1L	
ТАЙМЕР	TIM	---	3	1,30	6,4	---
	TIMX	550				
СЧЕТЧИК	CNT	---	3	1,30	6,7	---
	CNTX	546				
СКОРОСТНОЙ ТАЙМЕР	TIMH	15	3	1,80	6,4	---
	TIMHX	551				
1 мс ТАЙМЕР	TMHH	540	3	1,75	6,4	---
	TMHHX	552				
НАКАПЛИВАЮЩИЙ ТАЙМЕР	TTIM	87	3	24,81	12,9	---
				17,79	11,7	При сбросе
				13,97	10,2	При блокировке
	TTIMX	555		23,78	14,2	---
				17,76	11,6	При сбросе
				14,11	10,1	При блокировке
ДОЛГОСРОЧНЫЙ ТАЙМЕР	TIML	542	4	15,69	13,2	---
				13,61	11,5	При блокировке
	TIMLX	553		17,51	14,6	---
				13,11	11,2	При блокировке
МНОГОВЫХОДНОЙ ТАЙМЕР	MTIM	543	4	35,36	14,2	---
				12,81	7,6	При сбросе
	MTIMX	554		41,95	27,8	---
				17,42	11,5	При сбросе
РЕВЕРСИВНЫЙ СЧЕТЧИК	CNTR	12	3	29,03	14,5	---
	CNTRX	548		22,44	14,3	
СБРОС ТАЙМЕРА/ СЧЕТЧИКА	CNR	545	3	15,27	10,9	При сбросе 1 слова
				5,95 мс	2,70 мс	При сбросе 1000 слов
	CNRX	547		14,44	10,9	При сбросе 1 слова
				5,95 мс	2,70 мс	При сбросе 1000 слов

Примечание. Если используется операнд длиной в два слова, к значению размера, указанному в соответствующем столбце следующей таблицы, следует добавить 1.

Команды сравнения

Команда	Мнемоническое обозначение	Код	Размер (кол-во шагов) (см. примеч.)	Время выполнения (ВКЛ) (мкс)		Условия
				CP1H	CP1L	
Входные команды сравнения (без знака)	LD, AND, OR +=	300	4	0,35	6,0	---
	LD, AND, OR + <>	305				
	LD, AND, OR + <	310				
	LD, AND, OR +<=	315				
	LD, AND, OR + >	320				
	LD, AND, OR +>=	325				
Входные команды сравнения (двойное слово, без знака)	LD, AND, OR +=+L	301	4	0,35	6,9	---
	LD, AND, OR +<>+L	306				---
	LD, AND, OR +<+L	311				---
	LD, AND, OR +<=+L	316				---
	LD, AND, OR +>+L	321				---
	LD, AND, OR +>=+L	326				---

Команда	Мнемоническое обозначение	Код	Размер (кол-во шагов) (см. примеч.)	Время выполнения (ВКЛ) (мкс)		Условия
				CP1H	CP1L	
Входные команды сравнения (со знаком)	LD, AND, OR +=+S	302	4	0,35	9,8	---
	LD, AND, OR +<>+S	307				
	LD, AND, OR +<+S	312				
	LD, AND, OR +<=+S	317				
	LD, AND, OR +>+S	322				
	LD, AND, OR +>=+S	327				
Входные команды сравнения (двойное слово, со знаком)	LD, AND, OR +=+SL	303	4	0,35	10,5	---
	LD, AND, OR +<>+SL	308				
	LD, AND, OR +<+SL	313				
	LD, AND, OR +<=+SL	318				
	LD, AND, OR +>+SL	323				
	LD, AND, OR +>=+SL	328				
Команды сравнения значений времени	LD, AND, OR +DT	341	4	18,8	53,3	---
	LD, AND, OR +<>DT	342	4	45,6	53,3	---
	LD, AND, OR +<DT	343	4	45,6	53,3	---
	LD, AND, OR +<=DT	344	4	18,8	53,2	---
	LD, AND, OR +>DT	345	4	45,6	53,4	---
	LD, AND, OR +>=DT	346	4	18,8	53,1	---
СРАВНИТЬ	CMP	20	3	0,10	4,0	---
	!CMP	20	7	+45,2	+17,4	Увеличение при мгновенном обновлении
СРАВНИТЬ ДВОЙНЫЕ СЛОВА	CMPL	60	3	0,50	4,8	---
СРАВНИТЬ ДВОИЧНЫЕ ЗНАЧЕНИЯ СО ЗНАКОМ	CPS	114	3	0,30	11,6	---
	!CPS	114	7	+45,2	+32,5	Увеличение при мгновенном обновлении
СРАВНИТЬ ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ	CPSL	115	3	0,50	12,5	---
СРАВНИТЬ С ТАБЛИЦЕЙ	TCMP	85	4	27,66	32,2	---
СРАВНИТЬ НЕСКОЛЬКО ЗНАЧЕНИЙ	MCMP	19	4	42,33	52,4	---
СРАВНИТЬ С ДИАПАЗОНАМИ БЕЗ ЗНАКА	BCMP	68	4	47,21	45,8	---
РАСШИРЕННОЕ СРАВНЕНИЕ С ДИАПАЗОНАМИ	BCMP2	502	4	13,20	18,9	Кол-во слов данных: 1
				650,0	800,7	Кол-во слов данных: 255
СРАВНИТЬ СЛОВО С ДИАПАЗОНОМ	ZCP	88	3	11,53	13,1	---
СРАВНИТЬ ДВОЙНОЕ СЛОВО С ДИАПАЗОНОМ	ZCPL	116	3	11,28	14,4	---

Примечание. Если используется операнд длиной в два слова, к значению размера, указанному в соответствующем столбце следующей таблицы, следует добавить 1.

Команды передачи данных

Команда	Мнемоническое обозначение	Код	Размер (кол-во шагов) (см. примеч.)	Время выполнения (ВКЛ) (мкс)		Условия
				CP1H	CP1L	
ПЕРЕДАТЬ	MOV	21	3	0,30	4,1	---
	!MOV	21	7	+35,1	+17,3	Увеличение при мгновенном обновлении
ПЕРЕДАТЬ ДВОЙНОЕ СЛОВО	MOVL	498	3	0,60	4,8	---
ПЕРЕДАТЬ ДОПОЛНЕНИЕ	MVN	22	3	0,35	15,5	---
ПЕРЕДАТЬ ДОПОЛНЕНИЕ К ДВОЙНОМУ СЛОВУ	MVNL	499	3	0,60	16,6	---
ПЕРЕДАТЬ БИТ	MOVB	82	4	0,50	19,4	---
ПЕРЕДАТЬ РАЗРЯД	MOVD	83	4	0,50	22,1	---
ПЕРЕДАТЬ НЕСКОЛЬКО БИТОВ	XFRB	62	4	20,1	19,4	Передача 1 бита
				266,30	252,6	Передача 255 битов
ПЕРЕДАЧА БЛОКА	XFER	70	4	8,80	14,9	Передача 1 слова
				1,18 мс	1,72 мс	Передача 1000 слов
ЗАПОЛНЕНИЕ БЛОКА	BSET	71	4	14,63	14,5	Запись 1 слова
				570,17	549,3	Запись 1000 слов
ОБМЕН ДАННЫМИ	XCHG	73	3	0,80	20,7	---
ОБМЕН ДВОЙНЫМИ СЛОВАМИ	XCGL	562	3	1,5	22,0	---
РАЗМЕЩЕНИЕ ОДИНОЧНОГО СЛОВА	DIST	80	4	12,77	13,8	---
СБОР ДАННЫХ	COLL	81	4	12,85	11,0	---
ПЕРЕДАТЬ В РЕГИСТР	MOVR	560	3	0,60	13,8	---
ПЕРЕДАТЬ ТЕКУЩЕЕ ЗНАЧЕНИЕ ТАЙМЕРА/ СЧЕТЧИКА В РЕГИСТР	MOVRW	561	3	0,60	13,8	---

Примечание. Если используется операнд длиной в два слова, к значению размера, указанному в соответствующем столбце следующей таблицы, следует добавить 1.

Команды сдвига данных

Команда	Мнемоническое обозначение	Код	Размер (кол-во шагов) (см. примеч.)	Время выполнения (ВКЛ) (мкс)		Условия
				CP1H	CP1L	
РЕГИСТР СДВИГА	SFT	10	3	12,68	8,8	Сдвиг 1 слова
				1,49 мс	2,49 мс	Сдвиг 1000 слов
РЕВЕРСИВНЫЙ РЕГИСТР СДВИГА	SFTR	84	4	13,76	14,0	Сдвиг 1 слова
				1,54 мс	2,47 мс	Сдвиг 1000 слов
АСИНХРОННЫЙ РЕГИСТР СДВИГА	ASFT	17	4	14,21	14,3	Сдвиг 1 слова
				2,94 мс	2,49 мс	Сдвиг 1000 слов
СДВИГ ПО СЛОВАМ	WSFT	16	4	11,20	11,6	Сдвиг 1 слова
				1,47 мс	3,52 мс	Сдвиг 1000 слов
АРИФМЕТИЧЕСКИЙ СДВИГ ВЛЕВО	ASL	25	2	0,45	21,2	---

Команда	Мнемони-ческое обозначение	Код	Размер (кол-во шагов) (см. примеч.)	Время выполнения (ВКЛ) (мкс)		Условия
				CP1H	CP1L	
СДВИГ ДВОЙНОГО СЛОВА ВЛЕВО	ASLL	570	2	0,80	22,0	---
АРИФМЕТИЧЕС-КИЙ СДВИГ ВПРАВО	ASR	26	2	0,45	19,8	---
СДВИГ ДВОЙНОГО СЛОВА ВПРАВО	ASRL	571	2	0,80	20,4	---
ЦИКЛИЧЕСКИЙ СДВИГ ВЛЕВО	ROL	27	2	0,45	25,2	---
ЦИКЛИЧЕСКИЙ СДВИГ ДВОЙНОГО СЛОВА ВЛЕВО	ROLL	572	2	0,80	25,7	---
ЦИКЛИЧЕСКИЙ СДВИГ ВЛЕВО БЕЗ ПЕРЕНОСА	RLNC	574	2	0,45	23,1	---
ЦИКЛИЧЕСКИЙ СДВИГ ДВОЙНОГО СЛОВА ВЛЕВО БЕЗ ПЕРЕНОСА	RLNL	576	2	0,80	23,6	---
ЦИКЛИЧЕСКИЙ СДВИГ ВПРАВО	ROR	28	2	0,45	24,7	---
ЦИКЛИЧЕСКИЙ СДВИГ ДВОЙНОГО СЛОВА ВПРАВО	RORL	573	2	0,80	25,8	---
ЦИКЛИЧЕСКИЙ СДВИГ ВПРАВО БЕЗ ПЕРЕНОСА	RRNC	575	2	0,45	22,4	---
ЦИКЛИЧЕСКИЙ СДВИГ ДВОЙНОГО СЛОВА ВПРАВО БЕЗ ПЕРЕНОСА	RRNL	577	2	0,80	23,0	---
СДВИГ НА ОДИН РАЗРЯД ВЛЕВО	SLD	74	3	11,86	12,9	Сдвиг 1 слова
				1,24 мс	2,51 мс	Сдвиг 1000 слов
СДВИГ НА ОДИН РАЗРЯД ВПРАВО	SRD	75	3	13,95	13,1	Сдвиг 1 слова
				1,85 мс	3,35 мс	Сдвиг 1000 слов
СДВИГ N БИТОВ ДАННЫХ ВЛЕВО	NSFL	578	4	14,39	14,0	Сдвиг 1 бита
				90,10	90,5	Сдвиг 1000 битов
СДВИГ N БИТОВ ДАННЫХ ВПРАВО	NSFR	579	4	14,43	15,5	Сдвиг 1 бита
				130,27	194,1	Сдвиг 1000 битов
СДВИГ НА N БИТОВ ВЛЕВО	NASL	580	3	0,45	27,9	---
СДВИГ ДВОЙНОГО СЛОВА НА N БИТОВ ВЛЕВО	NSLL	582	3	0,80	29,2	---
СДВИГ НА N БИТОВ ВПРАВО	NASR	581	3	0,45	27,0	---
СДВИГ ДВОЙНОГО СЛОВА НА N БИТОВ ВПРАВО	NSRL	583	3	0,80	29,0	---

Примечание. Если используется операнд длиной в два слова, к значению размера, указанному в соответствующем столбце следующей таблицы, следует добавить 1.

Команды увеличения/уменьшения

Команда	Мнемоническое обозначение	Код	Размер (кол-во шагов) (см. примеч.)	Время выполнения (ВКЛ) (мкс)		Условия
				CP1H	CP1L	
УВЕЛИЧИТЬ ДВОИЧНОЕ СЛОВО	++	590	2	0,45	19,1	---
УВЕЛИЧИТЬ ДВОЙНОЕ ДВОИЧНОЕ СЛОВО	++L	591	2	0,80	23,8	---
УМЕНЬШИТЬ ДВОИЧНОЕ СЛОВО	– –	592	2	0,45	23,2	---
УМЕНЬШИТЬ ДВОЙНОЕ ДВОИЧНОЕ СЛОВО	– –L	593	2	0,80	23,6	---
УВЕЛИЧИТЬ ДВОИЧНО-ДЕСЯТИЧНОЕ СЛОВО	++B	594	2	12,09	10,2	---
УВЕЛИЧИТЬ ДВОЙНОЕ ДВОИЧНО-ДЕСЯТИЧНОЕ СЛОВО	++BL	595	2	10,59	10,6	---
УМЕНЬШИТЬ ДВОИЧНО-ДЕСЯТИЧНОЕ СЛОВО	– –B	596	2	11,63	11,9	---
УМЕНЬШИТЬ ДВОЙНОЕ ДВОИЧНО-ДЕСЯТИЧНОЕ СЛОВО	– –BL	597	2	9,59	10,6	---

Примечание. Если используется операнд длиной в два слова, к значению размера, указанному в соответствующем столбце следующей таблицы, следует добавить 1.

Символьные математические команды

Команда	Мнемоническое обозначение	Код	Размер (кол-во шагов) (см. примеч.)	Время выполнения (ВКЛ) (мкс)		Условия
				CP1H	CP1L	
СЛОЖИТЬ БЕЗ ПЕРЕНОСА ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ	+	400	4	0,30	14,1	---
СЛОЖИТЬ БЕЗ ПЕРЕНОСА ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ	+L	401	4	0,60	15,3	---
СЛОЖИТЬ С ПЕРЕНОСОМ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ	+C	402	4	0,40	14,2	---
СЛОЖИТЬ С ПЕРЕНОСОМ ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ	+CL	403	4	0,60	15,5	---
СЛОЖИТЬ БЕЗ ПЕРЕНОСА ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА	+B	404	4	18,14	16,3	---
СЛОЖИТЬ БЕЗ ПЕРЕНОСА ДВОЙНЫЕ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА	+BL	405	4	22,87	20,1	---
СЛОЖИТЬ С ПЕРЕНОСОМ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА	+BC	406	4	19,7	16,4	---

Предварительная версия
Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

Команда	Мнемони-ческое обозначение	Код	Размер (кол-во шагов) (см. примеч.)	Время выполнения (ВКЛ) (мкс)		Условия
				CP1H	CP1L	
СЛОЖИТЬ С ПЕРЕНОСОМ ДВОЙНЫЕ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА	+BCL	407	4	23,63	20,2	---
ВЫЧЕСТЬ БЕЗ ПЕРЕНОСА ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ	–	410	4	0,3	14,1	---
ВЫЧЕСТЬ БЕЗ ПЕРЕНОСА ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ	–L	411	4	0,60	15,4	---
ВЫЧЕСТЬ С ПЕРЕНОСОМ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ	–C	412	4	0,40	14,2	---
ВЫЧЕСТЬ С ПЕРЕНОСОМ ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ	–CL	413	4	0,60	15,5	---
ВЫЧЕСТЬ БЕЗ ПЕРЕНОСА ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА	–B	414	4	17,57	16,2	---
ВЫЧЕСТЬ БЕЗ ПЕРЕНОСА ДВОЙНЫЕ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА	–BL	415	4	22,09	19,9	---
ВЫЧЕСТЬ С ПЕРЕНОСОМ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА	–BC	416	4	18,37	16,3	---
ВЫЧЕСТЬ С ПЕРЕНОСОМ ДВОЙНЫЕ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА	–BCL	417	4	22,91	20,1	---
УМНОЖИТЬ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ	*	420	4	0,65	27,2	---
УМНОЖИТЬ ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ	*L	421	4	13,02	16,9	---
УМНОЖИТЬ ДВОИЧНЫЕ СЛОВА БЕЗ ЗНАКА	*U	422	4	0,75	27,2	---
УМНОЖИТЬ ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА БЕЗ ЗНАКА	*UL	423	4	13,23	16,8	---
УМНОЖИТЬ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА	*B	424	4	16,83	17,4	---
УМНОЖИТЬ ДВОЙНЫЕ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА	*BL	425	4	33,33	29,4	---
ДЕЛИТЬ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ	/	430	4	0,70	37,4	---
ДЕЛИТЬ ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ	/L	431	4	13,35	17,2	---
ДЕЛИТЬ ДВОИЧНЫЕ СЛОВА БЕЗ ЗНАКА	/U	432	4	0,8	36,5	---

Команда	Мнемони-ческое обозначение	Код	Размер (кол-во шагов) (см. примеч.)	Время выполнения (ВКЛ) (мкс)		Условия
				CP1H	CP1L	
ДЕЛИТЬ ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА БЕЗ ЗНАКА	/UL	433	4	12,91	17,1	---
ДЕЛИТЬ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА	/B	434	4	18,03	18,6	---
ДЕЛИТЬ ДВОЙНЫЕ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА	/BL	435	4	27,77	25,3	---

Примечание. Если используется операнд длиной в два слова, к значению размера, указанному в соответствующем столбце следующей таблицы, следует добавить 1.

Команды преобразования

Команда	Мнемони-ческое обозначение	Код	Размер (кол-во шагов) (см. примеч.)	Время выполнения (ВКЛ) (мкс)		Условия
				CP1H	CP1L	
BCD В ДВОИЧНОЕ	BIN	023	3	0,40	57,5	---
ДВОЙНОЕ BCD В ДВОЙНОЕ ДВОИЧНОЕ	BINL	058	3	10,41	14,1	---
ДВОИЧНОЕ В BCD	BCD	024	3	10,22	12,8	---
ДВОЙНОЕ ДВОИЧНОЕ В ДВОЙНОЕ BCD	BCDL	059	3	10,18	14,2	---
ДОПОЛНЕНИЕ ДО 2	NEG	160	3	0,35	22,8	---
ДОПОЛНЕНИЕ ДВОЙНОГО СЛОВА ДО 2	NEGL	161	3	0,60	24,4	---
16-БИТОВОЕ ДВОИЧНОЕ СО ЗНАКОМ В 32-БИТОВОЕ	SIGN	600	3	0,60	25,5	---
ДЕКОДИРОВАНИЕ ДАННЫХ	MLPX	076	4	12,09	15,1	Декодирование 1 разряда (4 -> 16)
				14,15	15,0	Декодирование 4 разрядов (4 -> 16)
				24,01	43,1	Декодирование 1 разряда (8 -> 256)
				37,72	76,9	Декодирование 2 разрядов (8 -> 256)
КОДИРОВАНИЕ ДАННЫХ	DMPX	077	4	11,90	14,4	Кодирование 1 разряда (16 -> 4)
				58,70	24,1	Кодирование 4 разрядов (16 -> 4)
				19,76	16,9	Кодирование 1 разряда (256 -> 8)
				80,32	16,9	Кодирование 2 разрядо (256 -> 8)
ПРЕОБРАЗОВАНИЕ В ASCII	ASC	086	4	12,49	12,1	Преобразование 1 разряда в ASCII
				18,03	20,0	Преобразование 4 разрядов в ASCII
ASCII В HEX	HEX	162	4	12,64	15,6	Преобразование 1 разряда
СТОЛБЕЦ В СТРОКУ	LINE	063	4	34,95	28,4	---
СТРОКУ В СТОЛБЕЦ	COLM	064	4	42,09	40,4	---

Команда	Мнемони-ческое обозначение	Код	Размер (кол-во шагов) (см. примеч.)	Время выполнения (ВКЛ) (мкс)		Условия
				CP1H	CP1L	
BCD СО ЗНАКОМ В ДВОИЧНОЕ	BINS	470	4	15,73	15,7	Формат данных 0
				15,93	15,7	Формат данных 1
				15,93	15,9	Формат данных 2
				16,00	16,0	Формат данных 3
ДВОЙНОЕ BCD СО ЗНАКОМ В ДВОИЧНОЕ	BISL	472	4	18,59	17,4	Формат данных 0
				18,66	17,5	Формат данных 1
				18,41	17,7	Формат данных 2
				18,47	17,7	Формат данных 3
ДВОИЧНОЕ СО ЗНАКОМ В BCD	BCDS	471	4	13,16	17,4	Формат данных 0
				13,18	17,5	Формат данных 1
				13,00	17,7	Формат данных 2
				13,12	17,7	Формат данных 3
ДВОЙНОЕ ДВОИЧНОЕ СО ЗНАКОМ В BCD	BDSL	473	4	13,74	16,3	Формат данных 0
				13,58	16,4	Формат данных 1
				13,79	16,5	Формат данных 2
				13,75	16,6	Формат данных 3
ПРЕОБРАЗОВАНИЕ КОДА ГРЕЯ	GRY	474	4	82,99	78,2	8-бит, двоичное число
				81,67	80,6	8-бит, двоично-десятичное число
				98,65	90,4	8-бит, угол
				97,67	100,5	15-бит, двоичное число
				98,99	114,5	15-бит, двоично-десятичное число
				110,67	112,9	15-бит, угол
				97,00	93,7	360°, двоичное число
				108,33	99,7	360°, двоично-десятичное число
				113,00	92,3	360°, угол

Примечание. Если используется операнд длиной в два слова, к значению размера, указанному в соответствующем столбце следующей таблицы, следует добавить 1.

Логические команды

Команда	Мнемони-ческое обозначение	Код	Размер (кол-во шагов) (см. примеч.)	Время выполнения (ВКЛ) (мкс)		Условия
				CP1H	CP1L	
ЛОГИЧЕСКОЕ И	ANDW	034	4	0,30	27,0	---
ДВОЙНОЕ ЛОГИЧЕСКОЕ И	ANDL	610	4	0,60	28,3	---
ЛОГИЧЕСКОЕ ИЛИ	ORW	035	4	0,45	27,3	---
ДВОЙНОЕ ЛОГИЧЕСКОЕ ИЛИ	ORWL	611	4	0,60	27,3	---
ИСКЛЮЧАЮЩЕЕ ИЛИ	XORW	036	4	0,45	26,6	---
ДВОЙНОЕ ИСКЛЮЧАЮЩЕЕ ИЛИ	XORL	612	4	0,60	27,6	---
ИСКЛЮЧАЮЩЕЕ НЕ-ИЛИ	XNRW	037	4	0,45	27,1	---
ДВОЙНОЕ ИСКЛЮЧАЮЩЕЕ НЕ-ИЛИ	XNRL	613	4	0,60	28,5	---
ДОПОЛНЕНИЕ	COM	029	2	0,45	22,1	---
ДВОЙНОЕ ДОПОЛНЕНИЕ	COML	614	2	0,80	22,5	---

Примечание. Если используется операнд длиной в два слова, к значению размера, указанному в соответствующем столбце следующей таблицы, следует добавить 1.

Специальные математические команды

Команда	Мнемоническое обозначение	Код	Размер (кол-во шагов) (см. примеч.)	Время выполнения (ВКЛ) (мкс)		Условия
				CP1H	CP1L	
КВАДРАТНЫЙ КОРЕНЬ ИЗ ДВОИЧНОГО ЧИСЛА	ROTB	620	3	43,99	44,3	---
КВАДРАТНЫЙ КОРЕНЬ ИЗ ДВОИЧНО-ДЕСЯТИЧНОГО ЧИСЛА	ROOT	072	3	49,32	43,6	---
МАТЕМАТИЧЕСКАЯ ФУНКЦИЯ	APR	069	4	13,96	20,4	Функция SIN и COS
				30,51	27,2	Кусочно-линейная интерполяция
ДЕЛЕНИЕ С ПЛАВАЮЩЕЙ ЗАПЯТОЙ	FDIV	079	4	222,90	169,6	---
ПОДСЧЕТ БИТОВ	BCNT	067	4	29,70	43,5	Подсчет битов в 1 слове

Примечание. Если используется операнд длиной в два слова, к значению размера, указанному в соответствующем столбце следующей таблицы, следует добавить 1.

Команды математических операций с плавающей запятой

Команда	Мнемоническое обозначение	Код	Размер (кол-во шагов) (см. примеч.)	Время выполнения (ВКЛ) (мкс)		Условия
				CP1H	CP1L	
ПЛАВ. ЗАПЯТАЯ В 16-БИТ	FIX	450	3	13,03	14,4	---
ПЛАВ. ЗАПЯТАЯ В 32-БИТ	FIXL	451	3	12,03	14,9	---
16-БИТ В ПЛАВ. ЗАПЯТАЯ	FLT	452	3	8,97	14,0	---
32-БИТ В ПЛАВ. ЗАПЯТАЯ	FLTL	453	3	9,92	14,4	---
СЛОЖЕНИЕ С ПЛАВ. ЗАПЯТОЙ	+F	454	4	12,60	14,3	---
ВЫЧИТАНИЕ С ПЛАВ. ЗАПЯТОЙ	–F	455	4	12,70	14,4	---
ДЕЛЕНИЕ С ПЛАВ. ЗАПЯТОЙ	/F	457	4	13,40	14,4	---
УМНОЖЕНИЕ С ПЛАВ. ЗАПЯТОЙ	*F	456	4	12,67	13,5	---
ГРАДУСЫ В РАДИАНЫ	RAD	458	3	15,00	19,7	---
РАДИАНЫ В ГРАДУСЫ	DEG	459	3	17,97	17,3	---
СИНУС	SIN	460	3	37,10	38,4	---
КОСИНУС	COS	461	3	41,97	39,8	---
ТАНГЕНС	TAN	462	3	30,86	46,7	---
АРКСИНУС	ASIN	463	3	65,14	59,0	---
АРККОСИНУС	ACOS	464	3	31,26	22,7	---
АРКТАНГЕНС	ATAN	465	3	53,07	34,1	---
КВАДРАТНЫЙ КОРЕНЬ	SQRT	466	3	20,73	22,7	---
ЭКСПОНЕНТА	EXP	467	3	53,07	55,0	---
ЛОГАРИФМ	LOG	468	3	50,08	49,1	---
ВОЗВЕДЕНИЕ В СТЕПЕНЬ	PWR	840	4	185,77	104,3	---

Команда	Мнемоническое обозначение	Код	Размер (кол-во шагов) (см. примеч.)	Время выполнения (ВКЛ) (мкс)		Условия
				CP1H	CP1L	
Сравнение чисел с плавающей запятой	LD, AND, OR +=F	329	3	11,01	13,0	---
	LD, AND, OR +<>F	330				
	LD, AND, OR +<F	331				
	LD, AND, OR +<=F	332				
	LD, AND, OR +>F	333				
	LD, AND, OR +>=F	334				
ПЛАВАЮЩАЯ ЗАПЯТАЯ В ASCII	FSTR	448	4	46,57	49,0	---
ASCII В ПЛАВАЮЩАЯ ЗАПЯТАЯ	FVAL	449	3	25,37	27,7	---

Примечание. Если используется операнд длиной в два слова, к значению размера, указанному в соответствующем столбце следующей таблицы, следует добавить 1.

Команды для чисел с плавающей запятой двойной точности

Команда	Мнемоническое обозначение	Код	Размер (кол-во шагов) (см. примеч.)	Время выполнения (ВКЛ) (мкс)		Условия
				CP1H	CP1L	
Сравнение чисел с плавающей запятой двойной точности	LD, AND, OR +=D	335	3	16,04	17,0	---
	LD, AND, OR +<>D	336				
	LD, AND, OR +<D	337				
	LD, AND, OR +<=D	338				
	LD, AND, OR +>D	339				
	LD, AND, OR +>=D	340				
ПЛАВ. ЗАПЯТАЯ ДВОЙНОЙ ТОЧНОСТИ В 16-БИТ ДВОИЧНОЕ	FIXD	841	3	15,63	18,4	---
ПЛАВ. ЗАПЯТАЯ ДВОЙНОЙ ТОЧНОСТИ В 32-БИТ ДВОИЧНОЕ	FIXLD	842	3	14,90	18,8	---
16-БИТ ДВОИЧНОЕ В ПЛАВ. ЗАПЯТАЯ ДВОЙНОЙ ТОЧНОСТИ	DBL	843	3	12,29	16,0	---
32-БИТ ДВОИЧНОЕ В ПЛАВ. ЗАПЯТАЯ ДВОЙНОЙ ТОЧНОСТИ	DBLL	844	3	14,13	16,5	---
СЛОЖЕНИЕ С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ	+D	845	4	17,89	23,5	---
ВЫЧИТАНИЕ С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ	–D	846	4	17,96	22,9	---
УМНОЖЕНИЕ С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ	*D	847	4	17,96	24,1	---
ДЕЛЕНИЕ С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ	/D	848	4	37,09	32,5	---
ГРАДУСЫ В РАДИАНЫ ДВОЙНОЙ ТОЧНОСТИ	RADD	849	3	32,07	31,3	---
РАДИАНЫ В ГРАДУСЫ ДВОЙНОЙ ТОЧНОСТИ	DEGD	850	3	33,76	31,1	---
СИНУС ДВОЙНОЙ ТОЧНОСТИ	SIND	851	3	66,97	64,2	---

Команда	Мнемоническое обозначение	Код	Размер (кол-во шагов) (см. примеч.)	Время выполнения (ВКЛ) (мкс)		Условия
				CP1H	CP1L	
КОСИНУС ДВОЙНОЙ ТОЧНОСТИ	COSD	852	3	55,89	41,4	---
ТАНГЕНС ДВОЙНОЙ ТОЧНОСТИ	TAND	853	3	85,56	85,3	---
АРКСИНУС ДВОЙНОЙ ТОЧНОСТИ	ASIND	854	3	22,64	26,6	---
АРККОСИНУС ДВОЙНОЙ ТОЧНОСТИ	ACOSD	855	3	15,64	28,6	---
АРКТАНГЕНС ДВОЙНОЙ ТОЧНОСТИ	ATAND	856	3	14,91	24,6	---
КВАДРАТНЫЙ КОРЕНЬ ДВОЙНОЙ ТОЧНОСТИ	SQRTD	857	3	46,97	45,5	---
ЭКСПОНЕНТА ДВОЙНОЙ ТОЧНОСТИ	EXPD	858	3	102,49	97,5	---
ЛОГАРИФМ ДВОЙНОЙ ТОЧНОСТИ	LOGD	859	3	19,03	22,3	---
ВОЗВЕДЕНИЕ В СТЕПЕНЬ ДВОЙНОЙ ТОЧНОСТИ	PWRD	860	4	182,83	183,0	---

Примечание. Если используется операнд длиной в два слова, к значению размера, указанному в соответствующем столбце следующей таблицы, следует добавить 1.

Команды обработки табличных данных

Команда	Мнемони-ческое обозначение	Код	Размер (кол-во шагов) (см. примеч.)	Время выполнения (ВКЛ) (мкс)		Условия
				CP1H	CP1L	
СОЗДАТЬ СТЕК	SSET	630	3	16,97	17,6	5 слов в области стека
				700,67	1200	1000 слов в области стека
ЗАПИСАТЬ В СТЕК	PUSH	632	3	14,20	10,8	---
ПЕРВЫМ ПРИШЕЛ — ПЕРВЫМ ВЫШЕЛ	FIFO	633	3	11,50	9,7	5 слов в области стека
				1,48 мс	1,80 мс	1000 слов в области стека
ПОСЛЕДНИМ ПРИШЕЛ — ПЕРВЫМ ВЫШЕЛ	LIFO	634	3	16,94	11,4	---
СОЗДАТЬ ТАБЛИЦУ УКАЗАННЫХ РАЗМЕРОВ	DIM	631	5	30,69	33,6	---
ЗАПИСАТЬ РАСПОЛОЖЕНИЕ СТРОКИ	SETR	635	4	12,82	10,7	---
СЧИТАТЬ НОМЕР СТРОКИ	GETR	636	4	15,78	12,7	---
ПОИСК ЗНАЧЕНИЯ	SRCH	181	4	29,11	41,6	Поиск 1 слова
				4,86 мс	5,12 мс	Поиск 1000 слов
ПОМЕНЯТЬ МЕСТАМИ БАЙТЫ	SWAP	637	3	22,67	27,8	Перестановка в 1 слове
				3,79 мс	6,50 мс	Перестановка в 1000 словах
НАЙТИ МАКСИМУМ	MAX	182	4	34,17	44,6	Поиск в 1 слове
				4,46 мс	6,01 мс	Поиск в 1000 слов
НАЙТИ МИНИМУМ	MIN	183	4	34,97	44,5	Поиск в 1 слове
				4,74 мс	5,98 мс	Поиск в 1000 слов
СУММА	SUM	184	4	46,63	49,3	Сложение 1 слова
				2,37 мс	2,95 мс	Сложение 1000 слов

Команда	Мнемони-ческое обозначение	Код	Размер (кол-во шагов) (см. примеч.)	Время выполнения (ВКЛ) (мкс)		Условия
				CP1H	CP1L	
КОНТРОЛЬНАЯ СУММА КАДРА	FCS	180	4	33,17	41,1	Для таблицы длиной в 1 слово
				3,30 мс	4,15 мс	Для таблицы длиной в 1000 слово
СЧИТАТЬ РАЗМЕР СТЕКА	SNUM	638	3	12,21	9,6	---
ПРОЧИТАТЬ ДАННЫЕ ИЗ СТЕКА	SREAD	639	4	14,24	12,5	---
ЗАПИСАТЬ ДАННЫЕ В СТЕК	SWRIT	640	4	13,20	12,3	---
ВСТАВИТЬ ДАННЫЕ В СТЕК	SINS	641	4	17,78	13,8	---
				758,04	480,1	Для таблицы длиной в 1000 слов
УДАЛИТЬ ДАННЫЕ ИЗ СТЕКА	SDEL	642	4	19,83	16,0	---
				763,61	471,8	Для таблицы длиной в 1000 слов

Примечание. Если используется операнд длиной в два слова, к значению размера, указанному в соответствующем столбце следующей таблицы, следует добавить 1.

Команды управления данными

Команда	Мнемони-ческое обозначение	Код	Размер (кол-во шагов) (см. примеч.)	Время выполнения (ВКЛ) (мкс)		Условия
				CP1H	CP1L	
ПИД-РЕГУЛЯТОР	PID	190	4	550,12	237,9	Первоначальное выполнение
				546,43	672,9	Измерение
				152,87	308,3	Нет измерения
ОГРАНИЧЕНИЕ	LMT	680	4	27,1	32,3	---
ЗОНА НЕЧУВСТВИТЕЛЬ-НОСТИ	BAND	681	4	27,23	33,2	---
МЕРТВАЯ ЗОНА	ZONE	682	4	26,43	32,6	---
ВЫХОД ШИМ	TPO	685	4	19,85	117,1	Условие выполнения выключено
				86,03	103,0	Условие выполнения включено, указан коэффициент заполнения или выключено ограничение выхода
				95,27	100,3	Условие выполнения включено, указана выходная переменная регулятора и включено ограничение выхода
ИЗМЕНЕНИЕ МАСШТАБА	SCL	194	4	23,30	22,0	---
ИЗМЕНЕНИЕ МАСШТАБА 2	SCL2	486	4	20,93	16,4	---
ИЗМЕНЕНИЕ МАСШТАБА 3	SCL3	487	4	24,37	21,4	---
СРЕДНЕЕ ЗНАЧЕНИЕ	AVG	195	4	63,4	70,1	Среднее за 1 цикл
				540,87	725,3	Среднее за 64 цикла
ПИД-РЕГУЛЯТОР С АВТОНАСТРОЙКОЙ	PIDAT	191	4	740,97	462,7	Первоначальное выполнение
				611,30	488,3	Измерение
				197,97	837,2	Нет измерения
				212,86	693,6	Первое выполнение автонастройки
				548,97	329,2	Автонастройка при измерении

Примечание. Если используется операнд длиной в два слова, к значению размера, указанному в соответствующем столбце следующей таблицы, следует добавить 1.

Команды управления подпрограммами

Команда	Мнемоническое обозначение	Код	Размер (кол-во шагов) (см. примеч.)	Время выполнения (ВКЛ) (мкс)		Условия
				CP1H	CP1L	
ВЫЗОВ ПОДПРОГРАММЫ	SBS	91	2	2,04	2,9	---
ВХОД В ПОДПРОГРАММУ	SBN	92	2	---	---	---
ВЫХОД ИЗ ПОДПРОГРАММЫ	RET	93	1	1,80	8,0	---
МАКРОС	MCRO	99	4	47,9	45,2	---
ВЫЗОВ ГЛОБАЛЬНОЙ ПОДПРОГРАММЫ	GSBN	751	2	---	---	---
ВХОД В ГЛОБАЛЬНУЮ ПОДПРОГРАММУ	GRET	752	1	2,04	7,9	---
ВЫХОД ИЗ ГЛОБАЛЬНОЙ ПОДПРОГРАММЫ	GSBS	750	2	1,80	2,9	---

Примечание. Если используется операнд длиной в два слова, к значению размера, указанному в соответствующем столбце следующей таблицы, следует добавить 1.

Команды управления прерываниями

Команда	Мнемоническое обозначение	Код	Размер (кол-во шагов) (см. примеч.)	Время выполнения (ВКЛ) (мкс)		Условия
				CP1H	CP1L	
УСТАНОВИТЬ МАСКУ ПРЕРЫВАНИЙ	MSKS	690	3	51,90	46,4	Установка
				63,09	77,5	Сброс
СЧИТАТЬ МАСКУ ПРЕРЫВАНИЙ	MSKR	692	3	19,99	27,4	Установка
				43,67	52,0	Сброс
ОЧИСТИТЬ ПРЕРЫВАНИЕ	CLI	691	3	49,46	52,7	Установка
				38,93	43,8	Сброс
ЗАПРЕТИТЬ ПРЕРЫВАНИЯ	DI	693	1	14,83	21,0	---
РАЗРЕШИТЬ ПРЕРЫВАНИЯ	EI	694	1	27,44	13,1	---

Примечание. Если используется операнд длиной в два слова, к значению размера, указанному в соответствующем столбце следующей таблицы, следует добавить 1.

Команды управления скоростными счетчиками и импульсными выходами

Команда	Мнемоническое обозначение	Код	Размер (кол-во шагов) (см. примеч.)	Время выполнения (ВКЛ) (мкс)		Условия
				CP1H	CP1L	
УПРАВЛЕНИЕ РЕЖИМОМ	INI	880	4	80,39	85,5	Запуск сравнения для скоростного счетчика
				47,99	55,0	Остановка сравнения для скоростного счетчика
				47,99	67,6	Изменение текущего значения импульсного выхода
				48,01	30,0	Изменение текущего значения скоростного счетчика
				27,92	36,4	Изменение текущ. знач. счетчика в режиме входа прерывания
				48,45	63,0	Прекращение выдачи импульсов
				26,08	37,8	Прекращение выдачи импульсов PWM(891)
				---	95,2	Изменение текущ. знач. для позиционирования на базе ПЧ
				---	104,7	Прекращение позиционирования на базе ПЧ
ЧТЕНИЕ ТЕКУЩ. ЗНАЧ. СКОРОСТНОГО СЧЕТЧИКА	PRV	881	4	80,39	55,2	Чтение текущего значения импульсного выхода
				40,92	79,2	Чтение текущего значения скоростного счетчика
				28,63	38,0	Чтение состояния скоростного счетчика
				39,20	48,6	Чтение состояния импульсного выхода
				66,43	78,9	Чтение состояния скоростного счетчика
				34,63	43,7	Чтение состояния выхода PWM(891)
				145,52	108,8	Чтение результатов сравнения с диапазонами для скоростного счетчика
				47,48	68,6	Чтение частоты для скоростного счетчика 0
				---	51,8	Чтение текущ. знач. для позиционирования на базе ПЧ
				---	48,0	Чтение состояния позиционирования на базе ПЧ
ПРЕОБРАЗОВАНИЕ ЧАСТОТЫ НА ВХОДЕ СЧЕТЧИКА	PRV2	883	4	20,03	28,3	---
ЗАГРУЗКА ТАБЛИЦЫ СРАВНЕНИЯ	CTBL	882	4	221,63	225,3	Регистрация таблицы заданных значений и запуск сравнения, 1 заданное значение
				9,578 мс	8,08 мс	Регистрация таблицы заданных значений и запуск сравнения, 48 заданных значений
				262,37	261,5	Регистрация таблицы диапазонов и запуск сравнения
				166,03	176,9	Только регистрация таблицы заданных значений, 1 заданное значение
				9,557 мс	8,04 мс	Только регистрация таблицы заданных значений, 48 заданных значений
				241,70	235,8	Только регистрация таблицы диапазонов
ВЫДАЧА ИМПУЛЬСОВ СКОРОСТИ	SPED	885	4	89,24	105,2	Непрерывный режим
				94,47	109,3	Независимый режим
ЗАДАТЬ КОЛ-ВО ИМПУЛЬСОВ	PULS	886	4	32,63	36,0	---
ВЫДАЧА ИМПУЛЬСОВ	PLS2	887	5	103,19	15,6	---
УПРАВЛЕНИЕ РАЗГОНОМ	ACC	888	4	111,26	119,1	Непрерывный режим
				121,73	130,9	Независимый режим

Команда	Мнемоническое обозначение	Код	Размер (кол-во шагов) (см. примеч.)	Время выполнения (ВКЛ) (мкс)		Условия
				CP1H	CP1L	
ПОИСК ИСХОДНОГО ПОЛОЖЕНИЯ	ORG	889	3	112,93	114,6	Поиск исходного положения
				98,65	113,4	Возврат в исходное положение
ИМПУЛЬСЫ С ПЕРЕМЕННОЙ СКВАЖНОСТЬЮ	PWM	891	4	30,26	21,9	---

Примечание. Если используется операнд длиной в два слова, к значению размера, указанному в соответствующем столбце следующей таблицы, следует добавить 1.

Команды для пошагового выполнения

Команда	Мнемоническое обозначение	Код	Размер (кол-во шагов) (см. примеч.)	Время выполнения (ВКЛ) (мкс)		Условия
				CP1H	CP1L	
ОПРЕДЕЛИТЬ ШАГ	STEP	008	2	36,10	15,0	Бит управления шагами включен
				18,77	24,4	Бит управления шагами выключен
ЗАПУСТИТЬ ШАГ	SNXT	009	2	10,35	8,6	---

Примечание. Если используется операнд длиной в два слова, к значению размера, указанному в соответствующем столбце следующей таблицы, следует добавить 1.

Команды для модулей ввода-вывода

Команда	Мнемоническое обозначение	Код	Разме (кол-во шагов) (см. примеч.)	Время выполнения (ВКЛ) (мкс)		Условия
				CP1H	CP1L	
ОБНОВЛЕНИЕ СЛОВ ВВОДА/ ВЫВОДА	IORF	097	3	119,50	109,7	Обновление 1 слова входа модуля CPM1A
				122,17	100,1	Обновление 1 слова выхода модуля CPM1A
				282,20	---	Обновление 1 слова входа специального модуля ввода-вывода серии CJ
				390,50	---	Обновление 1 слова выхода специального модуля ввода-вывода серии CJ
				1,58 мс	1,18 мс	Обновление 10 слов входов модуля CPM1A
				1,50 мс	1,09 мс	Обновление 10 слов выходов модуля CPM1A
				720,83	---	Обновление 60 слов входов специального модуля ввода-вывода серии CJ
				1,032 мс	---	Обновление 60 слов выходов специального модуля ввода-вывода серии CJ
ПРЕОБРАЗОВАТЬ В 7-СЕГМЕНТНЫЙ КОД	SDEC	078	4	12,53	99,8	---

Команда	Мнемони-ческое обозначение	Код	Разме (кол-во шагов) (см. примеч.)	Время выполнения (ВКЛ) (мкс)		Условия
				CP1H	CP1L	
ЧТЕНИЕ ЦИФРОВОГО ПЕРЕКЛЮЧАТЕЛЯ	DSW	210	6	85,43	93,7	4 разряда, входное значение: 0
				80,43	86,3	4 разряда, входное значение: F
				82,11	78,7	8 разрядов, входное значение: 0
				75,23	87,3	8 разрядов, входное значение: F
ЧТЕНИЕ ДЕСЯТИЧНОЙ КЛАВИАТУРЫ	TKY	211	4	17,49	25,3	Входное значение: 0
				18,69	23,7	Входное значение: F
ЧТЕНИЕ ШЕСТНАДЦАТЕ-РИЧНОЙ КЛАВИАТУРЫ	HKY	212	5	72,77	79,2	Входное значение: 0
				75,63	77,1	Входное значение: F
ЧТЕНИЕ МАТРИЦЫ	MTR	213	5	71,55	83,1	Входное значение: 0
				79,77	88,6	Входное значение: F
ВЫВОД НА 7-СЕГМЕНТНЫЙ ИНДИКАТОР	7SEG	214	5	88,23	112,5	4 разряда
				86,97	101,1	8 разрядов
ЧТЕНИЕ ИНТЕЛЛЕКТУАЛЬ-НЫХ ВХОДОВ/ ВЫХОДОВ	IORD	222	4	232,10	---	Первое выполнение
				237,10	---	Когда модуль занят
				229,57	---	В конце
ЗАПИСЬ ИНТЕЛЛЕКТУАЛЬ-НЫХ ВХОДОВ/ ВЫХОДОВ	IOWR	223	4	261,10	---	Первое выполнение
				259,10	---	Когда модуль занят
				259,77	---	В конце
ОБНОВЛЕНИЕ ШИНЫ ВВОДА/ ВЫВОДА ЦПУ	DLNK	226	4	425,69	---	Отведено 1 слово

Примечание. (1) Если используется операнд длиной в два слова, к значению размера, указанному в соответствующем столбце следующей таблицы, следует добавить 1.

(2) Время выполнения команд IORD(222) и IOWR(223) зависит от специального модуля ввода/вывода, для которого выполняются эти команды.

Команды последовательного интерфейса

Команда	Мнемони-ческое обозначение	Код	Размер (кол-во шагов) (см. примеч.)	Время выполнения (ВКЛ) (мкс)		Условия
				CP1H	CP1L	
МАКРОС ПРОТОКОЛА	PMCR	260	5	152,83	---	Передача 0 слов, прием 0 слов
				186,37	---	Передача 249 слов, прием 249 слов
ОТПРАВИТЬ	TXD	236	4	107,67	118,2	Передача 1 байта
				1,22 мс	1,31 мс	Передача 256 байтов
ПРИНЯТЬ	RXD	235	4	149,3	160,0	Сохранение 1 байта
				1,33 мс	1,45 мс	Сохранение 256 байтов
ОТПРАВИТЬ ЧЕРЕЗ МОДУЛЬ ПОСЛЕД. ИНТЕРФЕЙСА	TXDU	256	4	145,64	---	Передача 1 байта
ПРИНЯТЬ ЧЕРЕЗ МОДУЛЬ ПОСЛЕД. ИНТЕРФЕЙСА	RXDU	255	4	44,48	---	Сохранение 1 байта
ИЗМЕНИТЬ ПАРАМЕТРЫ ПОСЛЕД. ПОРТА	STUP	237	3	479,3	252,8	---

Примечание. Если используется операнд длиной в два слова, к значению размера, указанному в соответствующем столбце следующей таблицы, следует добавить 1.

Сетевые команды

Команда	Мнемоническое обозначение	Код	Размер (кол-во шагов) (см. примеч.)	Время выполнения (ВКЛ) (мкс)		Условия
				CP1H	CP1L	
ПЕРЕДАТЬ ПО СЕТИ	SEND	090	4	174,63	158,4	---
ПРИНЯТЬ ПО СЕТИ	RECV	098	4	173,97	153,8	---
ДОСТАВИТЬ КОМАНДУ	CMND	490	4	195,97	186,3	---
ОТПРАВИТЬ ЯВНОЕ СООБЩЕНИЕ	EXPLT	720	4	228,63	---	---
ЯВНЫЙ ЗАПРОС АТРИБУТА	EGATR	721	4	203,30	---	---
ЯВНАЯ УСТАНОВКА АТРИБУТА	ESATR	722	3	197,30	---	---
ЯВНОЕ ЧТЕНИЕ СЛОВА	ECHRD	723	4	188,63	---	---
ЯВНАЯ ЗАПИСЬ СЛОВА	ECHWR	724	4	181,97	---	---

Примечание. Если используется операнд длиной в два слова, к значению размера, указанному в соответствующем столбце следующей таблицы, следует добавить 1.

Команды дисплея

Команда	Мнемоническое обозначение	Код	Размер (кол-во шагов) (см. примеч.)	Время выполнения (ВКЛ) (мкс)		Условия
				CP1H	CP1L	
ОТОБРАЗИТЬ СООБЩЕНИЕ	MSG	046	3	17,16	22,6	Отображение сообщения
				15,43	22,6	Удаление отображаемого сообщения
ОТОБРАЗИТЬ ЗНАЧЕНИЕ СЛОВА НА 7-СЕГМЕНТНОМ ИНДИКАТОРЕ	SCH	047	3	48,13	---	---
УПРАВЛЕНИЕ 7-СЕГМЕНТНЫМ ИНДИКАТОРОМ	SCTRL	048	2	36,40	---	---

Примечание. Если используется операнд длиной в два слова, к значению размера, указанному в соответствующем столбце следующей таблицы, следует добавить 1.

Команды управления часами

Команда	Мнемоническое обозначение	Код	Размер (кол-во шагов) (см. примеч.)	Время выполнения (ВКЛ) (мкс)		Условия
				CP1H	CP1L	
ДОБАВИТЬ К КАЛЕНДАРНОЙ ДАТЕ	CADD	730	4	212,90	219,9	---
ВЫЧЕСТЬ ИЗ КАЛЕНДАРНОЙ ДАТЫ	CSUB	731	4	176,23	223,3	---
ЧАСЫ В СЕКУНДЫ	SEC	065	3	34,19	37,0	---
СЕКУНДЫ В ЧАСЫ	HMS	066	3	40,95	44,1	---
КОРРЕКТИРОВКА ЧАСОВ	DATE	735	2	134,67	123,7	---

Примечание. Если используется операнд длиной в два слова, к значению размера, указанному в соответствующем столбце следующей таблицы, следует добавить 1.

Команды для отладки программы

Команда	Мнемоническое обозначение	Код	Размер (кол-во шагов) (см. примеч.)	Время выполнения (ВКЛ) (мкс)		Условия
				CP1H	CP1L	
ОТБОР ДАННЫХ ДЛЯ ПАМЯТИ ПРОТОКОЛА ДАННЫХ	TRSM	045	1	201,33	30,3	Регистрация 1 бита и 0 слов
				1,12 мс	158,7	Регистрация 31 бита и 6 слов

Примечание. Если используется операнд длиной в два слова, к значению размера, указанному в соответствующем столбце следующей таблицы, следует добавить 1.

Команды для диагностики неисправностей

Команда	Мнемоническое обозначение	Код	Размер (кол-во шагов) (см. примеч.)	Время выполнения (ВКЛ) (мкс)		Условия
				CP1H	CP1L	
СИГНАЛИЗАЦИЯ НЕИСПРАВНОСТИ	FAL	006	3	23,24	27,3	Регистрация ошибок
				266,57	406,7	Удаление ошибок (в порядке приоритета)
				817,17	610,7	Удаление ошибок (всех)
				305,33	410,0	Удаление ошибок (отдельных)
СИГНАЛИЗАЦИЯ СЕРЬЕЗНОЙ НЕИСПРАВНОСТИ	FALS	007	3	---	---	---
ОБНАРУЖЕНИЕ НЕИСПРАВНОГО КАНАЛА	FPD	269	4	245,07	286,7	При выполнении
				258,2	362,4	В первый раз
				317,73	363,1	При выполнении
				316,4	400	В первый раз

Примечание. Если используется операнд длиной в два слова, к значению размера, указанному в соответствующем столбце следующей таблицы, следует добавить 1.

Прочие команды

Команда	Мнемоническое обозначение	Код	Размер (кол-во шагов) (см. примеч.)	Время выполнения (ВКЛ) (мкс)		Условия
				CP1H	CP1L	
УСТАНОВИТЬ ПЕРЕНОС	STC	040	1	0,15	10,0	---
СБРОСИТЬ ПЕРЕНОС	CLC	041	1	0,15	10,2	---
УВЕЛИЧИТЬ МАКС. ВРЕМЯ ЦИКЛА	WDT	094	2	23,94	24,2	---
СОХРАНИТЬ ФЛАГИ УСЛОВИЙ	CCS	282	1	14,97	17,5	---
ЗАГРУЗИТЬ ФЛАГИ УСЛОВИЙ	CCL	283	1	17,83	18,9	---
ПРЕОБРАЗОВАТЬ АДРЕС СЕРИИ CV	FRMCV	284	3	31,03	29,1	---
ПРЕОБРАЗОВАТЬ В АДРЕС СЕРИИ CV	TOCV	285	3	34,90	28,0	---

Примечание. Если используется операнд длиной в два слова, к значению размера, указанному в соответствующем столбце следующей таблицы, следует добавить 1.

Команды для программирования блоков

Команда	Мнемони-ческое обозначение	Код	Размер (кол-во шагов) (см. примеч.)	Время выполнения (ВКЛ) (мкс)		Условия
				CP1H	CP1L	
НАЧАЛО ПРОГРАММНОГО БЛОКА	BPRG	096	2	26,59	9,2	---
КОНЕЦ ПРОГРАММНОГО БЛОКА	BEND	801	1	24,19	6,4	---
ПРИОСТАНОВКА ПРОГРАММНОГО БЛОКА	BPPS	811	2	18,13	7,2	---
ПЕРЕЗАПУСК ПРОГРАММНОГО БЛОКА	BPRS	812	2	9,29	6,8	---
ВЫХОД ИЗ БЛОКА ПО УСЛОВИЮ	(Условие выполнения) EXIT	806	1	23,33	8,0	Условие ВЫХОДА выполняется
				9,33	5,8	Условие ВЫХОДА не выполняется
ВЫХОД ИЗ БЛОКА ПО УСЛОВИЮ	EXIT (адрес бита)	806	2	26,78	10,8	Условие ВЫХОДА выполняется
				11,47	8,4	Условие ВЫХОДА не выполняется
ВЫХОД ИЗ БЛОКА ПО УСЛОВИЮ (НЕ)	EXIT NOT (адрес бита)	806	2	26,74	9,0	Условие ВЫХОДА выполняется
				11,41	11,2	Условие ВЫХОДА не выполняется
Ветвление	IF (условие выполнения)	802	1	7,4	4,0	Если «истина»
				13,5	6,6	Если «ложь»
Ветвление	IF (адрес бита)	802	2	11,55	7,6	Если «истина»
				13,55	10,2	Если «ложь»
Ветвление (НЕ)	IF NOT (адрес бита)	802	2	11,61	10,6	Если «истина»
				13,61	8,0	Если «ложь»
Ветвление	ELSE	803	1	7,71	6,0	Если «истина»
				13,55	4,2	Если «ложь»
Ветвление	IEND	804	1	13,58	4,6	Если «истина»
				7,49	6,1	Если «ложь»
ОДИН ЦИКЛ И ОЖИДАНИЕ	WAIT (условие выполнения)	805	1	27,53	3,7	Условие ОЖИДАНИЯ выполняется
				6,15	9,3	Условие ОЖИДАНИЯ не выполняется
ОДИН ЦИКЛ И ОЖИДАНИЕ	WAIT (адрес бита)	805	2	28,78	7,4	Условие ОЖИДАНИЯ выполняется
				9,82	10,8	Условие ОЖИДАНИЯ не выполняется
ОДИН ЦИКЛ И ОЖИДАНИЕ (НЕ)	WAIT NOT (адрес бита)	805	2	26,27	11,2	Условие ОЖИДАНИЯ выполняется
				9,78	7,8	Условие ОЖИДАНИЯ не выполняется
СЧЕТЧИК ОЖИДАНИЯ	CNTW	814	4	36,57	17,6	Первое выполнение
				36,40	16,2	Нормальное выполнение
	CNTWX	818	4	43,69	18,1	Первое выполнение
				36,95	14,8	Нормальное выполнение
СКОРОСТНОЙ ТАЙМЕР ОЖИДАНИЯ	TMHW	815	3	48,37	21,4	Первое выполнение
				48,20	14,0	Нормальное выполнение
	TMHWX	817	3	50,59	19,6	Первое выполнение
				45,52	13,6	Нормальное выполнение
Управление циклом	LOOP	809	1	17,03	6,3	---

Команда	Мнемони-ческое обозначение	Код	Размер (кол-во шагов) (см. примеч.)	Время выполнения (ВКЛ) (мкс)		Условия
				CP1H	CP1L	
Управление циклом	LEND (условие выполнения)	810	1	17,13	6,5	Условие конца цикла выполняется
				18,07	6,0	Условие конца цикла не выполняется
Управление циклом	LEND (адрес бита)	810	2	20,77	9,5	Условие конца цикла выполняется
				23,63	9,2	Условие конца цикла не выполняется
Управление циклом	LEND NOT (адрес бита)	810	2	23,43	9,9	Условие конца цикла выполняется
				20,97	9,5	Условие конца цикла не выполняется
ТАЙМЕР ОЖИДАНИЯ	TIMW	813	3	48,40	19,5	Настройка по умолчанию
				46,33	13,6	Нормальное выполнение
	TIMWX	816	3	48,02	19,4	Настройка по умолчанию
				47,09	13,5	Нормальное выполнение

Примечание. Если используется операнд длиной в два слова, к значению размера, указанному в соответствующем столбце следующей таблицы, следует добавить 1.

Команды обработки текстовых строк

Команда	Мнемоническое обозначение	Код	Размер (кол-во шагов) (см. примеч.)	Время выполнения (ВКЛ) (мкс)		Условия
				CP1H	CP1L	
КОПИРОВАТЬ СТРОКУ	MOV$	664	3	68,44	78,9	Копирование строки длиной в 1 символ
ОБЪЕДИНИТЬ СТРОКИ	+$	656	4	145,10	146,0	1 символ + 1 символ
СКОПИРОВАТЬ СИМВОЛЫ СЛЕВА	LEFT$	652	4	87,81	94,7	Считывание 1 символа из строки длиной в 2 символа
СКОПИРОВАТЬ СИМВОЛЫ СПРАВА	RGHT$	653	4	91,81	95,4	Считывание 1 символа из строки длиной в 2 символа
СКОПИРОВАТЬ СИМВОЛЫ ВНУТРИ СТРОКИ	MID$	654	5	94,77	97,2	Считывание 1 символа из строки длиной в 3 символа
НАЙТИ В СТРОКЕ	FIND$	660	4	82,81	99,1	Поиск 1 символа в строке длиной в 2 символа
ДЛИНА СТРОКИ	LEN$	650	3	32,61	38,4	Для строки длиной в 1 символ
ЗАМЕНИТЬ В СТРОКЕ	RPLC$	661	6	269,43	289,7	Замена двух первых символов одним символом
УДАЛИТЬ СТРОКУ	DEL$	658	5	114,00	254,5	Удаление первого символа в строке из двух символов
ОБМЕН СТРОКАМИ	XCHG$	665	3	108,54	138,7	Обмен строками длиной в 1 символ
ОЧИСТИТЬ СТРОКУ	CLR$	666	2	37,33	39,3	Очистить строку длиной в 1 символ
ВСТАВИТЬ В СТРОКУ	INS$	657	5	199,43	253,3	Вставка 1 символа после двух первых символов

Команда	Мнемоническое обозначение	Код	Размер (кол-во шагов) (см. примеч.)	Время выполнения (ВКЛ) (мкс)		Условия
				CP1H	CP1L	
Команды сравнения строк	LD, AND, OR +=$	670	4	64,47	63,6	Сравнение строк длиной в 1 символ
	LD, AND, OR +<>$	671				
	LD, AND, OR +<$	672				
	LD, AND, OR +<=$	673				
	LD, AND, OR +>$	674				
	LD, AND, OR +>=$	675				

Примечание. Если используется операнд длиной в два слова, к значению размера, указанному в соответствующем столбце следующей таблицы, следует добавить 1.

Команды управления задачами

Команда	Мнемони-ческое обозначение	Код	Размер (кол-во шагов) (см. примеч.)	Время выполнения (ВКЛ) (мкс)		Условия
				CP1H	CP1L	
ВКЛЮЧИТЬ ЗАДАЧУ	TKON	820	2	30,65	23,2	---
ВЫКЛЮЧИТЬ ЗАДАЧУ	TKOF	821	2	18,30	23,4	---

Примечание. Если используется операнд длиной в два слова, к значению размера, указанному в соответствующем столбце следующей таблицы, следует добавить 1.

Команды преобразования моделей

Команда	Мнемони-ческое обозначение	Код	Размер (кол-во шагов) (см. примеч.)	Время выполнения (ВКЛ) (мкс)		Условия
				CP1H	CP1L	
ПЕРЕДАЧА БЛОКА	XFERC	565	4	37,04	24,5	Передача 1 слова
				2,922 мс	4,65 мс	Передача 1000 слов
РАЗМЕЩЕНИЕ ОДИНОЧНОГО СЛОВА	DISTC	566	4	24,80	34,9	Размещение значения
				35,57	45,0	Операция стека
СБОР ДАННЫХ	COLLC	567	4	29,83	36,6	Размещение значения
				30,13	39,2	Операция стека
				31,10	39,2	Операция стека Чтение 1 слова, FIFO
				8,100 мс	14,7 мс	Операция стека Чтение 1000 слова, FIFO
ПЕРЕДАТЬ БИТ	MOVBC	568	4	28,03	34,9	---
ПОДСЧЕТ БИТОВ	BCNTC	621	4	32,97	41,5	Подсчет битов в 1 слове
				5,703 мс	6,09 мс	Подсчет битов в 1000 слов

Примечание. Если используется операнд длиной в два слова, к значению размера, указанному в соответствующем столбце следующей таблицы, следует добавить 1.

Команды специальных функциональных блоков

Команда	Мнемони-ческое обозначение	Код	Размер (кол-во шагов) (см. примеч.)	Время выполнения (ВКЛ) (мкс)		Условия
				CP1H	CP1L	
СЧИТАТЬ ИДЕНТИФИКАТОР ПЕРЕМЕННОЙ	GETID	286	4	26,5	32,8	---

Примечание. Если используется операнд длиной в два слова, к значению размера, указанному в соответствующем столбце следующей таблицы, следует добавить 1.

4-2 Время выполнения экземпляра функционального блока

Влияние выполняемого экземпляра функционального блока на общую длительность цикла выполнения программы пользователя может быть определено по следующей формуле.

Влияние экземпляра функц. блока на время цикла
= Время запуска (A)
+ Время обмена входными/выходными параметрами (B)
+ Время выполнения команд функц. блока (C)

Длительность каждого из этапов A, B и C указана в следующей таблице.

Операция			Модуль ЦПУ CP1H	Модуль ЦПУ CP1L
A	Время запуска	Время запуска (не включает обмен входными и выходными параметрами)	6,8 мкс	320,4 мкс
B	Время обмена входными и выходными параметрами (В скобках указан тип данных.)	Вх./вых. переменная размером в 1 бит (BOOL)	0,4 мкс	59,52 мкс
		Вх./вых. переменная размером в 1 слово (INT, UINT, WORD)	0,3 мкс	13,16 мкс
		Вх./вых. переменная размером в 2 слова (DINT, UDINT, DWORD, REAL)	0,5 мкс	15,08 мкс
		Вх./вых. переменная размером в 4 слова (LINT, ULINT, LWORD, LREAL)	1,0 мкс	30,16 мкс
C	Время выполнения команд функц. блока	Общее время выполнения команд (такое же, как и у обычной программы пользователя)		

Пример:
Входные переменные длиной в 1 слово (INT): 3
Выходные переменные длиной в 1 слово (INT): 2
Общее время выполнения команд функц. блока: 10 мкс
Время выполнения 1 экземпляра = 6,8 мкс + (3 + 2) × 0,3 мкс + 10 мкс = 18,3 мкс

Примечание. Если одно и то же определение функционального блока используется в нескольких местах программы, время выполнения возрастает пропорционально количеству экземпляров.

Количество шагов программы, занимаемое функциональными блоками

Количество шагов программы, занимаемое всеми экземплярами функционального блока, которые используются в программе пользователя, можно рассчитать по следующей формуле.

Количество шагов
= Количество экземпляров × (Размер секции вызова (m) + Размер секции обмена вх./вых. параметрами (n) × Количество параметров) + Количество шагов команд в функциональном блоке (p) (см. примечание)

Примечание. Если в программе пользователя используется несколько экземпляров одного и того же функционального блока, второй, третий и все последующие экземпляры функционального блока не будут увеличивать общее число шагов команд. По этой причине в приведенной выше формуле количество шагов команд функционального блока не умножается на количество экземпляров (p).

Содержание			Модули ЦПУ серии CP/CS/CJ
m	Секция вызова		57 шагов
n	Секция обмена вх./вых. параметрами (В скобках указан тип данных.)	Вх./вых. переменная размером в 1 бит (BOOL)	6 шагов
		Вх./вых. переменная размером в 1 слово (INT, UINT, WORD)	6 шагов
		Вх./вых. переменная размером в 2 слова (DINT, UDINT, DWORD, REAL)	6 шагов
		Вх./вых. переменная размером в 4 слова (LINT, ULINT, LWORD, LREAL)	12 шагов
p	Количество шагов команд в функц. блоке	Общее количество шагов команд (такое же, как и у обычной программы пользователя) + 27 шагов.	

Пример:
Входные переменные длиной в 1 слово (INT): 5
Выходные переменные длиной в 1 слово (INT): 5
Определение функционального блока: 100 шагов
Количество шагов в 1 экземпляре = 57 + (5 + 5) × 6 шагов + 100 шагов + 27 шагов = 244 шага

Предварительная версия
Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

Приложение A

Классификация команд по функциям

В следующей таблице перечислены команды программируемых контроллеров серии CP, сгруппированные по выполняемым функциям (в таком же порядке команды описываются в разделе 3 *Команды*).

Классификация	Подкласс	Мнем. обознач.	Команда	Мнем. обознач.	Команда	Мнем. обознач.	Команда
Базовые команды	Вход	LD	ЗАГРУЗКА	LD NOT	ЗАГРУЗКА НЕ	AND	И
		AND NOT	И НЕ	OR	ИЛИ	OR NOT	ИЛИ НЕ
		AND LD	И ЗАГРУЗКА	OR LD	ИЛИ ЗАГРУЗКА	---	---
	Выход	OUT	ВЫВОД	OUT NOT	ВЫВОД НЕ	---	---
Входные битовые команды	---	NOT	НЕ	UP	УСЛОВИЕ ВКЛ	DOWN	УСЛОВИЕ ВЫКЛ
	Проверка бита	LD TST	ЗАГРУЗКА ПРОВЕРКА БИТА	LD TSTN	ЗАГРУЗКА ПРОВЕРКА БИТА НЕ	AND TST	И ПРОВЕРКА БИТА НЕ
		AND TSTN	И ПРОВЕРКА БИТА НЕ	OR TST	ИЛИ ПРОВЕРКА БИТА	OR TSTN	ИЛИ ПРОВЕРКА БИТА НЕ
Выходные битовые команды	---	KEEP	ЗАЩЕЛКА	DIFU	ПОЛОЖИТЕЛЬНЫЙ ФРОНТ	DIFD	ОТРИЦАТЕЛЬНЫЙ ФРОНТ
		OUTB	ВЫВОД В ОДИН БИТ	---	---	---	---
	Установка/сброс	SET	УСТАНОВКА	RSET	СБРОС	SETA	УСТАНОВИТЬ НЕСКОЛЬКО БИТОВ
		RSTA	СБРОСИТЬ НЕСКОЛЬКО БИТОВ	SETB	УСТАНОВИТЬ ОДИН БИТ	RSTB	СБРОСИТЬ ОДИН БИТ
Команды управления последовательностью выполнения	---	END	КОНЕЦ	NOP	ПУСТАЯ ОПЕРАЦИЯ	---	---
	Блокировка	IL	БЛОКИРОВКА	ILC	ОТМЕНА БЛОКИРОВКИ	MILH	ВЛОЖЕННАЯ БЛОКИРОВКА С ЗАПОМИНАНИЕМ ФРОНТА
		MILR	ВЛОЖЕННАЯ БЛОКИРОВКА БЕЗ ЗАПОМИНАНИЯ ФРОНТА	MILC	ОТМЕНА ВЛОЖЕННОЙ БЛОКИРОВКИ	---	---
	Переход	JMP	ПЕРЕХОД	JME	КОНЕЦ ПЕРЕХОДА	CJP	УСЛОВНЫЙ ПЕРЕХОД
		CJPN	УСЛОВНЫЙ ПЕРЕХОД	JMP0	МНОЖЕСТВЕННЫЙ ПЕРЕХОД	JME0	КОНЕЦ МНОЖЕСТВЕННОГО ПЕРЕХОДА
	Повтор	FOR	ЦИКЛЫ FOR-NEXT	BREAK	ВЫХОД ИЗ ЦИКЛА	NEXT	ЦИКЛЫ FOR-NEXT

Классификация	Подкласс		Мнем. обознач.	Команда	Мнем. обознач.	Команда	Мнем. обознач.	Команда
Команды управления таймерами и счетчиками	BCD (см. примеч.)	Таймер (с номером таймера)	TIM	ТАЙМЕР	TIMH	СКОРОСТНОЙ ТАЙМЕР	TMHH	1 мс ТАЙМЕР
			TTIM	НАКАПЛИВАЮЩИЙ ТАЙМЕР	---	---	---	---
		Таймер (без номера таймера)	TIML	ДОЛГОСРОЧНЫЙ ТАЙМЕР	MTIM	МНОГОВЫХОДНОЙ ТАЙМЕР	---	---
		Счетчик (с номером счетчика)	CNT	СЧЕТЧИК	CNTR	РЕВЕРСИВНЫЙ СЧЕТЧИК	CNR	СБРОС ТАЙМЕРА/СЧЕТЧИКА
	Двоичный (см. примеч.)	Таймер (с номером таймера)	TIMX	ТАЙМЕР	TIMHX	СКОРОСТНОЙ ТАЙМЕР	TMHHX	1 мс ТАЙМЕР
			TTIMX	НАКАПЛИВАЮЩИЙ ТАЙМЕР	---	---	---	---
		Таймер (без номера таймера)	TIMLX	ДОЛГОСРОЧНЫЙ ТАЙМЕР	MTIMX	МНОГОВЫХОДНОЙ ТАЙМЕР	---	---
		Счетчик (с номером счетчика)	CNTX	СЧЕТЧИК	CNTRX	РЕВЕРСИВНЫЙ СЧЕТЧИК	CNRX	СБРОС ТАЙМЕРА/СЧЕТЧИКА
Команды сравнения	Символы сравнения		LD, AND, OR + =, <>, <, <=, >, >=	Символы сравнения (без знака)	LD, AND, OR + =, <>, <, <=, >, >= + L	Символы сравнения (двойное слово, без знака)	LD, AND, OR + =, <>, <, <=, >, >= +S	Символы сравнения (со знаком)
			LD, AND, OR + =, <>, <, <=, >, >= + SL	Символы сравнения (двойное слово, со знаком)	LD, AND, OR + = DT, <> DT, < DT, <= DT, > DT, >= DT (см. примеч. 1)	Сравнение времени	---	---
	Сравнение значений (флаги условий)		CMP	СРАВНИТЬ БЕЗ ЗНАКА	CMPL	СРАВНИТЬ ДВОЙНЫЕ СЛОВА БЕЗ ЗНАКА	CPS	СРАВНИТЬ ДВОИЧНЫЕ ЗНАЧЕНИЯ СО ЗНАКОМ
			CPSL	СРАВНИТЬ ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ	ZCP	СРАВНИТЬ СЛОВО С ДИАПАЗОНОМ	ZCPL	СРАВНИТЬ ДВОЙНОЕ СЛОВО С ДИАПАЗОНОМ
	Таблица сравнения		MCMP	СРАВНИТЬ НЕСКОЛЬКО ЗНАЧЕНИЙ	TCMP	СРАВНИТЬ С ТАБЛИЦЕЙ	BCMP	СРАВНИТЬ С ДИАПАЗОНАМИ БЕЗ ЗНАКА
			BCMP2	РАСШИРЕННОЕ СРАВНЕНИЕ С ДИАПАЗОНАМИ	---	---	---	---

Классификация	Подкласс	Мнем. обознач.	Команда	Мнем. обознач.	Команда	Мнем. обознач.	Команда
Команды передачи данных	Одиночное/двойное слово	MOV	ПЕРЕДАТЬ	MOVL	ПЕРЕДАТЬ ДВОЙНОЕ СЛОВО	MVN	ПЕРЕДАТЬ ДОПОЛНЕНИЕ
		MVNL	ПЕРЕДАТЬ ДОПОЛНЕНИЕ К ДВОЙНОМУ СЛОВУ	---	---	---	---
	Бит/разряд	MOVB	ПЕРЕДАТЬ БИТ	MOVD	ПЕРЕДАТЬ РАЗРЯД	---	---
	Обмен	XCHG	ОБМЕН ДАННЫМИ	XCGL	ОБМЕН ДВОЙНЫМИ СЛОВАМИ	---	---
	Передача блока/бита	XFRB	ПЕРЕДАТЬ НЕСКОЛЬКО БИТОВ	XFER	ПЕРЕДАЧА БЛОКА	BSET	ЗАПОЛНЕНИЕ БЛОКА
	Распределение/сбор	DIST	РАЗМЕЩЕНИЕ ОДИНОЧНОГО СЛОВА	COLL	СБОР ДАННЫХ	---	---
	Регистр указателей	MOVR	ПЕРЕДАТЬ В РЕГИСТР	MOVRW	ПЕРЕДАТЬ ТЕКУЩЕЕ ЗНАЧЕНИЕ ТАЙМЕРА/СЧЕТЧИКА В РЕГИСТР	---	---
Команды сдвига данных	Сдвиг 1 бита	SFT	РЕГИСТР СДВИГА	SFTR	РЕВЕРСИВНЫЙ РЕГИСТР СДВИГА	ASLL	СДВИГ ДВОЙНОГО СЛОВА ВЛЕВО
		ASL	АРИФМЕТИЧЕСКИЙ СДВИГ ВЛЕВО	ASR	АРИФМЕТИЧЕСКИЙ СДВИГ ВПРАВО	ASRL	СДВИГ ДВОЙНОГО СЛОВА ВПРАВО
	Асинхронный сдвиг	ASFT	АСИНХРОННЫЙ РЕГИСТР СДВИГА	---	---	---	---
	Сдвиг слова	WSFT	СДВИГ ПО СЛОВАМ	---	---	---	---
	Циклический сдвиг 1 бита	ROL	ЦИКЛИЧЕСКИЙ СДВИГ ВЛЕВО	ROLL	ЦИКЛИЧЕСКИЙ СДВИГ ДВОЙНОГО СЛОВА ВЛЕВО	RLNC	ЦИКЛИЧЕСКИЙ СДВИГ ВЛЕВО БЕЗ ПЕРЕНОСА
		RLNL	ЦИКЛИЧЕСКИЙ СДВИГ ДВОЙНОГО СЛОВА ВЛЕВО БЕЗ ПЕРЕНОСА	ROR	ЦИКЛИЧЕСКИЙ СДВИГ ВПРАВО	RORL	ЦИКЛИЧЕСКИЙ СДВИГ ДВОЙНОГО СЛОВА ВПРАВО
		RRNC	ЦИКЛИЧЕСКИЙ СДВИГ ВПРАВО БЕЗ ПЕРЕНОСА	RRNL	ЦИКЛИЧЕСКИЙ СДВИГ ДВОЙНОГО СЛОВА ВПРАВО БЕЗ ПЕРЕНОСА	---	---
	Сдвиг 1 разряда	SLD	СДВИГ НА ОДИН РАЗРЯД ВЛЕВО	SRD	СДВИГ НА ОДИН РАЗРЯД ВПРАВО	---	---
	Сдвиг n-битового значения	NSFL	СДВИГ N БИТОВ ДАННЫХ ВЛЕВО	NSFR	СДВИГ N БИТОВ ДАННЫХ ВПРАВО	---	---
	Сдвиг n битов	NASL	СДВИГ НА N БИТОВ ВЛЕВО	NSLL	СДВИГ ДВОЙНОГО СЛОВА НА N БИТОВ ВЛЕВО	NASR	СДВИГ НА N БИТОВ ВПРАВО
		NSRL	СДВИГ ДВОЙНОГО СЛОВА НА N БИТОВ ВПРАВО	---	---	---	---

Классификация	Подкласс	Мнем. обознач.	Команда	Мнем. обознач.	Команда	Мнем. обознач.	Команда
Команды увеличения/ уменьшения	BCD	++B	УВЕЛИЧИТЬ ДВОИЧНО-ДЕСЯТИЧНОЕ СЛОВО	++BL	УВЕЛИЧИТЬ ДВОЙНОЕ ДВОИЧНО-ДЕСЯТИЧНОЕ СЛОВО	– –B	УМЕНЬШИТЬ ДВОИЧНО-ДЕСЯТИЧНОЕ СЛОВО
		– –BL	УМЕНЬШИТЬ ДВОЙНОЕ ДВОИЧНО-ДЕСЯТИЧНОЕ СЛОВО	---	---	---	---
	Двоичный	++	УВЕЛИЧИТЬ ДВОИЧНОЕ СЛОВО	++L	УВЕЛИЧИТЬ ДВОЙНОЕ ДВОИЧНОЕ СЛОВО	– –	УМЕНЬШИТЬ ДВОИЧНОЕ СЛОВО
		– –L	УМЕНЬШИТЬ ДВОЙНОЕ ДВОИЧНОЕ СЛОВО	---	---	---	---
Символьные математические команды	Сложение двоичных значений	+	СЛОЖИТЬ БЕЗ ПЕРЕНОСА ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ	+L	СЛОЖИТЬ БЕЗ ПЕРЕНОСА ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ	+C	СЛОЖИТЬ С ПЕРЕНОСОМ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ
		+CL	СЛОЖИТЬ С ПЕРЕНОСОМ ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ	---	---	---	---
	Сложение BCD-значений	+B	СЛОЖИТЬ БЕЗ ПЕРЕНОСА ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА	+BL	СЛОЖИТЬ БЕЗ ПЕРЕНОСА ДВОЙНЫЕ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА	+BC	СЛОЖИТЬ С ПЕРЕНОСОМ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА
		+BCL	СЛОЖИТЬ С ПЕРЕНОСОМ ДВОЙНЫЕ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА	---	---	---	---
	Вычитание двоичных значений	–	ВЫЧЕСТЬ БЕЗ ПЕРЕНОСА ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ	–L	ВЫЧЕСТЬ БЕЗ ПЕРЕНОСА ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ	–C	ВЫЧЕСТЬ С ПЕРЕНОСОМ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ
		–CL	ВЫЧЕСТЬ С ПЕРЕНОСОМ ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ	---	---	---	---
	Вычитание BCD-значений	–B	ВЫЧЕСТЬ БЕЗ ПЕРЕНОСА ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА	–BL	ВЫЧЕСТЬ БЕЗ ПЕРЕНОСА ДВОЙНЫЕ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА	–BC	ВЫЧЕСТЬ С ПЕРЕНОСОМ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА
		–BCL	ВЫЧЕСТЬ С ПЕРЕНОСОМ ДВОЙНЫЕ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА	---	---	---	---
	Умножение двоичных значений	*	УМНОЖИТЬ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ	*L	УМНОЖИТЬ ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ	*U	УМНОЖИТЬ ДВОИЧНЫЕ СЛОВА БЕЗ ЗНАКА
		*UL	УМНОЖИТЬ ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА БЕЗ ЗНАКА	---	---	---	---

Классификация	Подкласс	Мнем. обознач.	Команда	Мнем. обознач.	Команда	Мнем. обознач.	Команда
Символьные математические команды	Умножение BCD-значений	*B	УМНОЖИТЬ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА	*BL	УМНОЖИТЬ ДВОЙНЫЕ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА	---	---
	Деление двоичных значений	/	ДЕЛИТЬ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ	/L	ДЕЛИТЬ ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ	/U	ДЕЛИТЬ ДВОИЧНЫЕ СЛОВА БЕЗ ЗНАКА
		/UL	ДЕЛИТЬ ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА БЕЗ ЗНАКА	---	---	---	---
	Деление BCD-значений	/B	ДЕЛИТЬ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА	/BL	ДЕЛИТЬ ДВОЙНЫЕ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА	---	---
Команды преобразования	Преобразование BCD<->двоичное	BIN	BCD В ДВОИЧНОЕ	BINL	ДВОЙНОЕ BCD В ДВОЙНОЕ ДВОИЧНОЕ	BCD	ДВОИЧНОЕ В BCD
		BCDL	ДВОЙНОЕ ДВОИЧНОЕ В ДВОЙНОЕ BCD	NEG	ДОПОЛНЕНИЕ ДО 2	NEGL	ДОПОЛНЕНИЕ ДВОЙНОГО СЛОВА ДО 2
		SIGN	16-БИТОВОЕ ДВОИЧНОЕ СО ЗНАКОМ В 32-БИТОВОЕ	---	---	---	---
	Декодирование/ кодирование	MLPX	ДЕКОДИРОВАНИЕ ДАННЫХ	DMPX	КОДИРОВАНИЕ ДАННЫХ	---	---
	Преобразование ASCII<->HEX	ASC	ПРЕОБРАЗОВАНИЕ В ASCII	HEX	ASCII В HEX	---	---
	Преобразование строка<->столбец	LINE	СТОЛБЕЦ В СТРОКУ	COLM	СТРОКУ В СТОЛБЕЦ	---	---
	Преобразование двоичное со знаком<->BCD	BINS	BCD СО ЗНАКОМ В ДВОИЧНОЕ	BISL	ДВОЙНОЕ BCD СО ЗНАКОМ В ДВОИЧНОЕ	BCDS	ДВОИЧНОЕ СО ЗНАКОМ В BCD
		BDSL	ДВОЙНОЕ ДВОИЧНОЕ СО ЗНАКОМ В BCD	---	---	---	---
	Преобразование кода Грея	GRY	ПРЕОБРАЗОВАНИЕ КОДА ГРЕЯ	---	---	---	---
Логические команды	Логическое И/ИЛИ	ANDW	ЛОГИЧЕСКОЕ И	ANDL	ДВОЙНОЕ ЛОГИЧЕСКОЕ И	ORW	ЛОГИЧЕСКОЕ ИЛИ
		ORWL	ДВОЙНОЕ ЛОГИЧЕСКОЕ ИЛИ	XORW	ИСКЛЮЧАЮЩЕЕ ИЛИ	XORL	ДВОЙНОЕ ИСКЛЮЧАЮЩЕЕ ИЛИ
		XNRW	ИСКЛЮЧАЮЩЕЕ НЕ-ИЛИ	XNRL	ДВОЙНОЕ ИСКЛЮЧАЮЩЕЕ НЕ-ИЛИ	---	---
	Дополнение	COM	ДОПОЛНЕНИЕ	COML	ДВОЙНОЕ ДОПОЛНЕНИЕ	---	---
Специальные математические команды	---	ROTB	КВАДРАТНЫЙ КОРЕНЬ ИЗ ДВОИЧНОГО ЧИСЛА	ROOT	КВАДРАТНЫЙ КОРЕНЬ ИЗ ДВОИЧНО-ДЕСЯТИЧНОГО ЧИСЛА	APR	МАТЕМАТИЧЕСКАЯ ФУНКЦИЯ
		FDIV	ДЕЛЕНИЕ С ПЛАВАЮЩЕЙ ЗАПЯТОЙ	BCNT	ПОДСЧЕТ БИТОВ	---	---

Классификация	Подкласс	Мнем. обознач.	Команда	Мнем. обознач.	Команда	Мнем. обознач.	Команда
Команды математических операций с плавающей запятой	Преобразование плавающая запятая<->двоичный формат	FIX	ПЛАВ. ЗАПЯТАЯ В 16-БИТ	FIXL	ПЛАВ. ЗАПЯТАЯ В 32-БИТ	FLT	16-БИТ В ПЛАВ. ЗАПЯТАЯ
		FLTL	32-БИТ В ПЛАВ. ЗАПЯТАЯ	---	---	---	---
	Арифметические операции с плавающей запятой	+F	СЛОЖЕНИЕ С ПЛАВ. ЗАПЯТОЙ	–F	ВЫЧИТАНИЕ С ПЛАВ. ЗАПЯТОЙ	/F	ДЕЛЕНИЕ С ПЛАВ. ЗАПЯТОЙ
		*F	УМНОЖЕНИЕ С ПЛАВ. ЗАПЯТОЙ	---	---	---	---
	Тригонометрические функции с плавающей запятой	RAD	ГРАДУСЫ В РАДИАНЫ	DEG	РАДИАНЫ В ГРАДУСЫ	SIN	СИНУС
		COS	КОСИНУС	TAN	ТАНГЕНС	ASIN	АРКСИНУС
		ACOS	АРККОСИНУС	ATAN	АРКТАНГЕНС	---	---
	Математические операции с плавающей запятой	SQRT	КВАДРАТНЫЙ КОРЕНЬ	EXP	ЭКСПОНЕНТА	LOG	ЛОГАРИФМ
		PWR	ВОЗВЕДЕНИЕ В СТЕПЕНЬ	---	---	---	---
	Символы сравнения и преобразование	LD, AND, OR + =, <>, <, <=, >, >= + F	Символы сравнения (с плавающей запятой, одинарной точности)	FSTR	ПЛАВАЮЩАЯ ЗАПЯТАЯ В ASCII	FVAL	ASCII В ПЛАВАЮЩАЯ ЗАПЯТАЯ
Команды для чисел с плавающей запятой двойной точности	Преобразование плавающая запятая<->двоичный формат	FIXD	ПЛАВ. ЗАПЯТАЯ ДВОЙНОЙ ТОЧНОСТИ В 16-БИТ	FIXLD	ПЛАВ. ЗАПЯТАЯ ДВОЙНОЙ ТОЧНОСТИ В 32-БИТ	DBL	16-БИТ В ПЛАВ. ЗАПЯТАЯ ДВОЙНОЙ ТОЧНОСТИ
		DBLL	32-БИТ В ПЛАВ. ЗАПЯТАЯ ДВОЙНОЙ ТОЧНОСТИ	---	---	---	---
	Арифметические операции с плавающей запятой	+D	СЛОЖЕНИЕ С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ	–D	ВЫЧИТАНИЕ С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ	/D	ДЕЛЕНИЕ С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ
		*D	УМНОЖЕНИЕ С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ	---	---	---	---
	Тригонометрические функции с плавающей запятой	RADD	ГРАДУСЫ В РАДИАНЫ ДВОЙНОЙ ТОЧНОСТИ	DEGD	РАДИАНЫ В ГРАДУСЫ ДВОЙНОЙ ТОЧНОСТИ	SIND	СИНУС ДВОЙНОЙ ТОЧНОСТИ
		COSD	КОСИНУС ДВОЙНОЙ ТОЧНОСТИ	TAND	ТАНГЕНС ДВОЙНОЙ ТОЧНОСТИ	ASIND	АРКСИНУС ДВОЙНОЙ ТОЧНОСТИ
		ACOSD	АРККОСИНУС ДВОЙНОЙ ТОЧНОСТИ	ATAND	АРКТАНГЕНС ДВОЙНОЙ ТОЧНОСТИ	---	---
	Математические операции с плавающей запятой	SQRTD	КВАДРАТНЫЙ КОРЕНЬ ДВОЙНОЙ ТОЧНОСТИ	EXPD	ЭКСПОНЕНТА ДВОЙНОЙ ТОЧНОСТИ	LOGD	ЛОГАРИФМ ДВОЙНОЙ ТОЧНОСТИ
		PWRD	ВОЗВЕДЕНИЕ В СТЕПЕНЬ ДВОЙНОЙ ТОЧНОСТИ	---	---	---	---
	Символы сравнения	LD, AND, OR + =, <>, <, <=, >, >= + D	Символы сравнения (с плавающей запятой, двойной точности)	---	---	---	---

Классификация	Подкласс	Мнем. обознач.	Команда	Мнем. обознач.	Команда	Мнем. обознач.	Команда
Команды обработки табличных данных	Операции над стеком	SSET	СОЗДАТЬ СТЕК	PUSH	ЗАПИСАТЬ В СТЕК	LIFO	ПОСЛЕДНИМ ПРИШЕЛ — ПЕРВЫМ ВЫШЕЛ
		FIFO	ПЕРВЫМ ПРИШЕЛ — ПЕРВЫМ ВЫШЕЛ	SNUM	СЧИТАТЬ РАЗМЕР СТЕКА	SREAD	ПРОЧИТАТЬ ДАННЫЕ ИЗ СТЕКА
		SWRIT	ЗАПИСАТЬ ДАННЫЕ В СТЕК	SINS	ВСТАВИТЬ ДАННЫЕ В СТЕК	SDEL	УДАЛИТЬ ДАННЫЕ ИЗ СТЕКА
	Операции над 1 записью/несколькими словами	DIM	СОЗДАТЬ ТАБЛИЦУ УКАЗАННЫХ РАЗМЕРОВ	SETR	ЗАПИСАТЬ РАСПОЛОЖЕНИЕ СТРОКИ	GETR	СЧИТАТЬ НОМЕР СТРОКИ
	Операции над строками и словами	SRCH	ПОИСК ЗНАЧЕНИЯ	MAX	НАЙТИ МАКСИМУМ	MIN	НАЙТИ МИНИМУМ
		SUM	СУММА	FCS	КОНТРОЛЬНАЯ СУММА КАДРА	---	---
	Операции над байтами	SWAP	ПОМЕНЯТЬ МЕСТАМИ БАЙТЫ	---	---	---	---
Команды управления данными	---	PID	ПИД-РЕГУЛЯТОР	PIDAT	ПИД-РЕГУЛЯТОР С АВТОНАСТРОЙКОЙ	LMT	ОГРАНИЧЕНИЕ
		BAND	ЗОНА НЕЧУВСТВИТЕЛЬНОСТИ	ZONE	МЕРТВАЯ ЗОНА	TPO	ВЫХОД ШИМ
		SCL	ИЗМЕНЕНИЕ МАСШТАБА	SCL2	ИЗМЕНЕНИЕ МАСШТАБА 2	SCL3	ИЗМЕНЕНИЕ МАСШТАБА 3
		AVG	СРЕДНЕЕ ЗНАЧЕНИЕ	---	---	---	---
Команды подпрограмм	---	SBS	ВЫЗОВ ПОДПРОГРАММЫ	MCRO	МАКРОС	SBN	ВХОД В ПОДПРОГРАММУ
		RET	ВЫХОД ИЗ ПОДПРОГРАММЫ	GSBS	ВЫЗОВ ГЛОБАЛЬНОЙ ПОДПРОГРАММЫ	GSBN	ВХОД В ГЛОБАЛЬНУЮ ПОДПРОГРАММУ
		GRET	ВЫХОД ИЗ ГЛОБАЛЬНОЙ ПОДПРОГРАММЫ	---	---	---	---
Команды управления прерываниями	---	MSKS	УСТАНОВИТЬ МАСКУ ПРЕРЫВАНИЙ	MSKR	СЧИТАТЬ МАСКУ ПРЕРЫВАНИЙ	CLI	ОЧИСТИТЬ ПРЕРЫВАНИЕ
		DI	ЗАПРЕТИТЬ ПРЕРЫВАНИЯ	EI	РАЗРЕШИТЬ ПРЕРЫВАНИЯ	---	---
Команды управления скоростными счетчиками и импульсными выходами (см. примеч.)	---	INI	УПРАВЛЕНИЕ РЕЖИМОМ	PRV	ЧТЕНИЕ ТЕКУЩ. ЗНАЧ. СКОРОСТНОГО СЧЕТЧИКА	PRV2	ПРЕОБРАЗОВАНИЕ ЧАСТОТЫ НА ВХОДЕ СЧЕТЧИКА
		CTBL	ЗАГРУЗКА ТАБЛИЦЫ СРАВНЕНИЯ	SPED	ВЫДАЧА ИМПУЛЬСОВ СКОРОСТИ	PULS	ЗАДАТЬ КОЛВО ИМПУЛЬСОВ
		PLS2	ВЫДАЧА ИМПУЛЬСОВ	ACC	УПРАВЛЕНИЕ РАЗГОНОМ	ORG	ПОИСК ИСХОДНОГО ПОЛОЖЕНИЯ
		PWM	ИМПУЛЬСЫ С ПЕРЕМЕННОЙ СКВАЖНОСТЬЮ	---	---	---	---
Команды пошагового выполнения	---	STEP	ОПРЕДЕЛИТЬ ШАГ	SNXT	ЗАПУСТИТЬ ШАГ	---	---

Классификация	Подкласс	Мнем. обознач.	Команда	Мнем. обознач.	Команда	Мнем. обознач.	Команда
Команды для модулей ввода-вывода	---	IORF	ОБНОВЛЕНИЕ СЛОВ ВВОДА/ВЫВОДА	SDEC	ПРЕОБРАЗОВАТЬ В 7-СЕГМЕНТНЫЙ КОД	DSW	ЧТЕНИЕ ЦИФРОВОГО ПЕРЕКЛЮЧАТЕЛЯ
		TKY	ЧТЕНИЕ ДЕСЯТИЧНОЙ КЛАВИАТУРЫ	HKY	ЧТЕНИЕ ШЕСТНАДЦАТЕРИЧНОЙ КЛАВИАТУРЫ	MTR	ЧТЕНИЕ МАТРИЦЫ
		7SEG	ВЫВОД НА 7-СЕГМЕНТНЫЙ ИНДИКАТОР	IORD	ЧТЕНИЕ ИНТЕЛЛЕКТУАЛЬНЫХ ВХОДОВ/ВЫХОДОВ	IOWR	ЗАПИСЬ ИНТЕЛЛЕКТУАЛЬНЫХ ВХОДОВ/ВЫХОДОВ
		DLNK (см. примеч.)	ОБНОВИТЬ ВХОДЫ/ВЫХОДЫ МОДУЛЯ ШИНЫ ЦПУ	---	---	---	---
Команды последовательного интерфейса	---	PMCR	МАКРОС ПРОТОКОЛА	TXD	ОТПРАВИТЬ	RXD	ПРИНЯТЬ
		STUP	ИЗМЕНИТЬ ПАРАМЕТРЫ ПОСЛЕД. ПОРТА	---	---	---	---
Сетевые команды	---	SEND	ПЕРЕДАТЬ ПО СЕТИ	RECV	ПРИНЯТЬ ПО СЕТИ	CMND	ДОСТАВИТЬ КОМАНДУ
		EXPLT	ОТПРАВИТЬ ЯВНОЕ СООБЩЕНИЕ	EGATR	ЯВНЫЙ ЗАПРОС АТРИБУТА	ESATR	ЯВНАЯ УСТАНОВКА АТРИБУТА
		ECHRD	ЯВНОЕ ЧТЕНИЕ СЛОВА	ECHWR	ЯВНАЯ ЗАПИСЬ СЛОВА	---	---
Команды отображения	---	MSG	ОТОБРАЗИТЬ СООБЩЕНИЕ	SCH	ОТОБРАЗИТЬ ЗНАЧЕНИЕ СЛОВА НА 7-СЕГМЕНТНОМ ИНДИКАТОРЕ	SCTRL	УПРАВЛЕНИЕ 7-СЕГМЕНТНЫМ ИНДИКАТОРОМ
Команды для работы с часами	---	CADD	ДОБАВИТЬ К КАЛЕНДАРНОЙ ДАТЕ	CSUB	ВЫЧЕСТЬ ИЗ КАЛЕНДАРНОЙ ДАТЫ	SEC	ЧАСЫ В СЕКУНДЫ
		HMS	СЕКУНДЫ В ЧАСЫ	DATE	КОРРЕКТИРОВКА ЧАСОВ	---	---
Команды отладки программы	---	TRSM	ОТБОР ДАННЫХ ДЛЯ ПАМЯТИ ПРОТОКОЛА ДАННЫХ	---	---	---	---
Команды для диагностики неисправностей	---	FAL	СИГНАЛИЗАЦИЯ НЕИСПРАВНОСТИ	FALS	СИГНАЛИЗАЦИЯ СЕРЬЕЗНОЙ НЕИСПРАВНОСТИ	FPD	ОБНАРУЖЕНИЕ НЕИСПРАВНОГО КАНАЛА
Прочие команды	---	STC	УСТАНОВИТЬ ПЕРЕНОС	CLC	СБРОСИТЬ ПЕРЕНОС	---	---
		WDT	УВЕЛИЧИТЬ МАКС. ВРЕМЯ ЦИКЛА	CCS	СОХРАНИТЬ ФЛАГИ УСЛОВИЙ	CCL	ЗАГРУЗИТЬ ФЛАГИ УСЛОВИЙ
		FRMCV	ПРЕОБРАЗОВАТЬ АДРЕС СЕРИИ CV	TOCV	ПРЕОБРАЗОВАТЬ В АДРЕС СЕРИИ CV	---	---

Классификация	Подкласс		Мнем. обознач.	Команда	Мнем. обознач.	Команда	Мнем. обознач.	Команда
Команды программирования блоков	Определение области программного блока		BPRG	НАЧАЛО ПРОГРАММНОГО БЛОКА	BEND	КОНЕЦ ПРОГРАММНОГО БЛОКА	---	---
	Запуск/остановка программы блока		BPPS	ПРИОСТАНОВКА ПРОГРАММНОГО БЛОКА	BPRS	ПЕРЕЗАПУСК ПРОГРАММНОГО БЛОКА	---	---
	Выход		EXIT *адрес_бита*	ВЫХОД ИЗ БЛОКА ПО УСЛОВИЮ	EXIT NOT *адрес_бита*	ВЫХОД ИЗ БЛОКА ПО УСЛОВИЮ НЕ	Входное_условие EXIT	ВЫХОД ИЗ БЛОКА ПО УСЛОВИЮ
	Ветвление с командой IF		IF *адрес_бита*	БЛОК УСЛОВНОГО ВЕТВЛЕНИЯ	IF NOT *адрес_бита*	БЛОК УСЛОВНОГО ВЕТВЛЕНИЯ НЕ	ELSE	БЛОК УСЛОВНОГО ВЕТВЛЕНИЯ (ИНАЧЕ)
			IEND	КОНЕЦ БЛОКА УСЛОВНОГО ВЕТВЛЕНИЯ	---	---	---	---
	WAIT		WAIT *адрес_бита*	ОДИН ЦИКЛ И ОЖИДАНИЕ	WAIT NOT *адрес_бита*	ОДИН ЦИКЛ И ОЖИДАНИЕ НЕ	*Входное_условие* WAIT	ОДИН ЦИКЛ И ОЖИДАНИЕ
	Таймеры/счетчики	BCD (см. примеч.)	TIMW	ТАЙМЕР ОЖИДАНИЯ	CNTW	СЧЕТЧИК ОЖИДАНИЯ	TMHW	СКОРОСТНОЙ ТАЙМЕР ОЖИДАНИЯ
		Двоичный (см. примеч.)	TIMWX	ТАЙМЕР ОЖИДАНИЯ	CNTWX	СЧЕТЧИК ОЖИДАНИЯ	TMHWX	СКОРОСТНОЙ ТАЙМЕР ОЖИДАНИЯ
	Повтор		LOOP	Блок цикла	LEND *адрес_бита*	КОНЕЦ БЛОКА ЦИКЛА	LEND NOT *адрес_бита*	КОНЕЦ БЛОКА ЦИКЛА НЕ
			Входное_условие LEND	КОНЕЦ БЛОКА ЦИКЛА	---	---	---	---
Команды обработки текстовых строк	---		MOV$	КОПИРОВАТЬ СТРОКУ	+$	ОБЪЕДИНИТЬ СТРОКИ	LEFT$	СКОПИРОВАТЬ СИМВОЛЫ СЛЕВА
			RIGHT$	СКОПИРОВАТЬ СИМВОЛЫ СПРАВА	MID$	СКОПИРОВАТЬ СИМВОЛЫ ВНУТРИ СТРОКИ	FIND$	НАЙТИ В СТРОКЕ
			LEN$	ДЛИНА СТРОКИ	RPLC$	ЗАМЕНИТЬ В СТРОКЕ	DEL$	УДАЛИТЬ СТРОКУ
			XCHG$	ОБМЕН СТРОКАМИ	CLR$	ОЧИСТИТЬ СТРОКУ	INS$	ВСТАВИТЬ В СТРОКУ
			LD, AND, OR + =$, <>$, <$, <=$, >$, >=$	СРАВНЕНИЕ СТРОК	---	---	---	---
Команды управления задачами	---		TKON	ВКЛЮЧИТЬ ЗАДАЧУ	TKOF	ВЫКЛЮЧИТЬ ЗАДАЧУ	---	---
Команды преобразования моделей	---		XFERC	ПЕРЕДАЧА БЛОКА	DISTC	РАЗМЕЩЕНИЕ ОДИНОЧНОГО СЛОВА	COLLC	СБОР ДАННЫХ
			MOVBC	ПЕРЕДАТЬ БИТ	BCNTC	ПОДСЧЕТ БИТОВ	---	---
Команды специальных функциональных блоков	---		GETID	СЧИТАТЬ ИДЕНТИФИКАТОР ПЕРЕМЕННОЙ	---	---	---	---

Примечание. Переключение формата представления значений (двоичный или двоично-десятичный) для таймеров и счетчиков производится в CX-Programmer.

Предварительная версия
Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

Приложение B

Список команд в порядке возрастания кодов функций

Код функции	Мнемоническое обозначение	Команда	Выполнение по положительному фронту	Выполнение по отрицательному фронту	Модификатор мгновенного обновления	Стр.
---	LD	ЗАГРУЗКА	@LD	%LD	!LD	99
---	LD NOT	ЗАГРУЗКА НЕ	@LD NOT	%LD NOT	!LD NOT	101
---	AND	И	@AND	%AND	!AND	104
---	AND NOT	И НЕ	@AND NOT	%AND NOT	!AND NOT	105
---	OR	ИЛИ	@OR	%OR	!OR	107
---	OR NOT	ИЛИ НЕ	@OR NOT	%OR NOT	!OR NOT	109
---	AND LD	И ЗАГРУЗКА	---	---	---	111
---	OR LD	ИЛИ ЗАГРУЗКА	---	---	---	113
---	OUT	ВЫВОД	---	---	!OUT	126
---	OUT NOT	ВЫВОД НЕ	---	---	!OUT NOT	127
---	SET	УСТАНОВКА	@SET	%SET	!SET	135
---	RSET	СБРОС	@RSET	%RSET	!RSET	135
---	TIM	ТАЙМЕР	---	---	---	186
---	TIMX	ТАЙМЕР	---	---	---	186
---	CNT	СЧЕТЧИК	---	---	---	212
000	NOP	ПУСТАЯ ОПЕРАЦИЯ	---	---	---	147
001	END	КОНЕЦ	---	---	---	146
002	IL	БЛОКИРОВКА	---	---	---	150
003	ILC	ОТМЕНА БЛОКИРОВКИ	---	---	---	150
004	JMP	ПЕРЕХОД	---	---	---	168
005	JME	КОНЕЦ ПЕРЕХОДА	---	---	---	168
006	FAL	СИГНАЛИЗАЦИЯ НЕИСПРАВНОСТИ	@FAL	---	---	1025
007	FALS	СИГНАЛИЗАЦИЯ СЕРЬЕЗНОЙ НЕИСПРАВНОСТИ	---	---	---	1033
008	STEP	ОПРЕДЕЛИТЬ ШАГ	---	---	---	825
009	SNXT	ЗАПУСТИТЬ ШАГ	---	---	---	825
010	SFT	РЕГИСТР СДВИГА	---	---	---	302
011	ЗАЩЕЛКА	ЗАЩЕЛКА	---	---	!KEEP	128
012	CNTR	РЕВЕРСИВНЫЙ СЧЕТЧИК	---	---	---	215
013	DIFU	ПОЛОЖИТЕЛЬНЫЙ ФРОНТ	---	---	!DIFU	133
014	DIFD	ОТРИЦАТЕЛЬНЫЙ ФРОНТ	---	---	!DIFD	133
015	TIMH	СКОРОСТНОЙ ТАЙМЕР	---	---	---	191
016	WSFT	СДВИГ ПО СЛОВАМ	@WSFT	---	---	309
017	ASFT	АСИНХРОННЫЙ РЕГИСТР СДВИГА	@ASFT	---	---	306
019	MCMP	СРАВНИТЬ НЕСКОЛЬКО ЗНАЧЕНИЙ	@MCMP	---	---	253
020	CMP	СРАВНИТЬ БЕЗ ЗНАКА	---	---	!CMP	241
021	MOV	ПЕРЕДАТЬ	@MOV	---	!MOV	271
022	MVN	ПЕРЕДАТЬ ДОПОЛНЕНИЕ	@MVN	---	---	273
023	BIN	BCD В ДВОИЧНОЕ	@BIN	---	---	425
024	BCD	ДВОИЧНОЕ В BCD	@BCD	---	---	429
025	ASL	АРИФМЕТИЧЕСКИЙ СДВИГ ВЛЕВО	@ASL	---	---	311
026	ASR	АРИФМЕТИЧЕСКИЙ СДВИГ ВПРАВО	@ASR	---	---	314
027	ROL	ЦИКЛИЧЕСКИЙ СДВИГ ВЛЕВО	@ROL	---	---	317

Код функции	Мнемоническое обозначение	Команда	Выполнение по положительному фронту	Выполнение по отрицательному фронту	Модификатор мгновенного обновления	Стр.
028	ROR	ЦИКЛИЧЕСКИЙ СДВИГ ВПРАВО	@ROR	---	---	321
029	COM	ДОПОЛНЕНИЕ	@COM	---	---	490
034	ANDW	ЛОГИЧЕСКОЕ И	@ANDW	---	---	477
035	ORW	ЛОГИЧЕСКОЕ ИЛИ	@ORW	---	---	480
036	XORW	ИСКЛЮЧАЮЩЕЕ ИЛИ	@XORW	---	---	483
037	XNRW	ИСКЛЮЧАЮЩЕЕ НЕ-ИЛИ	@XNRW	---	---	487
040	STC	УСТАНОВИТЬ ПЕРЕНОС	@STC	---	---	1051
041	CLC	СБРОСИТЬ ПЕРЕНОС	@CLC	---	---	1052
045	TRSM	ОТБОР ДАННЫХ ДЛЯ ПАМЯТИ ПРОТОКОЛА ДАННЫХ	---	---	---	1020
046	MSG	ОТОБРАЗИТЬ СООБЩЕНИЕ	@MSG	---	---	997
047	SCH	ОТОБРАЗИТЬ ЗНАЧЕНИЕ СЛОВА НА 7-СЕГМЕНТНОМ ИНДИКАТОРЕ	@SCH	---	---	1000
048	SCTRL	УПРАВЛЕНИЕ 7-СЕГМЕНТНЫМ ИНДИКАТОРОМ	@SCTRL	---	---	1002
058	BINL	ДВОЙНОЕ BCD В ДВОЙНОЕ ДВОИЧНОЕ	@BINL	---	---	427
059	BCDL	ДВОЙНОЕ ДВОИЧНОЕ В ДВОЙНОЕ BCD	@BCDL	---	---	430
060	CMPL	СРАВНИТЬ ДВОЙНЫЕ СЛОВА БЕЗ ЗНАКА	---	---	---	244
062	XFRB	ПЕРЕДАТЬ НЕСКОЛЬКО БИТОВ	@XFRB	---	---	282
063	LINE	СТОЛБЕЦ В СТРОКУ	@LINE	---	---	454
064	COLM	СТРОКУ В СТОЛБЕЦ	@COLM	---	---	456
065	SEC	ЧАСЫ В СЕКУНДЫ	@SEC	---	---	1013
066	HMS	СЕКУНДЫ В ЧАСЫ	@HMS	---	---	1015
067	BCNT	ПОДСЧЕТ БИТОВ	@BCNT	---	---	514
068	BCMP	СРАВНИТЬ С ДИАПАЗОНАМИ БЕЗ ЗНАКА	@BCMP	---	---	258
069	APR	МАТЕМАТИЧЕСКАЯ ФУНКЦИЯ	@APR	---	---	499
070	XFER	ПЕРЕДАЧА БЛОКА	@XFER	---	---	285
071	BSET	ЗАПОЛНЕНИЕ БЛОКА	@BSET	---	---	287
072	ROOT	КВАДРАТНЫЙ КОРЕНЬ ИЗ ДВОИЧНО-ДЕСЯТИЧНОГО ЧИСЛА	@ROOT	---	---	495
073	XCHG	ОБМЕН ДАННЫМИ	@XCHG	---	---	289
074	SLD	СДВИГ НА ОДИН РАЗРЯД ВЛЕВО	@SLD	---	---	331
075	SRD	СДВИГ НА ОДИН РАЗРЯД ВПРАВО	@SRD	---	---	332
076	MLPX	ДЕКОДИРОВАНИЕ ДАННЫХ	@MLPX	---	---	437
077	DMPX	КОДИРОВАНИЕ ДАННЫХ	@DMPX	---	---	442
078	SDEC	ПРЕОБРАЗОВАТЬ В 7-СЕГМЕНТНЫЙ КОД	@SDEC	---	---	846
079	FDIV	ДЕЛЕНИЕ С ПЛАВАЮЩЕЙ ЗАПЯТОЙ	@FDIV	---	---	510
080	DIST	РАЗМЕЩЕНИЕ ОДИНОЧНОГО СЛОВА	@DIST	---	---	292
081	COLL	СБОР ДАННЫХ	@COLL	---	---	294
082	MOVB	ПЕРЕДАТЬ БИТ	@MOVB	---	---	277
083	MOVD	ПЕРЕДАТЬ РАЗРЯД	@MOVD	---	---	279
084	SFTR	РЕВЕРСИВНЫЙ РЕГИСТР СДВИГА	@SFTR	---	---	304

Код функции	Мнемоническое обозначение	Команда	Выполнение по положительному фронту	Выполнение по отрицательному фронту	Модификатор мгновенного обновления	Стр.
085	TCMP	СРАВНИТЬ С ТАБЛИЦЕЙ	@TCMP	---	---	256
086	ASC	ПРЕОБРАЗОВАНИЕ В ASCII	@ASC	---	---	446
087	TTIM	НАКАПЛИВАЮЩИЙ ТАЙМЕР	---	---	---	199
088	ZCP	СРАВНИТЬ СЛОВО С ДИАПАЗОНОМ	---	---	---	265
090	SEND	ПЕРЕДАТЬ ПО СЕТИ	@SEND	---	---	945
091	SBS	ВЫЗОВ ПОДПРОГРАММЫ	@SBS	---	---	730
092	SBN	ВХОД В ПОДПРОГРАММУ	---	---	---	740
093	RET	ВЫХОД ИЗ ПОДПРОГРАММЫ	---	---	---	743
094	WDT	УВЕЛИЧИТЬ МАКС. ВРЕМЯ ЦИКЛА	@WDT	---	---	1053
096	BPRG	НАЧАЛО ПРОГРАММНОГО БЛОКА	---	---	---	1072
097	IORF	ОБНОВЛЕНИЕ СЛОВ ВВОДА/ВЫВОДА	@IORF	---	---	843
098	RECV	ПРИНЯТЬ ПО СЕТИ	@RECV	---	---	952
099	MCRO	МАКРОС	@MCRO	---	---	736
114	CPS	СРАВНИТЬ ДВОИЧНЫЕ ЗНАЧЕНИЯ СО ЗНАКОМ	---	---	!CPS	247
115	CPSL	СРАВНИТЬ ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ	---	---	---	250
116	ZCPL	СРАВНИТЬ ДВОЙНОЕ СЛОВО С ДИАПАЗОНОМ	---	---	---	268
160	NEG	ДОПОЛНЕНИЕ ДО 2	@NEG	---	---	432
161	NEGL	ДОПОЛНЕНИЕ ДВОЙНОГО СЛОВА ДО 2	@NEGL	---	---	434
162	HEX	ASCII В HEX	@HEX	---	---	450
180	FCS	КОНТРОЛЬНАЯ СУММА КАДРА	@FCS	---	---	653
181	SRCH	ПОИСК ЗНАЧЕНИЯ	@SRCH	---	---	638
182	MAX	НАЙТИ МАКСИМУМ	@MAX	---	---	643
183	MIN	НАЙТИ МИНИМУМ	@MIN	---	---	646
184	SUM	СУММА	@SUM	---	---	649
190	ПИД	ПИД-РЕГУЛЯТОР	---	---	---	671
191	PIDAT	ПИД-РЕГУЛЯТОР С АВТОНАСТРОЙКОЙ	---	---	---	684
194	SCL	ИЗМЕНЕНИЕ МАСШТАБА	@SCL	---	---	713
195	AVG	СРЕДНЕЕ ЗНАЧЕНИЕ	---	---	---	726
210	DSW	ЧТЕНИЕ ЦИФРОВОГО ПЕРЕКЛЮЧАТЕЛЯ	---	---	---	849
211	TKY	ЧТЕНИЕ ДЕСЯТИЧНОЙ КЛАВИАТУРЫ	@TKY	---	---	853
212	HKY	ЧТЕНИЕ ШЕСТНАДЦАТЕРИЧНОЙ КЛАВИАТУРЫ	---	---	---	857
213	MTR	ЧТЕНИЕ МАТРИЦЫ	---	---	---	861
214	7SEG	ВЫВОД НА 7-СЕГМЕНТНЫЙ ИНДИКАТОР	---	---	---	865
222	IORD	ЧТЕНИЕ ИНТЕЛЛЕКТУАЛЬНЫХ ВХОДОВ/ВЫХОДОВ	@IORD	---	---	870
223	IOWR	ЗАПИСЬ ИНТЕЛЛЕКТУАЛЬНЫХ ВХОДОВ/ВЫХОДОВ	@IOWR	---	---	873
226	DLNK	ОБНОВИТЬ ВХОДЫ/ ВЫХОДЫ МОДУЛЯ ШИНЫ ЦПУ	@DLNK	---	---	877
235	RXD	ПРИНЯТЬ	@RXD	---	---	898

Код функции	Мнемоническое обозначение	Команда	Выполнение по положительному фронту	Выполнение по отрицательному фронту	Модификатор мгновенного обновления	Стр.
236	TXD	ОТПРАВИТЬ	@TXD	---	---	892
237	STUP	ИЗМЕНИТЬ ПАРАМЕТРЫ ПОСЛЕД. ПОРТА	@STUP	---	---	921
255	RXDU	ПРИНЯТЬ ЧЕРЕЗ МОДУЛЬ ПОСЛЕД. ИНТЕРФЕЙСА	@RXDU	---	---	912
256	TXDU	ОТПРАВИТЬ ЧЕРЕЗ МОДУЛЬ ПОСЛЕД. ИНТЕРФЕЙСА	@TXDU	---	---	904
260	PMCR	МАКРОС ПРОТОКОЛА	@PMCR	---	---	884
269	FPD	ОБНАРУЖЕНИЕ НЕИСПРАВНОГО КАНАЛА	---	---	---	1041
282	CCS	СОХРАНИТЬ ФЛАГИ УСЛОВИЙ	@CCS	---	---	1055
283	CCL	ЗАГРУЗИТЬ ФЛАГИ УСЛОВИЙ	@CCL	---	---	1057
284	FRMCV	ПРЕОБРАЗОВАТЬ АДРЕС СЕРИИ CV	@FRMCV	---	---	1059
285	TOCV	ПРЕОБРАЗОВАТЬ В АДРЕС СЕРИИ CV	@TOCV	---	---	1063
286	GETID	СЧИТАТЬ ИДЕНТИФИКАТОР ПЕРЕМЕННОЙ	@GETID	---	---	1162
300	AND=	И РАВНО	---	---	---	230
300	LD =	ЗАГРУЗКА РАВНО	---	---	---	230
300	OR =	ИЛИ РАВНО	---	---	---	230
301	AND =L	И ДВОЙНЫЕ СЛОВА РАВНЫ	---	---	---	230
301	LD =L	ЗАГРУЗКА ДВОЙНЫЕ СЛОВА РАВНЫ	---	---	---	230
301	OR =L	ИЛИ ДВОЙНЫЕ СЛОВА РАВНЫ	---	---	---	230
302	AND =S	И СЛОВА СО ЗНАКОМ РАВНЫ	---	---	---	230
302	LD =S	ЗАГРУЗКА СЛОВА СО ЗНАКОМ РАВНЫ	---	---	---	230
302	OR =S	ИЛИ СЛОВА СО ЗНАКОМ РАВНЫ	---	---	---	230
303	AND =SL	И ДВОЙНЫЕ СЛОВА СО ЗНАКОМ РАВНЫ	---	---	---	230
303	LD =SL	ЗАГРУЗКА ДВОЙНЫЕ СЛОВА СО ЗНАКОМ РАВНЫ	---	---	---	230
303	OR =SL	ИЛИ ДВОЙНЫЕ СЛОВА СО ЗНАКОМ РАВНЫ	---	---	---	230
305	AND<>	И НЕ РАВНО	---	---	---	230
305	LD<>	ЗАГРУЗКА НЕ РАВНО	---	---	---	230
305	OR<>	ИЛИ НЕ РАВНО	---	---	---	230
306	AND<>L	И ДВОЙНЫЕ СЛОВА НЕ РАВНЫ	---	---	---	230
306	LD<>L	ЗАГРУЗКА ДВОЙНЫЕ СЛОВА НЕ РАВНЫ	---	---	---	230
306	OR<>L	ИЛИ ДВОЙНЫЕ СЛОВА НЕ РАВНЫ	---	---	---	230
307	AND<>S	И СЛОВА СО ЗНАКОМ НЕ РАВНЫ	---	---	---	230
307	LD<>S	ЗАГРУЗКА СЛОВА СО ЗНАКОМ НЕ РАВНЫ	---	---	---	230
307	OR<>S	ИЛИ СЛОВА СО ЗНАКОМ НЕ РАВНЫ	---	---	---	230
308	AND<>SL	И ДВОЙНЫЕ СЛОВА СО ЗНАКОМ НЕ РАВНЫ	---	---	---	230
308	LD<>SL	ЗАГРУЗКА ДВОЙНЫЕ СЛОВА СО ЗНАКОМ НЕ РАВНЫ	---	---	---	230

Код функции	Мнемоническое обозначение	Команда	Выполнение по положительному фронту	Выполнение по отрицательному фронту	Модификатор мгновенного обновления	Стр.
308	OR<>SL	ИЛИ ДВОЙНЫЕ СЛОВА СО ЗНАКОМ НЕ РАВНЫ	---	---	---	230
310	AND<	И МЕНЬШЕ	---	---	---	230
310	LD<	ЗАГРУЗКА МЕНЬШЕ	---	---	---	230
310	OR<	ИЛИ МЕНЬШЕ	---	---	---	230
311	AND<L	И ДВОЙНОЕ СЛОВО МЕНЬШЕ	---	---	---	230
311	LD<L	ЗАГРУЗКА ДВОЙНОЕ СЛОВО МЕНЬШЕ	---	---	---	230
311	OR<L	ИЛИ ДВОЙНОЕ СЛОВО МЕНЬШЕ	---	---	---	230
312	AND<S	И СЛОВО СО ЗНАКОМ МЕНЬШЕ	---	---	---	230
312	LD<S	ЗАГРУЗКА СЛОВО СО ЗНАКОМ МЕНЬШЕ	---	---	---	230
312	OR<S	ИЛИ СЛОВО СО ЗНАКОМ МЕНЬШЕ	---	---	---	230
313	AND<SL	И ДВОЙНОЕ СЛОВО СО ЗНАКОМ МЕНЬШЕ	---	---	---	230
313	LD<SL	ЗАГРУЗКА ДВОЙНОЕ СЛОВО СО ЗНАКОМ МЕНЬШЕ	---	---	---	230
313	OR<SL	ИЛИ ДВОЙНОЕ СЛОВО СО ЗНАКОМ МЕНЬШЕ	---	---	---	230
315	AND <=	И МЕНЬШЕ ИЛИ РАВНО	---	---	---	230
315	LD <=	ЗАГРУЗКА МЕНЬШЕ ИЛИ РАВНО	---	---	---	230
315	OR <=	ИЛИ МЕНЬШЕ ИЛИ РАВНО	---	---	---	230
316	AND <=L	И ДВОЙНОЕ СЛОВО МЕНЬШЕ ИЛИ РАВНО	---	---	---	230
316	LD <=L	ЗАГРУЗКА ДВОЙНОЕ СЛОВО МЕНЬШЕ ИЛИ РАВНО	---	---	---	230
316	OR <=L	ИЛИ ДВОЙНОЕ СЛОВО МЕНЬШЕ ИЛИ РАВНО	---	---	---	230
317	AND <=S	И СЛОВО СО ЗНАКОМ МЕНЬШЕ ИЛИ РАВНО	---	---	---	230
317	LD <=S	ЗАГРУЗКА СЛОВО СО ЗНАКОМ МЕНЬШЕ ИЛИ РАВНО	---	---	---	230
317	OR <=S	ИЛИ СЛОВО СО ЗНАКОМ МЕНЬШЕ ИЛИ РАВНО	---	---	---	230
318	AND <=SL	И ДВОЙНОЕ СЛОВО СО ЗНАКОМ МЕНЬШЕ ИЛИ РАВНО	---	---	---	230
318	LD <=SL	ЗАГРУЗКА ДВОЙНОЕ СЛОВО СО ЗНАКОМ МЕНЬШЕ ИЛИ РАВНО	---	---	---	230
318	OR <=SL	ИЛИ СЛОВО СО ЗНАКОМ МЕНЬШЕ ИЛИ РАВНО	---	---	---	230
320	AND>	И БОЛЬШЕ	---	---	---	230
320	LD>	ЗАГРУЗКА БОЛЬШЕ	---	---	---	230
320	OR>	ИЛИ БОЛЬШЕ	---	---	---	230
321	AND>L	И ДВОЙНОЕ СЛОВО БОЛЬШЕ	---	---	---	230
321	LD>L	ЗАГРУЗКА ДВОЙНОЕ СЛОВО БОЛЬШЕ	---	---	---	230
321	OR>L	ИЛИ ДВОЙНОЕ СЛОВО БОЛЬШЕ	---	---	---	230
322	AND>S	И СЛОВО СО ЗНАКОМ БОЛЬШЕ	---	---	---	230
322	LD>S	ЗАГРУЗКА СЛОВО СО ЗНАКОМ БОЛЬШЕ	---	---	---	230
322	OR>S	ИЛИ СЛОВО СО ЗНАКОМ БОЛЬШЕ	---	---	---	230

Код функции	Мнемоническое обозначение	Команда	Выполнение по положительному фронту	Выполнение по отрицательному фронту	Модификатор мгновенного обновления	Стр.
323	AND>SL	И ДВОЙНОЕ СЛОВО СО ЗНАКОМ БОЛЬШЕ	---	---	---	230
323	LD>SL	ЗАГРУЗКА ДВОЙНОЕ СЛОВО СО ЗНАКОМ БОЛЬШЕ	---	---	---	230
323	OR>SL	ИЛИ ДВОЙНОЕ СЛОВО СО ЗНАКОМ БОЛЬШЕ	---	---	---	230
325	AND >=	И БОЛЬШЕ ИЛИ РАВНО	---	---	---	230
325	LD >=	ЗАГРУЗКА БОЛЬШЕ ИЛИ РАВНО	---	---	---	230
325	OR >=	ИЛИ БОЛЬШЕ ИЛИ РАВНО	---	---	---	230
326	AND >=L	И ДВОЙНОЕ СЛОВО БОЛЬШЕ ИЛИ РАВНО	---	---	---	230
326	LD >=L	ЗАГРУЗКА ДВОЙНОЕ СЛОВО БОЛЬШЕ ИЛИ РАВНО	---	---	---	230
326	OR >=L	ИЛИ ДВОЙНОЕ СЛОВО БОЛЬШЕ ИЛИ РАВНО	---	---	---	230
327	AND >=S	И СЛОВО СО ЗНАКОМ БОЛЬШЕ ИЛИ РАВНО	---	---	---	230
327	LD >=S	ЗАГРУЗКА СЛОВО СО ЗНАКОМ БОЛЬШЕ ИЛИ РАВНО	---	---	---	230
327	OR >=S	ИЛИ СЛОВО СО ЗНАКОМ БОЛЬШЕ ИЛИ РАВНО	---	---	---	230
328	AND >=SL	И ДВОЙНОЕ СЛОВО СО ЗНАКОМ БОЛЬШЕ ИЛИ РАВНО	---	---	---	230
328	LD >=SL	ЗАГРУЗКА ДВОЙНОЕ СЛОВО СО ЗНАКОМ БОЛЬШЕ ИЛИ РАВНО	---	---	---	230
328	OR >=SL	ИЛИ ДВОЙНОЕ СЛОВО СО ЗНАКОМ БОЛЬШЕ ИЛИ РАВНО	---	---	---	230
329	AND =F	И С ПЛАВАЮЩЕЙ ЗАПЯТОЙ РАВНО	---	---	---	559
329	LD =F	ЗАГРУЗКА С ПЛАВАЮЩЕЙ ЗАПЯТОЙ РАВНО	---	---	---	559
329	OR =F	ИЛИ С ПЛАВАЮЩЕЙ ЗАПЯТОЙ РАВНО	---	---	---	559
330	AND<>F	И С ПЛАВАЮЩЕЙ ЗАПЯТОЙ НЕ РАВНО	---	---	---	559
330	LD<>F	ЗАГРУЗКА С ПЛАВАЮЩЕЙ ЗАПЯТОЙ НЕ РАВНО	---	---	---	559
330	OR<>F	ИЛИ С ПЛАВАЮЩЕЙ ЗАПЯТОЙ НЕ РАВНО	---	---	---	559
331	AND<F	И С ПЛАВАЮЩЕЙ ЗАПЯТОЙ МЕНЬШЕ	---	---	---	559
331	LD<F	ЗАГРУЗКА С ПЛАВАЮЩЕЙ ЗАПЯТОЙ МЕНЬШЕ	---	---	---	559
331	OR<F	ИЛИ С ПЛАВАЮЩЕЙ ЗАПЯТОЙ МЕНЬШЕ	---	---	---	559
332	AND<=F	И С ПЛАВАЮЩЕЙ ЗАПЯТОЙ МЕНЬШЕ ИЛИ РАВНО	---	---	---	559
332	LD<=F	ЗАГРУЗКА С ПЛАВАЮЩЕЙ ЗАПЯТОЙ МЕНЬШЕ ИЛИ РАВНО	---	---	---	559
332	OR<=F	ИЛИ С ПЛАВАЮЩЕЙ ЗАПЯТОЙ МЕНЬШЕ ИЛИ РАВНО	---	---	---	559
333	AND>F	И С ПЛАВАЮЩЕЙ ЗАПЯТОЙ БОЛЬШЕ	---	---	---	559

Код функции	Мнемоническое обозначение	Команда	Выполнение по положительному фронту	Выполнение по отрицательному фронту	Модификатор мгновенного обновления	Стр.
333	LD>F	ЗАГРУЗКА С ПЛАВАЮЩЕЙ ЗАПЯТОЙ БОЛЬШЕ	---	---	---	559
333	OR>F	ИЛИ С ПЛАВАЮЩЕЙ ЗАПЯТОЙ БОЛЬШЕ	---	---	---	559
334	AND>=F	И С ПЛАВАЮЩЕЙ ЗАПЯТОЙ БОЛЬШЕ ИЛИ РАВНО	---	---	---	559
334	LD>=F	ЗАГРУЗКА С ПЛАВАЮЩЕЙ ЗАПЯТОЙ БОЛЬШЕ ИЛИ РАВНО	---	---	---	559
334	OR>=F	ИЛИ С ПЛАВАЮЩЕЙ ЗАПЯТОЙ БОЛЬШЕ ИЛИ РАВНО	---	---	---	559
335	AND =D	И С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ РАВНО	---	---	---	616
335	LD =D	ЗАГРУЗКА С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ РАВНО	---	---	---	616
335	OR =D	ИЛИ С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ РАВНО	---	---	---	616
336	AND<>D	И С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ НЕ РАВНО	---	---	---	616
336	LD<>D	ЗАГРУЗКА С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ НЕ РАВНО	---	---	---	616
336	OR<>D	ИЛИ С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ НЕ РАВНО	---	---	---	616
337	AND<D	И С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ МЕНЬШЕ	---	---	---	616
337	LD<D	ЗАГРУЗКА С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ МЕНЬШЕ	---	---	---	616
337	OR<D	ИЛИ С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ МЕНЬШЕ	---	---	---	616
338	AND<=D	И С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ МЕНЬШЕ ИЛИ РАВНО	---	---	---	616
338	LD<=D	ЗАГРУЗКА С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ МЕНЬШЕ ИЛИ РАВНО	---	---	---	616
338	OR<=D	ИЛИ С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ МЕНЬШЕ ИЛИ РАВНО	---	---	---	616
339	AND>D	И С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ БОЛЬШЕ	---	---	---	616
339	LD>D	ЗАГРУЗКА С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ БОЛЬШЕ	---	---	---	616
339	OR>D	ИЛИ С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ БОЛЬШЕ	---	---	---	616
340	AND>=D	И С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ БОЛЬШЕ ИЛИ РАВНО	---	---	---	616
340	LD>=D	ЗАГРУЗКА С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ БОЛЬШЕ ИЛИ РАВНО	---	---	---	616
340	OR>=D	ИЛИ С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ БОЛЬШЕ ИЛИ РАВНО	---	---	---	616
341	AND = DT	И ВРЕМЯ РАВНО	---	---	---	236

Код функции	Мнемоническое обозначение	Команда	Выполнение по положительному фронту	Выполнение по отрицательному фронту	Модификатор мгновенного обновления	Стр.
341	LD= DT	ЗАГРУЗКА ВРЕМЯ РАВНО	---	---	---	236
341	OR = DT	ИЛИ ВРЕМЯ РАВНО	---	---	---	236
342	AND<>DT	И ВРЕМЯ НЕ РАВНО	---	---	---	236
342	LD<>DT	ЗАГРУЗКА ВРЕМЯ НЕ РАВНО	---	---	---	236
342	OR<>DT	ИЛИ ВРЕМЯ НЕ РАВНО	---	---	---	236
343	AND<DT	И ВРЕМЯ МЕНЬШЕ	---	---	---	236
343	LD<DT	ЗАГРУЗКА ВРЕМЯ МЕНЬШЕ	---	---	---	236
343	OR<DT	ИЛИ ВРЕМЯ МЕНЬШЕ	---	---	---	236
344	AND<=DT	И ВРЕМЯ МЕНЬШЕ ИЛИ РАВНО	---	---	---	236
344	LD<=DT	ЗАГРУЗКА ВРЕМЯ МЕНЬШЕ ИЛИ РАВНО	---	---	---	236
344	OR<=DT	ИЛИ ВРЕМЯ МЕНЬШЕ ИЛИ РАВНО	---	---	---	236
345	AND>DT	И ВРЕМЯ БОЛЬШЕ	---	---	---	236
345	LD>DT	ЗАГРУЗКА ВРЕМЯ БОЛЬШЕ	---	---	---	236
345	OR>DT	ИЛИ ВРЕМЯ БОЛЬШЕ	---	---	---	236
346	AND>=DT	И ВРЕМЯ БОЛЬШЕ ИЛИ РАВНО	---	---	---	236
346	LD>=DT	ЗАГРУЗКА ВРЕМЯ БОЛЬШЕ ИЛИ РАВНО	---	---	---	236
346	OR>=DT	ИЛИ ВРЕМЯ БОЛЬШЕ ИЛИ РАВНО	---	---	---	236
350	AND TST	И ПРОВЕРКА БИТА	---	---	---	123
350	LD TST	ЗАГРУЗКА ПРОВЕРКА БИТА	---	---	---	123
350	OR TST	ИЛИ ПРОВЕРКА БИТА	---	---	---	123
351	AND TSTN	И ПРОВЕРКА БИТА НЕ	---	---	---	123
351	LD TSTN	ЗАГРУЗКА ПРОВЕРКА БИТА НЕ	---	---	---	123
351	OR TSTN	ИЛИ ПРОВЕРКА БИТА НЕ	---	---	---	123
400	+	СЛОЖИТЬ БЕЗ ПЕРЕНОСА ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ	@+	---	---	367
401	+L	СЛОЖИТЬ БЕЗ ПЕРЕНОСА ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ	@+L	---	---	370
402	+C	СЛОЖИТЬ С ПЕРЕНОСОМ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ	@+C	---	---	371
403	+CL	СЛОЖИТЬ С ПЕРЕНОСОМ ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ	@+CL	---	---	374
404	+B	СЛОЖИТЬ БЕЗ ПЕРЕНОСА ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА	@+B	---	---	376
405	+BL	СЛОЖИТЬ БЕЗ ПЕРЕНОСА ДВОЙНЫЕ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА	@+BL	---	---	377
406	+BC	СЛОЖИТЬ С ПЕРЕНОСОМ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА	@+BC	---	---	379
407	+BCL	СЛОЖИТЬ С ПЕРЕНОСОМ ДВОЙНЫЕ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА	@+BCL	---	---	381
410	–	ВЫЧЕСТЬ БЕЗ ПЕРЕНОСА ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ	@–	---	---	382

Код функции	Мнемоническое обозначение	Команда	Выполнение по положительному фронту	Выполнение по отрицательному фронту	Модификатор мгновенного обновления	Стр.
411	–L	ВЫЧЕСТЬ БЕЗ ПЕРЕНОСА ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ	@–L	---	---	385
412	–C	ВЫЧЕСТЬ С ПЕРЕНОСОМ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ	@–C	---	---	389
413	–CL	ВЫЧЕСТЬ С ПЕРЕНОСОМ ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ	@–CL	---	---	391
414	–B	ВЫЧЕСТЬ БЕЗ ПЕРЕНОСА ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА	@–B	---	---	394
415	–BL	ВЫЧЕСТЬ БЕЗ ПЕРЕНОСА ДВОЙНЫЕ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА	@–BL	---	---	396
416	–BC	ВЫЧЕСТЬ С ПЕРЕНОСОМ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА	@–BC	---	---	399
417	–BCL	ВЫЧЕСТЬ С ПЕРЕНОСОМ ДВОЙНЫЕ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА	@–BCL	---	---	401
420	*	УМНОЖИТЬ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ	@*	---	---	403
421	*L	УМНОЖИТЬ ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ	@*L	---	---	405
422	*U	УМНОЖИТЬ ДВОИЧНЫЕ СЛОВА БЕЗ ЗНАКА	@*U	---	---	407
423	*UL	УМНОЖИТЬ ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА БЕЗ ЗНАКА	@*UL	---	---	408
424	*B	УМНОЖИТЬ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА	@*B	---	---	410
425	*BL	УМНОЖИТЬ ДВОЙНЫЕ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА	@*BL	---	---	412
430	/	ДЕЛИТЬ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ	@/	---	---	413
431	/L	ДЕЛИТЬ ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ	@/L	---	---	415
432	/U	ДЕЛИТЬ ДВОИЧНЫЕ СЛОВА БЕЗ ЗНАКА	@/U	---	---	417
433	/UL	ДЕЛИТЬ ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА БЕЗ ЗНАКА	@/UL	---	---	419
434	/B	ДЕЛИТЬ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА	@/B	---	---	421
435	/BL	ДЕЛИТЬ ДВОЙНЫЕ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА	@/BL	---	---	422
448	FSTR	ПЛАВАЮЩАЯ ЗАПЯТАЯ В ASCII	@FSTR	---	---	563
449	FVAL	ASCII В ПЛАВАЮЩАЯ ЗАПЯТАЯ	@FVAL	---	---	568
450	FIX	ПЛАВ. ЗАПЯТАЯ В 16-БИТ	@FIX	---	---	579
451	FIXL	ПЛАВ. ЗАПЯТАЯ В 32-БИТ	@FIXL	---	---	581
452	FLT	16-БИТ В ПЛАВ. ЗАПЯТАЯ	@FLT	---	---	582
453	FLTL	32-БИТ В ПЛАВ. ЗАПЯТАЯ	@FLTL	---	---	584
454	+F	СЛОЖЕНИЕ С ПЛАВ. ЗАПЯТОЙ	@+F	---	---	528

Код функции	Мнемоническое обозначение	Команда	Выполнение по положительному фронту	Выполнение по отрицательному фронту	Модификатор мгновенного обновления	Стр.
455	–F	ВЫЧИТАНИЕ С ПЛАВ. ЗАПЯТОЙ	@–F	---	---	530
456	*F	УМНОЖЕНИЕ С ПЛАВ. ЗАПЯТОЙ	@*F	---	---	532
457	/F	ДЕЛЕНИЕ С ПЛАВ. ЗАПЯТОЙ	@/F	---	---	534
458	RAD	ГРАДУСЫ В РАДИАНЫ	@RAD	---	---	536
459	DEG	РАДИАНЫ В ГРАДУСЫ	@DEG	---	---	538
460	SIN	СИНУС	@SIN	---	---	540
461	COS	КОСИНУС	@COS	---	---	542
462	TAN	ТАНГЕНС	@TAN	---	---	543
463	ASIN	АРКСИНУС	@ASIN	---	---	545
464	ACOS	АРККОСИНУС	@ACOS	---	---	547
465	ATAN	АРКТАНГЕНС	@ATAN	---	---	549
466	SQRT	КВАДРАТНЫЙ КОРЕНЬ	@SQRT	---	---	551
467	EXP	ЭКСПОНЕНТА	@EXP	---	---	553
468	LOG	ЛОГАРИФМ	@LOG	---	---	555
470	BINS	BCD СО ЗНАКОМ В ДВОИЧНОЕ	@BINS	---	---	459
471	BCDS	ДВОИЧНОЕ СО ЗНАКОМ В BCD	@BCDS	---	---	465
472	BISL	ДВОЙНОЕ BCD СО ЗНАКОМ В ДВОИЧНОЕ	@BISL	---	---	462
473	BDSL	ДВОЙНОЕ ДВОИЧНОЕ СО ЗНАКОМ В BCD	@BDSL	---	---	468
474	GRY	ПРЕОБРАЗОВАНИЕ КОДА ГРЕЯ	@GRY	---	---	472
486	SCL2	ИЗМЕНЕНИЕ МАСШТАБА 2	@SCL2	---	---	718
487	SCL3	ИЗМЕНЕНИЕ МАСШТАБА 3	@SCL3	---	---	722
490	CMND	ДОСТАВИТЬ КОМАНДУ	@CMND	---	---	958
498	MOVL	ПЕРЕДАТЬ ДВОЙНОЕ СЛОВО	@MOVL	---	---	274
499	MVNL	ПЕРЕДАТЬ ДОПОЛНЕНИЕ К ДВОЙНОМУ СЛОВУ	@MVNL	---	---	276
502	BCMP2	РАСШИРЕННОЕ СРАВНЕНИЕ С ДИАПАЗОНАМИ	@BCMP2	---	---	261
510	CJP	УСЛОВНЫЙ ПЕРЕХОД	---	---	---	172
511	CJPN	УСЛОВНЫЙ ПЕРЕХОД	---	---	---	172
512	FOR	ЦИКЛЫ FOR-NEXT	---	---	---	178
513	NEXT	ЦИКЛЫ FOR-NEXT	---	---	---	178
514	BREAK	ВЫХОД ИЗ ЦИКЛА	---	---	---	182
515	JMP0	МНОЖЕСТВЕННЫЙ ПЕРЕХОД	---	---	---	176
516	JME0	КОНЕЦ МНОЖЕСТВЕННОГО ПЕРЕХОДА	---	---	---	176
517	MILH	ВЛОЖЕННАЯ БЛОКИРОВКА С ЗАПОМИНАНИЕМ ФРОНТА	---	---	---	155
518	MILR	ВЛОЖЕННАЯ БЛОКИРОВКА БЕЗ ЗАПОМИНАНИЯ ФРОНТА	---	---	---	155
519	MILC	ОТМЕНА ВЛОЖЕННОЙ БЛОКИРОВКИ	---	---	---	155
520	NOT	НЕ	---	---	---	120
521	UP	УСЛОВИЕ ВКЛ	---	---	---	121
522	DOWN	УСЛОВИЕ ВЫКЛ	---	---	---	121
530	SETA	УСТАНОВИТЬ НЕСКОЛЬКО БИТОВ	@SETA	---	---	138

Код функции	Мнемоническое обозначение	Команда	Выполнение по положительному фронту	Выполнение по отрицательному фронту	Модификатор мгновенного обновления	Стр.
531	RSTA	СБРОСИТЬ НЕСКОЛЬКО БИТОВ	@RSTA	---	---	138
532	SETB	УСТАНОВИТЬ ОДИН БИТ	@SETB	---	!SETB	141
533	RSTB	СБРОСИТЬ ОДИН БИТ	@RSTB	---	!RSTB	141
534	OUTB	ВЫВОД В ОДИН БИТ	@OUTB	---	!OUTB	144
540	TMHH	1 мс ТАЙМЕР	---	---	---	195
542	TIML	ДОЛГОСРОЧНЫЙ ТАЙМЕР	---	---	---	203
543	MTIM	МНОГОВЫХОДНОЙ ТАЙМЕР	---	---	---	206
545	CNR	СБРОС ТАЙМЕРА/ СЧЕТЧИКА	@CNR	---	---	220
546	CNTX	СЧЕТЧИК	---	---	---	212
547	CNRX	СБРОС ТАЙМЕРА/ СЧЕТЧИКА	---	---	---	220
548	CNTRX	РЕВЕРСИВНЫЙ СЧЕТЧИК	---	---	---	215
550	TIMX	ТАЙМЕР	---	---	---	186
551	TIMHX	СКОРОСТНОЙ ТАЙМЕР	---	---	---	191
552	TMHHX	1 мс ТАЙМЕР	---	---	---	195
553	TIMLX	ДОЛГОСРОЧНЫЙ ТАЙМЕР	---	---	---	203
554	MTIMX	МНОГОВЫХОДНОЙ ТАЙМЕР	---	---	---	206
555	TTIMX	НАКАПЛИВАЮЩИЙ ТАЙМЕР	---	---	---	199
560	MOVR	ПЕРЕДАТЬ В РЕГИСТР	@MOVR	---	---	296
561	MOVRW	ПЕРЕДАТЬ ТЕКУЩЕЕ ЗНАЧЕНИЕ ТАЙМЕРА/ СЧЕТЧИКА В РЕГИСТР	@MOVRW	---	---	298
562	XCGL	ОБМЕН ДВОЙНЫМИ СЛОВАМИ	@XCGL	---	---	290
565	XFERC	ПЕРЕДАЧА БЛОКА	@XFERC	---	---	1148
566	DISTC	РАЗМЕЩЕНИЕ ОДИНОЧНОГО СЛОВА	@DISTC	---	---	1150
567	COLLC	СБОР ДАННЫХ	@COLLC	---	---	1154
568	MOVBC	ПЕРЕДАТЬ БИТ	@MOVBC	---	---	1158
570	ASLL	СДВИГ ДВОЙНОГО СЛОВА ВЛЕВО	@ASLL	---	---	312
571	ASRL	СДВИГ ДВОЙНОГО СЛОВА ВПРАВО	@ASRL	---	---	316
572	ROLL	ЦИКЛИЧЕСКИЙ СДВИГ ДВОЙНОГО СЛОВА ВЛЕВО	@ROLL	---	---	319
573	RORL	ЦИКЛИЧЕСКИЙ СДВИГ ДВОЙНОГО СЛОВА ВПРАВО	@RORL	---	---	322
574	RLNC	ЦИКЛИЧЕСКИЙ СДВИГ ВЛЕВО БЕЗ ПЕРЕНОСА	@RLNC	---	---	324
575	RRNC	ЦИКЛИЧЕСКИЙ СДВИГ ВПРАВО БЕЗ ПЕРЕНОСА	@RRNC	---	---	327
576	RLNL	ЦИКЛИЧЕСКИЙ СДВИГ ДВОЙНОГО СЛОВА ВЛЕВО БЕЗ ПЕРЕНОСА	@RLNL	---	---	326
577	RRNL	ЦИКЛИЧЕСКИЙ СДВИГ ДВОЙНОГО СЛОВА ВПРАВО БЕЗ ПЕРЕНОСА	@RRNL	---	---	329
578	NSFL	СДВИГ N БИТОВ ДАННЫХ ВЛЕВО	@NSFL	---	---	334
579	NSFR	СДВИГ N БИТОВ ДАННЫХ ВПРАВО	@NSFR	---	---	336
580	NASL	СДВИГ НА N БИТОВ ВЛЕВО	@NASL	---	---	338

Код функции	Мнемоническое обозначение	Команда	Выполнение по положительному фронту	Выполнение по отрицательному фронту	Модификатор мгновенного обновления	Стр.
581	NASR	СДВИГ НА N БИТОВ ВПРАВО	@NASR	---	---	344
582	NSLL	СДВИГ ДВОЙНОГО СЛОВА НА N БИТОВ ВЛЕВО	@NSLL	---	---	341
583	NSRL	СДВИГ ДВОЙНОГО СЛОВА НА N БИТОВ ВПРАВО	@NSRL	---	---	346
590	++	УВЕЛИЧИТЬ ДВОИЧНОЕ СЛОВО	@++	---	---	349
591	++L	УВЕЛИЧИТЬ ДВОЙНОЕ ДВОИЧНОЕ СЛОВО	@++L	---	---	351
592	– –	УМЕНЬШИТЬ ДВОИЧНОЕ СЛОВО	@– –	---	---	353
593	– –L	УМЕНЬШИТЬ ДВОЙНОЕ ДВОИЧНОЕ СЛОВО	@– –L	---	---	355
594	++B	УВЕЛИЧИТЬ ДВОИЧНО-ДЕСЯТИЧНОЕ СЛОВО	@++B	---	---	358
595	++BL	УВЕЛИЧИТЬ ДВОЙНОЕ ДВОИЧНО-ДЕСЯТИЧНОЕ СЛОВО	@++BL	---	---	360
596	– –B	УМЕНЬШИТЬ ДВОИЧНО-ДЕСЯТИЧНОЕ СЛОВО	@– –B	---	---	362
597	– –BL	УМЕНЬШИТЬ ДВОЙНОЕ ДВОИЧНО-ДЕСЯТИЧНОЕ СЛОВО	@– –BL	---	---	364
600	SIGN	16-БИТОВОЕ ДВОИЧНОЕ СО ЗНАКОМ В 32-БИТОВОЕ	@SIGN	---	---	435
610	ANDL	ДВОЙНОЕ ЛОГИЧЕСКОЕ И	@ANDL	---	---	478
611	ORWL	ДВОЙНОЕ ЛОГИЧЕСКОЕ ИЛИ	@ORWL	---	---	482
612	XORL	ДВОЙНОЕ ИСКЛЮЧАЮЩЕЕ ИЛИ	@XORL	---	---	485
613	XNRL	ДВОЙНОЕ ИСКЛЮЧАЮЩЕЕ НЕ-ИЛИ	@XNRL	---	---	488
614	COML	ДВОЙНОЕ ДОПОЛНЕНИЕ	@COML	---	---	492
620	ROTB	КВАДРАТНЫЙ КОРЕНЬ ИЗ ДВОИЧНОГО ЧИСЛА	@ROTB	---	---	493
621	BCNTC	ПОДСЧЕТ БИТОВ	@BCNTC	---	---	1160
630	SSET	СОЗДАТЬ СТЕК	@SSET	---	---	621
631	DIM	СОЗДАТЬ ТАБЛИЦУ УКАЗАННЫХ РАЗМЕРОВ	@DIM	---	---	632
632	PUSH	ЗАПИСАТЬ В СТЕК	@PUSH	---	---	623
633	FIFO	ПЕРВЫМ ПРИШЕЛ — ПЕРВЫМ ВЫШЕЛ	@FIFO	---	---	626
634	LIFO	ПОСЛЕДНИМ ПРИШЕЛ — ПЕРВЫМ ВЫШЕЛ	@LIFO	---	---	629
635	SETR	ЗАПИСАТЬ РАСПОЛОЖЕНИЕ СТРОКИ	@SETR	---	---	634
636	GETR	СЧИТАТЬ НОМЕР СТРОКИ	@GETR	---	---	636
637	SWAP	ПОМЕНЯТЬ МЕСТАМИ БАЙТЫ	@SWAP	---	---	641
638	SNUM	СЧИТАТЬ РАЗМЕР СТЕКА	@SNUM	---	---	655
639	SREAD	ПРОЧИТАТЬ ДАННЫЕ ИЗ СТЕКА	@SREAD	---	---	658
640	SWRIT	ЗАПИСАТЬ ДАННЫЕ В СТЕК	@SWRIT	---	---	661
641	SINS	ВСТАВИТЬ ДАННЫЕ В СТЕК	@SINS	---	---	664
642	SDEL	УДАЛИТЬ ДАННЫЕ ИЗ СТЕКА	@SDEL	---	---	667

Код функции	Мнемоническое обозначение	Команда	Выполнение по положительному фронту	Выполнение по отрицательному фронту	Модификатор мгновенного обновления	Стр.
650	LEN$	ДЛИНА СТРОКИ	@LEN$	---	---	1118
652	LEFT$	СКОПИРОВАТЬ СИМВОЛЫ СЛЕВА	@LEFT$	---	---	1109
653	RGHT$	СКОПИРОВАТЬ СИМВОЛЫ СПРАВА	@RGHT$	---	---	1111
654	MID$	СКОПИРОВАТЬ СИМВОЛЫ ВНУТРИ СТРОКИ	@MID$	---	---	1113
656	+$	ОБЪЕДИНИТЬ СТРОКИ	@+$	---	---	1106
657	INS$	ВСТАВИТЬ В СТРОКУ	@INS$	---	---	1129
658	DEL$	УДАЛИТЬ СТРОКУ	@DEL$	---	---	1123
660	FIND$	НАЙТИ В СТРОКЕ	@FIND$	---	---	1116
661	RPLC$	ЗАМЕНИТЬ В СТРОКЕ	@RPLC$	---	---	1120
664	MOV$	КОПИРОВАТЬ СТРОКУ	@MOV$	---	---	1104
665	XCHG$	ОБМЕН СТРОКАМИ	@XCHG$	---	---	1125
666	CLR$	ОЧИСТИТЬ СТРОКУ	@CLR$	---	---	1127
670	AND =$	И СТРОКА РАВНА	---	---	---	1132
670	LD =$	ЗАГРУЗКА СТРОКА РАВНА	---	---	---	1132
670	OR =$	ИЛИ СТРОКА РАВНА	---	---	---	1132
671	AND <>$	И СТРОКА НЕ РАВНА	---	---	---	1132
671	LD <>$	ЗАГРУЗКА СТРОКА НЕ РАВНА	---	---	---	1132
671	OR <>$	ИЛИ СТРОКА НЕ РАВНА	---	---	---	1132
672	AND <$	И СТРОКА МЕНЬШЕ	---	---	---	1132
672	LD <$	ЗАГРУЗКА СТРОКА МЕНЬШЕ	---	---	---	1132
672	OR <$	ИЛИ СТРОКА МЕНЬШЕ	---	---	---	1132
673	AND <=$	И СТРОКА МЕНЬШЕ ИЛИ РАВНА	---	---	---	1132
673	LD <=$	ЗАГРУЗКА СТРОКА МЕНЬШЕ ИЛИ РАВНА	---	---	---	1132
673	OR <=$	ИЛИ СТРОКА МЕНЬШЕ ИЛИ РАВНА	---	---	---	1132
674	AND >$	И СТРОКА БОЛЬШЕ	---	---	---	1132
674	LD >$	ЗАГРУЗКА СТРОКА БОЛЬШЕ	---	---	---	1132
674	OR >$	ИЛИ СТРОКА БОЛЬШЕ	---	---	---	1132
675	AND >=$	И СТРОКА БОЛЬШЕ ИЛИ РАВНА	---	---	---	1132
675	LD >=$	ЗАГРУЗКА СТРОКА БОЛЬШЕ ИЛИ РАВНА	---	---	---	1132
675	OR >=$	ИЛИ СТРОКА БОЛЬШЕ ИЛИ РАВНА	---	---	---	1132
680	LMT	ОГРАНИЧЕНИЕ	@LMT	---	---	696
681	BAND	ЗОНА НЕЧУВСТВИТЕЛЬНОСТИ	@BAND	---	---	698
682	ZONE	МЕРТВАЯ ЗОНА	@ZONE	---	---	701
685	TPO	ВЫХОД ШИМ	---	---	---	704
690	MSKS	УСТАНОВИТЬ МАСКУ ПРЕРЫВАНИЙ	@MSKS	---	---	755
691	CLI	ОЧИСТИТЬ ПРЕРЫВАНИЕ	@CLI	---	---	763
692	MSKR	СЧИТАТЬ МАСКУ ПРЕРЫВАНИЙ	@MSKR	---	---	759
693	DI	ЗАПРЕТИТЬ ПРЕРЫВАНИЯ	@DI	---	---	766
694	EI	РАЗРЕШИТЬ ПРЕРЫВАНИЯ	---	---	---	768
720	EXPLT	ОТПРАВИТЬ ЯВНОЕ СООБЩЕНИЕ	@EXPLT	---	---	967
721	EGATR	ЯВНЫЙ ЗАПРОС АТРИБУТА	@EGATR	---	---	975
722	ESATR	ЯВНАЯ УСТАНОВКА АТРИБУТА	@ESATR	---	---	982

Код функции	Мнемоническое обозначение	Команда	Выполнение по положительному фронту	Выполнение по отрицательному фронту	Модификатор мгновенного обновления	Стр.
723	ECHRD	ЯВНОЕ ЧТЕНИЕ СЛОВА	@ECHRD	---	---	989
724	ECHWR	ЯВНАЯ ЗАПИСЬ СЛОВА	@ECHWR	---	---	993
730	CADD	ДОБАВИТЬ К КАЛЕНДАРНОЙ ДАТЕ	@CADD	---	---	1006
731	CSUB	ВЫЧЕСТЬ ИЗ КАЛЕНДАРНОЙ ДАТЫ	@CSUB	---	---	1009
735	DATE	КОРРЕКТИРОВКА ЧАСОВ	@DATE	---	---	1017
750	GSBS	ВЫЗОВ ГЛОБАЛЬНОЙ ПОДПРОГРАММЫ	@GSBS	---	---	743
751	GSBN	ВХОД В ГЛОБАЛЬНУЮ ПОДПРОГРАММУ	---	---	---	751
752	GRET	ВЫХОД ИЗ ГЛОБАЛЬНОЙ ПОДПРОГРАММЫ	---	---	---	754
801	BEND	КОНЕЦ ПРОГРАММНОГО БЛОКА	---	---	---	1072
802	IF	БЛОК УСЛОВНОГО ВЕТВЛЕНИЯ	---	---	---	1077
802	IF	БЛОК УСЛОВНОГО ВЕТВЛЕНИЯ	---	---	---	1077
802	IF NOT	БЛОК УСЛОВНОГО ВЕТВЛЕНИЯ НЕ	---	---	---	1077
803	ELSE	ИНАЧЕ	---	---	---	1077
804	IEND	КОНЕЦ ЕСЛИ	---	---	---	1077
805	WAIT	ОДИН ЦИКЛ И ОЖИДАНИЕ	---	---	---	1084
805	WAIT	ОДИН ЦИКЛ И ОЖИДАНИЕ	---	---	---	1084
805	WAIT NOT	ОДИН ЦИКЛ И ОЖИДАНИЕ НЕ	---	---	---	1084
806	EXIT	ВЫХОД ИЗ БЛОКА ПО УСЛОВИЮ	---	---	---	1081
806	EXIT	ВЫХОД ИЗ БЛОКА ПО УСЛОВИЮ	---	---	---	1081
806	EXIT NOT	ВЫХОД ИЗ БЛОКА ПО УСЛОВИЮ НЕ	---	---	---	1081
809	LOOP	ЦИКЛ	---	---	---	1099
810	LEND	КОНЕЦ ЦИКЛА	---	---	---	1099
810	LEND	КОНЕЦ ЦИКЛА	---	---	---	1099
810	LEND NOT	КОНЕЦ ЦИКЛА НЕ	---	---	---	1099
811	BPPS	ПРИОСТАНОВКА ПРОГРАММНОГО БЛОКА	---	---	---	1072
812	BPRS	ПЕРЕЗАПУСК ПРОГРАММНОГО БЛОКА	---	---	---	1075
813	TIMW	ТАЙМЕР ОЖИДАНИЯ	---	---	---	1088
814	CNTW	СЧЕТЧИК ОЖИДАНИЯ	---	---	---	1092
815	TMHW	СКОРОСТНОЙ ТАЙМЕР ОЖИДАНИЯ	---	---	---	1096
816	TIMWX	ТАЙМЕР ОЖИДАНИЯ	---	---	---	1088
817	TMHWX	СКОРОСТНОЙ ТАЙМЕР ОЖИДАНИЯ	---	---	---	1096
818	CNTWX	СЧЕТЧИК ОЖИДАНИЯ	---	---	---	1092
820	TKON	ВКЛЮЧИТЬ ЗАДАЧУ	@TKON	---	---	1137
821	TKOF	ВЫКЛЮЧИТЬ ЗАДАЧУ	@TKOF	---	---	1141
840	PWR	ВОЗВЕДЕНИЕ В СТЕПЕНЬ	@PWR	---	---	557
841	FIXD	ПЛАВ. ЗАПЯТАЯ ДВОЙНОЙ ТОЧНОСТИ В 16-БИТ ДВОИЧНОЕ	@FIXD	---	---	579
842	FIXLD	ПЛАВ. ЗАПЯТАЯ ДВОЙНОЙ ТОЧНОСТИ В 32-БИТ ДВОИЧНОЕ	@FIXLD	---	---	581
843	DBL	16-БИТ ДВОИЧНОЕ В ПЛАВ. ЗАПЯТАЯ ДВОЙНОЙ ТОЧНОСТИ	@DBL	---	---	582

Код функции	Мнемоническое обозначение	Команда	Выполнение по положительному фронту	Выполнение по отрицательному фронту	Модификатор мгновенного обновления	Стр.
844	DBLL	32-БИТ ДВОИЧНОЕ В ПЛАВ. ЗАПЯТАЯ ДВОЙНОЙ ТОЧНОСТИ	@DBLL	---	---	584
845	+D	СЛОЖЕНИЕ С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ	@+D	---	---	585
846	–D	ВЫЧИТАНИЕ С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ	@–D	---	---	587
847	*D	УМНОЖЕНИЕ С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ	@*D	---	---	589
848	/D	ДЕЛЕНИЕ С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ	@/D	---	---	591
849	RADD	ГРАДУСЫ В РАДИАНЫ ДВОЙНОЙ ТОЧНОСТИ	@RADD	---	---	593
850	DEGD	РАДИАНЫ В ГРАДУСЫ ДВОЙНОЙ ТОЧНОСТИ	@RADD	---	---	595
851	SIND	СИНУС ДВОЙНОЙ ТОЧНОСТИ	@SIND	---	---	597
852	COSD	КОСИНУС ДВОЙНОЙ ТОЧНОСТИ	@COSD	---	---	598
853	TAND	ТАНГЕНС ДВОЙНОЙ ТОЧНОСТИ	@TAND	---	---	600
854	ASIND	АРКСИНУС ДВОЙНОЙ ТОЧНОСТИ	@ASIND	---	---	602
855	ACOSD	АРККОСИНУС ДВОЙНОЙ ТОЧНОСТИ	@ACOSD	---	---	604
856	ATAND	АРКТАНГЕНС ДВОЙНОЙ ТОЧНОСТИ	@ATAND	---	---	606
857	SQRTD	КВАДРАТНЫЙ КОРЕНЬ ДВОЙНОЙ ТОЧНОСТИ	@SQRTD	---	---	608
858	EXPD	ЭКСПОНЕНТА ДВОЙНОЙ ТОЧНОСТИ	@EXPD	---	---	610
859	LOGD	ЛОГАРИФМ ДВОЙНОЙ ТОЧНОСТИ	@LOGD	---	---	612
860	PWRD	ВОЗВЕДЕНИЕ В СТЕПЕНЬ ДВОЙНОЙ ТОЧНОСТИ	@PWRD	---	---	581
880	INI	УПРАВЛЕНИЕ РЕЖИМОМ	@INI	---	---	770
881	PRV	ЧТЕНИЕ ТЕКУЩ. ЗНАЧ. СКОРОСТНОГО СЧЕТЧИКА	@PRV	---	---	775
882	CTBL	ЗАГРУЗКА ТАБЛИЦЫ СРАВНЕНИЯ	@CTBL	---	---	785
883	PRV2	ПРЕОБРАЗОВАНИЕ ЧАСТОТЫ НА ВХОДЕ СЧЕТЧИКА	@PRV2	---	---	781
885	SPED	ВЫДАЧА ИМПУЛЬСОВ СКОРОСТИ	@SPED	---	---	790
886	PULS	ЗАДАТЬ КОЛ-ВО ИМПУЛЬСОВ	@PULS	---	---	796
887	PLS2	ВЫДАЧА ИМПУЛЬСОВ	@PLS2	---	---	799
888	ACC	УПРАВЛЕНИЕ РАЗГОНОМ	@ACC	---	---	808
889	ORG	ПОИСК ИСХОДНОГО ПОЛОЖЕНИЯ	@ORG	---	---	817
891	PWN	ИМПУЛЬСЫ С ПЕРЕМЕННОЙ СКВАЖНОСТЬЮ	@PWN	---	---	822

Предварительная версия
Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

Приложение C

Список команд в алфавитном порядке

Символы

Мнемоническое обозначение	Код функции	Команда	Стр.
7SEG	214	ВЫВОД НА 7-СЕГМЕНТНЫЙ ИНДИКАТОР	865
+	400	СЛОЖИТЬ БЕЗ ПЕРЕНОСА ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ	367
+$	656	ОБЪЕДИНИТЬ СТРОКИ	1106
++	590	УВЕЛИЧИТЬ ДВОИЧНОЕ СЛОВО	349
++B	594	УВЕЛИЧИТЬ ДВОИЧНО-ДЕСЯТИЧНОЕ СЛОВО	358
++BL	595	УВЕЛИЧИТЬ ДВОЙНОЕ ДВОИЧНО-ДЕСЯТИЧНОЕ СЛОВО	360
++L	591	УВЕЛИЧИТЬ ДВОЙНОЕ ДВОИЧНОЕ СЛОВО	351
+B	404	СЛОЖИТЬ БЕЗ ПЕРЕНОСА ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА	376
+BC	406	СЛОЖИТЬ С ПЕРЕНОСОМ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА	379
+BCL	407	СЛОЖИТЬ С ПЕРЕНОСОМ ДВОЙНЫЕ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА	381
+BL	405	СЛОЖИТЬ БЕЗ ПЕРЕНОСА ДВОЙНЫЕ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА	377
+C	402	СЛОЖИТЬ С ПЕРЕНОСОМ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ	371
+CL	403	СЛОЖИТЬ С ПЕРЕНОСОМ ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ	374
+D	845	СЛОЖЕНИЕ С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ	585
+F	454	СЛОЖЕНИЕ С ПЛАВ. ЗАПЯТОЙ	528
+L	401	СЛОЖИТЬ БЕЗ ПЕРЕНОСА ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ	370
–	410	ВЫЧЕСТЬ БЕЗ ПЕРЕНОСА ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ	382
– –	592	УМЕНЬШИТЬ ДВОИЧНОЕ СЛОВО	353
– –B	596	УМЕНЬШИТЬ ДВОИЧНО-ДЕСЯТИЧНОЕ СЛОВО	362
– –BL	597	УМЕНЬШИТЬ ДВОЙНОЕ ДВОИЧНО-ДЕСЯТИЧНОЕ СЛОВО	364
– –L	593	УМЕНЬШИТЬ ДВОЙНОЕ ДВОИЧНОЕ СЛОВО	355
–B	414	ВЫЧЕСТЬ БЕЗ ПЕРЕНОСА ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА	394
–BC	416	ВЫЧЕСТЬ С ПЕРЕНОСОМ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА	399
–BCL	417	ВЫЧЕСТЬ С ПЕРЕНОСОМ ДВОЙНЫЕ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА	401
–BL	415	ВЫЧЕСТЬ БЕЗ ПЕРЕНОСА ДВОЙНЫЕ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА	396
–C	412	ВЫЧЕСТЬ С ПЕРЕНОСОМ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ	389
–CL	413	ВЫЧЕСТЬ С ПЕРЕНОСОМ ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ	391
–D	846	ВЫЧИТАНИЕ С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ	587
–F	455	ВЫЧИТАНИЕ С ПЛАВ. ЗАПЯТОЙ	530
*	420	УМНОЖИТЬ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ	403
*B	424	УМНОЖИТЬ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА	410
*BL	425	УМНОЖИТЬ ДВОЙНЫЕ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА	412
*D	847	УМНОЖЕНИЕ С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ	589
*F	456	УМНОЖЕНИЕ С ПЛАВ. ЗАПЯТОЙ	532
*L	421	УМНОЖИТЬ ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ	405
*U	422	УМНОЖИТЬ ДВОИЧНЫЕ СЛОВА БЕЗ ЗНАКА	407
*UL	423	УМНОЖИТЬ ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА БЕЗ ЗНАКА	408
–L	411	ВЫЧЕСТЬ БЕЗ ПЕРЕНОСА ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ	385
/	430	ДЕЛИТЬ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ	413
/B	434	ДЕЛИТЬ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА	421
/BL	435	ДЕЛИТЬ ДВОЙНЫЕ ДВОИЧНО-ДЕСЯТИЧНЫЕ СЛОВА	422

Мнемоническое обозначение	Код функции	Команда	Стр.
/D	848	ДЕЛЕНИЕ С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ	591
/F	457	ДЕЛЕНИЕ С ПЛАВ. ЗАПЯТОЙ	534
/L	431	ДЕЛИТЬ ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ	415
/U	432	ДЕЛИТЬ ДВОИЧНЫЕ СЛОВА БЕЗ ЗНАКА	417
/UL	433	ДЕЛИТЬ ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА БЕЗ ЗНАКА	419

A

Мнемоническое обозначение	Код функции	Команда	Стр.
ACC	888	УПРАВЛЕНИЕ РАЗГОНОМ	808
ACOS	464	АРККОСИНУС	547
ACOSD	855	АРККОСИНУС ДВОЙНОЙ ТОЧНОСТИ	606
AND	---	И	104
AND<	310	И МЕНЬШЕ	230
AND <$	672	И СТРОКА МЕНЬШЕ	1132
AND<>	305	И НЕ РАВНО	230
AND <>$	671	И СТРОКА НЕ РАВНА	1132
AND<>D	336	И С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ НЕ РАВНО	616
AND<>DT	342	И ВРЕМЯ НЕ РАВНО	236
AND<>F	330	И С ПЛАВАЮЩЕЙ ЗАПЯТОЙ НЕ РАВНО	559
AND<>L	306	И ДВОЙНЫЕ СЛОВА НЕ РАВНЫ	230
AND<>S	307	И СЛОВА СО ЗНАКОМ НЕ РАВНЫ	230
AND<>SL	308	И ДВОЙНЫЕ СЛОВА СО ЗНАКОМ НЕ РАВНЫ	230
AND<D	337	И С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ МЕНЬШЕ	616
AND<DT	343	И ВРЕМЯ МЕНЬШЕ	236
AND<F	331	И С ПЛАВАЮЩЕЙ ЗАПЯТОЙ МЕНЬШЕ	559
AND<L	311	И ДВОЙНОЕ СЛОВО МЕНЬШЕ	230
AND<S	312	И СЛОВО СО ЗНАКОМ МЕНЬШЕ	230
AND<SL	313	И ДВОЙНОЕ СЛОВО СО ЗНАКОМ МЕНЬШЕ	230
AND=	300	И РАВНО	230
AND =$	670	И СТРОКА РАВНА	1132
AND =D	335	И С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ РАВНО	616
AND = DT	341	И ВРЕМЯ РАВНО	236
AND =F	329	И С ПЛАВАЮЩЕЙ ЗАПЯТОЙ РАВНО	559
AND =L	301	И ДВОЙНЫЕ СЛОВА РАВНЫ	230
AND =S	302	И СЛОВА СО ЗНАКОМ РАВНЫ	230
AND =SL	303	И ДВОЙНЫЕ СЛОВА СО ЗНАКОМ РАВНЫ	230
AND>	320	И БОЛЬШЕ	230
AND >$	674	И СТРОКА БОЛЬШЕ	1132
AND>D	339	И С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ БОЛЬШЕ	616
AND>DT	345	И ВРЕМЯ БОЛЬШЕ	236
AND>F	333	И С ПЛАВАЮЩЕЙ ЗАПЯТОЙ БОЛЬШЕ	559
AND>L	321	И ДВОЙНОЕ СЛОВО БОЛЬШЕ	230
AND>S	322	И СЛОВО СО ЗНАКОМ БОЛЬШЕ	230
AND>SL	323	И ДВОЙНОЕ СЛОВО СО ЗНАКОМ БОЛЬШЕ	230
AND LD	---	И ЗАГРУЗКА	111
AND NOT	---	И НЕ	105
AND TST	350	И ПРОВЕРКА БИТА	123
AND TSTN	351	И ПРОВЕРКА БИТА НЕ	123
AND<=	315	И МЕНЬШЕ ИЛИ РАВНО	230
AND <=$	673	И СТРОКА МЕНЬШЕ ИЛИ РАВНА	1132
AND<=D	338	И С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ МЕНЬШЕ ИЛИ РАВНО	616
AND<=DT	344	И ВРЕМЯ МЕНЬШЕ ИЛИ РАВНО	236
AND<=F	332	И С ПЛАВАЮЩЕЙ ЗАПЯТОЙ МЕНЬШЕ ИЛИ РАВНО	559
AND<=L	316	И ДВОЙНОЕ СЛОВО МЕНЬШЕ ИЛИ РАВНО	230

Мнемоническое обозначение	Код функции	Команда	Стр.
AND<=S	317	И СЛОВО СО ЗНАКОМ МЕНЬШЕ ИЛИ РАВНО	230
AND<=SL	318	И ДВОЙНОЕ СЛОВО СО ЗНАКОМ МЕНЬШЕ ИЛИ РАВНО	230
AND>=	325	И БОЛЬШЕ ИЛИ РАВНО	230
AND >=$	675	И СТРОКА БОЛЬШЕ ИЛИ РАВНА	1132
AND>=D	340	И С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ БОЛЬШЕ ИЛИ РАВНО	616
AND>=DT	346	И ВРЕМЯ БОЛЬШЕ ИЛИ РАВНО	236
AND>=F	334	И С ПЛАВАЮЩЕЙ ЗАПЯТОЙ БОЛЬШЕ ИЛИ РАВНО	230
AND>=L	326	И ДВОЙНОЕ СЛОВО БОЛЬШЕ ИЛИ РАВНО	230
AND>=S	327	И СЛОВО СО ЗНАКОМ БОЛЬШЕ ИЛИ РАВНО	230
AND>=SL	328	И ДВОЙНОЕ СЛОВО СО ЗНАКОМ БОЛЬШЕ ИЛИ РАВНО	230
ANDL	610	ДВОЙНОЕ ЛОГИЧЕСКОЕ И	478
ANDW	034	ЛОГИЧЕСКОЕ И	477
APR	069	МАТЕМАТИЧЕСКАЯ ФУНКЦИЯ	499
ASC	086	ПРЕОБРАЗОВАНИЕ В ASCII	446
ASFT	017	АСИНХРОННЫЙ РЕГИСТР СДВИГА	306
ASIN	463	АРКСИНУС	545
ASIND	854	АРКСИНУС ДВОЙНОЙ ТОЧНОСТИ	602
ASL	025	АРИФМЕТИЧЕСКИЙ СДВИГ ВЛЕВО	311
ASLL	570	СДВИГ ДВОЙНОГО СЛОВА ВЛЕВО	312
ASR	026	АРИФМЕТИЧЕСКИЙ СДВИГ ВПРАВО	314
ASRL	571	СДВИГ ДВОЙНОГО СЛОВА ВПРАВО	316
ATAN	465	АРКТАНГЕНС	549
ATAND	856	АРКТАНГЕНС ДВОЙНОЙ ТОЧНОСТИ	606
AVG	195	СРЕДНЕЕ ЗНАЧЕНИЕ	726

B

Мнемоническое обозначение	Код функции	Команда	Стр.
BAND	681	ЗОНА НЕЧУВСТВИТЕЛЬНОСТИ	698
BCD	024	ДВОИЧНОЕ В BCD	429
BCDL	059	ДВОЙНОЕ ДВОИЧНОЕ В ДВОЙНОЕ BCD	430
BCDS	471	ДВОИЧНОЕ СО ЗНАКОМ В BCD	465
BCMP	068	СРАВНИТЬ С ДИАПАЗОНАМИ БЕЗ ЗНАКА	258
BCMP2	502	РАСШИРЕННОЕ СРАВНЕНИЕ С ДИАПАЗОНАМИ	261
BCNT	067	ПОДСЧЕТ БИТОВ	514
BCNTC	621	ПОДСЧЕТ БИТОВ	1160
BDSL	473	ДВОЙНОЕ ДВОИЧНОЕ СО ЗНАКОМ В BCD	468
BEND	801	КОНЕЦ ПРОГРАММНОГО БЛОКА	1072
BIN	023	BCD В ДВОИЧНОЕ	425
BINL	058	ДВОЙНОЕ BCD В ДВОЙНОЕ ДВОИЧНОЕ	427
BINS	470	BCD СО ЗНАКОМ В ДВОИЧНОЕ	459
BISL	472	ДВОЙНОЕ BCD СО ЗНАКОМ В ДВОИЧНОЕ	462
BPPS	811	ПРИОСТАНОВКА ПРОГРАММНОГО БЛОКА	1072
BPRG	096	НАЧАЛО ПРОГРАММНОГО БЛОКА	1072
BPRS	812	ПЕРЕЗАПУСК ПРОГРАММНОГО БЛОКА	1075
BREAK	514	ВЫХОД ИЗ ЦИКЛА	182
BSET	071	ЗАПОЛНЕНИЕ БЛОКА	287

C

Мнемоническое обозначение	Код функции	Команда	Стр.
CADD	730	ДОБАВИТЬ К КАЛЕНДАРНОЙ ДАТЕ	1006
CCL	283	ЗАГРУЗИТЬ ФЛАГИ УСЛОВИЙ	1057
CCS	282	СОХРАНИТЬ ФЛАГИ УСЛОВИЙ	1055
CJP	510	УСЛОВНЫЙ ПЕРЕХОД	172

Мнемоническое обозначение	Код функции	Команда	Стр.
CJPN	511	УСЛОВНЫЙ ПЕРЕХОД НЕ	172
CLC	041	СБРОСИТЬ ПЕРЕНОС	1052
CLI	691	ОЧИСТИТЬ ПРЕРЫВАНИЕ	763
CLR$	666	ОЧИСТИТЬ СТРОКУ	1127
CMND	490	ДОСТАВИТЬ КОМАНДУ	958
CMP	020	СРАВНИТЬ	241
CMPL	060	СРАВНИТЬ ДВОЙНЫЕ СЛОВА	244
CNR	545	СБРОС ТАЙМЕРА/СЧЕТЧИКА	220
CNRX	547	СБРОС ТАЙМЕРА/СЧЕТЧИКА	220
CNT	---	СЧЕТЧИК	212
CNTX	546	СЧЕТЧИК	212
CNTR	012	РЕВЕРСИВНЫЙ СЧЕТЧИК	215
CNTRX	548	РЕВЕРСИВНЫЙ СЧЕТЧИК	215
CNTW	814	СЧЕТЧИК ОЖИДАНИЯ	1092
CNTWX	818	СЧЕТЧИК ОЖИДАНИЯ	1092
COLL	081	СБОР ДАННЫХ	294
COLLC	567	СБОР ДАННЫХ	1154
COLM	064	СТРОКУ В СТОЛБЕЦ	456
COM	029	ДОПОЛНЕНИЕ	490
COML	614	ДВОЙНОЕ ДОПОЛНЕНИЕ	492
COS	461	КОСИНУС	542
COSD	852	КОСИНУС ДВОЙНОЙ ТОЧНОСТИ	598
CPS	114	СРАВНИТЬ ДВОИЧНЫЕ ЗНАЧЕНИЯ СО ЗНАКОМ	247
CPSL	115	СРАВНИТЬ ДВОЙНЫЕ ДВОИЧНЫЕ СЛОВА СО ЗНАКОМ	250
CSUB	731	ВЫЧЕСТЬ ИЗ КАЛЕНДАРНОЙ ДАТЫ	1009
CTBL	882	ЗАГРУЗКА ТАБЛИЦЫ СРАВНЕНИЯ	785

D

Мнемоническое обозначение	Код функции	Команда	Стр.
DATE	735	КОРРЕКТИРОВКА ЧАСОВ	1017
DBL	843	16-БИТ ДВОИЧНОЕ В ПЛАВ. ЗАПЯТАЯ ДВОЙНОЙ ТОЧНОСТИ	582
DBLL	844	32-БИТ ДВОИЧНОЕ В ПЛАВ. ЗАПЯТАЯ ДВОЙНОЙ ТОЧНОСТИ	584
DEG	459	РАДИАНЫ В ГРАДУСЫ	538
DEGD	850	РАДИАНЫ В ГРАДУСЫ ДВОЙНОЙ ТОЧНОСТИ	595
DEL$	658	УДАЛИТЬ СТРОКУ	1123
DI	693	ЗАПРЕТИТЬ ПРЕРЫВАНИЯ	766
DIFD	014	ОТРИЦАТЕЛЬНЫЙ ФРОНТ	133
DIFU	013	ПОЛОЖИТЕЛЬНЫЙ ФРОНТ	133
DIM	631	СОЗДАТЬ ТАБЛИЦУ УКАЗАННЫХ РАЗМЕРОВ	632
DIST	080	РАЗМЕЩЕНИЕ ОДИНОЧНОГО СЛОВА	292
DISTC	566	РАЗМЕЩЕНИЕ ОДИНОЧНОГО СЛОВА	1150
DLNK	226	ОБНОВИТЬ ВХОДЫ/ВЫХОДЫ МОДУЛЯ ШИНЫ ЦПУ	877
DMPX	077	КОДИРОВАНИЕ ДАННЫХ	442
DOWN	522	УСЛОВИЕ ВЫКЛ	121
DSW	210	ЧТЕНИЕ ЦИФРОВОГО ПЕРЕКЛЮЧАТЕЛЯ	849

E

Мнемоническое обозначение	Код функции	Команда	Стр.
ECHRD	723	ЯВНОЕ ЧТЕНИЕ СЛОВА	989
ECHWR	724	ЯВНАЯ ЗАПИСЬ СЛОВА	993
EGATR	721	ЯВНЫЙ ЗАПРОС АТРИБУТА	975
EI	694	РАЗРЕШИТЬ ПРЕРЫВАНИЯ	768
ELSE	803	ИНАЧЕ	1077

Мнемоническое обозначение	Код функции	Команда	Стр.
END	001	КОНЕЦ	146
ESATR	722	ЯВНАЯ УСТАНОВКА АТРИБУТА	982
EXIT NOT (операнд)	806	ВЫХОД ИЗ БЛОКА ПО УСЛОВИЮ НЕ	1081
EXIT (входное условие)	806	ВЫХОД ИЗ БЛОКА ПО УСЛОВИЮ	1081
EXIT (операнд)	806	ВЫХОД ИЗ БЛОКА ПО УСЛОВИЮ	1081
EXP	467	ЭКСПОНЕНТА	553
EXPD	858	ЭКСПОНЕНТА ДВОЙНОЙ ТОЧНОСТИ	610
EXPLT	720	ОТПРАВИТЬ ЯВНОЕ СООБЩЕНИЕ	967

F

Мнемоническое обозначение	Код функции	Команда	Стр.
FAL	006	СИГНАЛИЗАЦИЯ НЕИСПРАВНОСТИ	1025
FALS	007	СИГНАЛИЗАЦИЯ СЕРЬЕЗНОЙ НЕИСПРАВНОСТИ	1033
FCS	180	КОНТРОЛЬНАЯ СУММА КАДРА	653
FDIV	079	ДЕЛЕНИЕ С ПЛАВАЮЩЕЙ ЗАПЯТОЙ	510
FIFO	633	ПЕРВЫМ ПРИШЕЛ — ПЕРВЫМ ВЫШЕЛ	626
FIND$	660	НАЙТИ В СТРОКЕ	1116
FIX	450	ПЛАВ. ЗАПЯТАЯ В 16-БИТ	579
FIXD	841	ПЛАВ. ЗАПЯТАЯ ДВОЙНОЙ ТОЧНОСТИ В 16-БИТ ДВОИЧНОЕ	579
FIXL	451	ПЛАВ. ЗАПЯТАЯ В 32-БИТ	581
FIXLD	842	ПЛАВ. ЗАПЯТАЯ ДВОЙНОЙ ТОЧНОСТИ В 32-БИТ ДВОИЧНОЕ	581
FLT	452	16-БИТ В ПЛАВ. ЗАПЯТАЯ	582
FLTL	453	32-БИТ В ПЛАВ. ЗАПЯТАЯ	584
FOR	512	ЦИКЛЫ FOR-NEXT	178
FPD	269	ОБНАРУЖЕНИЕ НЕИСПРАВНОГО КАНАЛА	1041
FRMCV	284	ПРЕОБРАЗОВАТЬ АДРЕС СЕРИИ CV	1059
FSTR	448	ПЛАВАЮЩАЯ ЗАПЯТАЯ В ASCII	563
FVAL	449	ASCII В ПЛАВАЮЩАЯ ЗАПЯТАЯ	568

G

Мнемоническое обозначение	Код функции	Команда	Стр.
GETID	286	СЧИТАТЬ ИДЕНТИФИКАТОР ПЕРЕМЕННОЙ	1162
GETR	636	СЧИТАТЬ НОМЕР СТРОКИ	636
GRET	752	ВЫХОД ИЗ ГЛОБАЛЬНОЙ ПОДПРОГРАММЫ	754
GRY	474	ПРЕОБРАЗОВАНИЕ КОДА ГРЕЯ	472
GSBN	751	ВХОД В ГЛОБАЛЬНУЮ ПОДПРОГРАММУ	751
GSBS	750	ВЫЗОВ ГЛОБАЛЬНОЙ ПОДПРОГРАММЫ	743

H

Мнемоническое обозначение	Код функции	Команда	Стр.
HEX	162	ASCII В HEX	450
HKY	212	ЧТЕНИЕ ШЕСТНАДЦАТЕРИЧНОЙ КЛАВИАТУРЫ	857
HMS	066	СЕКУНДЫ В ЧАСЫ	1015

I

Мнемоническое обозначение	Код функции	Команда	Стр.
IEND	804	КОНЕЦ ЕСЛИ	1077
IF NOT (операнд)	802	ЕСЛИ НЕ	1077
IF (входное условие)	802	IF	1077
IF (операнд)	802	IF	1077

Мнемоническое обозначение	Код функции	Команда	Стр.
IL	002	БЛОКИРОВКА	150
ILC	003	ОТМЕНА БЛОКИРОВКИ	150
INI	880	УПРАВЛЕНИЕ РЕЖИМОМ	770
INS$	657	INS$	1129
IORD	222	ЧТЕНИЕ ИНТЕЛЛЕКТУАЛЬНЫХ ВХОДОВ/ВЫХОДОВ	870
IORF	097	ОБНОВЛЕНИЕ СЛОВ ВВОДА/ВЫВОДА	843
IOWR	223	ЗАПИСЬ ИНТЕЛЛЕКТУАЛЬНЫХ ВХОДОВ/ВЫХОДОВ	873

J

Мнемоническое обозначение	Код функции	Команда	Стр.
JME	005	КОНЕЦ ПЕРЕХОДА	168
JME0	516	КОНЕЦ МНОЖЕСТВЕННОГО ПЕРЕХОДА	176
JMP	004	ПЕРЕХОД	168
JMP0	515	МНОЖЕСТВЕННЫЙ ПЕРЕХОД	176

K

Мнемоническое обозначение	Код функции	Команда	Стр.
ЗАЩЕЛКА	011	ЗАЩЕЛКА	128

L

Мнемоническое обозначение	Код функции	Команда	Стр.
LD	---	ЗАГРУЗКА	99
LD<	310	ЗАГРУЗКА МЕНЬШЕ	230
LD <$	672	ЗАГРУЗКА СТРОКА МЕНЬШЕ	1132
LD<D	337	ЗАГРУЗКА С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ МЕНЬШЕ	616
LD<DT	343	ЗАГРУЗКА ВРЕМЯ МЕНЬШЕ	236
LD<F	331	ЗАГРУЗКА С ПЛАВАЮЩЕЙ ЗАПЯТОЙ МЕНЬШЕ	559
LD<>	305	ЗАГРУЗКА НЕ РАВНО	230
LD <>$	671	ЗАГРУЗКА СТРОКА НЕ РАВНА	1132
LD<>D	336	ЗАГРУЗКА С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ НЕ РАВНО	616
LD<>DT	342	ЗАГРУЗКА ВРЕМЯ НЕ РАВНО	236
LD<>F	330	ЗАГРУЗКА С ПЛАВАЮЩЕЙ ЗАПЯТОЙ НЕ РАВНО	559
LD<>L	306	ЗАГРУЗКА ДВОЙНЫЕ СЛОВА НЕ РАВНЫ	230
LD<>S	307	ЗАГРУЗКА СЛОВА СО ЗНАКОМ НЕ РАВНЫ	230
LD<>SL	308	ЗАГРУЗКА ДВОЙНЫЕ СЛОВА СО ЗНАКОМ НЕ РАВНЫ	230
LD<L	311	ЗАГРУЗКА ДВОЙНОЕ СЛОВО МЕНЬШЕ	230
LD<S	312	ЗАГРУЗКА СЛОВО СО ЗНАКОМ МЕНЬШЕ	230
LD<SL	313	ЗАГРУЗКА ДВОЙНОЕ СЛОВО СО ЗНАКОМ МЕНЬШЕ	230
LD =	300	ЗАГРУЗКА РАВНО	230
LD =$	670	ЗАГРУЗКА СТРОКА РАВНА	1132
LD =D	335	ЗАГРУЗКА С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ РАВНО	616
LD= DT	341	ЗАГРУЗКА ВРЕМЯ РАВНО	236
LD =F	329	ЗАГРУЗКА С ПЛАВАЮЩЕЙ ЗАПЯТОЙ РАВНО	559
LD =L	301	ЗАГРУЗКА ДВОЙНЫЕ СЛОВА РАВНЫ	230
LD =S	302	ЗАГРУЗКА СЛОВА СО ЗНАКОМ РАВНЫ	230
LD =SL	303	ЗАГРУЗКА ДВОЙНЫЕ СЛОВА СО ЗНАКОМ РАВНЫ	230
LD>	320	ЗАГРУЗКА БОЛЬШЕ	230
LD >$	674	ЗАГРУЗКА СТРОКА БОЛЬШЕ	1132
LD>D	339	ЗАГРУЗКА С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ БОЛЬШЕ	616
LD>DT	345	ЗАГРУЗКА ВРЕМЯ БОЛЬШЕ	236
LD>F	333	ЗАГРУЗКА С ПЛАВАЮЩЕЙ ЗАПЯТОЙ БОЛЬШЕ	559
LD>L	321	ЗАГРУЗКА ДВОЙНОЕ СЛОВО БОЛЬШЕ	230
LD>S	322	ЗАГРУЗКА СЛОВО СО ЗНАКОМ БОЛЬШЕ	230

Мнемоническое обозначение	Код функции	Команда	Стр.
LD>SL	323	ЗАГРУЗКА ДВОЙНОЕ СЛОВО СО ЗНАКОМ БОЛЬШЕ	230
LD NOT	---	ЗАГРУЗКА НЕ	101
LD TST	350	ЗАГРУЗКА ПРОВЕРКА БИТА	123
LD TSTN	351	ЗАГРУЗКА ПРОВЕРКА БИТА	123
LD<=	315	ЗАГРУЗКА МЕНЬШЕ ИЛИ РАВНО	230
LD <=$	673	ЗАГРУЗКА СТРОКА МЕНЬШЕ ИЛИ РАВНА	1132
LD<=D	338	ЗАГРУЗКА С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ МЕНЬШЕ ИЛИ РАВНО	616
LD<=DT	344	ЗАГРУЗКА ВРЕМЯ МЕНЬШЕ ИЛИ РАВНО	236
LD<=F	332	ЗАГРУЗКА С ПЛАВАЮЩЕЙ ЗАПЯТОЙ МЕНЬШЕ ИЛИ РАВНО	559
LD<=L	316	ЗАГРУЗКА ДВОЙНОЕ СЛОВО МЕНЬШЕ ИЛИ РАВНО	230
LD<=S	317	ЗАГРУЗКА СЛОВО СО ЗНАКОМ МЕНЬШЕ ИЛИ РАВНО	230
LD<=SL	318	ЗАГРУЗКА ДВОЙНОЕ СЛОВО СО ЗНАКОМ МЕНЬШЕ ИЛИ РАВНО	230
LD>=	325	ЗАГРУЗКА БОЛЬШЕ ИЛИ РАВНО	230
LD >=$	675	ЗАГРУЗКА СТРОКА БОЛЬШЕ ИЛИ РАВНА	1132
LD>=D	340	ЗАГРУЗКА С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ БОЛЬШЕ ИЛИ РАВНО	616
LD>=DT	346	ЗАГРУЗКА ВРЕМЯ БОЛЬШЕ ИЛИ РАВНО	236
LD>=F	334	ЗАГРУЗКА С ПЛАВАЮЩЕЙ ЗАПЯТОЙ БОЛЬШЕ ИЛИ РАВНО	559
LD>=L	326	ЗАГРУЗКА ДВОЙНОЕ СЛОВО БОЛЬШЕ ИЛИ РАВНО	230
LD>=S	327	ЗАГРУЗКА СЛОВО СО ЗНАКОМ БОЛЬШЕ ИЛИ РАВНО	230
LD>=SL	328	ЗАГРУЗКА ДВОЙНОЕ СЛОВО СО ЗНАКОМ БОЛЬШЕ ИЛИ РАВНО	230
LEFT$	652	СКОПИРОВАТЬ СИМВОЛЫ СЛЕВА	1109
LEN$	650	ДЛИНА СТРОКИ	1118
LEND NOT (операнд)	810	КОНЕЦ ЦИКЛА НЕ	1099
LEND (входное условие)	810	КОНЕЦ ЦИКЛА	1099
LEND (операнд)	810	КОНЕЦ ЦИКЛА	1099
LIFO	634	ПОСЛЕДНИМ ПРИШЕЛ — ПЕРВЫМ ВЫШЕЛ	629
LINE	063	СТОЛБЕЦ В СТРОКУ	454
LMT	680	ОГРАНИЧЕНИЕ	696
LOG	468	ЛОГАРИФМ	555
LOGD	859	ЛОГАРИФМ ДВОЙНОЙ ТОЧНОСТИ	612
LOOP	809	ЦИКЛ	1099

M

Мнемоническое обозначение	Код функции	Команда	Стр.
MAX	182	НАЙТИ МАКСИМУМ	643
MCMP	019	СРАВНИТЬ НЕСКОЛЬКО ЗНАЧЕНИЙ	253
MCRO	099	МАКРОС	736
MID$	654	СКОПИРОВАТЬ СИМВОЛЫ ВНУТРИ СТРОКИ	1113
MILC	519	ОТМЕНА ВЛОЖЕННОЙ БЛОКИРОВКИ	155
MILH	517	ВЛОЖЕННАЯ БЛОКИРОВКА С ЗАПОМИНАНИЕМ ФРОНТА	155
MILR	518	ВЛОЖЕННАЯ БЛОКИРОВКА БЕЗ ЗАПОМИНАНИЯ ФРОНТА	155
MIN	183	НАЙТИ МИНИМУМ	646
MLPX	076	ДЕКОДИРОВАНИЕ ДАННЫХ	437
MOV	021	ПЕРЕДАТЬ	271
MOV$	664	КОПИРОВАТЬ СТРОКУ	1104
MOVB	082	ПЕРЕДАТЬ БИТ	277
MOVBC	568	ПЕРЕДАТЬ БИТ	1158
MOVD	083	ПЕРЕДАТЬ РАЗРЯД	279
MOVL	498	ПЕРЕДАТЬ ДВОЙНОЕ СЛОВО	274
MOVR	560	ПЕРЕДАТЬ В РЕГИСТР	296

Мнемоническое обозначение	Код функции	Команда	Стр.
MOVRW	561	ПЕРЕДАТЬ ТЕКУЩЕЕ ЗНАЧЕНИЕ ТАЙМЕРА/СЧЕТЧИКА В РЕГИСТР	298
MSG	046	ОТОБРАЗИТЬ СООБЩЕНИЕ	997
MSKR	692	СЧИТАТЬ МАСКУ ПРЕРЫВАНИЙ	759
MSKS	690	УСТАНОВИТЬ МАСКУ ПРЕРЫВАНИЙ	755
MTIM	543	МНОГОВЫХОДНОЙ ТАЙМЕР	206
MTIMX	554	МНОГОВЫХОДНОЙ ТАЙМЕР	206
MTR	213	ЧТЕНИЕ МАТРИЦЫ	861
MVN	022	ПЕРЕДАТЬ ДОПОЛНЕНИЕ	273
MVNL	499	ПЕРЕДАТЬ ДОПОЛНЕНИЕ К ДВОЙНОМУ СЛОВУ	276

N

Мнемоническое обозначение	Код функции	Команда	Стр.
NASL	580	СДВИГ НА N БИТОВ ВЛЕВО	338
NASR	581	СДВИГ НА N БИТОВ ВПРАВО	344
NEG	160	ДОПОЛНЕНИЕ ДО 2	432
NEGL	161	ДОПОЛНЕНИЕ ДВОЙНОГО СЛОВА ДО 2	434
NEXT	513	ЦИКЛЫ FOR-NEXT	178
NOP	000	ПУСТАЯ ОПЕРАЦИЯ	147
NOT	520	НЕТ	120
NSFL	578	СДВИГ N БИТОВ ДАННЫХ ВЛЕВО	334
NSFR	579	СДВИГ N БИТОВ ДАННЫХ ВПРАВО	336
NSLL	582	СДВИГ ДВОЙНОГО СЛОВА НА N БИТОВ ВЛЕВО	341
NSRL	583	СДВИГ ДВОЙНОГО СЛОВА НА N БИТОВ ВПРАВО	346

O

Мнемоническое обозначение	Код функции	Команда	Стр.
OR	---	ИЛИ	107
OR<	310	ИЛИ МЕНЬШЕ	230
OR <$	672	ИЛИ СТРОКА МЕНЬШЕ	1132
OR<>	305	ИЛИ НЕ РАВНО	230
OR <>$	671	ИЛИ СТРОКА НЕ РАВНА	1132
OR<>D	336	ИЛИ С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ НЕ РАВНО	616
OR<>DT	342	ИЛИ ВРЕМЯ НЕ РАВНО	236
OR<>F	330	ИЛИ С ПЛАВАЮЩЕЙ ЗАПЯТОЙ НЕ РАВНО	559
OR<>L	306	ИЛИ ДВОЙНЫЕ СЛОВА НЕ РАВНЫ	230
OR<>S	307	ИЛИ СЛОВА СО ЗНАКОМ НЕ РАВНЫ	230
OR<>SL	308	ИЛИ ДВОЙНЫЕ СЛОВА СО ЗНАКОМ НЕ РАВНЫ	230
OR<D	337	ИЛИ С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ МЕНЬШЕ	616
OR<DT	343	ИЛИ ВРЕМЯ МЕНЬШЕ	236
OR<F	331	ИЛИ С ПЛАВАЮЩЕЙ ЗАПЯТОЙ МЕНЬШЕ	559
OR<L	311	ИЛИ ДВОЙНОЕ СЛОВО МЕНЬШЕ	230
OR<S	312	ИЛИ СЛОВО СО ЗНАКОМ МЕНЬШЕ	230
OR<SL	313	ИЛИ ДВОЙНОЕ СЛОВО СО ЗНАКОМ МЕНЬШЕ	230
OR =	300	ИЛИ РАВНО	230
OR =$	670	ИЛИ СТРОКА РАВНА	1132
OR =D	335	ИЛИ С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ РАВНО	616
OR = DT	341	ИЛИ ВРЕМЯ РАВНО	236
OR =F	329	ИЛИ С ПЛАВАЮЩЕЙ ЗАПЯТОЙ РАВНО	559
OR =L	301	ИЛИ ДВОЙНЫЕ СЛОВА РАВНЫ	230
OR =S	302	ИЛИ СЛОВА СО ЗНАКОМ РАВНЫ	230
OR =SL	303	ИЛИ ДВОЙНЫЕ СЛОВА СО ЗНАКОМ РАВНЫ	230
OR>	320	ИЛИ БОЛЬШЕ	230
OR >$	674	ИЛИ СТРОКА БОЛЬШЕ	1132

Мнемоническое обозначение	Код функции	Команда	Стр.
OR>D	339	ИЛИ С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ БОЛЬШЕ	616
OR>DT	345	ИЛИ ВРЕМЯ БОЛЬШЕ	236
OR>F	333	ИЛИ С ПЛАВАЮЩЕЙ ЗАПЯТОЙ БОЛЬШЕ	559
OR>L	321	ИЛИ ДВОЙНОЕ СЛОВО БОЛЬШЕ	230
OR>S	322	ИЛИ СЛОВО СО ЗНАКОМ БОЛЬШЕ	230
OR>SL	323	ИЛИ ДВОЙНОЕ СЛОВО СО ЗНАКОМ БОЛЬШЕ	230
OR LD	---	ИЛИ ЗАГРУЗКА	113
OR NOT	---	ИЛИ НЕ	109
OR TST	350	ИЛИ ПРОВЕРКА БИТА	123
OR TSTN	351	ИЛИ ПРОВЕРКА БИТА НЕ	123
OR<=	315	ИЛИ МЕНЬШЕ ИЛИ РАВНО	230
OR <=$	673	ИЛИ СТРОКА МЕНЬШЕ ИЛИ РАВНА	1132
OR<=D	338	ИЛИ С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ МЕНЬШЕ ИЛИ РАВНО	616
OR<=DT	344	ИЛИ ВРЕМЯ МЕНЬШЕ ИЛИ РАВНО	236
OR<=F	332	ИЛИ С ПЛАВАЮЩЕЙ ЗАПЯТОЙ МЕНЬШЕ ИЛИ РАВНО	559
OR<=L	316	ИЛИ ДВОЙНОЕ СЛОВО МЕНЬШЕ ИЛИ РАВНО	230
OR<=S	317	ИЛИ СЛОВО СО ЗНАКОМ МЕНЬШЕ ИЛИ РАВНО	230
OR<=SL	318	ИЛИ СЛОВО СО ЗНАКОМ МЕНЬШЕ ИЛИ РАВНО	230
OR>=	325	ИЛИ БОЛЬШЕ ИЛИ РАВНО	230
OR >=$	675	ИЛИ СТРОКА БОЛЬШЕ ИЛИ РАВНА	1132
OR>=D	340	ИЛИ С ПЛАВ. ЗАПЯТОЙ ДВОЙНОЙ ТОЧНОСТИ БОЛЬШЕ ИЛИ РАВНО	616
OR>=DT	346	ИЛИ ВРЕМЯ БОЛЬШЕ ИЛИ РАВНО	236
OR>=F	334	ИЛИ С ПЛАВАЮЩЕЙ ЗАПЯТОЙ БОЛЬШЕ ИЛИ РАВНО	559
OR>=L	326	ИЛИ ДВОЙНОЕ СЛОВО БОЛЬШЕ ИЛИ РАВНО	230
OR>=S	327	ИЛИ СЛОВО СО ЗНАКОМ БОЛЬШЕ ИЛИ РАВНО	230
OR>=SL	328	ИЛИ ДВОЙНОЕ СЛОВО СО ЗНАКОМ БОЛЬШЕ ИЛИ РАВНО	230
ORG	889	ПОИСК ИСХОДНОГО ПОЛОЖЕНИЯ	817
ORW	035	ЛОГИЧЕСКОЕ ИЛИ	480
ORWL	611	ДВОЙНОЕ ЛОГИЧЕСКОЕ ИЛИ	482
OUT	---	ВЫВОД	126
OUTB	534	ВЫВОД В ОДИН БИТ	144
OUT NOT	---	ВЫВОД НЕ	127

P

Мнемоническое обозначение	Код функции	Команда	Стр.
PID	190	ПИД-РЕГУЛЯТОР	671
PIDAT	191	ПИД-РЕГУЛЯТОР С АВТОНАСТРОЙКОЙ	684
PMCR	260	МАКРОС ПРОТОКОЛА	884
PRV	881	ЧТЕНИЕ ТЕКУЩ. ЗНАЧ. СКОРОСТНОГО СЧЕТЧИКА	775
PRV2	883	ПРЕОБРАЗОВАНИЕ ЧАСТОТЫ НА ВХОДЕ СЧЕТЧИКА	781
PULS	886	ЗАДАТЬ КОЛ-ВО ИМПУЛЬСОВ	796
PLS2	887	ВЫДАЧА ИМПУЛЬСОВ	799
PUSH	632	ЗАПИСАТЬ В СТЕК	623
PWM	891	ИМПУЛЬСЫ С ПЕРЕМЕННОЙ СКВАЖНОСТЬЮ	822
PWR	840	ВОЗВЕДЕНИЕ В СТЕПЕНЬ	557
PWRD	860	ВОЗВЕДЕНИЕ В СТЕПЕНЬ ДВОЙНОЙ ТОЧНОСТИ	581

R

Мнемоническое обозначение	Код функции	Команда	Стр.
RAD	458	ГРАДУСЫ В РАДИАНЫ	536
RADD	849	ГРАДУСЫ В РАДИАНЫ ДВОЙНОЙ ТОЧНОСТИ	593
RECV	098	ПРИНЯТЬ ПО СЕТИ	952
RET	093	ВЫХОД ИЗ ПОДПРОГРАММЫ	743

Мнемоническое обозначение	Код функции	Команда	Стр.
RGHT$	653	СКОПИРОВАТЬ СИМВОЛЫ СПРАВА	1111
RLNC	574	ЦИКЛИЧЕСКИЙ СДВИГ ВЛЕВО БЕЗ ПЕРЕНОСА	324
RLNL	576	ЦИКЛИЧЕСКИЙ СДВИГ ДВОЙНОГО СЛОВА ВЛЕВО БЕЗ ПЕРЕНОСА	326
ROL	027	ЦИКЛИЧЕСКИЙ СДВИГ ВЛЕВО	317
ROLL	572	ЦИКЛИЧЕСКИЙ СДВИГ ДВОЙНОГО СЛОВА ВЛЕВО	319
ROOT	072	КВАДРАТНЫЙ КОРЕНЬ ИЗ ДВОИЧНО-ДЕСЯТИЧНОГО ЧИСЛА	495
ROR	028	ЦИКЛИЧЕСКИЙ СДВИГ ВПРАВО	321
RORL	573	ЦИКЛИЧЕСКИЙ СДВИГ ДВОЙНОГО СЛОВА ВПРАВО	322
ROTB	620	КВАДРАТНЫЙ КОРЕНЬ ИЗ ДВОИЧНОГО ЧИСЛА	493
RPLC$	661	ЗАМЕНИТЬ В СТРОКЕ	1120
RRNC	575	ЦИКЛИЧЕСКИЙ СДВИГ ВПРАВО БЕЗ ПЕРЕНОСА	327
RRNL	577	ЦИКЛИЧЕСКИЙ СДВИГ ДВОЙНОГО СЛОВА ВПРАВО БЕЗ ПЕРЕНОСА	329
RSET	---	СБРОС	135
RSTA	531	СБРОСИТЬ НЕСКОЛЬКО БИТОВ	138
RSTB	533	СБРОСИТЬ ОДИН БИТ	141
RXD	235	ПРИНЯТЬ	898
RXDU	255	ПРИНЯТЬ ЧЕРЕЗ МОДУЛЬ ПОСЛЕД. ИНТЕРФЕЙСА	904

S

Мнемоническое обозначение	Код функции	Команда	Стр.
SBN	092	ВХОД В ПОДПРОГРАММУ	740
SBS	091	ВЫЗОВ ПОДПРОГРАММЫ	730
SCH	047	УПРАВЛЕНИЕ 7-СЕГМЕНТНЫМ ИНДИКАТОРОМ	1000
SCL	194	ИЗМЕНЕНИЕ МАСШТАБА	713
SCL2	486	ИЗМЕНЕНИЕ МАСШТАБА 2	718
SCL3	487	ИЗМЕНЕНИЕ МАСШТАБА 3	722
SCTRL	048	ОТОБРАЗИТЬ ЗНАЧЕНИЕ СЛОВА НА 7-СЕГМЕНТНОМ ИНДИКАТОРЕ	1002
SDEC	078	ПРЕОБРАЗОВАТЬ В 7-СЕГМЕНТНЫЙ КОД	846
SDEL	642	УДАЛИТЬ ДАННЫЕ ИЗ СТЕКА	667
SEC	065	ЧАСЫ В СЕКУНДЫ	1013
SEND	090	ПЕРЕДАТЬ ПО СЕТИ	945
SET	---	УСТАНОВКА	135
SETA	530	УСТАНОВИТЬ НЕСКОЛЬКО БИТОВ	138
SETB	532	УСТАНОВИТЬ ОДИН БИТ	141
SETR	635	ЗАПИСАТЬ РАСПОЛОЖЕНИЕ СТРОКИ	634
SFT	010	РЕГИСТР СДВИГА	302
SFTR	084	РЕВЕРСИВНЫЙ РЕГИСТР СДВИГА	304
SIGN	600	16-БИТОВОЕ ДВОИЧНОЕ СО ЗНАКОМ В 32-БИТОВОЕ	435
SIN	460	СИНУС	540
SIND	851	СИНУС ДВОЙНОЙ ТОЧНОСТИ	597
SINS	641	ВСТАВИТЬ ДАННЫЕ В СТЕК	664
SLD	074	СДВИГ НА ОДИН РАЗРЯД ВЛЕВО	331
SNUM	638	СЧИТАТЬ РАЗМЕР СТЕКА	655
SNXT	009	ЗАПУСТИТЬ ШАГ	825
SPED	885	ВЫДАЧА ИМПУЛЬСОВ СКОРОСТИ	790
SQRT	466	КВАДРАТНЫЙ КОРЕНЬ	551
SQRTD	857	КВАДРАТНЫЙ КОРЕНЬ ДВОЙНОЙ ТОЧНОСТИ	608
SRCH	181	ПОИСК ЗНАЧЕНИЯ	638
SRD	075	СДВИГ НА ОДИН РАЗРЯД ВПРАВО	332
SREAD	639	ПРОЧИТАТЬ ДАННЫЕ ИЗ СТЕКА	658
SSET	630	СОЗДАТЬ СТЕК	621
STC	040	УСТАНОВИТЬ ПЕРЕНОС	1051
STEP	008	ОПРЕДЕЛИТЬ ШАГ	825
STUP	237	ИЗМЕНИТЬ ПАРАМЕТРЫ ПОСЛЕД. ПОРТА	921

Мнемоническое обозначение	Код функции	Команда	Стр.
SUM	184	СУММА	649
SWAP	637	ПОМЕНЯТЬ МЕСТАМИ БАЙТЫ	641
SWRIT	640	ЗАПИСАТЬ ДАННЫЕ В СТЕК	661

T

Мнемоническое обозначение	Код функции	Команда	Стр.
TAN	462	ТАНГЕНС	543
TAND	853	ТАНГЕНС ДВОЙНОЙ ТОЧНОСТИ	600
TCMP	085	СРАВНИТЬ С ТАБЛИЦЕЙ	256
TIM	---	ТАЙМЕР	186
TIMH	015	СКОРОСТНОЙ ТАЙМЕР	191
TIMHX	551	СКОРОСТНОЙ ТАЙМЕР	191
TIML	542	ДОЛГОСРОЧНЫЙ ТАЙМЕР	203
TIMLX	553	ДОЛГОСРОЧНЫЙ ТАЙМЕР	203
TIMW	813	ТАЙМЕР ОЖИДАНИЯ	1088
TIMWX	816	ТАЙМЕР ОЖИДАНИЯ	1088
TIMX	---	ТАЙМЕР	186
TKOF	821	ВЫКЛЮЧИТЬ ЗАДАЧУ	1141
TKON	820	ВКЛЮЧИТЬ ЗАДАЧУ	1137
TKY	211	ЧТЕНИЕ ДЕСЯТИЧНОЙ КЛАВИАТУРЫ	853
TMHH	540	1 мс ТАЙМЕР	195
TMHHX	552	1 мс ТАЙМЕР	195
TMHW	815	СКОРОСТНОЙ ТАЙМЕР ОЖИДАНИЯ	1096
TMHWX	817	СКОРОСТНОЙ ТАЙМЕР ОЖИДАНИЯ	1096
TOCV	285	ПРЕОБРАЗОВАТЬ В АДРЕС СЕРИИ CV	1162
TPO	685	ВЫХОД ШИМ	704
TRSM	045	ОТБОР ДАННЫХ ДЛЯ ПАМЯТИ ПРОТОКОЛА ДАННЫХ	1020
TTIM	087	НАКАПЛИВАЮЩИЙ ТАЙМЕР	199
TTIMX	555	НАКАПЛИВАЮЩИЙ ТАЙМЕР	199
TXD	236	ОТПРАВИТЬ	892
TXDU	256	ОТПРАВИТЬ ЧЕРЕЗ МОДУЛЬ ПОСЛЕД. ИНТЕРФЕЙСА	904

U

Мнемоническое обозначение	Код функции	Команда	Стр.
UP	521	УСЛОВИЕ ВКЛ	121

W

Мнемоническое обозначение	Код функции	Команда	Стр.
WAIT NOT (операнд)	805	ОДИН ЦИКЛ И ОЖИДАНИЕ НЕ	1084
WAIT (входное условие)	805	ОДИН ЦИКЛ И ОЖИДАНИЕ	1084
WAIT (операнд)	805	ОДИН ЦИКЛ И ОЖИДАНИЕ	1084
WDT	094	УВЕЛИЧИТЬ МАКС. ВРЕМЯ ЦИКЛА	1053
WSFT	016	СДВИГ ПО СЛОВАМ	309

X

Мнемоническое обозначение	Код функции	Команда	Стр.
XCGL	562	ОБМЕН ДВОЙНЫМИ СЛОВАМИ	290
XCHG	073	ОБМЕН ДАННЫМИ	289
XCHG$	665	ОБМЕН СТРОКАМИ	1125
XFER	070	ПЕРЕДАЧА БЛОКА	285
XFERC	565	ПЕРЕДАЧА БЛОКА	1150

Мнемоническое обозначение	Код функции	Команда	Стр.
XFRB	062	ПЕРЕДАТЬ НЕСКОЛЬКО БИТОВ	282
XNRL	613	ДВОЙНОЕ ИСКЛЮЧАЮЩЕЕ НЕ-ИЛИ	488
XNRW	037	ИСКЛЮЧАЮЩЕЕ НЕ-ИЛИ	487
XORL	612	ДВОЙНОЕ ИСКЛЮЧАЮЩЕЕ ИЛИ	485
XORW	036	ИСКЛЮЧАЮЩЕЕ ИЛИ	483

Z

Мнемоническое обозначение	Код функции	Команда	Стр.
ZCP	088	СРАВНИТЬ СЛОВО С ДИАПАЗОНОМ	265
ZCPL	116	СРАВНИТЬ ДВОЙНОЕ СЛОВО С ДИАПАЗОНОМ	268
ZONE	682	МЕРТВАЯ ЗОНА	701

Предварительная версия
Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

Указатель

7-сегментный индикатор
  Преобразование значений, 846

А

Адрес в памяти ПЛК
  См. также адрес памяти внутреннего ввода/вывода
Адрес памяти ввода/вывода
  См. также адрес памяти внутреннего ввода/вывода
Адрес памяти внутреннего ввода/вывода
  Указание адреса слова/бита в регистре указателя, 296
  Указание адреса текущего значения таймера/счетчика в регистре указателя, 298
Адресация
  Адреса памяти, 7
  Косвенные адреса, 9
  Операнды, 9
  См. также Регистры указателей
  Счетчики, 227
  Таймеры, 227

Б

Базовые модули ввода/вывода
  Команды для базовых модулей ввода/вывода, 843, 877
Батарея
  Меры предосторожности, xxx
Бит удержания состояний входов/выходов, xxx
Биты
  Установка и сброс, 141
Биты управления
  Бит запуска отбора данных, 1023
  Бит запуска протоколирования, 1023
Блокировки, 5, 21, 41, 150, 168

В

Внешние прерывания
  Задачи, 61, 81
Время
  Команды для преобразования времени, 1013, 1015
Время цикла
  Увеличение максимальной длительности цикла, 1053
Входные команды
  Время выполнения, 1167
Входные команды сравнения чисел с плавающей запятой двойной точности, 616
Входные команды сравнения чисел с плавающей запятой одинарной точности, 559
Выходные команды
  Время выполнения, 1168
Выходы
  Меры предосторожности, xxv, xxx
Выходы скорости, 790

Г

Градусы
  Преобразование из градусов в радианы, 536, 593

Д

Данные
  Поиск, 638
  Преобразование
    Радианы и градусы, 536, 538, 593, 595
Двоично-десятичное значение, 14
Двоичное значение без знака, 14
Двоичное значение со знаком, 14
  Перемещение знака, 435
Директива по низковольтному оборудованию, xxxi
Директивы по ЭМС, xxxi
Директивы ЕС, xxxi
Дополнительные циклические задачи, 1137, 1141

Ж

Журнал ошибок
  Запрет регистрации ошибок пользователя, 1029

З

Задачи, xvi, 2, 55
  Введение, 60
  Взаимосвязь с программными блоками, 75
  Время выполнения команд, 1190
  Выполнение, 67
  Задачи обработки прерываний, 61, 77
  Команды управления задачами, 1137, 1146
  Номера задач, 68
  Ограничения, 70
  Преимущества, 56
  Примеры, 73
  Программные блоки в задачах, 1068
  Работа флагов условий, 69
  Свойства, 56
  См. также Задачи обработки прерываний
  См. также Циклические задачи

Создание, 74
Создание задач, 85
Таймеры, 69
Условия выполнения, 63
Флаги, 71
Циклические задачи, 61
Задачи обработки прерываний, 61, 77, 85
Меры предосторожности, 83
Флаги и слова, 82
Запланированные прерывания
Задачи, 61, 81
Значение с плавающей запятой, 517, 574
Деление, 510
Квадратный корень, 551, 608
Команды математических операций с плавающей запятой, 516, 559, 573, 616
Логарифм, 555, 612
Математические функции, 573
Преобразование, 573
Преобразование в ASCII, 563, 568
Сравнение, 559
Тригонометрические функции, 573
формат, 573
Экспонента, 553, 610

И

Имитация системных ошибок, 1026, 1033
Импульсные выходы, 770
Управление, 770, 808
Источник питания, xxvii
Меры предосторожности, xxx

К

Квадратный корень
Двоично-десятичные данные, 495
Двоичное значение со знаком
См. также Математические операции
Значение с плавающей запятой, 551, 608
Коды функций
Команды в порядке возрастания кодов функций, 1203
Команда FALS, xxv, xxx
Команды, 87, 229
Базовые команды, 4
В порядке возрастания кодов функций, 1203
Варианты выполнения, 18
Входные битовые команды, 99, 125
Входные и выходные команды, 4, 7
Входные команды сравнения, 230, 236, 559, 616
Входные, с различением фронтов, 19
Выходные битовые команды, 126, 141
Команды дисплея, 997, 1000
Команды для базовых модулей ввода/вывода, 843, 877
Команды для диагностики неисправностей, 1025, 1050
Команды для работы с часами, 1006, 1067
Команды для создания программных блоков, 1067, 1102
Команды математических операций с плавающей запятой, 516, 559, 573, 616
Команды обработки табличных данных, 620, 655, 1179
Команды обработки текстовых строк, 1103, 1137
Команды отладки программы, 1020, 1024
Команды передачи данных, 271
Команды подпрограмм, 730, 754
Команды последовательного интерфейса, 882, 923
Команды пошагового выполнения, 824, 842
Команды преобразования, 425, 471
Команды с различением фронтов, 21
Команды сдвига данных, 301, 348
Команды сравнения, 230, 265
Команды сравнения строк, 1132, 1137
Команды счетчиков, 184, 229
Команды таймеров, 184, 229
Команды увеличения, 349, 366
Команды уменьшения, 349, 366
Команды управления данными, 671, 729
Команды управления задачами, 1137, 1146
Команды управления импульсными выходами, 770
Команды управления последовательностью выполнения, 146, 183
Команды управления прерываниями, 755
Команды управления скоростными счетчиками, 770
Логические команды, 477, 493
Местоположение в программе, 7
Обновление входов/выходов, 20
Ограничения на применение в задачах, 70
Операнды, 6
Программные блоки, 44
Сетевые команды, 924
Символьные математические команды, 366, 424
Специальные математические команды, 493, 1162
Управление задачами, 65
Управление скоростными счетчиками и импульсными выходами, 770
Управление условиями выполнения UP(521) и DOWN(522), 121
Условия выполнения, 18
Условия выполнения команд, 5
Циклы, 5, 41

Команды вычисления контрольной суммы, 620
Команды диагностики неисправностей
  Время выполнения, 1187
Команды дисплея
  Время выполнения, 1186
Команды для чисел с плавающей запятой двойной точности, 573
Команды математических операций с плавающей запятой
  Время выполнения, 1178
Команды отладки программы
  Время выполнения, 1187
Команды передачи данных
  Время выполнения, 1172
Команды подпрограмм
  Время выполнения, 1182
Команды поиска, 620
Команды последовательного интерфейса
  Время выполнения, 1185
Команды пошагового выполнения
  Время выполнения, 1183, 1184
Команды работы со стеком, 620
  Время выполнения, 1180
Команды с различением фронтов, 21
Команды сдвига данных
  Время выполнения, 1172
Команды сравнения
  Время выполнения, 1170, 1171
Команды увеличения
  Время выполнения, 1174
Команды уменьшения
  Время выполнения, 1174
Команды управления данными
  Время выполнения, 1181
Команды управления задачами
  Время выполнения, 1190
Команды управления последовательностью выполнения
  Время выполнения, 1169
Команды управления прерываниями
  Время выполнения, 1182
Команды управления скоростными счетчиками и импульсными выходами, 770
Команды управления часами
  Время выполнения, 1186
Команды FINS, 958
Команды, выполняемые по отрицательному фронту, 20
Команды, выполняемые по положительному фронту, 19
Контрольная сумма
  Вычисления, 653
Контрольная сумма кадра
  Вычисления, 653
Коэффициент заполнения
  Импульсы с переменным коэффициентом заполнения, 822
Критические ошибки в работе
  Формирование и сброс, 1033

Л

Лестничные диаграммы
  Управление состоянием бита
    С помощью DIFU(013) и DIFD(014), 133, 135
    С помощью KEEP(011), 128, 133
    С помощью SET и RSET, 135, 138
    С помощью SETA(530) и RSTA(531), 138, 141, 144
Логарифм, 555, 612
Логические команды
  Время выполнения, 1177

М

Макрос протокола, 884
Максимальное время цикла
  Увеличение, 1053
Математические операции
  Вычитание чисел с плавающей запятой, 530, 587
  Деление с плавающей запятой, 510, 534
  Квадратный корень, 493, 495, 551, 608
  Команды математических операций с плавающей запятой, 516, 559, 573, 616
  Линейная экстраполяция, 501
  Логарифм, 555, 612
  Поиск максимального значения в массиве, 643
  Поиск минимального значения в массиве, 646
  Символьные математические команды, 366, 424
  Сложение массива слов, 649
  Сложение чисел с плавающей запятой, 528, 585
  См. также Тригонометрические функции
  Специальные математические команды, 493, 1162
  Среднее значение, 726
  Тригонометрические функции, 499
  Умножение чисел с плавающей запятой, 532, 589
  Экспонента, 553, 610
Мгновенное обновление, 18, 23
Меры защиты
  Помехи, xxxii

Меры предосторожности, xxiii
Задачи обработки прерываний, 83
Обеспечение безопасности, xxiv
Общие, xxiv
Программирование, 36
Условия эксплуатации, xxvi
Эксплуатация, xxvii
Меры предосторожности и обеспечения безопасности, xxiv
Мнемоническое представление, 25
Ввод, 29
Модули шины ЦПУ
Обновление, 877
Модуль ЦПУ
Объем программы, 24
Основные принципы работы, 59
Монтаж
Место, xxvi

H

Набор команд
7SEG(214), 865
DSW(210), 849
HKY(212), 857
TKY(211), 853
Наборы команд
Входные команды сравнения чисел с плавающей запятой двойной точности (335...340), 616
Входные команды сравнения чисел с плавающей запятой одинарной точности (329...334), 559
Проверка состояния бита, 123
-(410), 382
--(592), 353
*(420), 403
*B(424), 410
*BL(425), 412
*D(847), 589
*F(456), 532, 589
*L(421), 405
*U(422), 407
*UL(423), 408
+$(656), 1106
+(400), 367
++(590), 349
++B(594), 358
++BL(595), 360
++L(591), 351
+B(404), 376
+BC(406), 379
+BCL(407), 381
+BL(405), 377
+C(402), 371
+CL(403), 374
+D(845), 585
+F(454), 528, 585
+L(401), 370
/(430), 413
/B(434), 421
/BL(435), 422
/D(848), 591
/F(457), 534
/L(431), 415
/U(432), 417
/UL(433), 419
ACC(888), 808
ACOS(464), 547, 604
ACOSD(855), 604
AND, 104
AND LD, 111
AND NOT, 105
ANDL(610), 478
ANDW(034), 477
APR(069), 499
ASC(086), 446
ASIN(463), 545, 602
ASIND(854), 602
ATAN(465), 549, 606
ATAND(856), 606
AVG(195), 726
-B(414), 394
--B(596), 362
BAND(681), 698
-BC(416), 399
BCD(024), 429
BCDL(059), 430
BCDS(471), 465
-BCL(417), 401
BCMP(068), 258
BCNT(067), 514
BDSL(473), 468
BIN(023), 425
BINL(058), 427
BINS(470), 459
BISL(472), 462
-BL(415), 396
--BL(597), 364
BPPS(811), 1075
BPRS(812), 1075
BREAK(514), 182
BSET(071), 287
-C(412), 389
CADD(730), 1006

Предварительная версия
Замечания и предложения просьба направлять по адресу: omron_russia@eu.omron.com

CCL(283), 1057
CCS(282), 1055
CJP(510), 172
CJPN(511), 172
-CL(413), 391
CLC(041), 1052
CLI(691), 763
CLR$(666), 1127
CMND(490), 924
CMP(020), 241
CMPL(060), 244
CNR(545), 220
CNT, 212
CNTR(012), 215
CNTRX(548), 215
CNTW(814), 1092
CNTWX(818), 1092
CNTX(546), 212
COLL(081), 294, 1154
COLM(064), 456
COM(029), 490
COML(614), 492
COS(461), 542, 598
COSD(852), 598
CPS(114), 247
CPSL(115), 250
CSUB(731), 1009
CTBL(882), 785
–D(846), 587
DBL(843), 582
DBLL(844), 584
DEG(459), 538, 595
DEGD(850), 595
DEL$(658), 1123
DI(693), 766
DIFD(014), 133, 135
  Применение в блокировках, 152
  Применение в переходах, 171, 175, 178
DIFU(013), 133, 135
  Применение в блокировках, 152
  Применение в переходах, 171, 175, 178
DIM(631), 632
DIST(080), 292
DLNK(226), 877
DMPX(077), 442
DOWN(522), 121
EI(694), 768
ELSE(803), 1077
END(001), 146
EXIT(806), 1081
EXP(467), 553, 610
EXPD(858), 610
-F(455), 530, 587
FAL(006), 1025
FALS(007), 1033
FCS(180), 653
FDIV(079), 510
FIFO(633), 626
FIND$(660), 1116
FIX(450), 522, 579
FIXD(841), 579
FIXL(451), 524, 563, 581
FIXLD(842), 581
FLT(452), 525, 582
FLTL(453), 527, 584
FOR(512), 178
FRMCV(284), 1059
FSTR(448), 563
FVAL(449), 568
GETR(636), 636
GRET(752), 754
GSBN(751), 751
GSBS(750), 743
HEX(162), 450
HMS(066), 1015
IEND(804), 1077
IF(802), 1077, 1084
IL(002), 150, 168
ILC(003), 150, 168
INI(880), 770
INS$(657), 1129
IORD(222), 870
IORF(097), 843
IOWR(223), 873
JME(005), 168
JME0(516), 176
JMP(004), 168
JMP0(515), 176
KEEP(011), 128
-L(411), 385
--L(593), 355
LD, 99
LD NOT, 101
LEFT$(652), 1109
LEN$(650), 1118
LEND(810), 1099
LIFO(634), 629
LINE(063), 454
LMT(680), 696
LOG(468), 555, 612
LOGD(859), 612
LOOP(809), 1099
MAX(182), 643
MCMP(019), 253, 268

MCRO(099), 736
MID$(654), 1113
MIN(183), 646
MLPX(076), 437
MOV$(664), 1000, 1002, 1104
MOV(021), 271
MOVB(082), 277
MOVD(083), 279
MOVL(498), 274
MOVR(560), 296
MOVRW(561), 298
MSG(046), 997
MSKR(692), 759
MSKS(690), 755
MTIM(543), 206
MTIMX(554), 206
MVN(022), 273
MVNL(499), 276
NEG(160), 432
NEGL(161), 434
NEXT(513), 178
NOP(000), 147
NOT(520), 120
OR, 107
OR LD, 113
OR NOT, 109
ORG(889), 817
ORW(035), 480
ORWL(611), 482
OUT, 126
OUT NOT, 127
OUTB(534), 144
PID(190), 671, 684, 1059, 1063
PIDAT(191), 684
PLS2(887), 799
PMCR(260), 884
PRV(881), 775, 781
PULS(886), 796
PUSH(632), 623
PWM(891), 822
PWRD(860), 614
RAD(458), 536, 593
RADD(849), 593
RECV(098), 924
RET(093), 743, 754
RGHT$(653), 1111
ROOT(072), 495
ROTB(620), 493
RPLC$(661), 1120
RSET, 135
RSTA(531), 138, 141, 144
RSTB(533), 141
RXD(235), 898
SBN(092), 740, 751
SBS(091), 730, 743, 877
SCL(194), 713
SCL2(486), 718
SCL3(487), 722
SDEC(078), 846
SDEL(642), 667
SEC(065), 1013
SEND(090), 924
SET, 135
SETA(530), 138, 141, 144
SETB(532), 141
SETR(635), 634
SIGN(600), 435
SIN(460), 540, 597
SIND(851), 597
SINS(641), 664
SNUM(638), 655
SNXT(009), 825
SPED(885), 790
SQRT(466), 551, 608
SQRTD(857), 608
SRCH(181), 638
SREAD(639), 658
SSET(630), 621
STEP(008), 825
STUP(237), 921
SUM(184), 649
SWAP(637), 641, 655, 658, 661, 664, 667
SWRIT(640), 661
TAN(462), 543
TAND(853), 600
TCMP(085), 256
TIM, 186
TIMH(015), 191
TIMHWX(817), 1096
TIMHX(551), 191
TIML(542), 203
TIMLX(553), 203
TIMW(813), 1088
TIMWX(816), 1088
TIMX(550), 186
TKOF(821), 1141
TKON(820), 1137
TMHH(540), 195
TMHHX(552), 195
TMHW(815), 1096
TOCV(285), 1063
TRSM(045), 1020
TST(350), 123
TSTN(351), 123

TTIM(087), 199
TTIMX(555), 199
TXD(236), 892
UP(521), 121
WDT(094), 1053
XCGL(562), 290
XCHG$(665), 1125
XCHG(073), 289
XFER(070), 285
XFRB(062), 282
XNRL(613), 488
XNRW(037), 487
XORL(612), 485
XORW(036), 483
ZCP(088), 265
ZCPL(116), 268
ZONE(682), 701
Некритические ошибки в работе
Формирование и сброс, 1025

О

Области данных
Адресация, 7
Область DM
Биты области DM в качестве условий выполнения, 123
Обновление
Командой IORF(097), 843
Мгновенное обновление, 18, 23, 116
Обновление входов/выходов, 22
Однократное выполнение по фронту, 116
Циклическое обновление, 22
IORF(097), 24, 83
Обновление входов/выходов, 22
Обновление по команде IORF(097), 24
Задачи обработки прерываний, 83
Объем программы, 24
Операнды
Описание, 6
Постоянные, 12
Текстовые строки, 13
Указание, 9
Операции над стеком
Время выполнения, 1180
Отладка программы
Команды для диагностики неисправностей, 1025, 1050
Команды отладки программы, 1020, 1024
Ошибки
Коды
Программирование, 1025, 1033
Критические, 50
Сброс, 1033
Формирование, 1033
Некритические
Сброс, 1025
Формирование, 1025
Ошибка выполнения команды, 48
Ошибка выхода за границу UM, 48
Ошибка доступа, 48
Ошибка недопустимой команды, 48
Ошибки программы, 50
Ошибки, программируемые пользователем, 1025, 1033
Программирование сообщений, 997
Сообщения
Программирование, 997
Флаги ошибок связи, 909, 918, 933
Ошибки программы, 50

П

Память ввода/вывода
Адресация, 7
Задачи, 68
Переходы, 22, 41, 168, 176
CJP(510) и CJPN(511), 172
ПИД-регулятор, 671, 684, 1059, 1063
ПЛК серии CV
Преобразование адресов памяти, 1059, 1063
Подпрограммы, 41
Время выполнения, 1182
Помехи, xxvii
Подавление, xxxii
Попадание в диапазон, 265, 268, 789
Порт RS-232C
Передача данных через порт RS-232C, 892
Прием данных через порт RS-232C, 898
Последовательный интерфейс
Описание, 882
Постоянные
Операнды, 12
Пошаговые программы
Создание, 824
Преобразование
см. также Данные, преобразование
Преобразование адресов памяти, 1059, 1063
Прерывания

Запланированные
Чтение интервала, 759
Запрет всех прерываний, 766
Маскирование, 755
Отключение, 84
Разрешение всех прерываний, 768
Сброс, 763
См. также Внешние прерывания
Чтение состояния маски, 759
Программирование
Задачи и программы, 58
Использование битов TR, 119
Меры предосторожности, 36
Местоположение команд, 7
Мнемоническое представление, 25
Объем программы, 24
Ограничения, 27
Основные принципы, 25
Подготовка данных в областях данных, 287
Примеры, 31
Приостановка/перезапуск программных блоков, 1075
Проверка программ, 45
Программирование сообщений, 997
Программирование шагов, 41
Ограничения, 43
Программные блоки, 5, 41
Ограничения, 44
Программы и задачи, 2
Путь для тока, 4
См. также Блочные программы
Создание задач, 74
Создание пошаговых программ, 824
Структура программы, 2, 25
Программирование шагов, 41
Программные блоки, 5, 41, 44
Ветвление, 1077, 1084, 1088, 1092, 1096, 1099
Взаимосвязь с задачами, 75
Время выполнения команд, 1188
Команды для создания программных блоков, 1067, 1102
Описание, 1068, 1071
Приостановка и перезапуск, 1075
Программы
См. также Программирование
Протоколирование
Флаги и биты управления, 1023
Протоколирование данных
См. также Протоколирование
Путь для тока
Описание, 4

P

Работа
Основные принципы, 59
Радианы
Преобразование из радиан в градусы, 538, 595
Радиоактивные излучения, xxvii
Регистры указателей, 10
Указание адреса слова/бита в регистре указателя, 296
Указание адреса текущего значения таймера/счетчика в регистре указателя, 298
Резервное копирование
Меры предосторожности, xxvi
Реле с защелкой
С помощью KEEP(011), 128

C

С плавающей запятой десятичное, 14, 15
Самоблокирующиеся биты
С помощью KEEP(011), 130
Сброс битов, 141
Свойства, 51
Связь
Время выполнения команд, 1185
Время выполнения сетевых команд, 1186
Команды последовательного интерфейса, 882, 923
Описание последовательного интерфейса, 882
Передача данных через порт RS-232C, 892
Прием данных через порт RS-232C, 898
Сетевые команды
Время выполнения, 1186
Сети
Сетевые команды, 924
Символы ASCII, 13
Символьные математические команды
Время выполнения, 1174
Система связи SYSMAC NET
Связь, 924, 931
Система SYSMAC LINK
Связь, 924, 931
Системные ошибки
Запрет регистрации в журнале ошибок, 1027
Системные требования к ПК, 53
Скоростной счет
Чтение текущего значения, 775, 781
Сообщения
Программирование, 997

Спец. модули ввода/вывода
Запись в память модуля, 873
Чтение памяти модуля, 870
Специальные математические команды
Время выполнения, 1178
Сравнение, 785
Статическое электричество, xxvii
Стеки
Команды работы со стеком, 620
Сторожевой таймер
Увеличение, 1053
Структура программы, 25
Счетчики, 184, 228
Время выполнения, 1170
Примеры применения, 223
Реверсивный счетчик, 215
Сброс с помощью CNR(545), 220

Т

Таблицы сравнения, 785
Таймеры, 184, 229
Время выполнения, 1170
Примеры применения, 223
Сброс с помощью CNR(545), 220
Таймер задержки для программного блока, 1096
Текстовые строки
Время выполнения команд, 1190
Команды обработки текстовых строк, 1103, 1137
Операнды, 13
Тригонометрические функции
Арккосинус, 547, 604
Арксинус, 545, 602
Арктангенс, 549, 606
Косинус, 542, 598
Преобразование из градусов в радианы, 536, 593
Преобразование из радиан в градусы, 538, 595
Синус, 540, 597
Тангенс, 543, 600

У

Условие выполнения
Вывод, 144
Условия выполнения
Варианты выполнения, 18
Задачи, 63
Условия выполнения команд
Описание, 5
Условия эксплуатации, xxvii
Меры предосторожности, xxvi
Установка битов, 141

Ф

Файлы
Библиотека, 54
Текстовые файлы проекта, 54
Флаг большего значения («Больше»), 40
Флаг меньшего значения («Меньше»), 40
Флаг отрицательного значения, 40
Флаг ошибки батареи, xxvi
Флаг ошибки задачи, 72
Флаг первого выполнения задачи, 71
Флаг переноса, 40
Флаг равенства, 40
Флаги, 6
Флаг активности протоколирования, 1023
Флаг завершения протоколирования, 1023
Флаг контроля события протоколирования, 1023
Флаги условий, 36
CY
Сброс, 1052
Флаги задач, 71
Флаги условий, 36
Загрузка состояния, 1057
Работа в задачах, 69
Сохранение состояния, 1055
Формат данных
Значение с плавающей запятой, 573
Форматы данных, 14
Функции, 51
Ограничения, 53

Х

Характеристики
CX-Programmer версии 5.0, 52

Ц

Цикл FOR-NEXT, 41
Циклические задачи, 61
Отключенное состояние (INI), 63
Состояние, 63
Состояние выполнения (RUN), 64
Состояние готовности (READY), 64

Состояние ожидания (WAIT), 64
Циклическое обновление, 22
Циклы
Цикл FOR-NEXT, 41
BREAK(514), 182
FOR(512) и NEXT(513), 178

Ч

Часы
Вычитание значения времени, 1009
Добавление значения времени, 1006
Команды для работы с часами, 1006, 1067

Э

Эксплуатация
Меры предосторожности, xxvii
Экспонента, 553, 610
Электромагнитные поля, xxvii

A

ASCII
Обработка текстовых строк, 1103
Преобразование в значение с плавающей запятой, 563, 568
Преобразование шестнадцатеричного значения в ASCII-символ, 446
ASCII В HEX, 450

C

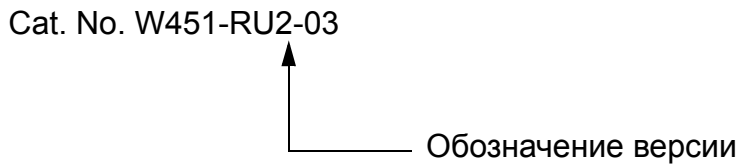
CX-Programmer
Функции для работы с задачами, 85

I

IEC 61131-3, 52, 53

Перечень версий

Версия руководства указывается в конце номера каталога на титульной странице руководства.



В таблице ниже показаны изменения, внесенные в настоящее руководство после выхода его оригинальной версии. Номера страниц соответствуют предыдущим версиям.

Обозначение версии	Дата	Изменения
01	Октябрь 2005	Оригинальная версия
02	Май 2006	Добавлена информация о модулях ЦПУ CP1H и версиях модулей. **Стр xiii и 46, 47:** обновлена версия CX-Programmer. **Стр. 694, 698 и 700:** исправлено примечание. **Стр. 707 и 711:** добавлено примечание. **Стр. 714:** расширена таблица. **Стр. 718:** расширен *Диапазон результатов преобразования* и добавлена информация о флаге переполнения. **Стр. 724, 732 и 739:** актуализирована частота на иллюстрации внизу страницы. **Стр. 732:** «S1» изменено на «S». **Стр. 1080:** из таблицы удалены команды IOSP и IORS.
03	Май 2007	Добавлена информация о модулях ЦПУ CP1L.

Поскольку компания OMRON неуклонно стремится к совершенствованию своей продукции, информация, содержащаяся в настоящем руководстве, может быть изменена без предупреждения. Подготовка настоящего руководства выполнялась с надлежащей тщательностью. Тем не менее, данная версия является предварительной, компания OMRON не несет ответственности за какие-либо ошибки и упущения в данном руководстве. Компания OMRON не несет юридической ответственности за повреждения, явившиеся результатом использования информации, содержащейся в настоящем руководстве.

Россия
ООО «Омрон Электроникс»
улица Правды, дом 26
Москва, Россия, 125040
Тел.: +7 495 648 94 50
Факс: +7 495 648 94 51
www.industrial.omron.ru

Официальный дистрибьютор:

По мере внесения в изделие изменений и улучшений его технические характеристики могут быть изменены без предварительного уведомления.
Cat. No. W451-RU2-03