



Allen-Bradley

**Усовершенствованные
(Enhanced) и Ethernet
программируемые
контроллеры PLC-5**

**(№ 1785-L11B, -L20B, -L30B,
-L40B, -L40L, -L60B, -L60L, -L80B,
-L20E, -L40E, -L80E, -L26B, -L46B,
-L46L, -L86B)**

РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ



Важная информация для пользователя

Параметры полупроводникового электрооборудования отличаются от электромеханического оборудования. Публикация Allen-Bradley (**SIG-1.1 “Safety Guidelines for the Application, Installation and Maintenance of Solid State Controls”**) описывает некоторые важные различия между полупроводниковым оборудованием и электромеханическими устройствами. Из-за этого различия, а так же из-за разнообразия используемого полупроводникового оборудования, все лица, ответственные за применение этого оборудования, должны убедиться, что применение этого оборудования оправдано.

Allen-Bradley не несет ответственности за прямой или косвенный ущерб, возникающий при использовании или применении этого оборудования.

Примеры и диаграммы в этом руководстве приведены исключительно для примера. Так как имеется много переменных и требований, связанных с любой конкретной установкой, компания **Allen-Bradley** не может принять на себя ответственность или обязательства за фактическое использование примеров, показанных в этой публикации.

Никакая патентная ответственность не принимается компанией **Allen-Bradley** относительно использования информации, токов, оборудования или программного обеспечения, описанных в этом руководстве.

Воспроизведение содержания этой публикации, полное или частичное, без письменного разрешения **Allen-Bradley Company, Inc.**, запрещено.

В этом руководстве мы используем примечания, чтобы обратить ваше внимание на требования безопасности:



ВНИМАНИЕ: Идентифицирует информацию относительно методов или обстоятельств, которые могут привести к ранениям или смерти персонала, повреждению оборудования или экономическим потерям.

Пометки "**внимание**" помогут Вам:

- идентифицировать опасность
- избежать опасности
- предвидеть последствия

Важно: Идентифицирует информацию, которая является особенно важной для успешного применения и понимания изделия.

Ethernet - является торговой маркой Intel Corporation, Xerox Corporation, и Digital Equipment Corporation.

PLC and PLC-5 - являются торговой маркой Allen-Bradley Company, Inc.

Data Highway Plus, DH+, PLC-5/11, -5/20, -5/26, -5/30, -5/40, -5/46, -5/40L, -5/60, -5/60L, -5/80, -5/86, -5/20E, -5/40E, и -5/80E - является торговой маркой Allen-Bradley Company, Inc.

Введение

Эта версия содержит новую и модифицируемую информацию. Наиболее значительные изменения - следующее:

Для этой новой информации:	См. главу:
Процессоры серии E поддерживают до 2000 программных файлов	4

Дополнительная информация

Чтобы Вам найти новую и модифицированную информацию, ищите области изменения, а направо от них номер главы.

Для этой измененной информации:	См. главу:
Канал 1A возможности подключения для процессоров PLC-5/20E, - 5/40E, и - 5/80E	1
Ссылка по использованию служебного бита S26:6 для управления форсирования и изменения таблицы данных , в то время как процессор находится в режиме работа	1
Краткий обзор шагов, требуемых для связи процессора с резидентными Вх/Вых.	5
Совет по размещению модулей Вх/Вых непосредственных Вх/Вых	5
Краткий обзор шагов, требуемых для связи с удаленными Вх/Вых	6
Советы для поиска неисправностей связи с удаленными Вх/Вых	6
Краткий обзор шагов, требуемых, для связи с каналом адаптера	7
Краткий обзор шагов, требуемых, для связи с расширенными локальными Вх/Вых	8
Проектирование сети DH +	10
Использование глобальных флагов файла статуса	10
Создание подпрограммы ошибок для обработки неисправностей удаленных Вх/Вых	16
Использование ПФС вложений требующих MCR A	17
Использование .ТО бита блок-трансфера для предотвращения задержки завершения STI	18
Программирование блок-трансферов в PIs	19
Разъяснение сообщений о неисправностях	21

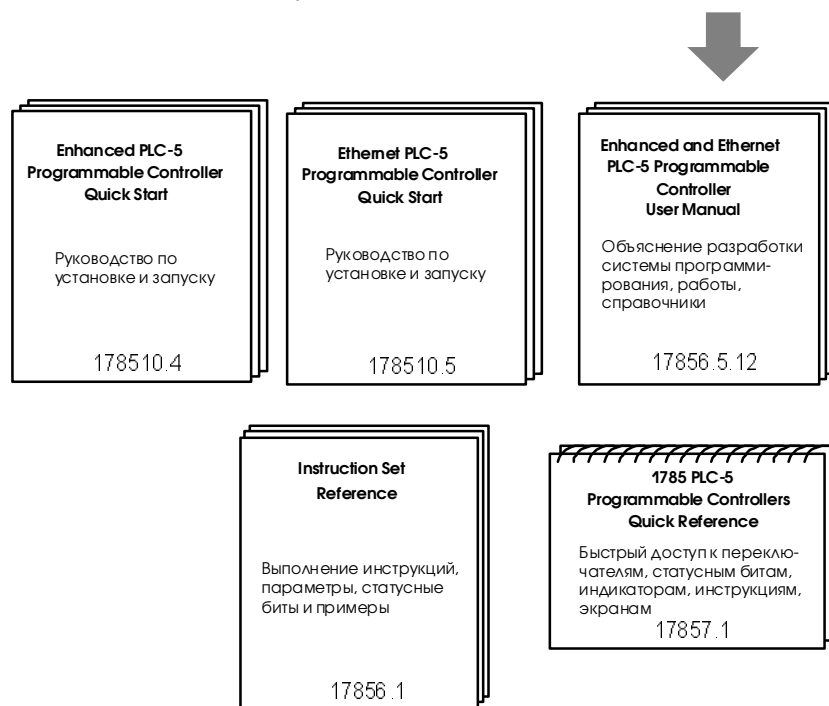
Для заметок: _____

Использование этого руководства

Как пользоваться Вашей документацией

Ваша документация по программируемым контроллерам PLC-5 организована согласно задачам, которые Вы выполняете. Эта организация позволяет Вам находить необходимую информацию без чтения всей информации, не связанной с текущей задачей. Стрелка на Рисунке 1 указывает на книгу, которую Вы в настоящее время используете.

Рис. 1
Документация по усовершенствованным и Ethernet PLC-5 программируемым контроллерам



Для более подробной информации относительно программируемых контроллеров PLC-5 или вышеупомянутых публикаций, или других связанных публикаций, свяжитесь с вашим местным торговым представительством, дистрибьютором или интегратором системы.

Для детализированной информации относительно TCP/IP протокола и работы с сетями, см. следующие публикации:

- Comer, Douglas E. *Internetworking with TCP-IP, Volume 1: Protocols and Architecture*. Englewood Cliffs, N.J.: Prentice-Hall, 1990. ISBN 0-13-468505-9.
- Tanenbaum, Andrew S. *Computer Networks*, 2nd ed. Englewood Cliffs, N.J.: Prentice-Hall, 1989. ISBN 0-13-162959-X.



Цель этого руководства

Это руководство предназначено, чтобы помочь Вам разработать и эксплуатировать усовершенствованный и/или Ethernet PLC-5 программируемые контроллеры. Использование руководства поможет Вам:

- как определить свойства процессоров и как использовать их;
- выбрать соответствующие аппаратные элементы для вашей системы;
- планировать вашу PLC-5 систему;
- эксплуатировать вашу PLC-5 систему.

Соглашения

Это руководство использует следующие соглашения:

Эта пиктограмма:	Указывает:
	Информация относится к разработчику проекта Эти советы также вызваны в индексе.
	Тема рассматривается далее в указанной публикации

- Слова в квадратных скобках представляют фактические клавиши, которые Вы нажимаете. Например: [Enter] или [F1] -онлайн программирование/ документация
- Слова, которые описывают информацию, которую Вы должны обеспечить, показаны курсивом. Например, если Вы должны напечатать имя файла, это показывается как: *filename*
- Сообщения и подсказки терминал показывает как:
Press a function key

Термины, используемые в этом руководстве

Ознакомьтесь со следующими терминами и определениями:

Термин	Определение
Блок-трансфер данных	передача данных, блоками до 64 слов, из/в блок-трансфер Вх/Вых модулей (например, для аналоговых модулей)
Передача дискретных данных	данные (слова) передаваемые из/в модули дискретных Вх/Вых
Усовершенствованные PLC-5 процессоры	ссылки на PLC-5/11, -5/20, -5/26, -5/30, -5/40, 5-46, -5/40L, -5/60, -5/60L, -5/80, и -5/86 процессоры PLC-5/26, -5/46, и -5/86 защищенные процессоры. См. публикацию PLC-5 Protected Processor Supplement 1785-6.5.13 Этот термин также относится к PLC-5/V30B, -5/40B, -5/V40B, -5/V40L процессорам. Для более подробной информации См. руководство пользователя PLC-5/VME VMEbus Programmable Controllers User Manual, 1785-6.5.9.
Ethernet	локальная сеть со скоростью обмена 10Mb, разработанная для высокоскоростного обмена информацией между компьютерами и связанными устройствами
Процессор PLC-5 Ethernet	ссылка на процессоры PLC-5/20E, -5/40E, -5/80E
Расширенные локальные Вх/Вых	Вх/Вых подключены к процессору по паралельной связи для повышения производительности, однако эта связь ограничена по длине
Расширенная локальная сеть Вх/Вых	паралельная связь для передачи данных Вх/Вых между PLC-5/40L или -5/60L и расширенным локальным адаптером Вх/Вых
PLC-5 процессор	использует характерные ссылки на усовершенствованные процессоры PLC-5 и Ethernet в этом руководстве
PLC-5/30	только применительно к PLC-5/V30B
PLC-5/40	только применительно к PLC-5/V40B
PLC-5/40L	только применительно к PLC-5/V40L
Резидентное локальное шасси	шасси Вх/Вых с установленным в нем процессором PLC-5
Резидентные локальные Вх/Вых	модули Вх/Вых расположенные в том же шасси где и процессор PLC-5
Сеть удаленных Вх/Вых	последовательная сеть связи между портом процессора PLC-5 в режиме сканера и адаптером, а также удаленными от процессора PLC-5 модулями Вх/Вых
Удаленное шасси Вх/Вых	аппаратно выполнены так, что состоят из адаптера и модулей Вх/Вых, подключенных по последовательной сети к PLC-5 в режиме сканера
MCP	главная программа управления
ПФС (SFC)	последовательная функциональня схема (язык блок графов)
Релейно-контактная	Язык программирования в релейно-контактных символах
Онлайн	режим работы (программирования) в диалоговом режиме с PLC-5
Офлайн	режим работы (программирования) при отключенном PLC-5

Краткий обзор руководства

Руководство состоит из трех основных секций:

- проектирование;
- функционирование;
- справочники.

Секция	Информация	См. главу	Название
Проектирование	Обзор возможностей процессора PLC-5	1	Знакомство с процессором
	Руководство по выбору и размещению модулей Вх/Вых	2	Выбор и размещение Вх/Вых
	Окружающая среда для системы PLC-5	3	Установка аппаратуры
	Выбор способа адресации, назначение номера рэка, обзор памяти процессора PLC-5	4	Память процессора и адресация Вх/Вых
Функционирование	Конфигурирование процессора для резидентных Вх/Вых, пересылка данных, монитор статуса	5	Связь с резидентными Вх/Вых процессора
	Конфигурация системы для связи с удаленными Вх/Вых, проектирование линии связи удаленных Вх/Вых, пересылка данных и контроль состояния	6	Связь с удаленными Вх/Вых
	Конфигурация канала адаптера PLC-5, пересылка данных и контроль состояния	7	Связь с PLC-5 через канал адаптера
	Только для процессоров PLC-5/40L и PLC-5/60L. Конфигурация системы расширенных локальных Вх/Вых, пересылка данных, контроль состояния	8	Связь с расширенными локальными Вх/Вых
	Общие и специфические соображения по производительности	9	Оптимизация производительности системы
	Конфигурация системы для Data Highway Plus и монитор статуса канала	10	Связь с оборудованием по сети DH+
	Конфигурация системы для последовательной связи и контроль состояния канала	11	Связь с оборудованием по последовательному порту
	Только для процессоров PLC-5/20E, -5/40E, -5/80E. Конфигурация системы для сети Ethernet, контроль состояния канала	12	Связь с устройствами по сети Ethernet
	Назначение паролей и привилегий	13	Защита Ваших программ
	Краткий обзор программирования PLC-5	14	Соглашения по программированию
	Определение процедуры подачи питания	15	Установка программ подачи питания
	Определение, программирование и контроль программы ошибок	16	Установка подпрограмм ошибок
	Конфигурация и контроль главных программ управления	17	Использование главных программ управления
	Использование, задание и контроль прерываний по времени	18	Использование прерываний по времени
Использование, задание и контроль прерываний по входу	19	Использование прерываний процессора по входу	
Справочники	Характеристики системы	20	Характеристики системы
	Распечатка статусных слов процессора и назначение	21	Файл состояния процессора
	Руководство по инструкциям релейно-контактной логики и временам выполнения	22	Краткий справочник по комплекту инструкций
	Как установить системные переключатели	23	Установка переключателей
	Проблемы и рекомендуемые решения	24	Поиск и устранение неисправностей
	Руководящие принципы для выбора и использования кабелей	25	Справочник по кабелям

Поддержка Rockwell Automation

Rockwell Automation предлагает услуги поддержки во всем мире, с более чем 75 офисами по продаже, 512 авторизованных дистрибьюторов и 260 авторизованных интеграторов расположенных только в США, плюс представители Rockwell Automation во всех ведущих странах мира.

Локальная поддержка продуктов

Свяжитесь с Вашим локальным представителем Rockwell Automation для:

- по продажам и поддержке;
- по обучению;
- по гарантийному обслуживанию;
- по поддержке сервисных соглашений.

Поддержка технической помощи

Если Вам необходим контакт с Rockwell Automation для технической помощи, пожалуйста просмотрите сначала раздел Поиск и устранение неисправностей. Затем вызовите Вашего локального представителя Rockwell Automation.

Ваши вопросы или комментарии относительно этого руководства

Если у Вас появились проблемы с этим руководством, пожалуйста сообщите нам о них на вложенном листе отчета «Проблемы публикации».

Важная информация для пользователя.....	1
Введение.....	S - 1
Дополнительная информация	S - 1
Использование этого руководства	P - 1
Как пользоваться Вашей документацией	P - 1
Цель этого руководства	P - 2
Соглашения	P - 2
Термины, используемые в этом руководстве	P - 3
Краткий обзор руководства	P - 4
Поддержка Rockwell Automation	P - 5
Локальная поддержка продуктов	P - 5
Поддержка технической помощи	P - 5
Ваши вопросы или комментарии относительно этого руководства	P - 5
Знакомство с процессором	1-1
Назначение этой главы	1-1
Разработка систем	1-1
Назначение частей процессора PLC-5	1-2
Возможности программирования	1-10
Использование канала процессора PLC-5 как сканера удаленных Вх/Вых	1-11
Использование канала процессора PLC-5 как адаптера удаленных Вх/Вых	1-12
Использование процессоров PLC-5/40L -5/60L как сканера локальных расширенных Вх/Вых	1-14
Выбор и размещение Вх/Вых	2-1
Назначение этой главы	2-1
Выбор модулей Вх/Вых	2-1
Выбор плотности модуля Вх/Вых	2-2
Размещение модулей Вх/Вых в шасси	2-3
Установка аппаратуры.....	3-1
Назначение этой главы	3-1
Определение надлежащих условий эксплуатации	3-1
Обеспечение механической защиты процессора	3-3
Предотвращение повреждения статическим электричеством	3-3
Разметка прокладки кабелей	3-3
Трассировка соединений	3-4
Распределение проводов по категориям	3-4
Компоновка расположения шасси	3-5
Заземление систем	3-6
Память процессора и адресация входов-выходов	4-1
Назначение этой главы	4-1
Концепция адресации Вх/Вых	4-1
Выбор режима адресации	4-4
Адресация в модулях блок-трансферов	4-8
Основы адресации	4-8
Назначение рэков	4-9
Понятие памяти процессора PLC-5.....	4-10
Понятие области данных (Файлов данных)	4-11

Адресуемые типы файлов	4-13
Понятие памяти программных файлов	4-14
Способы адресации	4-15
Адресация Вх/Вых в таблицах отображения	4-15
Указание логических адресов	4-16
Косвенная адресация	4-18
Индексная адресация	4-19
Символическая адресация	4-20
Оптимизация времени выполнения инструкций и процессорной памяти	4-21
Эффективное использование памяти Вх/Вых	4-22
Коммуникации с резидентными Вх/Вых процессора	5-1
Назначение этой главы	5-1
Введение в сканирование процессора PLC-5	5-1
Сканирование программы	5-2
Передача данных к резидентным Вх/Вых процессора	5-3
Передача дискретных данных к резидентным Вх/Вых процессора	5-3
Передача запросов немедленных Вх/Вых	5-3
Конфигурирование системы для резидентных Вх/Вых процессора	5-4
Передача данных блок-трансферов в резидентные Вх/Вых процессора	5-4
Связь с удаленными входами/выходами (Remote I/O)	6-1
Назначение этой главы	6-1
Выбор устройств, которые вы можете подключить	6-2
Введение в удаленные Вх/Вых (Remote I/O)	6-3
Проектирование сети удаленных Вх/Вых	6-4
Требования по проектировании сети удаленных Вх/Вых	6-4
Проектирование кабелей	6-5
Настройка канала процессора для работы в режиме сканера	6-6
Определение файла состояния Вх/Вых	6-7
Определение информации о конфигурации канала	6-8
Определение списка сканирования	6-9
Связь с адаптером удаленных Вх/Вых	6-11
Трудности поиска неисправностей при связи с удаленными Вх/Вых	6-12
Пересылка блоков данных	6-13
Биты неосновных ошибок блок-трансфера	6-15
Блок-трансфер данных удаленных Вх/Вых	6-15
Последовательность работы блок-трансфера и биты состояния	6-16
Требования к программированию блок-трансферов	6-20
Основные требования	6-20
Для процессорного шасси	6-20
Просмотр каналов сканера удаленных Вх/Вых	6-21
Адресация файла состояния удаленных Вх/Вых	6-23
Связь с PLC-5 через канал адаптера	7 - 1
Назначение этой главы	7 - 1
Конфигурация канала связи адаптера к PLC-5	7 - 1
Конфигурирование канала связи адаптера к PLC-5	7 - 2
Задание для канала адаптера скорости связи, адреса и размера река	7 - 3
Задание файла конфигурации дискретных пересылок	7 - 4
Программирование дискретных пересылок для режима адаптера	7 - 8
Программирование блок-трансферов для канала адаптера	7 - 8
Конфигурирование запросов блок-трансферов	7 - 9
Пример блок-трансфера релейно-контактной логики	7 - 11
Эффект программирования блок-трансферов на канал процессора в режиме адаптера для дискретных пересылок	7 - 14

Монитор статуса канала адаптера	7 - 15
Монитор статуса супервизорного процессора	7 - 16
Монитор удаленных Вх/Вых канала адаптера	7 - 17
Подключение локальных расширенных Вх/Вых	8-1
Назначение этой главы	8-1
Выбор устройств для подключения	8-1
Кабельные соединения	8-2
Размещение и адресация Вх/Вых	8-2
Передача данных	8-4
Передача дискретных данных	8-5
Расчет времени выполнения блок-трансферов	8-6
Передача блоков данных	8-6
Конфигурирование процессора в качестве сканера локальных расширенных Вх/Вых	8-9
Требования для локальных расширенных рэков	8-9
Мониторинг статуса локальных расширенных Вх/Вых	8-13
Оптимизация производительности системы	9-1
Назначение этой главы	9-1
Сканирование программы	9-1
Эффекты от ложной и истинной логики на время сканирования логики	9-2
Эффекты от различных инструкций на время сканирования логики	9-2
Эффекты от различных состояний входов на время сканирования логики	9-2
Эффекты от использования прерываний на время сканирования логики	9-3
Эффекты служебного времени	9-4
Редактирование в режиме REMOTE RUN	9-4
Размещение блок-трансфер модулей в резидентном шасси процессора	9-4
Использование файлов глобальных флагов состояния	9-5
Расчет производительности	9-5
Задержка входных и выходных модулей	9-5
Пересылка через монтажную плату Вх/Вых	9-5
Время сканирования удаленных Вх/Вых	9-6
Скорость связи	9-6
Количество рэков	9-7
Блок-трансферы	9-7
Вычисление наихудшего времени сканирования удаленных Вх/Вых	9-8
Оптимизация времени сканирования удаленных Вх/Вых	9-8
Время процессора	9-10
Пример расчета	9-11
Влияние онлайн операций на производительность	9-11
Эффект вставки релейно-контактных цепей около предела в 56 кслов	9-12
Связь с оборудованием по сети DH+	10-1
Назначение этой главы	10-1
Выбор устройств, которые вы хотите подключить	10-1
Проектирование сети	10-2
Конфигурация каналов для связи по DH+	10-2
Использования файла основных флагов состояния	10-4
Просмотр каналов связи DH+	10-5
Оценка работы сети DH+	10-9
Узлы	10-9
Размер и число сообщений	10-10
Кому предназначено сообщение	10-11
Внутреннее время обработки	10-12
Результаты тестирования лучшего времени ответа сети DH+	10-13
Требования к приложениям	10-14

Связь с оборудованием по последовательному порту	11-1
Назначение этой главы	11-1
Выбор между RS-232C, RS-422A и RS-423	11-1
Настройка последовательного порта процессора	11-2
Использование канала 0	11-2
Режим User («Пользователь»)	11-2
Режим System («Системный»)	11-3
Метод связи станции Master с удаленной станцией	11-4
Изменение режима	11-4
Подключение	11-5
Конфигурация канала 0	11-5
Конфигурация канала 0 для связи DF1 «точка-точка»	11-6
Конфигурация канала 0 как станции Slave	11-8
Конфигурация канала 0 как станции Master	11-10
Конфигурация канала 0 для режима User (протокол ASCII)	11-15
Конфигурация канала 0 для изменения метода связи	11-17
Просмотр состояния канала 0	11-18
Использование Экрана Состояния Режимы System	11-18
Использование экрана состояния режима User (ASCII)	11-20
Связь с устройствами в сети Ethernet	12 - 1
Назначение этой главы	12 - 1
Средства информации и носители	12 - 1
Установка и назначение IP адреса	12 - 2
Адресация сети	12 - 2
Конфигурирование канала 2 для сети Ethernet	12 - 3
Ручная конфигурация канала 2	12 - 3
Динамическая конфигурация Канала 2 с помощью BOOTP	12 - 5
Использование DOS/Windows BOOTP	12 - 6
Инсталляция DOS/Windows BOOTP сервера	12 - 7
Редактирование файла конфигурации DOS/Windows BOOTP	12 - 7
Запуск утилиты Boot Server	12 - 9
Запуск DOS утилиты	12 - 10
Запуск Windows утилиты	12 - 10
Использование дополнительных функций Ethernet	12 - 11
Использование широковещательной адресации	12 - 11
Использование маски сегмента и шлюзов	12 - 12
Ручная конфигурация канала 2 для процессора сегмента	12 - 13
Использование BOOTP для конфигурации канала 2 процессора сегмента	12 - 13
Расшифровка кодов ошибок	12 - 16
Расшифровка данных состояния сети Ethernet	12 - 17
Характеристики Ethernet PLC-5	12 - 18
Защита ваших программ	13 - 1
Использование этой главы	13 - 1
Относительно паролей и привилегий	13 - 1
Определение классов привилегий	13 - 3
Назначение класса привилегии на канал или файл офлайн	13 - 4
Назначение класса привилегий для узлов	13 - 4
Назначение привилегий чтения/записи для программных файлов	13 - 5
Назначение привилегии чтения/записи файлу данных	13 - 5
Использование защиты процессоров	13 - 5
Соглашения по программированию	14 - 1
Назначение этой главы	14 - 1
Форсирование	14 - 1
Форсирование входов и выходов	14 - 1
Форсирование переходов ПФС	14 - 2

Использование специальных подпрограмм	14 - 2
Задание приоритетов для прерываний и МСР	14 - 4
Состояние выполняемой программы	14 - 4
Влияние на планирование Приоритетов	14 - 6
Определение и программирование подпрограмм прерывания	14 - 6
Установка подпрограмм подачи питания.....	15 - 1
Назначение этой главы	15 - 1
Установка защиты подачи питания	15 - 1
Разрешение или запрещение старта	15 - 1
Определение процедуры подачи питания	15 - 2
Установка подпрограмм ошибок	16 - 1
Назначение этой главы	16 - 1
Понятие концепции подпрограммы ошибок	16 - 1
Реакция на основное повреждение	16 - 1
Понятие обнаружения процессором основных повреждений.....	16 - 2
Ошибка процессора в резидентном или расширенном локальном рэке Вх/Вых	16 - 3
Ошибка удаленного шасси Вх/Вых	16 - 3
Описание программы ошибок	16 - 4
Описание программного сторожа	16 - 5
Уход от многочисленных ошибок программного сторожа	16 - 5
Создание подпрограммы ошибок	16 - 6
Установка аварии.....	16 - 6
Очистка главной ошибки	16 - 6
Выбор программы ошибок из релейно-контактной логики	16 - 8
Использование релейно-контактной логики для устранения причин ошибок	16 - 9
Блок-трансфер в программе ошибок	16 - 10
Проверка программы ошибок.....	16 - 10
Монитор ошибок	16 - 10
Монитор основных и не основных ошибок и кодов ошибок	16 - 11
Интерпретация основной ошибки	16 - 11
Интерпретация неосновной ошибки	16 - 11
Монитор статусных битов	16 - 11
Использование главных программ управления.....	17 - 1
Назначение этой главы	17 - 1
Выбор главных программ управления	17 - 1
Понимание обработки процессором МСР	17 - 1
Конфигурация МСР	17 - 3
Монитор МСР	17 - 4
Использование прерываний по времени	18 - 1
Назначение этой главы	18 - 1
Использование прерываний по времени	18 - 1
Задание STI в релейно-контактной логике	18 - 1
Пример выполнения STI	18 - 2
Блок-трансферы в файле прерывания по времени	18 - 2
Назначение прерываний по времени	18 - 3
Монитор прерываний по времени	18 - 4
Применение прерываний процессора по входу	19-1
Назначение этой главы	19-1
Использование прерывания процессора по входу	19-1
Написание релейно-контактной логики PII.	19-2

Примеры PII приложения	19-2
Блок-трансфер в прерываниях по вводу (PII)	19-3
Требования при проектировании	19-4
Установка прерывания процессора по входу	19-5
Контроль прерываний процессора по входу	19-6
Характеристики системы.....	20-1
Характеристики процессора	20-1
Характеристики процессора (продолжение)	20-3
Характеристики батареи (1770-ХУС)	20-4
Файл состояния процессора	21 - 1
S:0 - S:2	21 - 1
S:3 - S:10	21 - 2
S:11	21 - 4
S:12	21 - 5
S:13 - S:24	21 - 7
S:26 - S:35	21 - 9
S:36 - S:78	21 - 10
S:79 - S:127	21 - 11
Краткий справочник по комплекту инструкций	22-1
Назначение этой главы	22-1
Релейные инструкции	22-2
Инструкции таймера	22-3
Инструкции счетчика	22-4
Инструкции сравнения	22-5
Инструкции вычисления	22-7
Логические инструкции	22-14
Инструкции преобразования	22-15
Инструкции перемещения и изменения битов	22-16
Файловые инструкции	22-17
Инструкции диагностики	22-18
Инструкции регистрового сдвига	22-19
Инструкции секвенсора	22-20
Инструкции управления программой	22-21
Инструкции передачи сообщений и управления процессами	22-23
Инструкции блок-трансферов	22-24
Инструкции ASCII	22-25
Требования к памяти и времена выполнения	22-28
Инструкции бита и слова	22-28
Инструкции: файловые, управления программой и ASCII	22-31
Установка переключателей	23-1
Назначение этой главы	23-1
Переключатели процессора	23-2
Переключатель 1	23-2
Переключатель 2	23-3
Монтажная плата шасси Вх/Вых	23-4
Процессор PLC-5 в шасси Вх/Вых	23-4
Адаптер удаленных Вх/Вых 1771-ASB или	23-5
адаптер локальных расширенных Вх/Вых 1771-ALX	23-5
Разъем шасси для конфигурации источника питания	23-6
Модуль адаптера удаленных Вх/Вых	23-7
Модули 1771-ASB серий С и D без комплементарных Вх/Вых	23-7

Номер рэков Вх/Вых (1771-ASB серий С и D) без комплементарных Вх/Вых	23-8
Модуль адаптера локальных расширенных Вх/Вых	23-9
Переключатель SW-1 модуля 1771-ALX	23-9
Конфигурационные контакты модуля 1771-ALX	23-10
Поиск и устранение неисправностей	24-1
Назначение этой главы	24-1
Процессор PLC-5	24-2
Общие проблемы	24-2
Поиск неисправностей в коммуникационном канале процессора	24-3
Поиск и устранение неисправностей в локальных расширенных Вх/Вых	24-4
Индикатор состояния Ethernet	24-4
Система удаленных Вх/Вых	24-5
Поиск и устранение неисправностей в модулях адаптеров 1771-ASB серий С и D (продолжение)	24-6
Система локальных расширенных Вх/Вых	24-7
Поиск и устранение неисправностей для модуля адаптера 1771-ALX	24-7
Неожиданное поведение программы при переходе в режим «RUN»	24-8
Инструкции уникальных операций прескана	24-8
Предлагаемые действия	24-9
Справочник по кабелям	25-1
Назначение этой главы	25-1
Назначение контактов канала 0 процессора	25-1
Назначение контактов кабеля последовательной связи	25-2
Диаграммы соединений	25-3
Характеристики кабеля программирования	25-5
Кабельные соединения Ethernet	25-10

Для заметок

Знакомство с процессором

Назначение этой главы

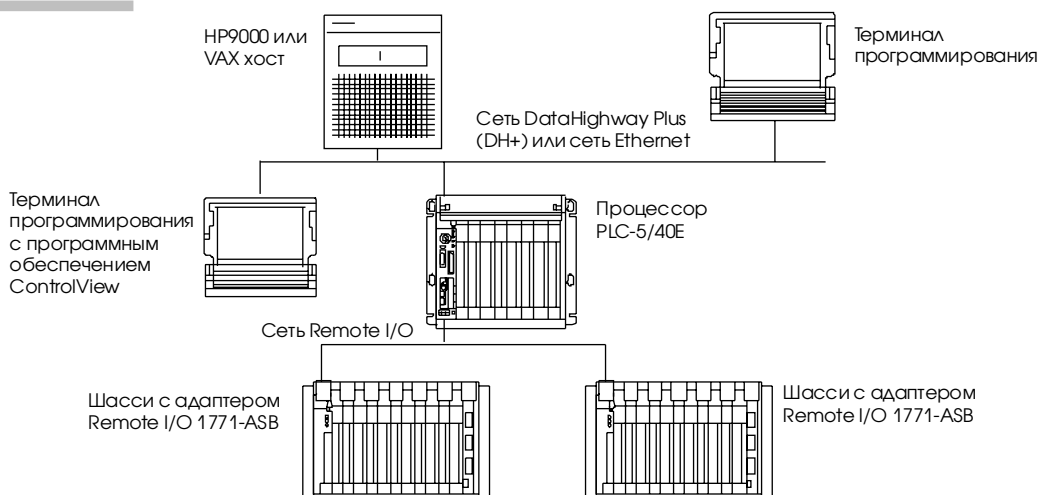
Сведения	Страница
Разработка систем	1-1
Назначение частей процессора PLC-5	1-2
Возможности программирования	1-10
Использование процессора PLC-5 как сканера удаленных Вх/Вых	1-11
Использования процессора PLC-5 как адаптера удаленных Вх/Вых	1-12
Использования процессора PLC-5/40L, -5/60L как сканера расширенных локальных Вх/Вых	1-14

Разработка систем

Вы можете использовать процессоры PLC-5 в системе, разработанной для централизованного или распределенного управления.

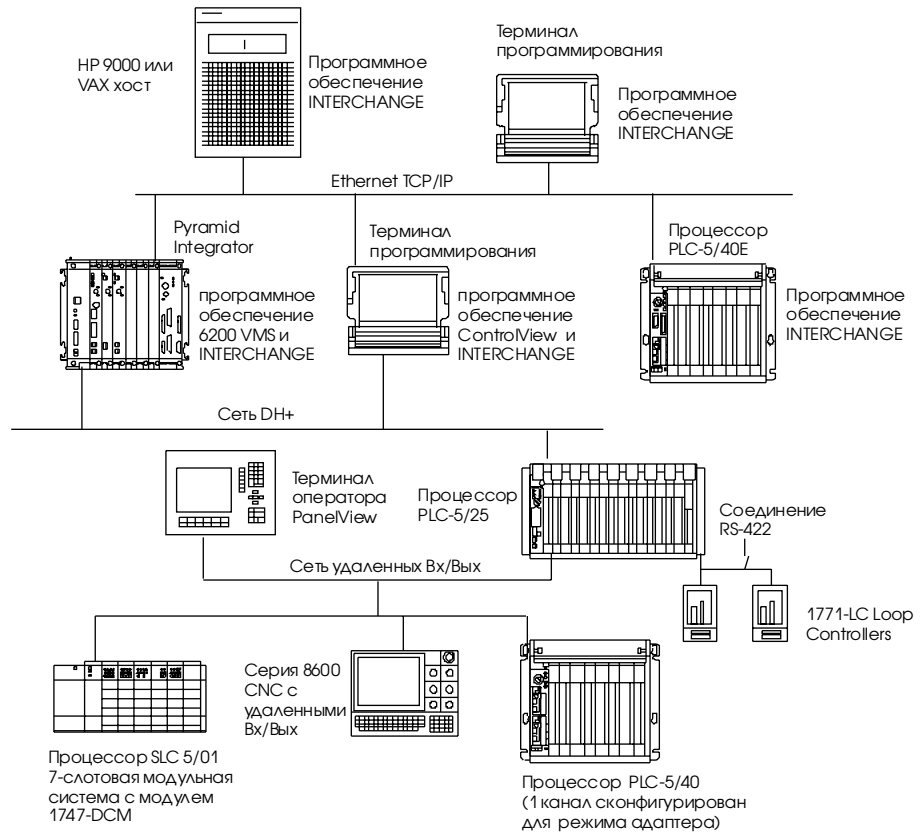
Централизованное управление -

это иерархическая система, в котором управление технологическим процессом сосредоточено в одном процессоре.



Распределенное управление

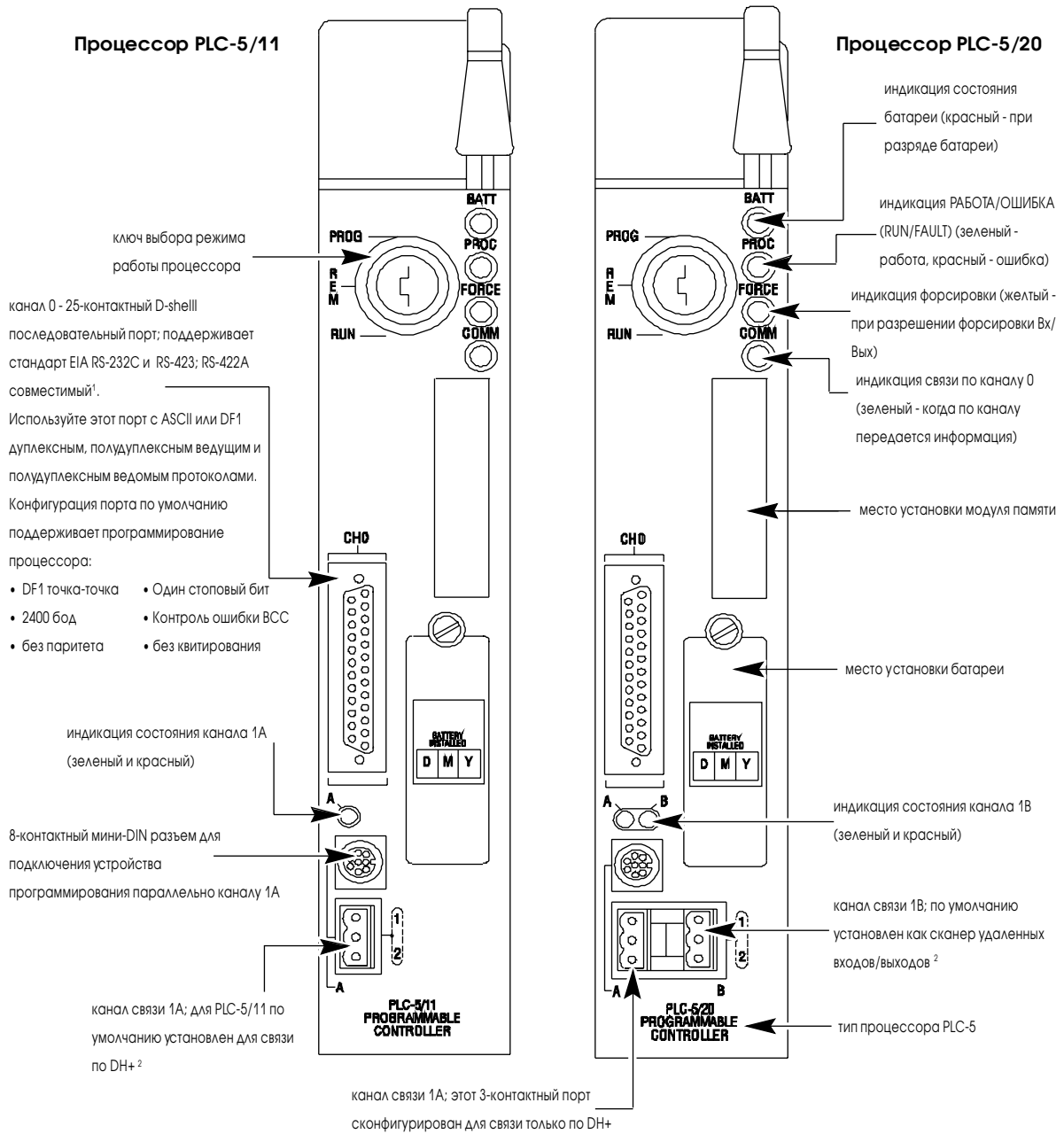
- это система, в которой функции контроля и управления распределены по производству. Несколько процессоров осуществляют функции контроля и управления, используя для связи сеть Data Highway, Ethernet или системную шину.

**Назначение частей процессора PLC-5**

Для ознакомления с лицевыми панелями процессоров используйте следующие рисунки:

Процессор:	Рисунок:	Страница:
PLC-5/11, -5/20 и -5/26	1.1	1-3
PLC- 5/30	1.2	1-4
PLC-5/40, -5/46, -5/60, -5/80 и -5/86	1.3	1-5
PLC-5/20E	1.4	1-6
PLC-5/40E и -5/80E	1.5	1-7
PLC-5/40L и -5/60L	1.6	1-8

Рисунок 1.1
Лицевые панели процессоров PLC-5/11, -5/20 и -5/26



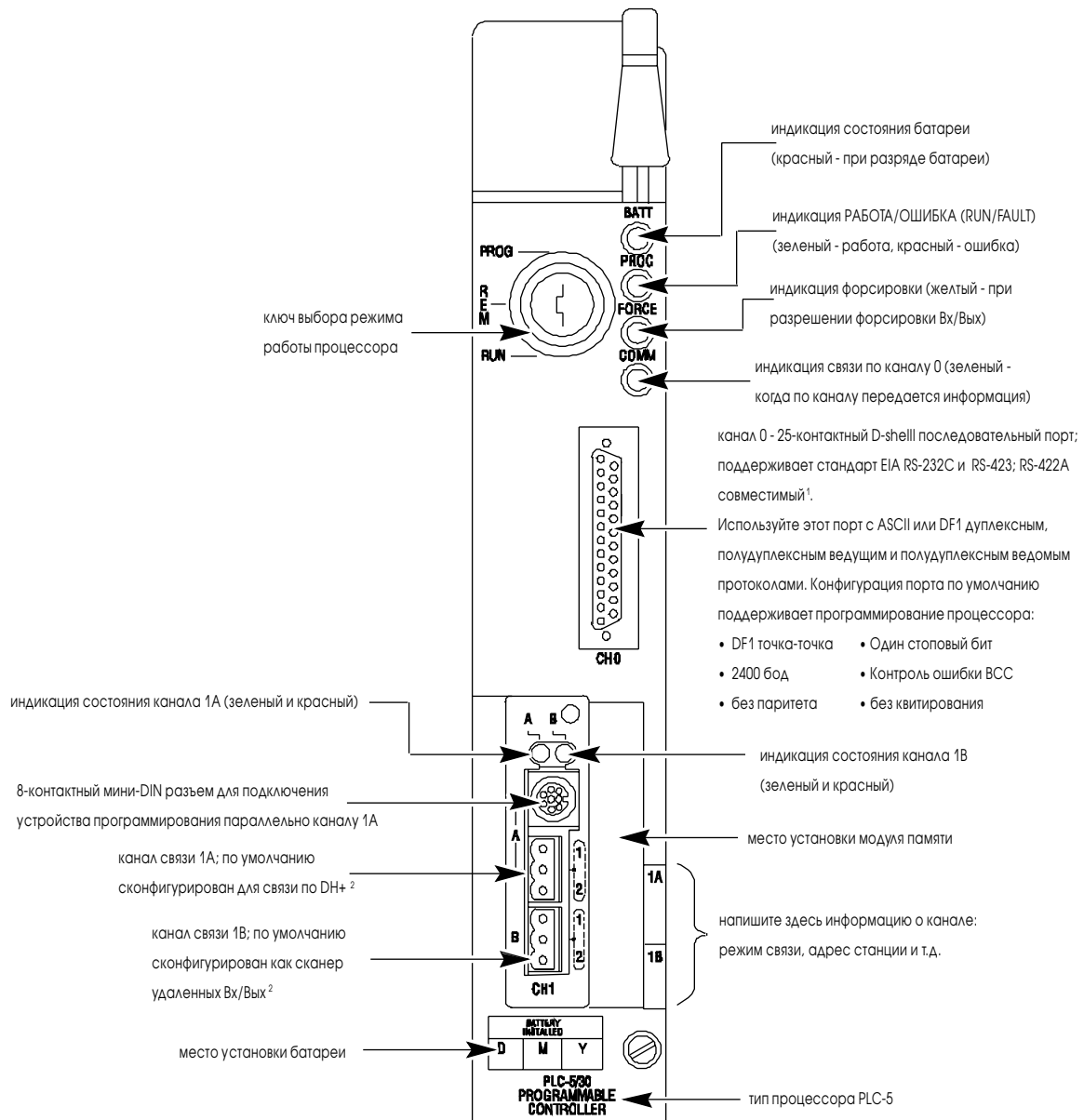
¹ Канал 0 имеет оптоэлектронную развязку (что обеспечивает высокую помехоустойчивость) и может быть использован с большей частью оборудования, поддерживающего протокол RS-422A, со следующими ограничениями:

- не используются терминаторы (termination resistors);
- расстояние и скорости передачи уменьшены до величин, приемлемых для оборудования RS-423.

² Этот 3-контактный порт конфигурируется:

- как сканер удаленных Вх/Вых;
- как адаптер удаленных Вх/Вых;
- для связи по сети DH+;
- как неиспользуемый канал.

Рисунок 1.2
Лицевая панель процессора PLC-5/30



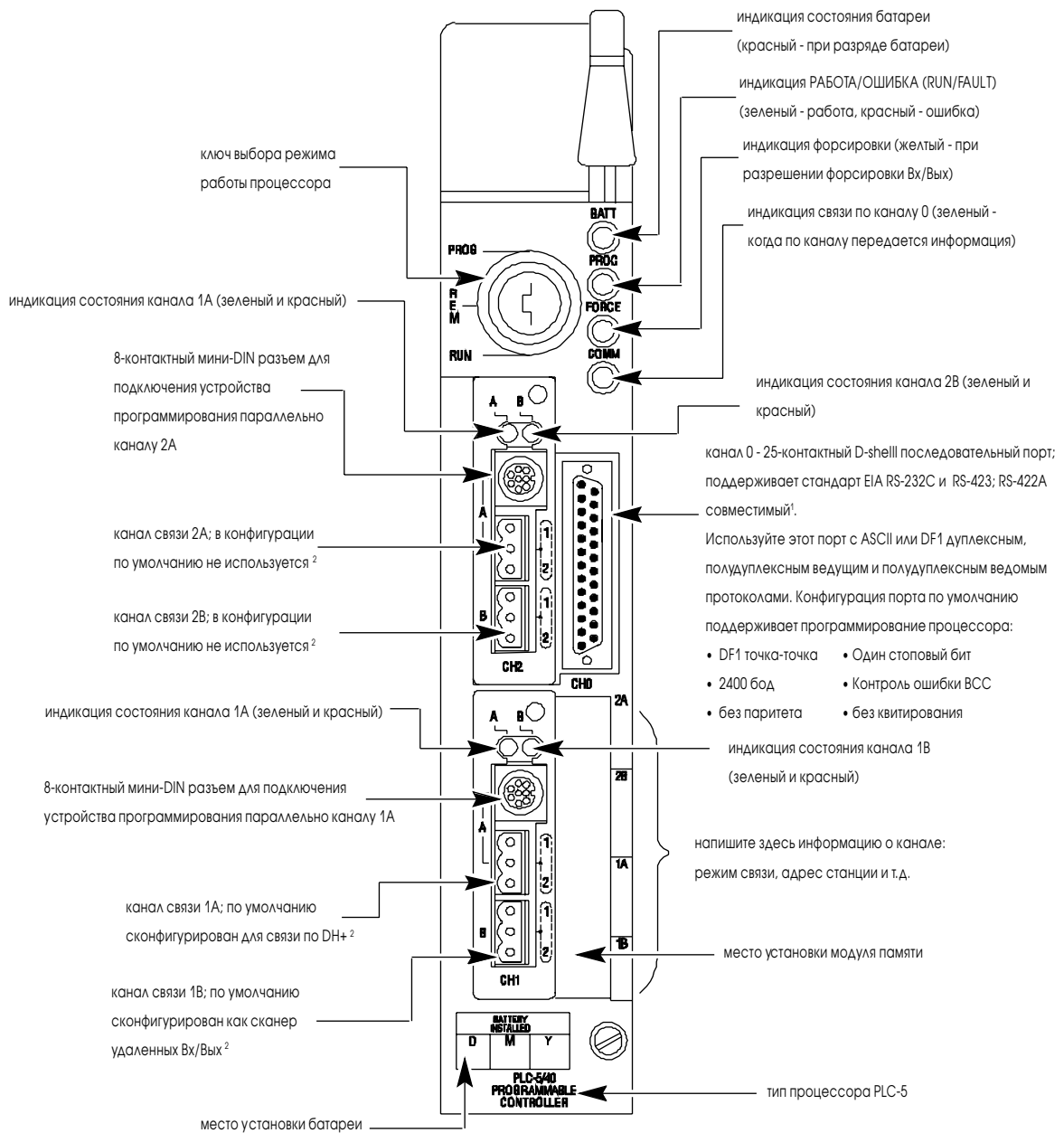
¹ Канал 0 имеет оптоэлектронную развязку (что обеспечивает высокую помехоустойчивость) и может быть использован с большей частью оборудования, поддерживающего протокол RS-422A, со следующими ограничениями:

- не используются терминаторы (termination resistors);
- расстояние и скорости передачи уменьшены до величин, приемлемых для оборудования RS-423.

² Этот 3-контактный порт конфигурируется:

- как сканер удаленных Вх/Вых;
- как адаптер удаленных Вх/Вых;
- для связи по сети DH+;
- как неиспользуемый канал.

Рисунок 1.3
Лицевые панели процессоров PLC-5/40, -5/46, -5/60, -5/80 и -5/86



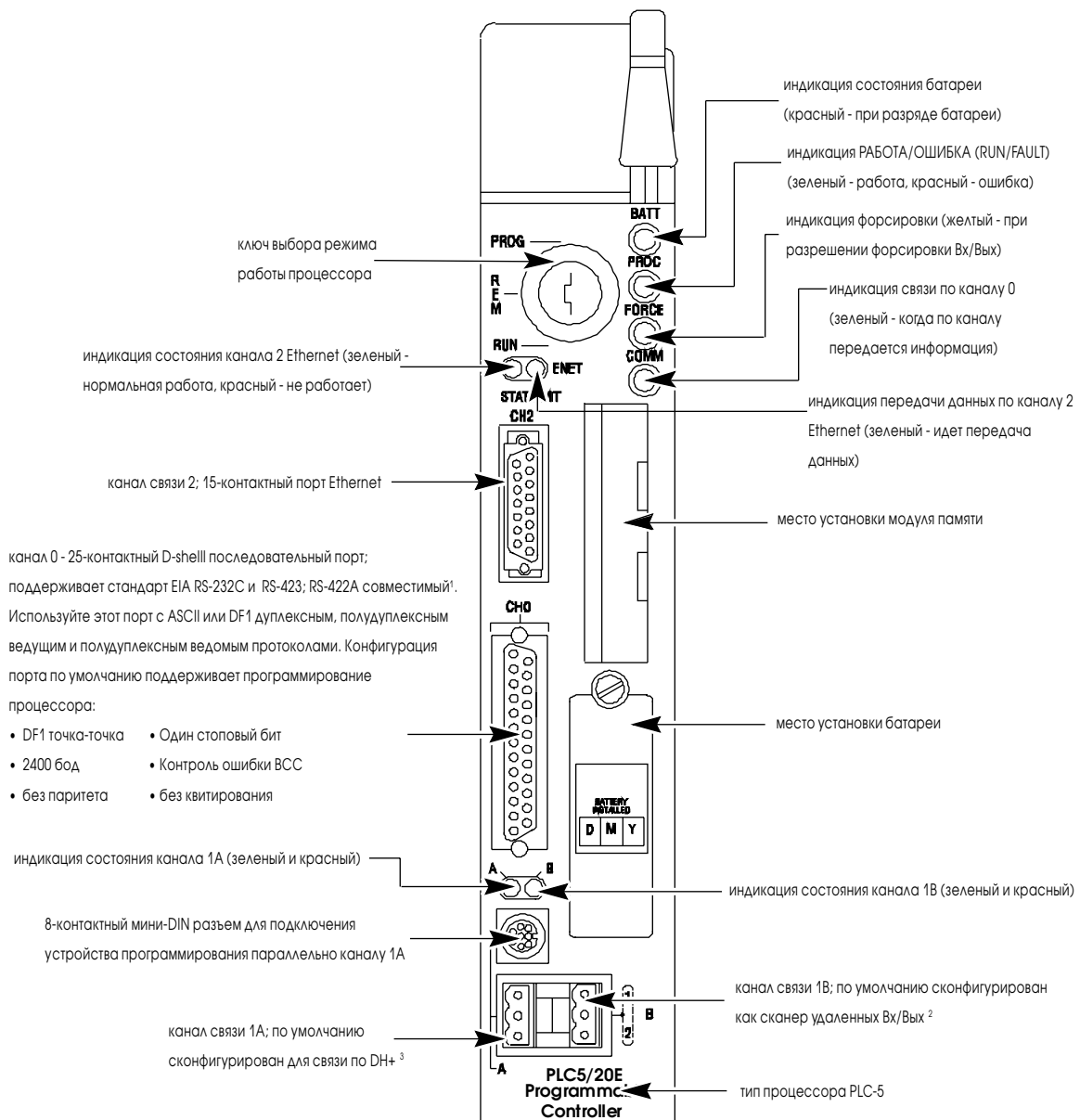
¹ Канал 0 имеет оптоэлектронную развязку (что обеспечивает высокую помехоустойчивость) и может быть использован с большей частью оборудования, поддерживающего протокол RS-422A, со следующими ограничениями:

- не используются терминаторы (termination resistors);
- расстояние и скорости передачи уменьшены до величин, приемлемых для оборудования RS-423.

² Этот 3-контактный порт конфигурируется:

- как сканер удаленных Вх/Вых;
- как адаптер удаленных Вх/Вых;
- для связи по сети DH+;
- как неиспользуемый канал.

Рисунок 1.4
Передняя панель процессора PLC-5/20E



¹ Канал 0 имеет оптоэлектронную развязку (что обеспечивает высокую помехоустойчивость) и может быть использован с большей частью оборудования, поддерживающего протокол RS-422A, со следующими ограничениями:

- не используются терминаторы (termination resistors);
- расстояние и скорости передачи уменьшены до величин, приемлемых для оборудования RS-423.

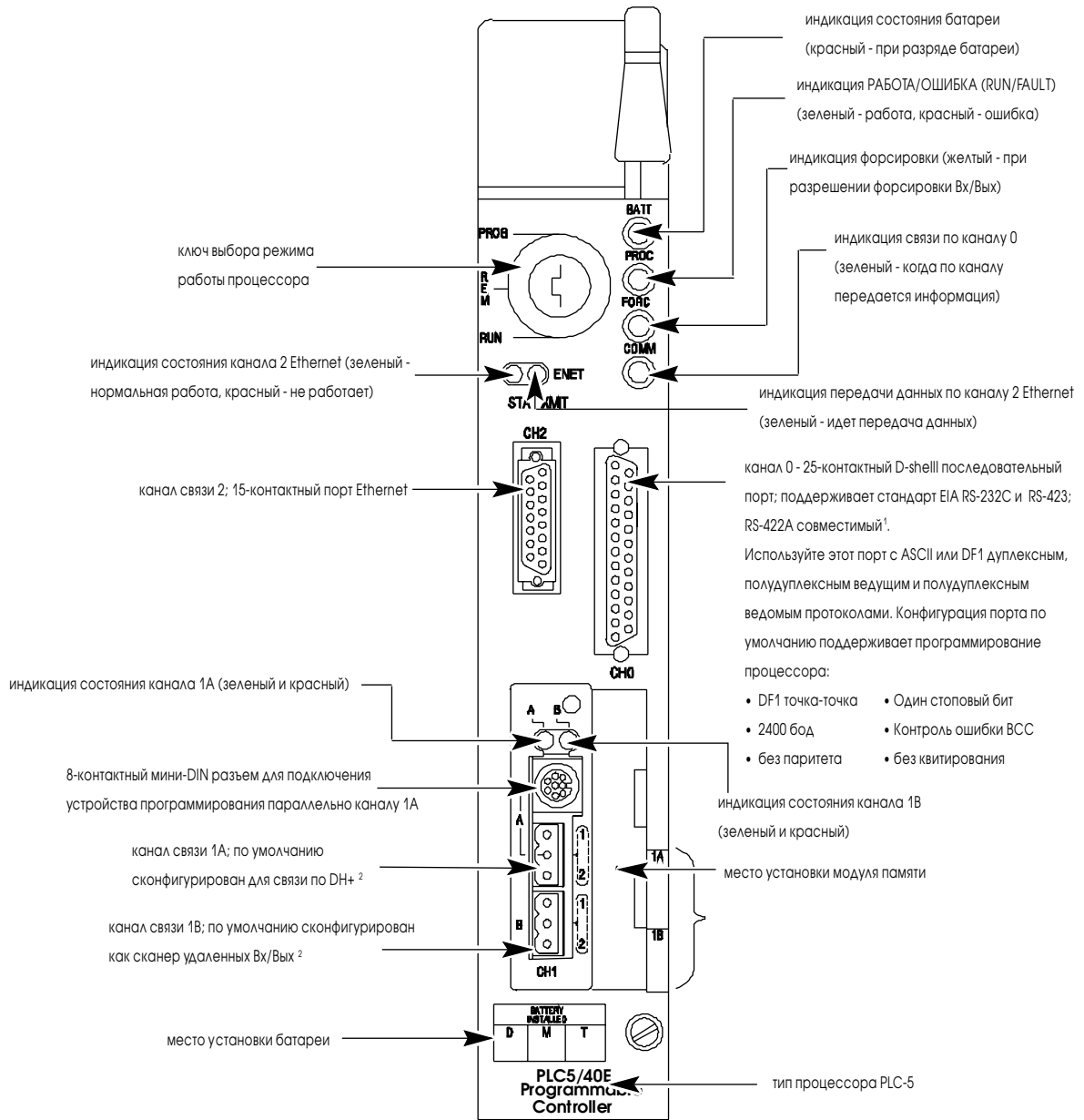
² Этот 3-контактный порт конфигурируется:

- как сканер удаленных Вх/Вых;
- как адаптер удаленных Вх/Вых;
- для связи по сети DH+;
- как неиспользуемый канал.

³ Этот 3-контактный порт конфигурируется:

- как адаптер удаленных Вх/Вых;
- для связи по сети DH+.

Рисунок 1.5
Лицевые панели процессоров PLC-5/40E и -5/80E



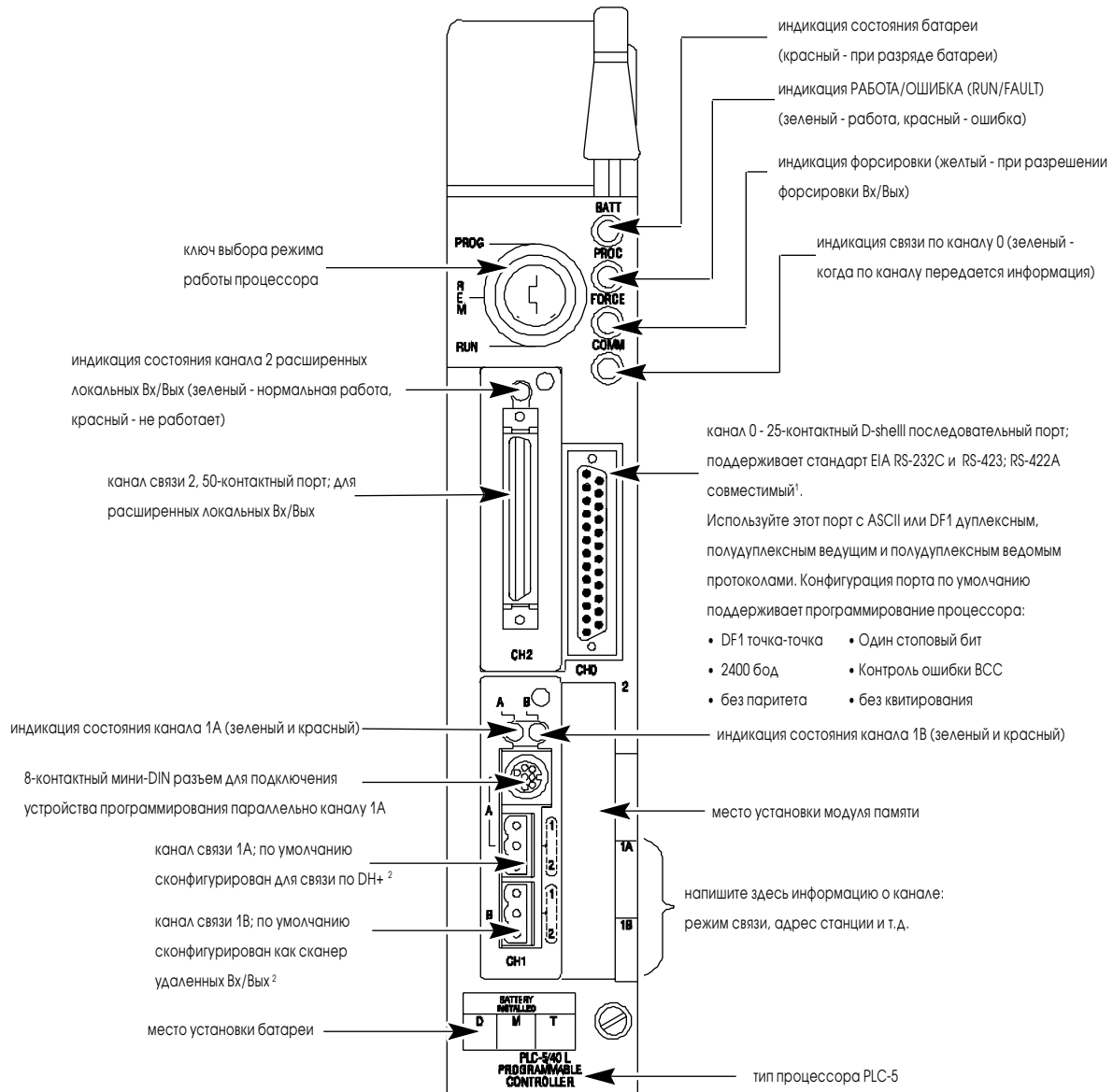
¹ Канал 0 имеет оптоэлектронную развязку (что обеспечивает высокую помехоустойчивость) и может быть использован с большей частью оборудования, поддерживающего протокол RS-422A, со следующими ограничениями:

- не используются терминаторы (termination resistors);
- расстояния и скорости передачи уменьшены до величин, приемлемых для оборудования RS-423.

² Этот 3-контактный порт конфигурируется:

- как сканер удаленных Вх/Вых;
- как адаптер удаленных Вх/Вых;
- для связи по сети DH+;
- как неиспользуемый канал.

Рисунок 1.6
Лицевые панели процессоров PLC-5/40L и -5/60L



¹ Канал 0 имеет оптоэлектронную развязку (что обеспечивает высокую помехоустойчивость) и может быть использован с большей частью оборудования, поддерживающего протокол RS-422A, со следующими ограничениями:

- не используются терминаторы (termination resistors);
- расстояние и скорости передачи уменьшены до величин, приемлемых для оборудования RS-423.

² Этот 3-контактный порт конфигурируется:

- как сканер удаленных Вх/Вых;
- как адаптер удаленных Вх/Вых;
- для связи по сети DH+;
- как неиспользуемый канал.

Используйте ключ для выбора режима работы процессора, в котором он будет находиться.

Если вы желаете:

- запустить вашу программу;

Работа выходов разрешена. (Оборудование управляется Вх/Вых, адресуемыми выполняемой программой релейно-контактной логики).

- форсировать входы/выходы;
- сохранить вашу программу на жестком диске;
- разрешить работу выходов;
- редактировать величины в таблице данных.

Примечания:

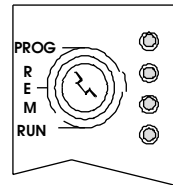
- вы не можете создавать или удалять программные файлы, создавать или удалять файлы данных, редактировать в режиме онлайн или изменять режим работы с помощью пакета программирования, пока установлен режим **RUN**;
- вы можете предупредить форсировку Вх/Вых или изменение таблицы данных, используя пакет программирования контроллера для установки пользовательского бита управления S:26/6.

Установите ключ:
RUN ("Работа")


-
- запретить работу выходов (выходы отключаются);
 - создавать, изменять и стирать файлы релейно-контактной логики, файлы SFC или файлы данных;
 - записывать/читать модуль памяти;
 - сохранять, восстанавливать программы.

Примечания:

- процессор не сканирует программу;
- вы не можете изменять режим работы с помощью пакета программирования, пока установлен режим **PROGRAM**.

PROG
("Программирование")


изменить режим между дистанционным **PROGRAM**, дистанционным **TEST** и дистанционным **RUN** с помощью пакета программирования.

Дистанционный RUN

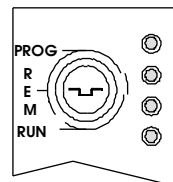
- разрешить работу выходов;
- сохранить/восстановить программу;
- редактирование *онлайн*.

Дистанционный PROGRAM

Смотрите описание режима программирования выше.

Дистанционный TEST

- выполнить программу релейно-контактной логики с отключенными выходами;
 - *нельзя* создавать или стирать файлы программы или таблицы данных;
 - сохранить/восстановить программу;
 - редактирование *онлайн*.
-

REM (Remote -
Дистанционное)


Возможности программирования

Ниже в таблице показаны возможности программирования процессора PLC-5.

Возможность	Позволяет
Релейно-контактная логика (ladder logic)	<p>Программировать, используя язык, который подобен языку релейной логики. Выберите этот язык, если Вы</p> <ul style="list-style-type: none"> • более знакомы с релейно-контактной логикой, чем с языками программирования, такими как БЭЙСИК. Ваш заводской персонал может быть более знаком с релейно-контактной логикой, учтите также и их мнение; • хотите выполнения диагностики; • хотите программировать дискретное управление.
Подпрограммы (subroutines)	<p>Использовать повторяющиеся части программной логики, доступ к которым возможен из нескольких программных файлов. Подпрограммы экономят объем памяти процессора, так как Вы программируете повторяющуюся логику только один раз. Инstrukция JSR заставляет процессор перейти к выбранному файлу подпрограммы, затем процессор этот файл однократно сканирует и возвращается в точку перехода.</p>
Последовательные функциональные схемы (ПФС) (Sequential Function Charts) (SFCs)	<p>Используйте язык последовательных функциональных схем для управления и отображения состояния логики последовательного процесса. Вместо использования для вашего приложения одной длинной логической программы разделите логику на шаги и переходы. Шаг соответствует задаче управления; переход соответствует условию, которое должно выполняться перед тем, как программируемый контроллер может выполнить следующую задачу. Просмотр этих шагов и переходов позволяет вам видеть в форме графа, каково состояние процесса в настоящее время. ПФСы позволяют конструкции, дающие возможность выполнения нескольких или одного логических путей, так же как и возможность перехода вперед или назад. Поиск неисправностей может быть сужен применением небольшой логической программы вместо всего файла релейно-контактной логики. ПФСы являются наиболее оптимальными для управления порядком событий последовательного процесса.</p>
Структурированный текст (structured text)	<p>Программа использует язык программирования, подобный БЭЙСИКу. Структурированный текст выбирается, если:</p> <ul style="list-style-type: none"> • вы более знакомы с языками программирования подобными БЭЙСИКу, чем с релейно-контактной логикой; • используется сложный математический алгоритм; • используются программные конструкции, которые зацикливаются или создают "петлю"; • создаются выборочные экраны просмотра таблицы данных.
Основные программы управления (Main Control Programs) (MCP)	<p>Отделяют последовательную логику от релейно-контактной логики и структурированного текста для модульности вашего процесса и облегчения путей поиска неисправностей. Используйте несколько основных программ управления (MCP), назначив отдельную MCP для каждого отдельного механизма или функции вашего процесса. MCP обеспечивают независимые или не связанные последовательностью действия. Основная программа управления может быть файлом ПФС (номер 1-999), релейно-контактной логики, или программой структурированного текста (номер 2-999). Одна таблица данных используется всеми MCP (т.е. у вас нет отдельной таблицы данных для каждой MCP).</p>

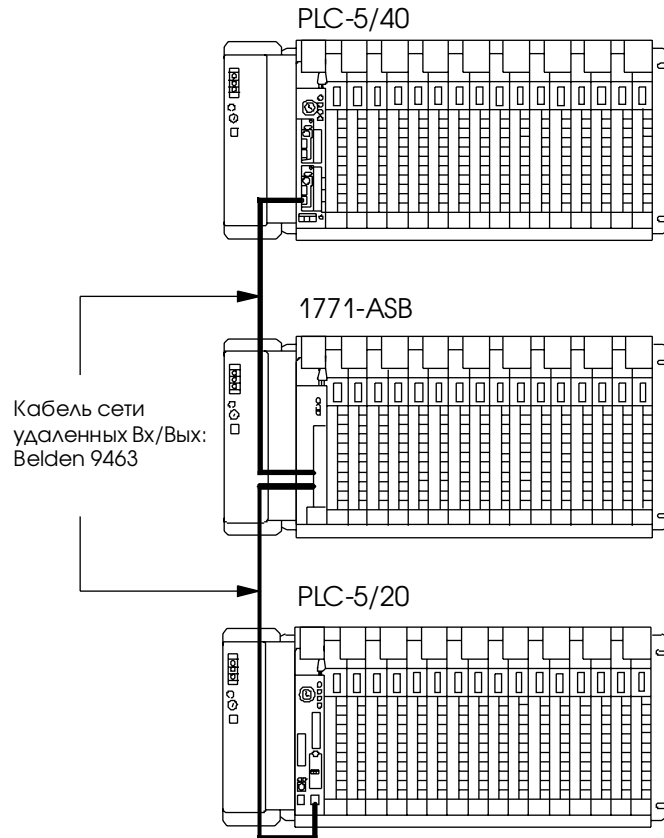
Использование канала процессора PLC-5 как сканера удаленных Вх/Вых

Сконфигурируйте канал удаленных Вх/Вых для работы в режиме сканера, чтобы читать и записывать информацию о состоянии Вх/Вых между процессором PLC-5 и устройствами Вх/Вых, расположенными удаленно от процессора.

Процессор с каналом, установленным в режим сканера, работает как супервизорный процессор для других процессоров, установленных в режим адаптера, (также как для модулей адаптеров удаленных Вх/Вых).

В режиме сканера процессор PLC-5 может:

- собирать данные от устройств в удаленных шасси Вх/Вых;
- обрабатывать данные Вх/Вых от 8-, 16-или 32-точечных модулей Вх/Вых;
- выполнять 2-, 1- или 1/2-слотовую адресацию группы Вх/Вых;
- поддерживать конфигурацию дополнительных (complementary) Вх/Вых;
- поддерживать передачу блок-трансферов для любых шасси Вх/Вых.

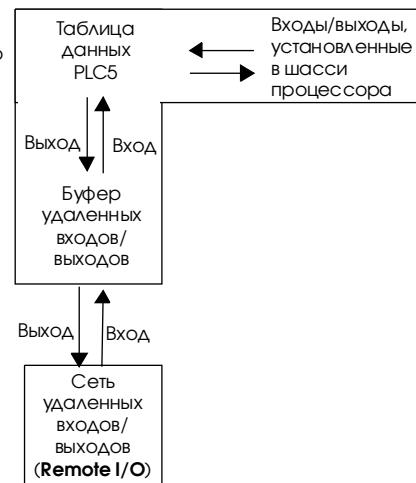


В режиме сканера процессор PLC-5:

- передает дискретные данные и данные блок-трансфера в модули и из модулей шасси удаленных Вх/Вых, а также процессоров установленных в режим адаптера;
- сканирует **буферы удаленных Вх/Вых** асинхронно к циклу программы.
- обновляет **таблицу образа данных Вх/Вых** из буфера удаленных Вх/Вых синхронно с циклом программы.

Таблица данных PLC-5 изменяется синхронно с циклом программы (при служебных действиях).

Буфер удаленных входов/выходов изменяется асинхронно с циклом программы.



Процессор PLC-5 пересылает данные Вх/Вых и состояния, используя:

• Пересылка дискретных данных	Данные передаются по 8 слов на рэк. Производится автоматически для сети удаленных Вх/Вых.
• Блок-трансферы	Специальная передача данных, для которой требуются соответствующие инструкции, обслуживающие передачу данных. Позволяют передавать максимум 64 слова. Так же используются для обмена информацией между каналом сканера и каналом процессора, установленным в режим адаптера.



Для более подробной информации об использовании процессора как сканера удаленных Вх/Вых смотрите главу 6.

Использование канала процессора PLC-5 как адаптера удаленных Вх/Вых

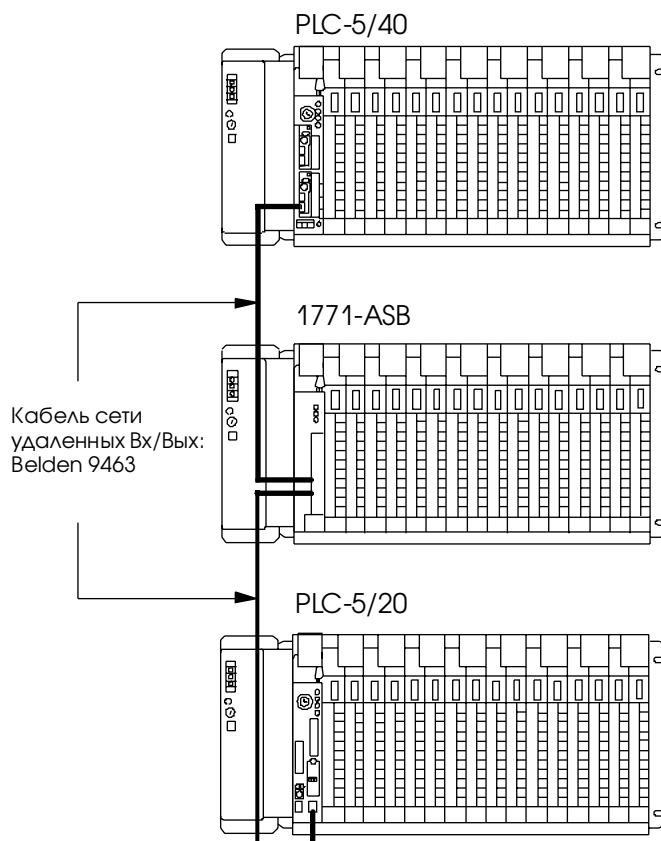
Сконфигурируйте канал процессора PLC-5 для работы в режиме адаптера, если вам необходим предсказуемый, в реальном масштабе времени, обмен данными между процессором, канал которого установлен в режим адаптера, и супервизорным процессором. Канал адаптера удаленных Вх/Вых обменивается данными с супервизорным процессором.

В этом примере канал процессора PLC-5/40, установленный в режим сканера, является супервизорным для модуля 1771-ASB и процессора PLC-5/20.

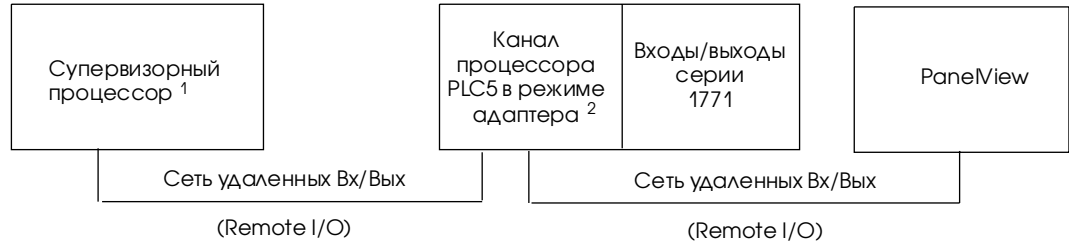
Соединение процессоров производится по сети удаленных Вх/Вых (Remote I/O).

Вы можете просматривать состояние между супервизорным процессором и процессором, канал которого установлен в режиме адаптера, с постоянным периодом (т.е. скорость обмена данными по сети удаленных входов/выходов (Remote I/O) не изменяется терминалом программирования и другим, не связанным с управлением обменом).

Процессор PLC-5, работающий в режиме адаптера, может просматривать и управлять собственными локальными входами/выходами во время связи с управляющим процессором по сети удаленных Вх/Вых (Remote I/O).



Для каналов расширенных (Enhanced) и Ethernet процессоров PLC-5, работающих в режиме адаптера, нет необходимости применять релейно-контактную логику для инструкций блок-трансферов. Вы определяете блок-трансферы в экране конфигурации адаптера, и путем определения файлов блок-трансферов.



¹ В качестве супервизорных процессоров могут применяться следующие программируемые контроллеры:

процессоры PLC-2/20 и PLC-2/30

процессоры PLC-3 и PLC-3/10

PLC-5/15 и PLC-5/25

Все расширенные (enhanced) и Ethernet процессоры PLC-5; отдельные каналы могут быть сконфигурированы для режимов сканера или адаптера удаленных Вх/Вы.

процессоры PLC-5/VME, PLC-5/30V, PLC-5/40V и PLC-5/40LV

PLC-5/250

² Все процессоры семейства PLC-5, кроме PLC-5/10, могут работать как адаптер удаленных входов/выходов.



Для более подробной информации об использовании процессора как адаптера удаленных Вх/Вых смотрите главу 7.

Использование процессоров PLC-5/40L -5/60L как сканера локальных расширенных Вх/Вых

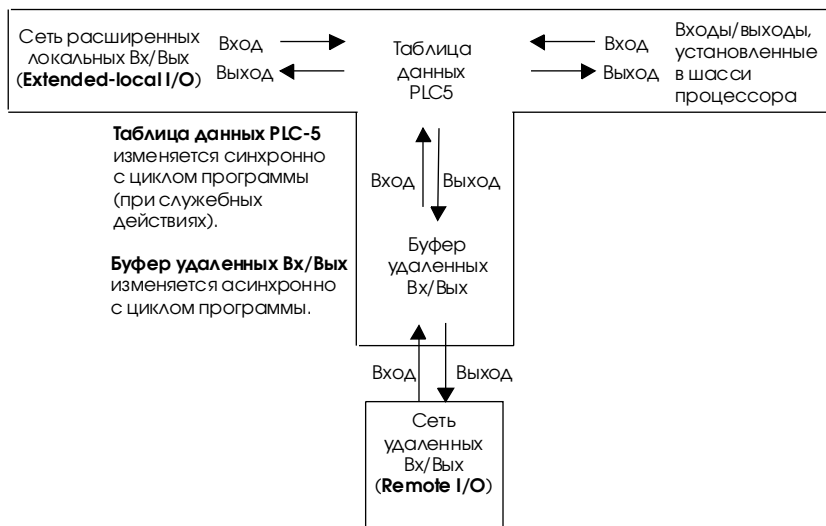
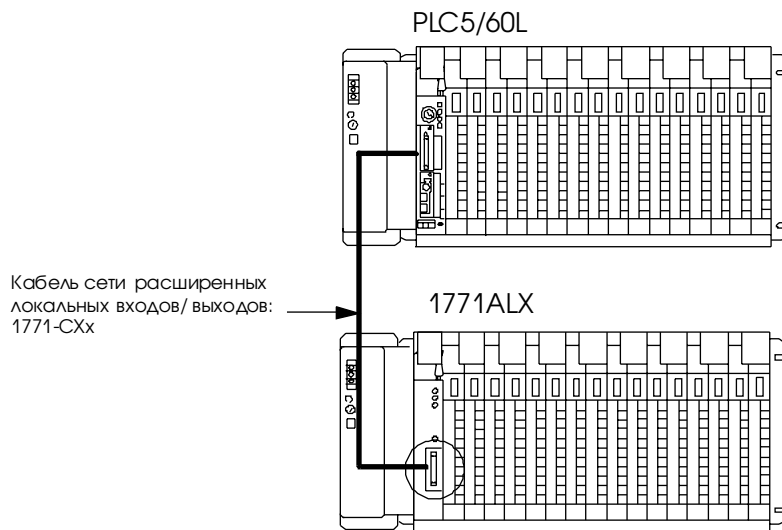
Используйте сеть расширенных локальных Вх/Вых, если вам необходимо обновлять входы/выходы намного быстрее, чем это возможно по сети удаленных Вх/Вых. Сеть расширенных локальных Вх/Вых обеспечивает более быстрое сканирование и обновление входов/выходов, чем сеть удаленных Вх/Вых. Длина сети расширенных локальных Вх/Вых ограничена 30.5 м (100 футов). Если входы/выходы размещены от процессора на расстояние, превышающее 30.5 м, вы должны использовать сеть удаленных Вх/Вых.

Процессор PLC-5/40L или -5/60L (канал 2) и модуль адаптера расширенных локальных Вх/Вых (1771-ALX) образуют сеть расширенных локальных Вх/Вых.

Сеть расширенных локальных Вх/Вых является параллельной сетью, позволяющей процессорам PLC-5/40L или -5/60L опрашивать максимум до 16 шасси расширенных локальных Вх/Вых.

Кабели выполнены таким образом, что Вы можете убирать модуль адаптера из шасси сети расширенных локальных Вх/Вых, не нарушая связи с другими шасси сети расширенных локальных входов/выходов.

Важно: Процессоры PLC-5/40L и -5/60L не могут использоваться как адаптеры расширенных локальных входов/выходов



Для более подробной информации об использовании расширенных локальных входов/выходов смотрите главу 8.

Выбор и размещение Вх/Вых

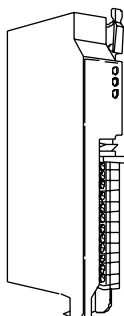
Назначение этой главы

Сведения	Страница
Выбор модулей Вх/Вых	2-1
Выбор плотности модуля Вх/Вых	2-2
Размещение Вх/Вых в шасси	2-3

Выбор модулей Вх/Вых

Выберите модули Вх/Вых для соединения вашего процессора PLC-5 с машинами или оборудованием, которые вы определили во время анализа функционирования вашего производства.

Используйте следующие список и таблицу как руководящие принципы для выбора модулей Вх/Вых и интерфейса(ов) управления оператором.



- Сколько Вх/Вых требуется, чтобы управлять вашим процессом(ами)?
- Где вы будете концентрировать точки Вх/Вых для участков полного производства, когда процесс распределен по большому физическому пространству?
- Какие типы Вх/Вых требуются, чтобы управлять вашим производством?
- Каков требуемый диапазон напряжения для каждого модуля Вх/Вых?
- Каков потребляемый от объединительной платы ток для каждого модуля Вх/Вых?
- Каковы ограничения по помехам и длине подключений для каждого модуля Вх/Вых?
- Какая изоляция требуется для каждого модуля Вх/Вых?

Таблица 2А
Указания по выбору модулей Вх/Вых

Выберите этот тип модуля Вх/Вых:	Для этих типов внешних устройств или операций (примеры):	Объяснение:
Дискретный входной модуль и комбинированный модуль Вх/Вых	Переключатели, кнопки, фотодатчики, конечные выключатели, силовые выключатели, бесконтактные датчики, датчики уровня, пускатели, контакты реле, командоконтроллеры	Входные модули воспринимают сигналы Вкл/Откл или разомкнут/замкнут. Дискретные сигналы могут быть переменного или постоянного тока.
Дискретный выходной модуль и комбинированный модуль Вх/Вых	Сигнализация, катушки реле, вентиляторы, лампы, сирены, клапаны, пускатели или соленоиды	Выходные модули взаимодействуют с устройствами типа Вкл/Откл или открыт/закрыт. Дискретные сигналы могут быть переменного или постоянного тока.
Аналоговый входной модуль	Преобразователи температуры, давления, влажности, расхода, тензопреобразователи и потенциометры	Преобразуют непрерывные аналоговые сигналы во входные значения для процессора PLC.
Аналоговый выходной модуль	Регулируемые задвижки, активаторы, регистраторы, электрические приводы, аналоговые измерители	Интерпретируют выходы процессора PLC в аналоговые сигналы (обычно через преобразователи) для внешних устройств.
Специальные модули Вх/Вых	Датчики положения, измерители расхода, связь с Вх/Вых, ASCII, радиоустройства, весы, считывание штрихового кода и меток, устройства отображения	Обычно используются для специфических приложений, таких как контроль положения, ПИД и связь с внешними устройствами.

Выбор плотности модуля Вх/Вых

Плотность модуля Вх/Вых - это количество битов в таблице образа входов или выходов, которым он (модуль) соответствует. Двухнаправленный модуль с 8-ю входами и 8-ю выходами имеет плотность 8. Плотность модуля Вх/Вых помогает определить вашу схему адресации Вх/Вых. Смотрите главу 4 для дополнительной информации по адресации Вх/Вых.

Используйте эти указания для выбора плотности модулей Вх/Вых:

Таблица 2Б
Указания по выбору плотности модулей Вх/Вых

Выберите эту плотность Вх/Вых:	Если вы:
8-ми точечный модуль Вх/Вых	<ul style="list-style-type: none"> • уже используете 8-ми точечные модули • нуждаетесь в интегрированных, отдельно защищенных предохранителями выходах • хотите минимизировать цену за модуль
16-ти точечный модуль Вх/Вых	<ul style="list-style-type: none"> • уже используете 16-ти точечные модули • нуждаетесь в отдельно защищенных предохранителями выходах со специальной клеммной колодкой
32-х точечный модуль Вх/Вых	<ul style="list-style-type: none"> • уже используете 32-х точечные модули • хотите минимизировать число модулей • хотите минимизировать место, требуемое для шасси Вх/Вых • хотите минимизировать цену за одну точку Вх/Вых

Размещение модулей Вх/Вых в шасси

Месторасположение модулей Вх/Вых зависит от электрических характеристик модуля. Размещение выполняется слева направо, с левой ближайшей к процессору PLC-5 или адаптеру Вх/Вых позиции в шасси. Порядок размещения нижеследующий:



Размещайте блок-трансфер модули в соответствии с этими указаниями:

- размещайте наибольшее количество модулей, для которых вам нужны быстрые блок-трансферы, в локальном шасси процессора;
- размещайте модули, для которых быстрые блок-трансферы не столь критичны, в удаленном шасси Вх/Вых;
- выходные модули переменного тока всегда должны быть самыми дальними модулями Вх/Вых от любого блок-трансфер модуля в том же шасси.

Размещайте входные и выходные модули в соответствии с этими указаниями:

- слева направо;
- от низкого напряжения к высокому.

Для оптимального быстродействия при использовании дискретных Вх/Вых, следуйте нижеследующей схеме приоритетов расположения модулей:

1. шасси процессора;
2. расширенное локальное шасси Вх/Вых;
3. удаленное шасси Вх/Вых.

Для заметок _____

Установка аппаратуры

Назначение этой главы

Сведения	Страница
Определение условий эксплуатации	3-1
Механическая защита процессора	3-3
Предотвращение повреждения статическим электричеством	3-3
Разметка прокладки кабелей	3-3
Компоновка расположения шасси	3-5
Заземление систем	3-6

Определение надлежащих условий эксплуатации

Поместите процессор в условия, которые удовлетворяют следующим характеристикам :

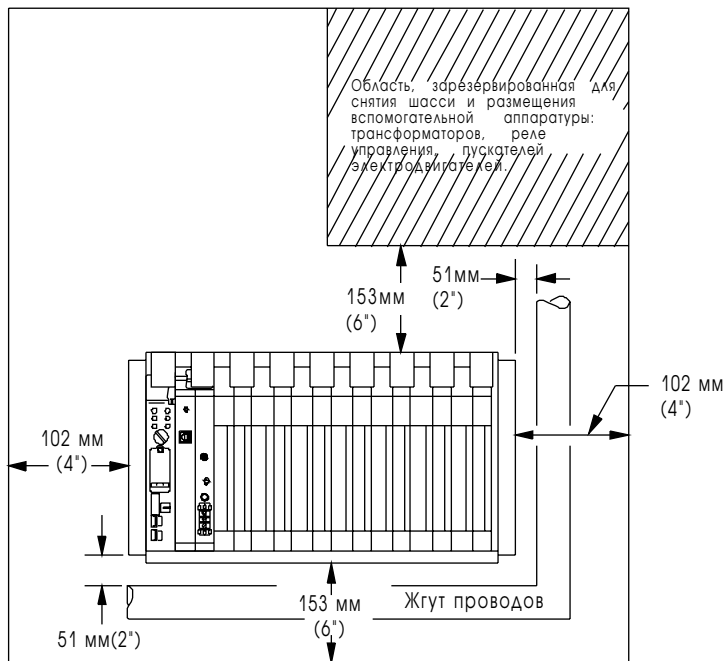
Условия окружающей среды	Допустимый диапазон
Эксплуатационная температура	0...60 °C
Температура хранения	-40...85 °C
Относительная влажность	5...95 % (без конденсата)

Разместите ваш программируемый контроллер на определенном расстоянии от другого оборудования и стен для обеспечения конвекционного охлаждения. Конвекционное охлаждение вытягивает вертикальный поток воздуха через процессор. Температура этого охлаждающего воздуха не должна превышать 60°C в любой точке непосредственно под процессором. Если температура воздуха превышает 60°C, следует устанавливать вентиляторы, обеспечивающие обдув оборудования фильтрованным воздухом извне или обеспечивающими рециркуляцию внутреннего воздуха. Можно использовать, также, кондиционеры или теплообменники.

Чтобы обеспечить требуемое конвекционное охлаждение процессорного шасси и шасси удаленных Вх\Вых (remote I\O), следуйте данному руководству.

Минимальные требования по размещению процессорного шасси:

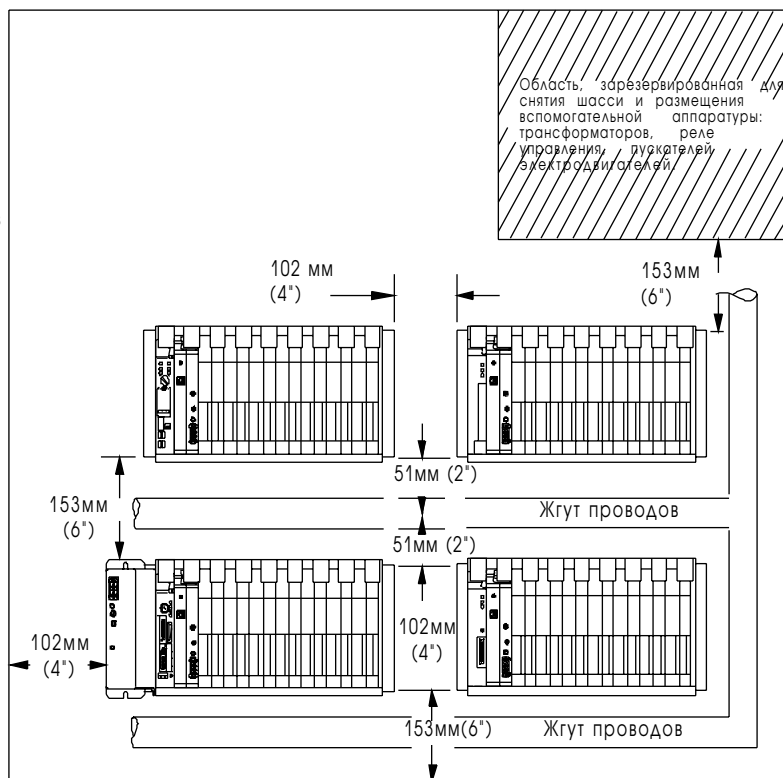
- Устанавливать шасси следует горизонтально.
- Минимальное расстояние сверху и снизу – 153 мм (6").
- Минимальное расстояние слева и справа – 102 мм (4").
- Минимальное расстояние до проводов или клеммников, с любой стороны – 51 мм (2").
- Следует оставлять дополнительное пространство над корпусом, в месте, где температура будет наивысшей.



13081

Минимальные требования по размещению удаленных шасси и локальных расширенных шасси Вх/Вых:

- Устанавливать шасси следует горизонтально.
- Минимальное расстояние сверху и снизу любого шасси – 153 мм (6").
При использовании более одного шасси в шкафу минимальное расстояние между любыми двумя шасси по вертикали – 153 мм (6").
- Минимальное расстояние слева и справа каждого шасси – 102 мм (4").
При использовании более одного шасси в шкафу минимальное расстояние между любыми двумя шасси по горизонтали – 153 мм (6").
- Минимальное расстояние каждого шасси до проводов или клеммников, с любой стороны – 51 мм (2").
- Следует оставлять дополнительное пространство над корпусом, в месте, где температура будет наивысшей.



18749

Обеспечение механической защиты процессора

Для защиты процессора ему необходим дополнительный защитный корпус. Он защитит процессор от атмосферных загрязнений, таких как масло, влага, пыль, вредные пары или другие опасные содержащиеся в воздухе вещества. Для того чтобы защитить процессор от электромагнитных и радиочастотных волн, мы рекомендуем использовать металлический корпус.

Установите корпус в месте, удобном для полного открывания дверей. Вам необходим легкий доступ к электропроводке процессора и дополнительным компонентам, для быстрого и своевременного устранения неполадок.

При выборе размера корпуса предусматривайте дополнительное место для трансформаторов, элементов электрической защиты, разъединительного выключателя, пускателя и клеммников.

Предотвращение повреждения статическим электричеством



ВНИМАНИЕ : При некоторых условиях электростатический разряд может ухудшить работу или повредить процессорный модуль. Прочтите и соблюдайте следующие меры предосторожности, для защиты от электростатических повреждений.

- При обслуживании процессорного модуля следует применять аттестованный заземляющий браслет.
- Перед обслуживанием процессорного модуля прикоснитесь к заземленному предмету, чтобы снять с себя заряд.
- Не трогайте контакты разъемов на задней стороне процессора.

Разметка прокладки кабелей

Трассировка системы зависит от расположения модулей Вх/Вых в шасси Вх/Вых. Поэтому перед трассировкой проводов следует определить размещение Вх/Вых модулей. При размещении Вх/Вых модулей разделяйте модули на группы с одинаковыми категориями проводников, указанными в документации на каждый модуль Вх/Вых. Настоящие указания соответствуют руководству по “установке электрического оборудования с целью минимизации электрических помех на входы контроллера от внешних источников”, опубликованному в стандарте IEEE 518-1982.

Разработка плана кабельных соединений выполняется в следующей последовательности:

- распределение проводов по категориям
- прокладка маршрута соединений.

Распределение проводов по категориям

Разделите все провода и кабели по категориям, как описано в “Руководстве по прокладке кабелей и заземлению в системах промышленной автоматизации”, публикация 1770-4.1.

Для классификации модулей пользуйтесь документацией по установке каждого модуля Вх/Вых.

Трассировка соединений

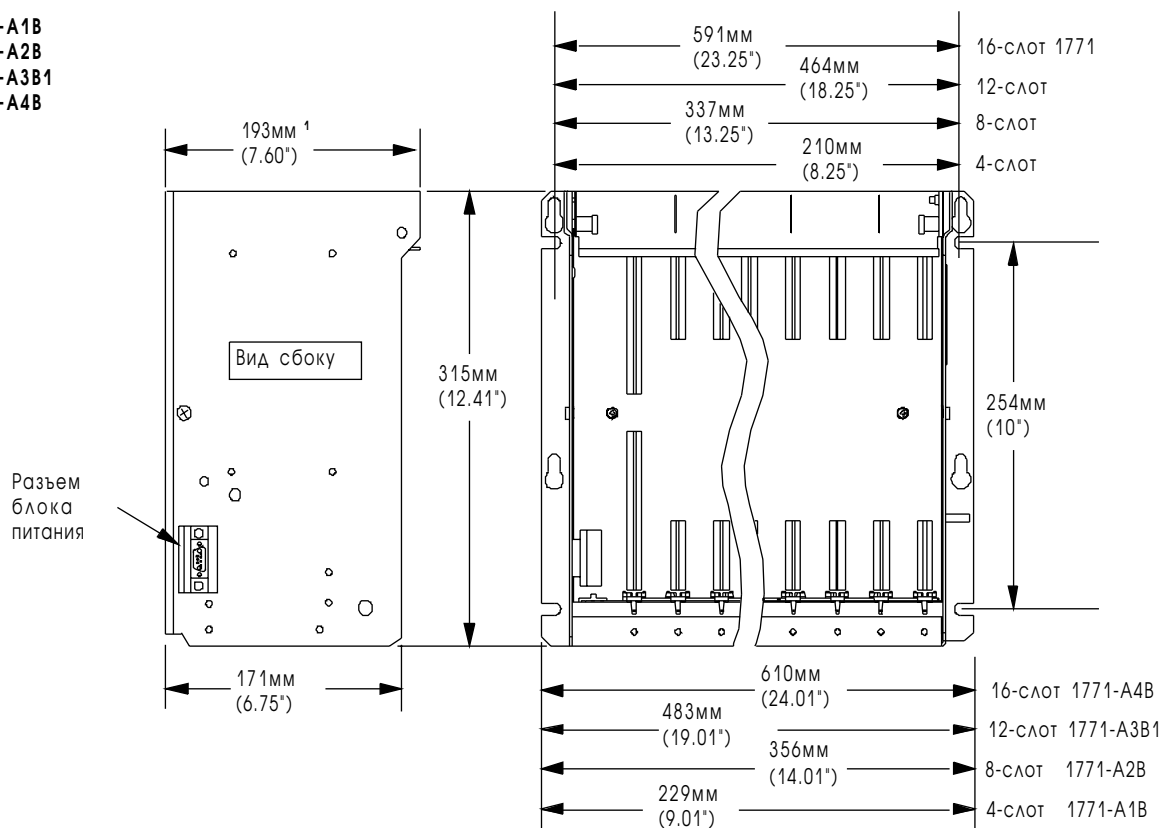
Для защиты от передачи помех от одного проводника к другому, следуйте общим указаниям по прокладке кабелей описанным в “Руководстве по прокладке кабелей и заземлению в системах промышленной автоматизации”, публикация 1770-4.1. Вы должны следовать, также, руководящим указаниям по защитному заземлению и прокладке кабелей Национального Электротехнического Кодекса (NEC, опубликованного Национальной Ассоциацией Противопожарной Защиты, Куинс, штат Массачусетс) и местным правилам устройства электроустановок.

Компоновка расположения шасси

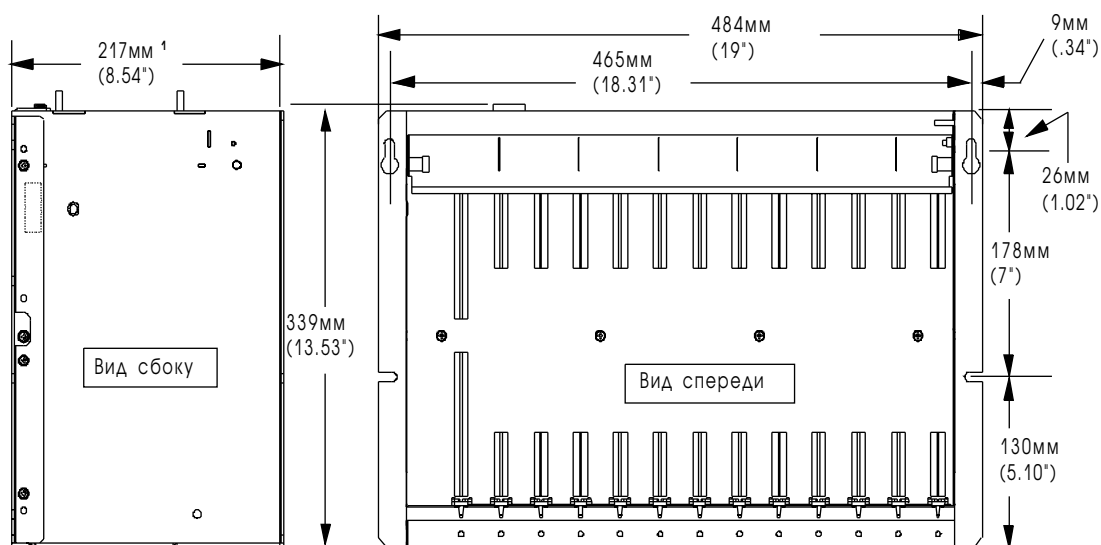
Для установки шасси Вх/Вых на задней панели шкафа следует применять болты диаметром 6 мм.

Рис. 3.1
Размеры шасси (Серии В)

1771-A1B
1771-A2B
1771-A3B1
1771-A4B

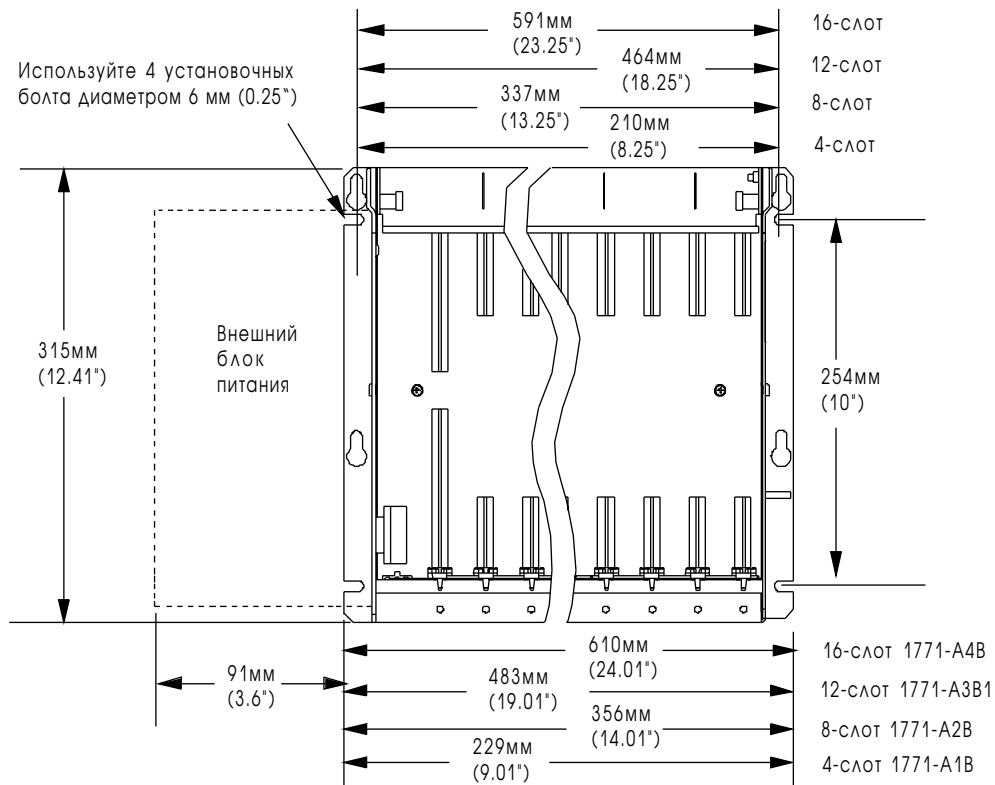


1771-A3B



¹ Полная максимальная глубина зависит от используемых разъемов и величины проводников.

Рис. 3.2
Размеры шасси Вх/Вых (I/O Chassis) и блока питания



Глубина шасси для модулей на 8 Вх/Вых составляет 204 мм (8") .

12451-1

Заземление систем

Для нижеследующей конфигурации заземления:

Смотри:

Заземление удаленных шасси Вх/вых

[Фиг. 3.3](#)

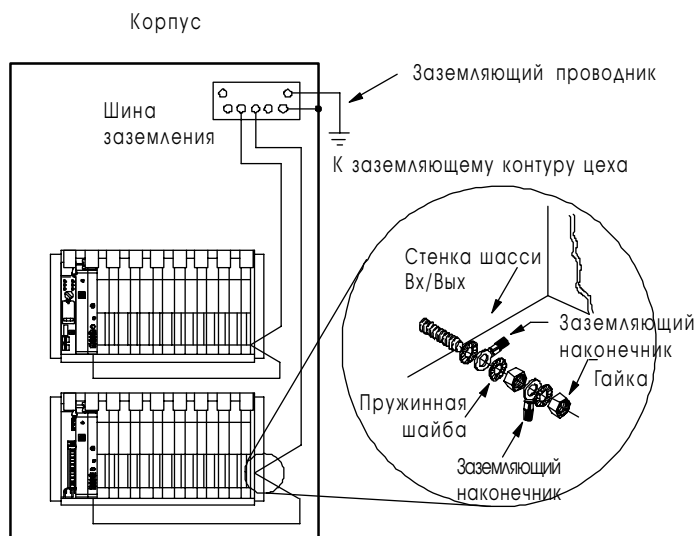
Заземление локальных расширенных Вх/Вых

[Фиг. 3.4](#)



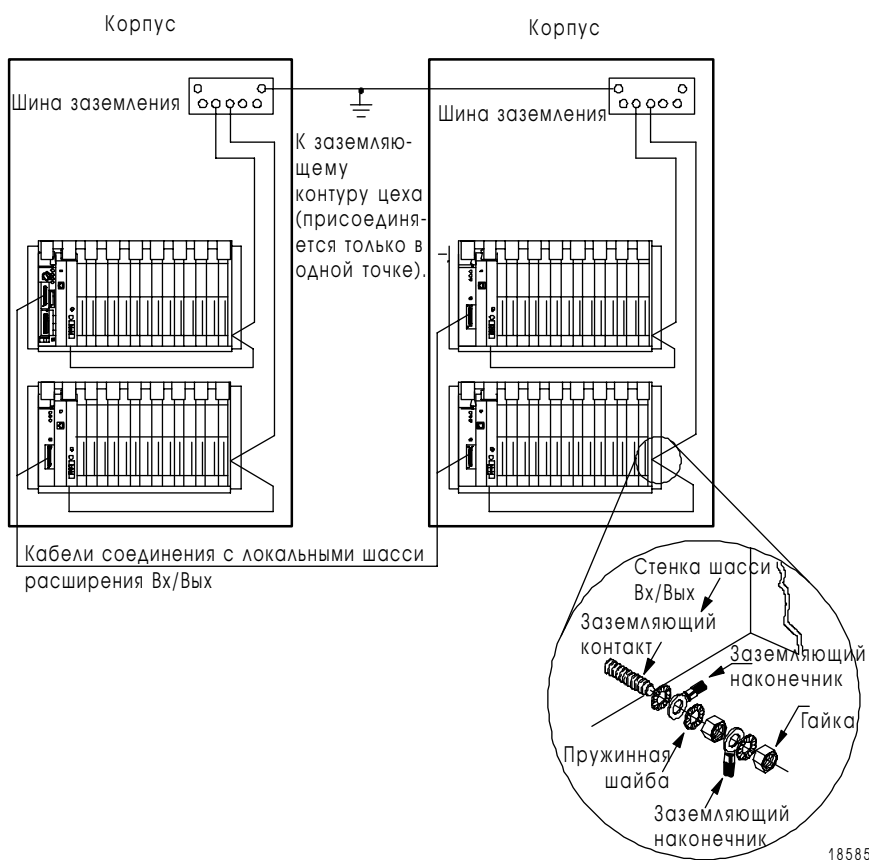
Дополнительная информация по заземлению систем дана в "Руководстве по прокладке кабелей и заземлению в системах промышленной автоматизации", публикация 1770-4.1.

Рис. 3.3
Рекомендуемая схема заземления для удаленных систем Вх/Вых



15561

Рис. 3.4
Требуемая конфигурация заземления для локальных расширенных систем Вх/Вых (Extended Local I/O Systems)



18585

Для заметок

Память процессора и адресация входов-выходов

Назначение этой главы

Сведения	Страница
Концепция адресации Вх/Вых	4-1
Выбор режима адресации	4-4
Адресация в модулях блок-трансферов	4-8
Основы адресации	4-8
Назначение рэков	4-9
Понятие памяти процессора PLC-5	4-10
Способы адресации	4-15
Эффективное использование памяти Вх/Вых	4-22

Концепция адресации Вх/Вых

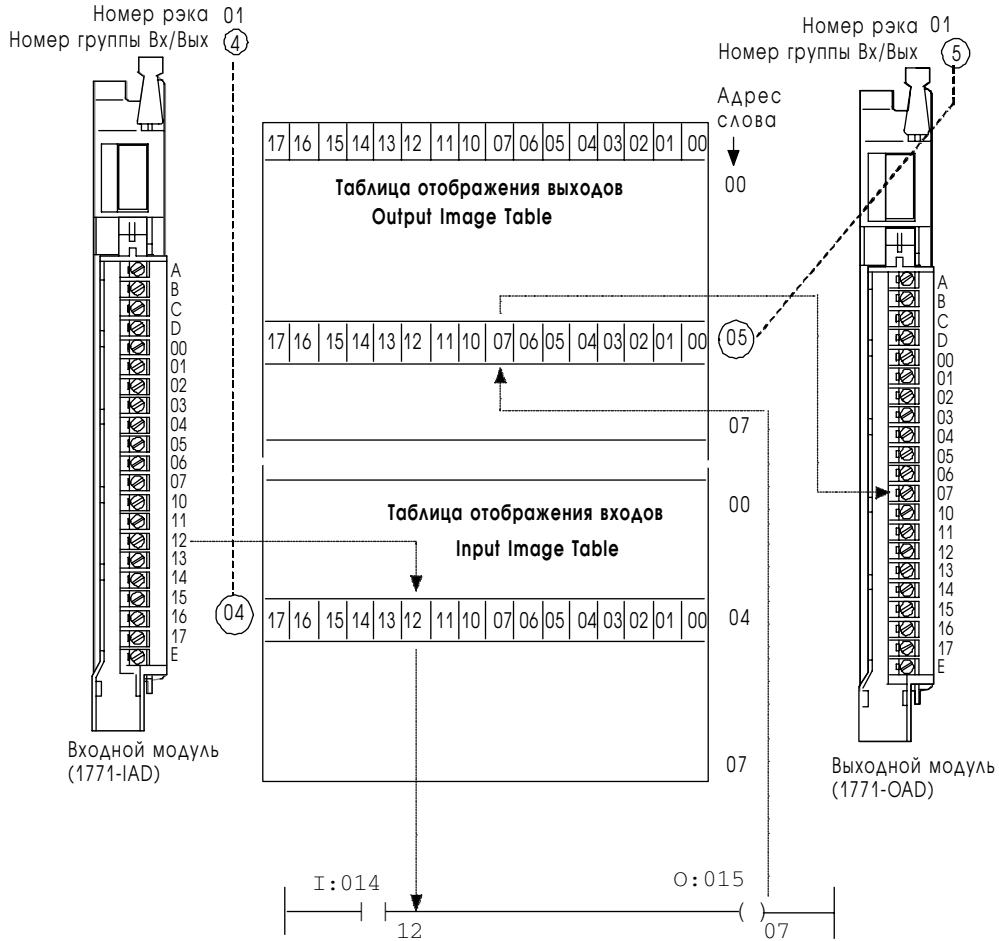
Основной целью применения программируемого контроллера является слежение за состоянием входов и управление внешними устройствами, такими как контактные датчики, клапаны и терморпары - поэтому, для адресации этих входов и выходов, они должны занимать место в памяти процессора. Каждый контакт входного или выходного модуля, соединенного с внешними устройствами занимает в памяти процессора соответствующий бит. Часть памяти процессора, содержащая адреса входов/выходов, является **таблицей отображения входов** (input image table) и **таблицей отображения выходов** (output image table).

Адресация Вх/Вых помогает соединить физическое расположение контакта с битовым расположением в памяти процессора. Адресация Вх/Вых является лишь средством разбиения памяти процессора. Различают следующие понятия:

Понятие	Термин	Отношение к памяти процессора
Отдельный контакт на модуле Вх/Вых, занимающий место в памяти процессора.	контакт или точка	Емкость модуля Вх/вых, например, модули на 8-, 16-, 32 точки непосредственно определяют объем памяти в битах, которую модуль занимает в памяти процессора. Например, модуль на 16 точек занимает 16 бит в таблице отображения входов процессора.
Клеммник Вх/Вых, занимающий 1 слово в таблице отображения входов и 1 слово в таблице отображения выходов процессора.	группа Вх/Вых	16 входных бит = 1 слово в таблице входов процессора 16 выходных бит = 1 слово в таблице выходов процессора
Память процессора должна быть организована таким образом, чтобы связанные группы Вх/вых могли адресоваться как единое целое.	рэк Вх/Вых	128 входных бит и 128 выходных бит или 8 входных слов и 8 выходных слов или 8 групп Вх/Вых. Каждый PLC-5 процессор имеет определенное количество поддерживаемых рэков. Например PLC-5/30 может иметь до 8 рэков. Сам процессор занимает один рэк с номером 0, по умолчанию.

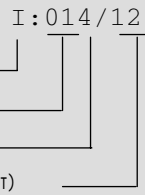
На рис. 4.1 показано соответствие между контактами Вх/Вых и их расположением в памяти процессора.

Рис. 4.1
Адресация Вх/Вых по отношению к расположению контактов Вх/Вых



Обратите внимание, как адреса, отображающие входы и выходы, связаны с оборудованием.

I для входа или O для выхода
2 цифры - номер рэка Вх/Вых
Номер группы Вх/Вых (0-7)
Номер Входа/Выхода(0-7,10-17) (бит)



Таблицы отображения входов/выходов адресуются только в восьмеричном коде

Теперь вы ознакомлены с тем, как память процессора адресует конкретный контакт Вх/Вых.

В следующей части объясняются допустимые режимы адресации. Эти режимы позволят вам определить взаимосвязи между местом, занимаемым в шасси Вх/Вых (slot) и номером группы (16 входных и 16 выходных бит).

Выбор режима адресации

Для каждого шасси Вх/Вых вы должны определить, сколько Вх/Вых слотов будут составлять Вх/Вых группу (по 1 слову в таблицах отображения входов и выходов), это называется режимом адресации. Существуют следующие режимы адресации:

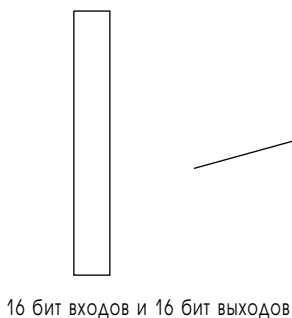
• 2-слотовая адресация

2 слота шасси Вх/Вых=1 группа Вх/Вых=1 слово Входов и 1 слово Выходов=16 входных бит и 16 выходных бит



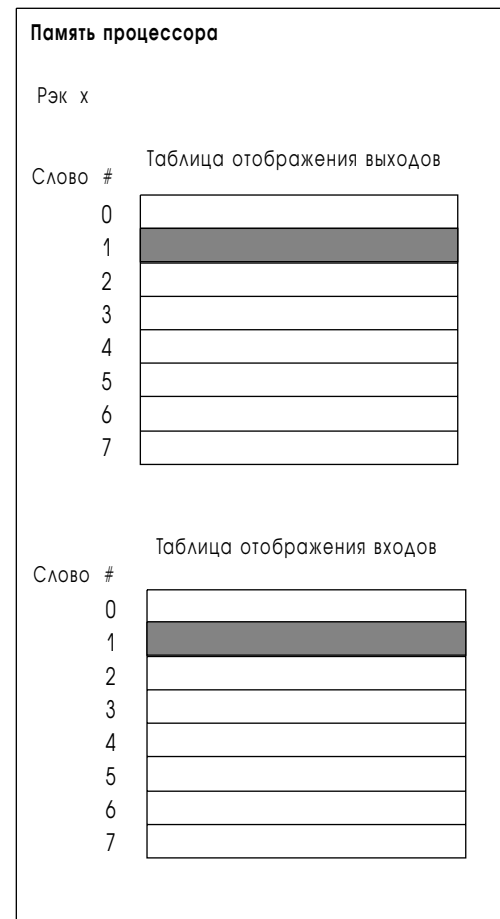
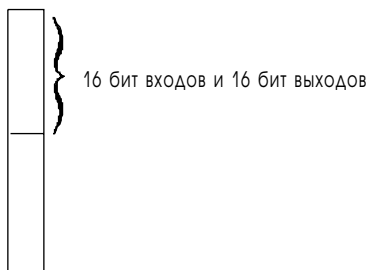
• 1-слотовая адресация

1 слот шасси Вх/Вых=1 группа Вх/Вых=1 слово Входов и 1 слово Выходов=16 входных бит и 16 выходных бит



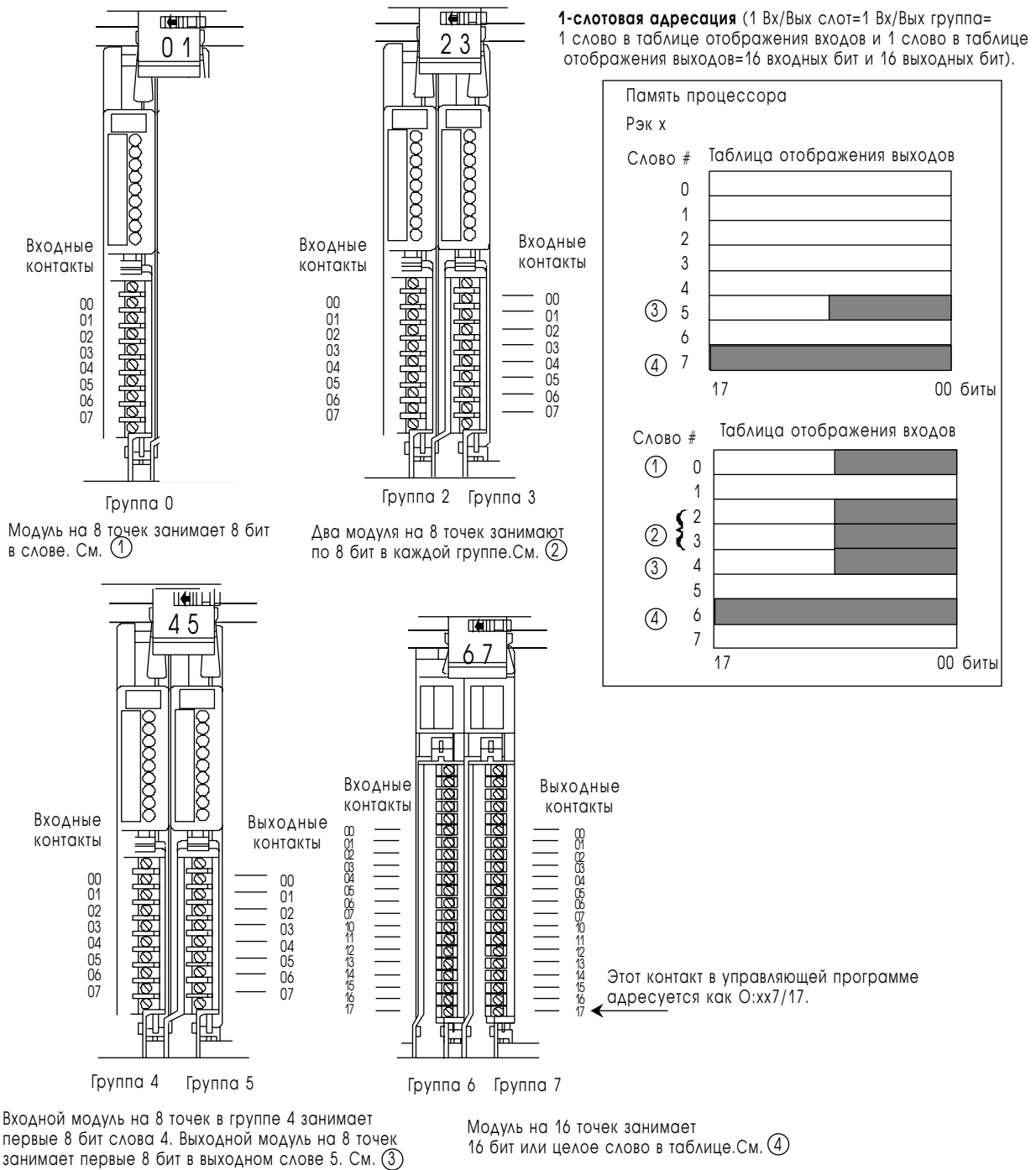
• 1/2-слотовая адресация

1/2 слота шасси Вх/Вых=1 группа Вх/Вых=1 слово Входов и 1 слово Выходов=16 входных бит и 16 выходных бит



Когда вы помещаете Вх/Вых модуль в шасси Вх/Вых емкость модуля будет определять, как быстро будут формироваться группы Вх/Вых. Например, давайте возьмем режим 1-слотовой адресации и рассмотрим, как будет заполняться память процессора модулями на 8, 16 и 32 точки.

Пример с модулями на 8 и 16 точек



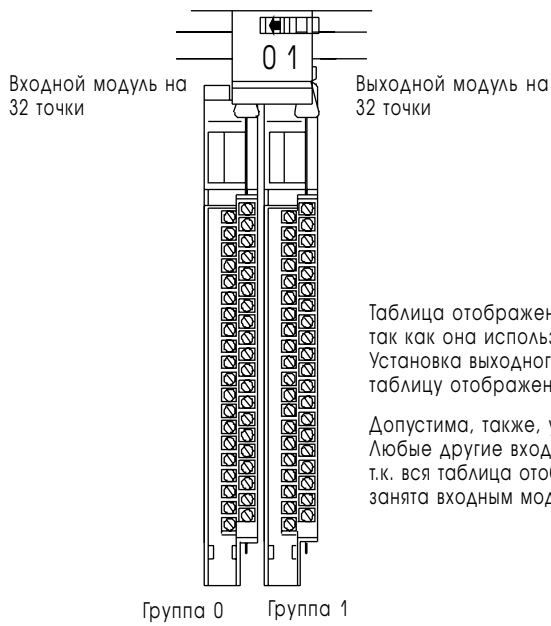
Пример для модуля на 32 точки



32-х точечные Вх/Вых модули используют все слово своей группы и заимствуют слово следующей группы. См.①

Входной модуль, находящийся в группе 0, использует входы группы 0 и группы 1, поэтому вы должны:

- в группу 1 установить выходной модуль;
- или оставить слот пустым.



1-слововая адресация (1 Вх/Вых слот=1 Вх/Вых группа= 1 слово в таблице отображения входов и 1 слово в таблице отображения выходов=16 входных бит и 16 выходных бит).

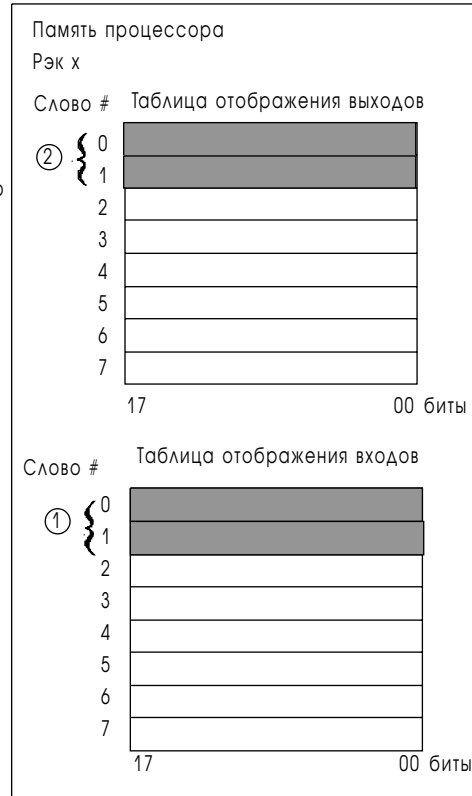
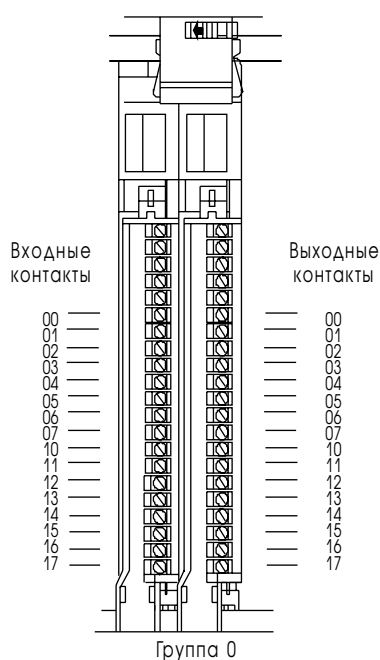


Таблица отображения входов для группы 1 недоступна, так как она используется входным модулем группы 0. Установка выходного модуля на 32 точки позволяет использовать таблицу отображения выходов для групп 0 и 1. См.②

Допустима, также, установка выходных модулей на 8 или 16 точек. Любые другие входные модули использовать нельзя, т.к. вся таблица отображения входов для групп 0 и 1 занята входным модулем группы 0.

При проектировании системы учитывайте емкость модулей и выбирайте метод адресации, наиболее полно использующий память процессора.

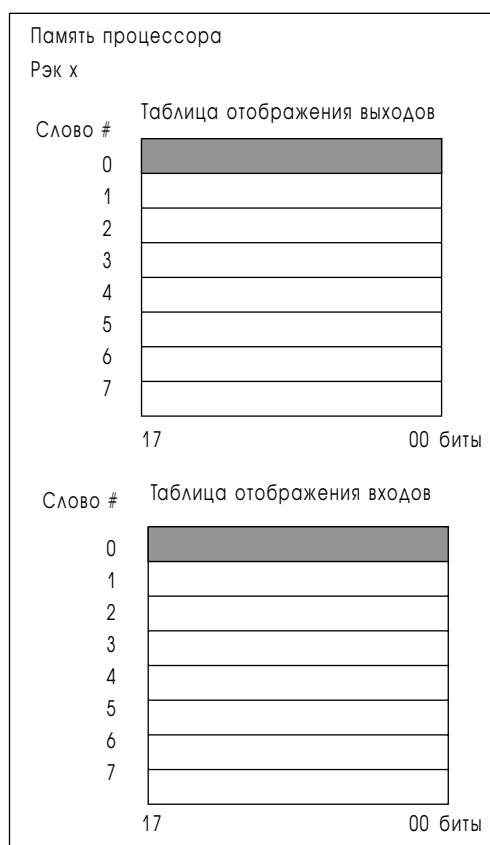
Пример эффективного использования таблицы отображения Вх/Вых



Вх/Вых модули на 16 точек занимают 16 бит или одно слово в таблице отображения.

Установленные в паре входной и выходной модуль на 16 точек эффективно используют пространство таблицы отображения.

2-слововая адресация (Вх/Вых слота=1 группа=1 слово в таблице отображения входов и 1 слово в таблице отображения выходов=16 входных бит и 16 выходных бит.



Метод адресации определяется для каждого шасси Вх/Вых установкой соответствующих переключателей. Дополнительную информацию см. в разделе 23 или в документе по быстрому запуску процессора в работу:

- Быстрый запуск усовершенствованного программируемого контроллера PLC-5 (Enhanced PLC-5 Programmable Controller Quick Start), публикация 1785-10.4.
- Быстрый запуск программируемого контроллера Ethernet PLC-5 (Ethernet PLC-5 Programmable Controller Quick Start), публикация 1785-10.5.

Адресация в модулях блок-трансферов

Модули с блочной передачей информации занимают 8 бит в таблице отображения Вх/Вых процессора. Передача информации в таких модулях производится в обоих направлениях, поэтому они не могут использоваться для дополнения входных или выходных модулей.

Для адресации:	Используйте:
Однослотовых модулей	назначенный номер рэка Вх/Вых и номер группы для слота, где размещается модуль и 0 для номера модуля. При использовании 1/2-слотовой адресации используйте назначенный номер рэка и младший номер группы и 0 для номера модуля.
Двухслотовых модулей	назначенный номер рэка и младший номер группы и 0 для номера модуля.

Основы адресации

Используйте приведенную таблицу для получения быстрой справки.

Метод адресации	Указания по использованию
2-слотовая	<ul style="list-style-type: none"> • Два Вх/Вых слота = 1 группа. • Каждая 2-слотовая группа соответствует 1 слову (16 бит) в таблице отображения входов и 1 слову (16 бит) в таблице отображения выходов. • При использовании модулей на 16 точек Вх/Вых требуется установка пары из входного и выходного модуля в группе; если в слоте 0 вы используете входной модуль, в слот 1 вы должны установить выходной модуль (или он должен остаться свободным). Данная конфигурация позволяет максимально использовать входы/выходы. • Вы не можете использовать модули блок-трансферов и 16-точечные модули в одной группе, т.к. модули блок-трансферов используют 8 бит в таблицах входов и выходов. Поэтому 8 бит 16-точечного модуля будут конфликтовать с модулем блок-трансфера. • Вы не можете использовать 32-точечные модули. • Один рэк объединяет 8 групп Вх/Вых.
1-слотовая	<ul style="list-style-type: none"> • Один Вх/Вых слот = 1 группа. • Каждый физический слот шасси соответствует одному слову (16 бит) в таблице отображения входов и одному слову (16 бит) в таблице отображения выходов. • В случае использования модулей на 32 точки, модули следует устанавливать парами, входному модулю должен соответствовать выходной в каждой четной/нечетной паре Вх/Вых группы; при использовании входного модуля в слоте 0, в слот 1 следует установить выходной (или оставить его свободным). Эта конфигурация дает максимальное использование Вх/Вых. • В одном шасси Вх/Вых допустимо использование любой комбинации модулей на 8 и 16 точек, модулей блок-трансферов или интеллектуальных модулей. Использование 8-точечных модулей приводит к уменьшению общего числа Вх/Вых. • Один рэк объединяет 8 групп Вх/Вых.
1/2-слотовая	<ul style="list-style-type: none"> • Половина слота Вх/Вых = 1 группа. • Каждый физический слот шасси соответствует 2 словам (32 бита) в таблице отображения входов и 2 словам в таблице отображения выходов. • Допустимо использование любой комбинации модулей на 8, 16 и 32 точки, модулей с блочной передачей информации и интеллектуальных модулей. Модули на 8 и 16 точек Вх/Вых приводят к некоторому уменьшению общего числа Вх/Вых. • В процессорно-резидентном локальном рэке, невозможно форсировать входные биты старшего слова любого слота, оставленного свободным, или слота, использующего 8- или 16-точечные модули. Например, если на первом месте локального рэка установлен 8- или 16-точечный модуль Вх/Вых (слова 0 и 1 таблицы отображения Вх/Вых при 1/2-слотовой адресации), вы не сможете форсировать биты слова 1 (1:001) в состоянии Вкл. или Выкл. • Один рэк объединяет 8 групп Вх/Вых.

Назначение рэков

Число рэков в шасси зависит от размеров шасси и выбранного метода адресации.

При использовании шасси следующего размера:	Размер шасси в рэках при 2-слотовой адресации	Размер шасси в рэках при 1-слотовой адресации	Размер шасси в рэках при 1/2-слотовой адресации
4-слотовое	1/4	1/2	1
8-слотовое	1/2	1	2
12-слотовое	3/4	1-1/2	3
16-слотовое	1	2	4

Совет

При назначении номеров рэкам пользуйтесь следующими рекомендациями:

- объем одного рэка составляет 8 групп Вх/Вых, независимо от применяемого метода адресации;
- в процессоро-резидентном локальном шасси вы можете назначить **от одного до четырех рэков (128 входов и 128 выходов)** в зависимости от размера шасси и метода адресации. Недопустимо разделять процессоро-резидентный локальный рэк Вх/Вых между двумя или большим числом шасси и назначать неиспользуемые процессоро-резидентные группы Вх/Вых удаленным рэкам Вх/Вых;
- адрес резидентного локального рэка процессора, по-умолчанию, есть 0;
- невозможно разделить рэки между удаленными и локальными расширенными Вх/Вых связями. Например, если 8-слотовое локальное шасси расширения Вх/Вых сконфигурировано как группы Вх/Вых 0-3 рэка 2, то 8-слотовое удаленное шасси Вх/Вых не может быть сконфигурировано как группы 4-7 рэка 2. Дополнительную информацию об адресации локальных расширенных шасси Вх/Вых см. в разделе 8;
- при использовании комплементарной адресации Вх/Вых, при группировке рэков рассматривайте адреса комплементарного рэка индивидуально; первичные номера рэков отделяйте от комплементарных;
- в случае неиспользования функции автоконфигурации, в экране конфигурации вашего программного обеспечения группируйте вместе 1/4 и 1/2 части каждого логического рэка. Не смешивайте их с другими логическими номерами рэков. Например, для определения рэков ваш пакет программирования имеет окно со следующей информацией:

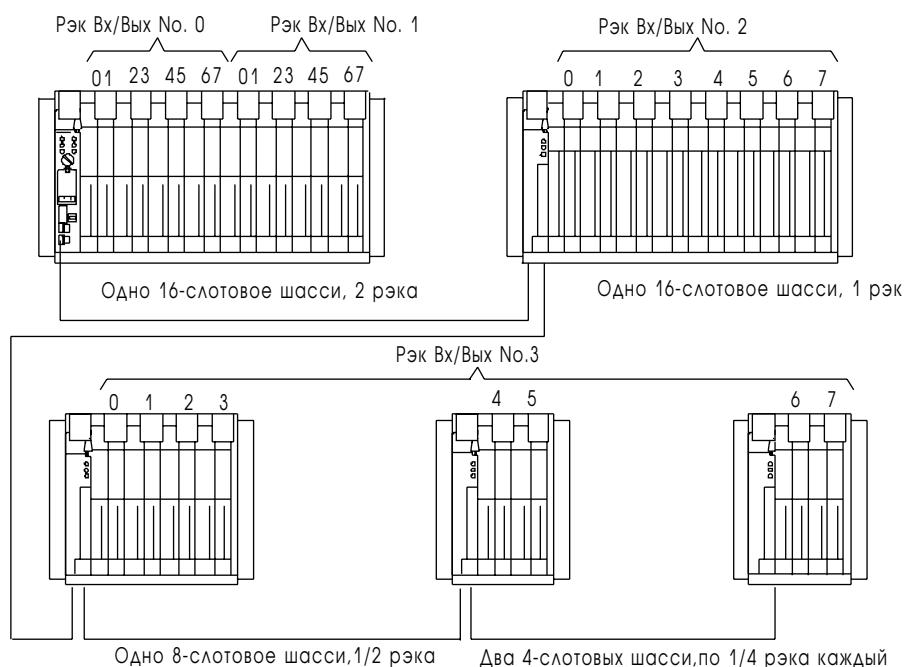
Группируйте вместе
1/4 и 1/2 рэков

Rack Address	Starting Group	Rack Size	Range	Fault	Inhibit	Reset	Retry
1	0	1/4	010-011		1	0	0
1	2	1/4	012-013		0	0	0
1	4	1/4	014-015		0	0	0
2	0	1/4	020-021		0	0	0
2	2	1/4	022-023		0	0	0
2	4	1/2	024-027		0	0	0
3	0	1/4	030-031		0	0	0
17	0	Full	170-177		0	0	0

Совет

При назначении номеров рэкам удаленных Вх/Вых пользуйтесь следующими указаниями:

- один удаленный канал сканера Вх/Вых может поддерживать до 32 устройств, но только 16 рэков. Дополнительную информацию см. в главе 6;
- максимальное количество удаленных Вх/Вых рэков зависит от используемого типа процессора PLC-5;
- процессор PLC-5 и 1771-ASB модуль адаптера автоматически определяет следующий больший номер(а) рэка(ов) остающимся Вх/Вых группам шасси. Например, при 1/2-слотовой адресации в вашем локальном шасси процессора (16-местное шасси типа 1771-A4B), процессор назначит этому шасси рэки 0,1,2 и 3;
- вы можете назначить удаленный рэк Вх/Вых части шасси, одному или нескольким шасси.



16466

Понятие памяти процессора PLC-5

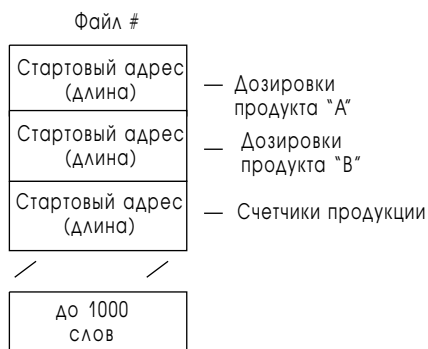
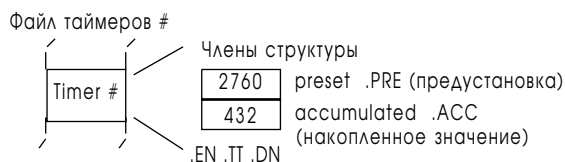
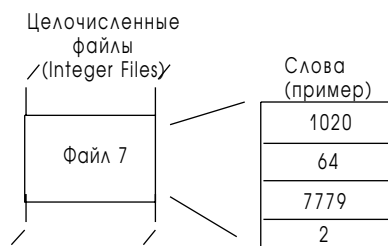
Память процессора PLC-5 подразделяется на две основные части:

Область памяти	Описание
Данные	<p>Все данные, которые процессор проверяет или изменяет в файлах, в областях памяти, служащих для хранения данных:</p> <ul style="list-style-type: none"> • данные полученные от входных модулей; • данные, которые должны быть посланы в выходные модули, или решения сделанные логикой; • промежуточные результаты вычислений; • постоянные данные, такие, как настройки или рецепты; • управляющие данные; • системный статус.
Файлы программ	<p>Создаваемые файлы для программирования логики, в зависимости от используемого метода: релейно-контактная логика, последовательные функциональные схемы и/или структурированный текст. Эти файлы содержат инструкции по проверке входов и выходов и возвращают результаты.</p>

Понятие области данных (Файлов данных)

Процессор разделяет области данных на:

- **типы**, позволяющие определять различные форматы и диапазоны данных,
Дополнительную информацию о различных типах файлов данных см. в табл.4.A на стр. 4-13.
- вы можете создать множество **файлов** данного типа. Файлы позволят вам сгруппировать и организовать логически связанные данные. При необходимости доступа к данным вы указываете файл, в котором находятся данные;
- некоторые типы файлов состоят из 16-битных **слов**. Значения с плавающей точкой представляются 32 битами. При необходимости доступа к данным вы указываете их форматированный адрес;
- каждое слово состоит из **бит**, который является наименьшей мерой информации. Бит имеет значение «ноль» или «единица». При необходимости доступа к данным вы указываете их форматированный адрес.
- Некоторые типы файлов, подразделены на **структуры**, используемые для управления элементами инструкций. Структуры, в свою очередь, разделены на **члены**, на уровне отдельных бит или слов. При необходимости доступа к данным вы указываете их форматированный адрес.



Для организации логически связанных данных информацию в файлах можно разбить на **блоки данных**. При необходимости доступа к таким данным вы указываете только стартовый адрес внутри файла и длину, вместо указания индивидуального адреса.

При организации данных группируйте данные по принципу единообразности, например:

- результаты вычислений;
- рецепты дозировок.

Т. к. инструкции блок-трансферов также структурированы, следует группировать такие данные как:

- входы аналоговых модулей;
- выходы на аналоговые модули.

Совет

Вероятно, что при организации данных вы захотите оставить место для дальнейшего развития. Делайте это оставлением незанятых интервалов между:

- блоками данных внутри файла;
- группами последовательно пронумерованных файлов;
- модулями в шасси Вх/Вых.

Важно: Если вы планируете работать ONLINE, вы должны оставлять промежутки между файлами данных или в системе нумерации программных файлов (т.к. создание пользовательской памяти в режиме RUN невозможно). Эти «пустоты», тем не менее, занимают 6 слов дополнительной памяти для каждого пропускаемого блока памяти. Планирование свободной памяти следует производить с осторожностью.

Совет

При организации файлов данных руководствуйтесь следующими указаниями:

- группируйте большие объемы связанных данных в файлы;
- адресуйте файлы данных в пределах номеров 3-999,
(См. таблицу 4.A на стр. 4-13);
- адресуйте слова данных внутри файла данных последовательно по номерам 0-999;
- адресуйте слова таблиц отображения входов/выходов согласно конфигурации вашей системы Вх/Вых:
 - 0-37(восьмеричное) для PLC-5/11, -5/20, -5/20E;
 - 0-77(восьмеричное) для PLC-5/30;
 - 0-177(восьмеричное) для PLC-5/40, -5/40L, -5/40E;
 - 0-277(восьмеричное) для PLC-5/60, -5/60L, -5/80, -5/80E;
- при адресации таблиц отображения входов/выходов используйте восьмеричные адреса 00-07, 10-17;
- адресация бит в каждом слове производится десятичными числами 0-15.

В дополнении к данным? хранящимся в файле, каждый файл содержит дополнительно 6 служебных слов.

Адресуемые типы файлов

В таблице снизу показаны допустимые типы файлов и количество памяти в файлах разных типов.

Таблица 4.А
Типы файлов и использование памяти

Тип файла	Идентификатор типа файла	Номер файла	Максимальный размер файла в 16-бит словах и структурах				Дополнительная служебная память (в 16-бит словах)	Используемая память (в 16-бит словах) на слово, символ или структуру
			PLC-5/11, -5/20, -5/20E	PLC-5/30	PLC-5/40, -5/40E, -5/40L	PLC-5/60, -5/60L, -5/80, -5/80E		
Выходы	O	0	32	64	128	192	6	1/слово
Входы	I	1	32	64	128	192	6	1/слово
Статус	S	2	128	128	128	128	6	1/слово
Битовый (Двоичный)	B	3 ¹	1000 слов				6	1/слово
Таймер	T	4 ¹	3000 слов/1000структур				6	3/структуру
Счетчик	C	5 ¹	3000 слов/1000 структур				6	3/структуру
Управления	R	6 ¹	3000 слов/1000 структур				6	3/структуру
Целый	N	7 ¹	1000 слов				6	1/слово
С плавающей точкой	F	8 ¹	2000 слов/1000 структур				6	2/структуру
ASCII	A	3-999 ²	1000 слов				6	1/2 на символ
BCD	D	3-999 ²	1000 слов				6	1/слово
Блок-трансфер	BT	3-999 ²	6000 слов/1000 структур				6	6/структуру
Сообщений	MG	3-999 ²	32760 слов/585 структур ³				6	56/структуру
PID	PD	3-999 ²	32718 слов/399 структур ³				6	82/структуру
ПФСстатус	SC	3-999 ²	3000 слов/1000 структур				6	3/структуру
ASCII строки	ST	3-999 ²	32769 слов/780 структур ³				6	42/структуру
Не использовано	--	9-999	6				6	0

¹ Это номера файлов, назначаемые по умолчанию. Для этих типов возможно назначение любого номера от 3 до 999.

² Эти файлы содержат только число слов в соответствии с наибольшим назначенным адресом.

³ Максимальный размер одного файла составляет 32К слов.

Максимальный размер всей области данных составляет 64К слов.

Таблица 4.В
Допустимые типы данных/имеют следующие значения :

Тип данных :	Может содержать значения :
Прямой (immediate) (программные константы)	Значение между -32768 и 32767 (Константа большая, чем 1024 использует 2 слова для хранения; значения с плавающей точкой использует 3 слова).
Целочисленный	Целочисленный тип данных: целочисленный, таймер, счетчик, статус, битовый, вход, выход, ASCII, BCD, управления (например N7:0,C4:0, и т.д.).
С плавающей точкой	Формат с плавающей точкой (допустимый диапазон от $+1.175494e^{-38}$ до $+3.402823e^{+38}$).
Блочный	Блок-трансферы (например BT14:0) или целочисленные (например N7:0).
Сообщения	Формат сообщений (например MG15:0) или целочисленный тип данных (N7:0).
Строковый	Строковый формат (например ST12:0).
SFC статус	Формат файлов статуса последовательных функциональных схем (например SC17:0).

Понятие памяти программных файлов

Создавайте программные файлы, основываясь на используемом методе программирования. В приведенной внизу таблице перечислены виды программных файлов и занимаемое количество слов:

Программный файл	Используемое число слов
Релейно-контактные схемы	6/файл +1/слово
ПФС	6/файл
Структурированный текст	6/файл +1/слово

Чем больше программных файлов вы создаете, тем больше времени требуется процессору для выполнения определенной команды, например, переход в режим RUN, выполнение программирования в режиме ONLINE, сохранение программы. Некоторые инструкции, такие как JMP, LBL, FOR, NXT, также, требуют большего времени выполнения в больших по объему файлах программ.

Серия E процессоров PLC-5 поддерживают 2000 программных файлов, что позволяет разместить больше программных шагов ПФС. Это усовершенствование практически удваивает количество шагов/переходов в вашей программе ПФС.

Каждый создаваемый вами файл занимает 6 служебных слов памяти. Это, помимо любой программы внутри файла. Если вы создадите программный файл с номером 1999, то это займет 12000 слов в памяти программ и уменьшит объем памяти, доступной для кода программы.

Способы адресации

Допустимые форматы адресации см. в нижепредставленной таблице.

Если вы хотите иметь доступ к:	Используйте формат адреса:	И см. стр. :
Входной или выходной бит в таблице отображения Вх/Вых	Адрес Вх/Вых	4-15
Бит, слово, под-член, блок данных, файл, или бит Вх/Вых	Логический адрес	4-16
Компонент внутри логического адреса, заменяющий значение, которое находится по другому адресу	Косвенный адрес	4-18
Смещение адреса, указанное некоторым числом элементов	Индексный адрес	4-19
Имя, заменяющее адрес	Символический адрес	4-20



Для получения дополнительной информации о способах адресации см. документацию на используемое вами программное обеспечение.

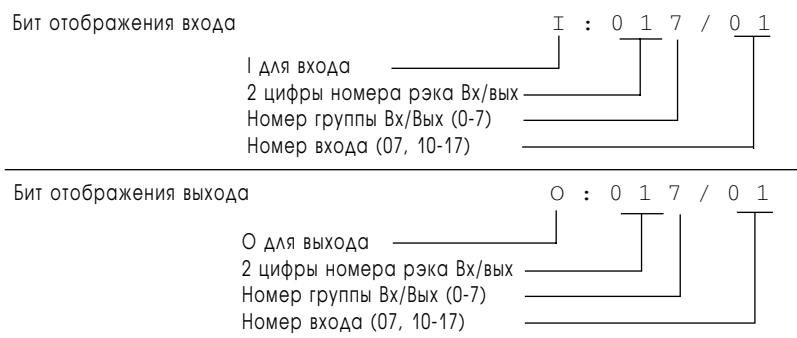
Адресация Вх/Вых в таблицах отображения

Адреса таблиц отображения входов/выходов соответствуют физическому расположению цепи Вх/Вых в шасси Вх/Вых: **a:bbc/dd**

a	Идентификатор Вх/Вых	I = устройство входа O = устройство выхода	
bb	Номер рэка Вх/Вых	PLC-5/11, -5/20, -5/20E PLC-5/30 PLC-5/40, -5/40L, -5/40L PLC-5/60, -5/60L, -5/80L, -5/80E	00-03 (восьмеричный) 00-07 (восьмеричный) 00-17 (восьмеричный) 00-17 (восьмеричный)
c	Номер группы Вх/Вых	0-7 (восьмеричный)	
dd	Номер контакта (бита)	00-17 (восьмеричный)	

Чтобы указать этот адрес:

Пример:



Указание логических адресов

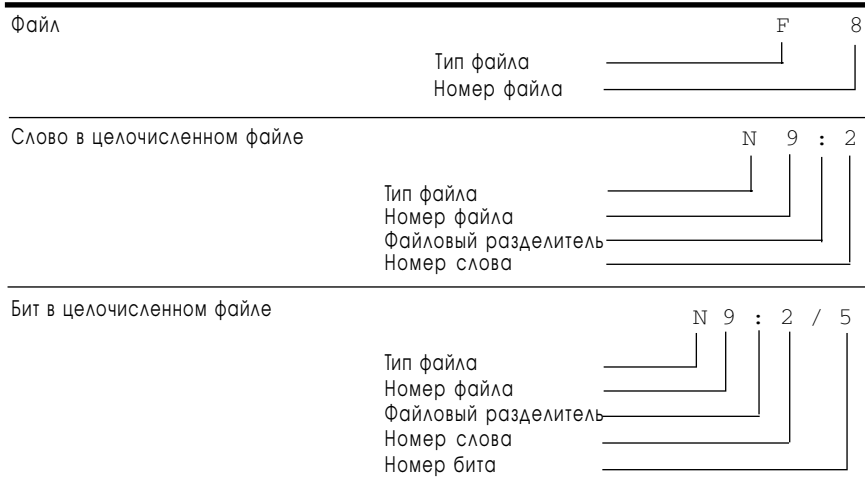
Формат логической адресации прямо соответствует расположению операндов в памяти: **# X F : e . s / b**

Где: Есть:

#	Адрес файла. Опускается для бита, слова и структуры (также указывает на индексную адресацию, см. следующую стр.)				
X	Тип файла:	B-Binary(двоичный) C-счетчик F-Floating point (с плавающей точкой) ST-строка ASCII	N-integer(целочисленный) O-output(выход) R-control(управления) I-Input(входы)	T-таймер A-ASCII(символьный) D-BCD S-status(статус)	MG-message(сообщение) PD-PID SF-статус ПФС BT-блок-трансфер (блоковой передачи)
F	номер файла	0-выходы 1-входы 3-999 - любого другого типа			
:	двоеточие или точка с запятой, разделяющие номер файла от номера структуры/слова				
e	Номер структуры/слова	0-277	восьмеричные для вход/выходных файлов		
		0-127	десятичное для статусных файлов		
		0-999	для всех типов файлов, кроме MG, PD, и ST.		
.	Разделитель «точка» используется только в обозначении членов структур в счетчиках, таймерах и файлах управления				
s	Обозначение члена структуры только в таймерах, счетчиках, файлах управления, BT, MG, PD, SC, и ST файлах				
/	Битовый разделитель, отделяющий номер бита				
b	Номер бита:	00-07 или 10-17 для файлов входа/выхода 00-15 для всех других типов 00-15,999 для двоичных файлов, в случае использования прямого битового адреса			

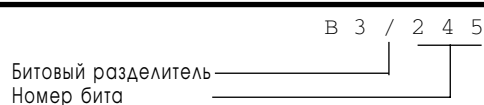
Для указания адреса:

Используйте следующие параметры:



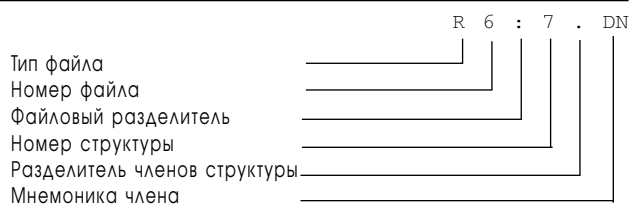
Для указания адреса:**Используйте следующие параметры:**

Бит в двоичном файле



Двоичный файл представляет собой непрерывную последовательность бит, и поэтому, может адресоваться двумя способами: указанием слова и бита или просто номером бита.

Бит в файлах структур



Вы можете использовать, также, мнемонику для адресации членов структур на уровне слов или бит. Допустимость применения мнемоники определяется типом данных (таймер, счетчик или структуры управления) и инструкцией программы. Например:

Тип инструкции	Уровень слова	Пример	Битовый уровень	Пример
Таймер	preset	.PRE T4:1.PRE	enable .EN	T4:0.EN
TON, TOF, RTO	accumulated	.ACC	timing .TT	
			done .DN	

Для получения дополнительной информации о мнемонике конкретных инструкций см. раздел 22, или PLC-5 «Программное обеспечение. Справочник по комплекту инструкций», публикация 1785-6.1.

Косвенная адресация

При косвенной адресации для формирования адреса процессор использует адрес дополнительного указателя. Используя релейно-контактную логику вы можете изменять значения в этом заменяющем значении адресе.

При формировании косвенного адреса следуйте рекомендациям:

- косвенно адресуются номер файла, номер слова, номер бита;
- адрес, заменяющий значение, должен быть типов: N, T, C, R, B, I, O, or S. Любой T, C, или R адрес должен быть адресом члена структур с длиной равной одному слову, например: T4:0.ACC;
- адрес указателя помещается в скобки [].

Пример	Переменная	Пояснение
N(N7:0):0	Номер файла	Номер файла помещен в целочисленном адресе N7:0.
N7:(C5:7.ACC)	Номер структуры	Номер слова представляет собой накопленное значение счетчика 7 в файле 5.
B3/(I:017)	Номер бита	Номер бита адресуется входным словом 17.
N(N7:0):(N9:1)	Номер файла и номер слова	Номер файла находится в целочисленном файле по адресу N7:0, а номер слова - по адресу N9:1.

Индексная адресация

Процессор начинает операции со стартового адреса плюс смещение. Величина смещения располагается в слове файла статуса процессора. В программе лестничной логики вы имеете возможность адресовать слово смещения.

Символом индексной адресации является # . В логическом адресе этот символ помещается непосредственно перед идентификатором типа файла. Введите слово смещения в слово S:24 статусного файла. Все индексные инструкции используют в качестве смещения слово S:24.

Для формирования индексного адреса следуйте указаниям:

- Значение индекса (положительное или отрицательное) не должно привести к выходу индексного адреса в файл другого типа.



ВНИМАНИЕ: Процессор не проверяет индексный адрес на предмет выхода в файл другого типа (например из N7 в F8). Вы даже можете модифицировать статусный файл, который является физически последним файлом таблицы данных. Но если индексный адрес выйдет за пределы таблицы данных, процессор инициирует ошибку выполнения и войдет в состояние основной ошибки.

- Когда инструкция использует больше, чем 2 индексных адреса, процессор использует одно и то же значение индекса для каждого из индексных адресов.
- Устанавливайте значение индексного смещения непосредственно перед использованием инструкции, в которой применяется индексный адрес.



ВНИМАНИЕ: Инструкции со знаком # в адресе манипулируют значением смещения, содержащимся в слове S:24. Перед использованием индексного адреса проконтролируйте или загрузите значение смещения. В противном случае, может возникнуть непредсказуемое состояние процесса с возможностью поломки оборудования и/или травмирования персонала.

Последующая инструкция MVM использует индексный адрес в источнике и получателе. Если значение смещения равно 10 (в слове S:24), процессор манипулирует данными следующим образом:

```

MVM
MASKED MOVE
Source      #N7:10
Mask       00110011
Destination #N11:5
  
```

Значение	Базовый адрес	Смещенный адрес
Источник	N7:10	N7:20
Получатель	N11:5	N11:15

Символическая адресация

При применении символической адресации, пользуйтесь нижеследующими рекомендациями:

- начинайте имя переменной с буквы (не с числа);
- имя может содержать до 10 символов, допустимыми являются:
 - A-Z (в верхнем или нижнем регистре)
 - 0-9
 - знак подчеркивания (_);
- допустимо применять символическую адресацию для слова или бита.

Важно: Символы - это особенность пакета программирования, а не процессора, они сохраняются в базе данных на жестком диске используемого программного терминала. При использовании другого компьютера вы не будете иметь доступа к базе данных символов.

Пример	Логический адрес	Символический адрес
Входы (биты)	I:015/00	LS1
	I:015/03	AUTO1
	I:015/06	SW1
Выход (биты)	O:013/00	M1
	O:013/02	CL1
	O:013/04	L1
Слово	F10:0	Calc_1
	F10:1	Calc_2

Совет**Оптимизация времени выполнения инструкций и процессорной памяти**

Для достижения наибольшего быстродействия, помещайте наиболее часто используемые адреса, в соответствии со следующими рекомендациями:

- Адресуйте битовые инструкции между концом таблицы отображения входов и физическим словом 256. Т.к. адреса слов с большим адресом требуют дополнительного слова в процессорной памяти и выполняются на 0.16 мкс медленнее, чем битовые адреса, помещенные в слова 0-255.
- Адресуйте элементарные инструкции в области от конца таблицы отображения входов до адреса 2048. Т.к. адресация за пределами этого диапазона требует для хранения дополнительного пространства в памяти процессора.

	PLC-5/11, -5/20, -5/20E Физическое слово #	PLC-5/30 Физическое слово #	PLC-5/40, -5/40L, -5/40E Физическое слово #	PLC-5/60, -5/60L, -5/80, -5/80E Физическое слово #	Тип файла	# файла по умолчанию
Наиболее часто исполь- зуемые битовые адреса	031	063	0127	0191	Таблица выходов ¹	0
	3263	32127	32255	32383	Таблица входов ¹	1
Наиболее часто исполь- зуемые элементарные адреса	----- слово 256				Двоичный, таймер, счетчик, управления, целочислен- ный, с плавающей точкой блоковой передачи, сообщений, PID, статуса SFC, ASCII строк	3-999 в соответствии с вашим проектом
	----- слово 2048					
					статус ²	2

¹ Минимальный размер файла 32 слова

² Файл статуса является всегда физически последним в памяти данных.

Следующий пример иллюстрирует вышесказанное.

Пример битовой адресации

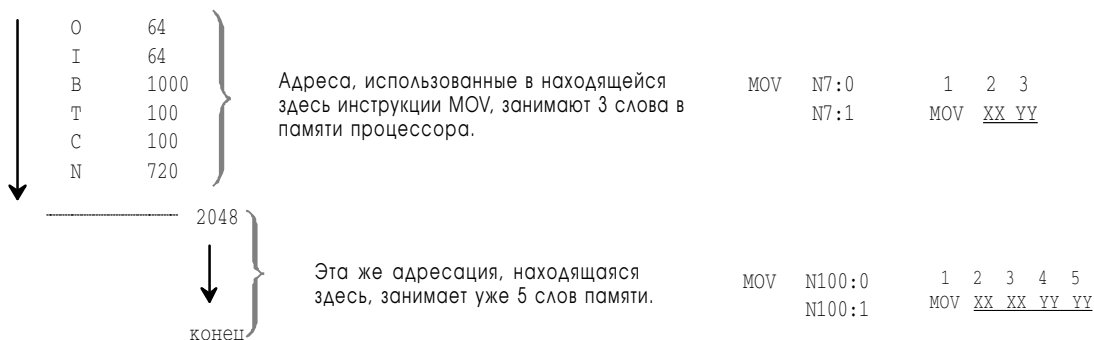
Если ваша область данных выглядит, как здесь:

↓	O	32	}	Использованный в инструкции OTE адрес находящийся здесь:	• занимает одно слово в памяти процессора • выполняется за время 0.48 мкс	1 OTE	
	I	32					
	B	64					
	T	32					
	C	32					
	R	32					
	N	32	}	Этот же адрес, находящийся здесь:	• занимает 2 слова в памяти • выполняется за время 0.64 мкс	1 2 OTE XX	
	----- 256						
			}				↓
							конец

Приведенный пример использует характеристики инструкций по времени исполнения и объему памяти из раздела 22. В этом же разделе приведены характеристики и для других инструкций.

Пример адресации в элементарных инструкциях

Если ваша область данных выглядит, как здесь:



Приведенный пример использует характеристики инструкций по времени исполнения и объему памяти из раздела 22. В этом-же разделе приведены характеристики и для других инструкций.

Эффективное использование памяти Вх/Вых

Процессор PLC-5 автоматически назначает входам и выходам их отображение в памяти. В основном модули используют либо входные, либо выходные цепи. Для более эффективного использования памяти ввода/вывода используйте следующие методы установки модулей Вх/Вых.

Используйте:	В применениях:
2-слота	Устанавливайте Вх/Вых модули на 16 точек в паре в одной группе. Например, если входной модуль устанавливаете в слот 0, выходной поместите в слот 1.
1-слот	Устанавливайте Вх/Вых модули на 32 точки в паре в одной группе. Например, если входной модуль устанавливаете в слот 0, выходной поместите в слот 1.
комплементарные шасси Вх/Вых	<p>Конфигурируйте комплементарные шасси как основное и дополнительное шасси в паре. Модули Вх/Вых в комплементарном шасси выполняют противоположную функцию, по отношению к модулям, установленным в основном шасси.</p> <p>Назначив канал сканера PLC-5 как комплементарный, вы сможете дополнить рэки 1-7. Канал, сконфигурированный для комплементарных Вх/Вых не может сканировать рэки по номеру больше 7. Типы процессоров PLC-5, способные адресовать рэки больше 7, могут адресовать эти рэки на другом канале сканера, которые не были сконфигурированы как комплементарные. Устройства удаленного ввода/вывода (такие, как адаптер 1771-ASB), также, должны быть сконфигурированы для комплементарности.</p> <p>Для получения дополнительной информации см. PLC-5 Reference Guide: Конфигурирование комплементарных Вх/Вых для процессоров PLC-5, публикация 1785-6.8.3</p>

Коммуникации с резидентными Вх/Вых процессора

Назначение этой главы

Сведения	Страница
Введение в сканирование процессора PLC-5	5-1
Сканирование программы	5-2
Передача данных к резидентным Вх/Вых процессора	5-3
Конфигурирование системы для резидентных Вх/Вых процессора	5-4

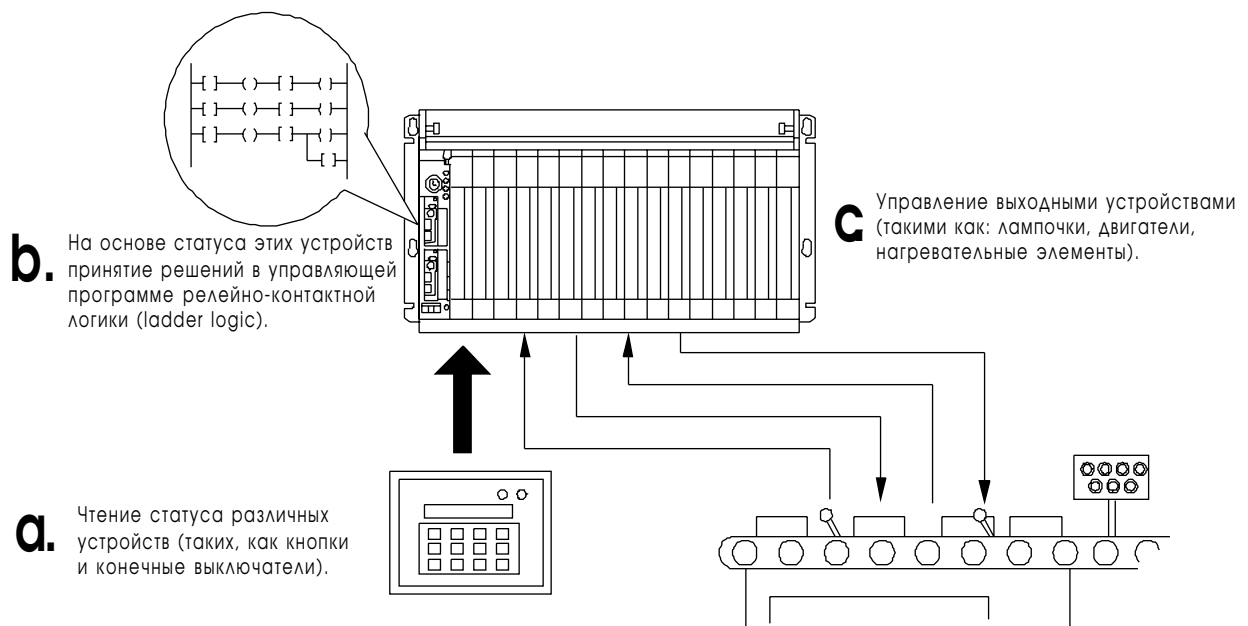
В этом разделе объясняется, как конфигурировать процессор для связи с резидентными входами/выходами:

1. Установка переключателями шасси Вх/Вых режима адресации.
2. Установка адреса рэка.

По умолчанию адрес рэка установлен равным 0. Если вы хотите изменить адрес рэка на 1, установите бит S:26/2.

Введение в сканирование процессора PLC-5

Основными функциями программируемого контроллера являются:

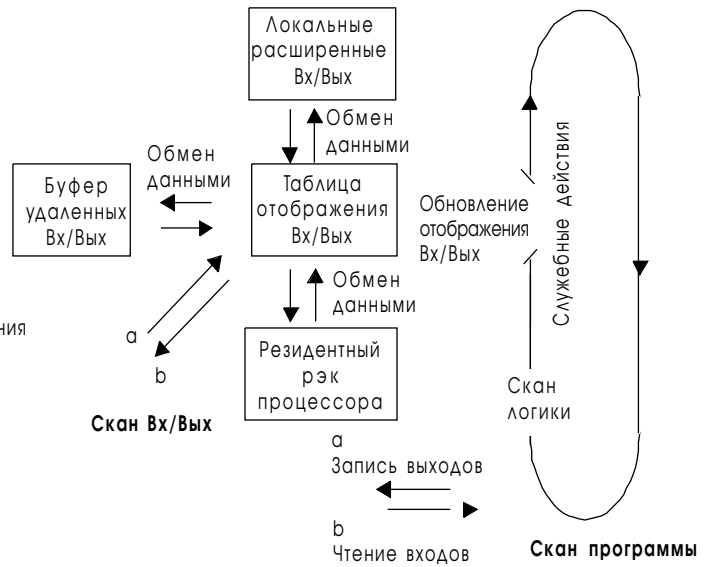


Процессор выполняет 2 основные операции :

- сканирование программы, где:
 - выполняется логика программы;
 - выполняются служебные действия,
- сканирование Вх/Вых - при этом, читаются входные данные и устанавливаются выходы.

Во время **сканирования логики**, входы читаются из таблицы, а выходы записываются в таблицу отображения Вх/Вых.

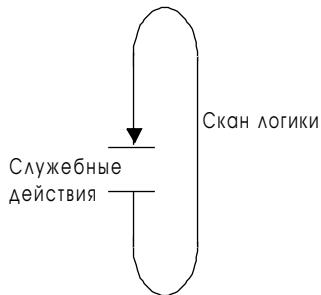
Во время **служебных действий** происходит обмен данными между таблицей отображения Вх/Вых и буфером удаленных Вх/Вых, локальными расширенными Вх/Вых, и резидентным рэком процессора.



Сканирование программы

Программный скан - это время, затрачиваемое процессором на выполнение логики программы один раз и выполнение служебных действий и, затем, снова переход к выполнению логики программы.

Процессор периодически выполняет сканирование логики и служебные действия, которые для процессора PLC-5 заключаются в :

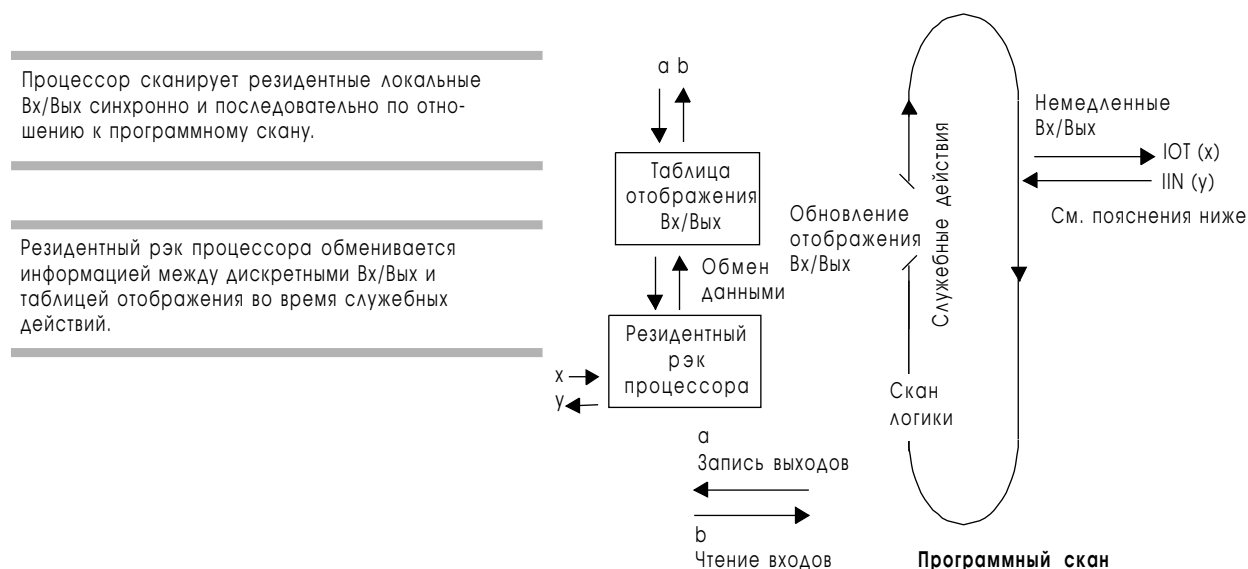


- выполнении внутренних проверок процессора;
- обновлении таблицы отображения входов в:
 - резидентном входном модуле процессора;
 - удаленном входном модуле (данных, содержащихся в буфере удаленных Вх/Вых);
 - входные модули локальных Вх/Вых расширения,
- посылке данных таблицы отображения выходов в:
 - резидентные выходные модули процессора;
 - буфер удаленных Вх/Вых;
 - выходные модули локальных Вх/Вых расширения.

Передача данных к резидентным Вх/Вых процессора

Ниже дано описание передачи дискретных данных и данных блок-трансферов для процессора PLC-5.

Передача дискретных данных к резидентным Вх/Вых процессора



Передача запросов немедленных Вх/Вых

Процессор обслуживает запросы немедленных входов (immediate input-IIN) и немедленных выходов (immediate output-IOT) во время выполнения скана логики. Скан логики задерживается на время запроса от немедленных Вх/Вых. Возобновление скана логики продолжается после получения данных и завершения запросов.

Инструкции IIN передают данные непосредственно в, а инструкции IOT передают данные непосредственно от модулей Вх/Вых процессора-резидентного и локального расширенного шасси Вх/Вых. Для удаленных Вх/Вых обновляется только буфер удаленных Вх/Вых. Для дополнительной информации см. раздел 22.

Совет

При размещении модулей Вх/Вых, не устанавливайте модули блок-трансферов в следующей или в той-же группе, в которой находится входной модуль, планируемый для использования в качестве немедленных Вх/Вых. Устанавливайте модули не в соседние слоты. Установка входных модулей для немедленных Вх/Вых сразу за модулями блок-трансферов может вызвать Ошибку чтения при передаче блоков -5 (Block-Transfer Read error).

Если ваш проект не может поддержать такую конфигурацию, помещайте инструкции немедленных Вх/Вых вместе с битами управления соседнего модуля блок-трансфера. Такая техника помогает убедиться, что соседний модуль блок-трансфера не будет выполнять передачу блоков во время выполнения инструкции немедленных Вх/Вых в соседнем входном модуле.

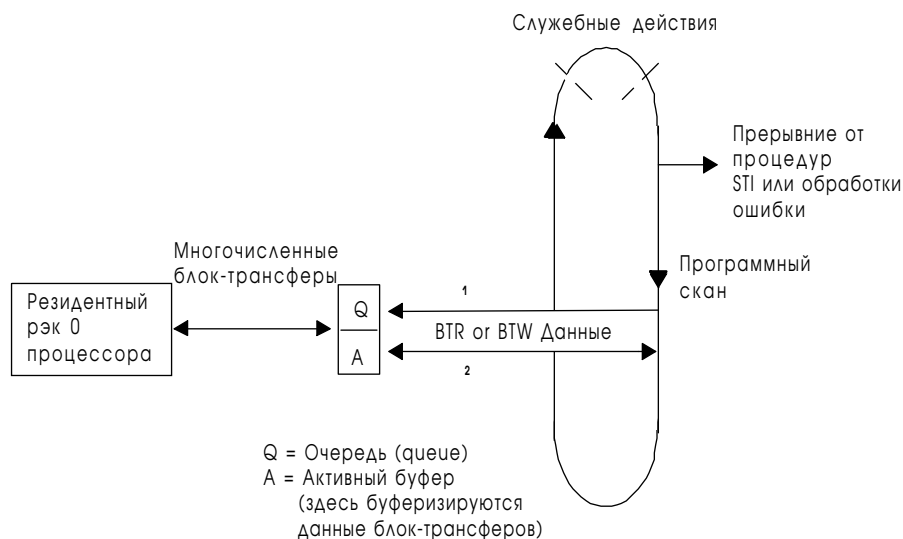
Передача данных блок-трансферов в резидентные Вх/Вых процессора

Процессор выполняет блок-трансферы в то же время, когда он сканирует программу.

Передача блок-трансферов к локальным резидентным Вх/Вых процессора происходит в следующей последовательности:

- запросы на передачу блоков помещаются в очередь адресуемого локального резидентного рэка Вх/Вых процессора ¹;
- активный буфер непрерывно обслуживает все модули блок-трансферов, чьи инструкции блок-трансферов были активированы в программном скане с помощью сканирования очереди, в порядке поступления запросов ²;
- передача блоков данных Вх/Вых может закончиться и бит готовности может быть установлен в любое время в течение программного скана.

Процессор непрерывно исполняет все активированные блок-трансферы Вх/Вых данных к резидентным Вх/Вых процессора, по мере поступления запроса на передачу блока в активный буфер.



Конфигурирование системы для резидентных Вх/Вых процессора

Для конфигурирования системы локальных резидентных Вх/Вых, Вам необходимо установить переключатель шасси Вх/Вых в соответствии с адресацией рэка. Режим адресации определяет число используемых номеров резидентных рэков процессора, основанное на количестве слотов в шасси. Для дополнительной информации см. раздел 4. Для установки переключателей шасси Вх/Вых см. раздел 23.

По умолчанию резидентный рэк процессора имеет адрес 0. Если необходимо, есть возможность установки этого адреса равным 1, путем установки пользовательского бита 2 (S26:2) на экране конфигурации процессора, используемого вами программного обеспечения. Если для резидентного рэка процессора вы выберете 1, рэк 0 станет для вашей системы недоступным.

Связь с удаленными входами/ выходами (Remote I/O)

Назначение этой главы

Сведения	Страница
Выбор устройств, которые вы можете подключить	6-2
Введение в удаленные Вх/Вых (Remote I/O)	6-3
Проектирование сети удаленных Вх/Вых	6-4
Настройка канала процессора для работы в режиме сканера	6-6
Связь с адаптером узла удаленных Вх/Вых	6-11
Пересылка блоков данных	6-13
Блок-трансфер данных удаленных Вх/Вых	6-15
Последовательность работы блок-трансфера и биты состояния	6-16
Требования к программированию блок-трансферов	6-20
Просмотр каналов сканера удаленных входов/выходов	6-21
Адресация файла состояния удаленных входов/выходов	6-23

В этой главе показано, как сконфигурировать процессор для связи с удаленными входами/выходами.

1. Выбор канала, настраиваемого для работы в режиме сканера.
2. Определение файла состояния Вх/Вых.

Определите отдельный, не использованный целочисленный файл. Вы должны определить файл состояния, если вы хотите провести автоконфигурацию для своей системы.

3. Определите файл диагностики.
Определите отдельный, не использованный целочисленный файл.
4. Определите список сканирования.

Выбор устройств, которые вы можете подключить

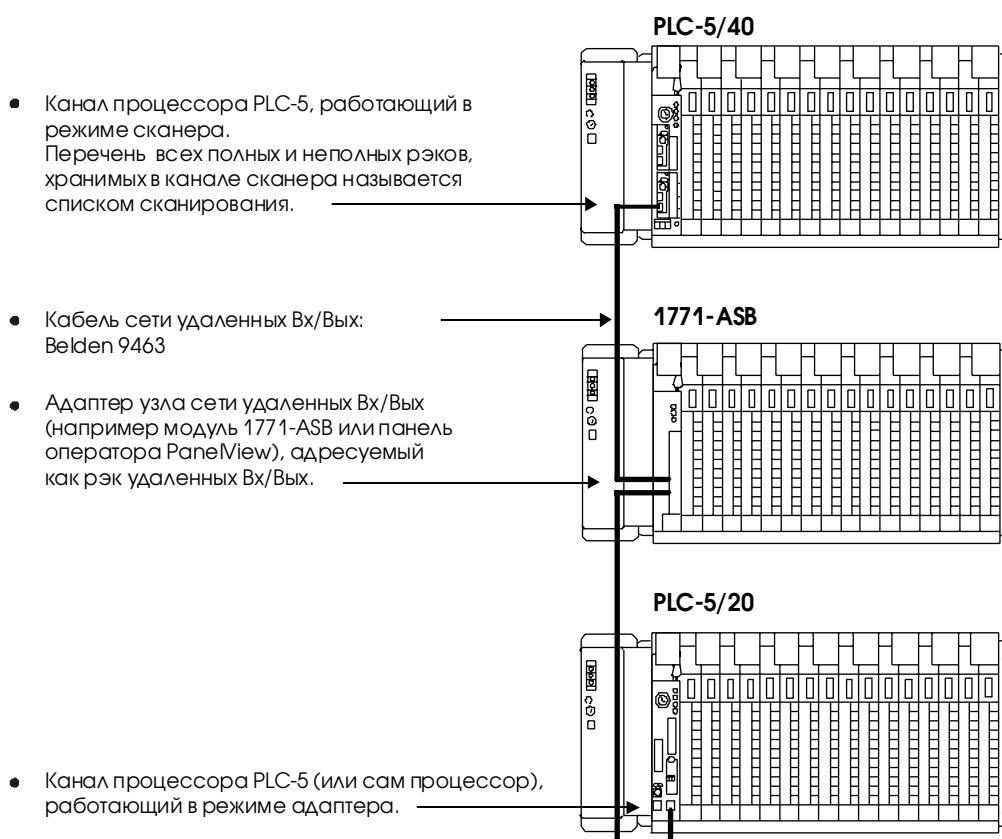
Следующая таблица, приведенная ниже, перечисляет часть из устройств, которые Вы можете использовать в сети удаленных Вх/Вых:

тип оборудования	Наименование	Номер по каталогу
Другой процессор (в режиме адаптера)	Расширенный процессор PLC-5	1785-LxxB
	Процессор PLC-5 Ethernet	1785-LxxE
	Процессор PLC-5 ControlNet	1785-LxxC
	Процессор PLC-5 VMEbus	1785-VxxB
	Процессор PLC-5 с расширенными локальными Вх/Вых	1785-LxxL
	Классический процессор PLC-5	1785-LTx
Другой процессор (в режиме адаптера)	Модуль связи для процессора SLC	1747-DCM
Для удаленных Вх/Вых	Модуль адаптера удаленных Вх/Вых процессора SLC	1747-ASB
	Блок Вх/Вых серии 1791	серия 1791
	Адаптер удаленных Вх/Вых	1771-ASB
	1-слотовое шасси Вх/Вых со встроенным источником питания и адаптером	1771-AM1
	2-слотовое шасси Вх/Вых со встроенным источником питания и адаптером	1771-AM2
	Модуль связи	1771-DCM
Интерфейсы оператора	DL40 Dataliner	2706-xxxx
	RediPANEL	2705-xxx
	Терминал PanelView	2711-xxx
Приводы	Адаптер удаленных Вх/Вых для промышленного привода 1336AC	1336-RIO
	Адаптер удаленных Вх/Вых для промышленного привода 1395AC	1395-NA

Введение в удаленные Вх/Вых (Remote I/O)

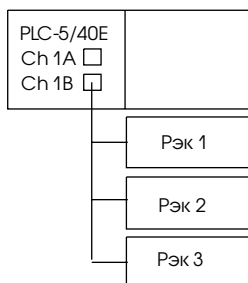
Использование сети удаленных Вх/Вых дает Вам возможность контролировать входы/выходы, не входящие в состав шасси процессора. Канал процессора PLC-5 в режиме сканера обменивается дискретными данными и данными блок-трансферов с удаленными Вх/Вых.

Ниже приведен пример системы удаленных Вх/Вых.



Для канала сканера удаленных Вх/Вых формируется перечень всего оборудования, подключенного к сети удаленных Вх/Вых и называемый списком сканирования. Пример списка сканирования приведен ниже:

Список сканирования канала 1B



Адрес рэка	Начальная группа	Размер рэка	Диапазон
1	0	Полный	010-017
2	0	1/2	020-023
3	0	Полный	030-037

В этом примере канал 1B постоянно сканирует три рэка списка сканирования и помещает данные в буфер удаленных Вх/Вых процессора. Процессор обновляет свой собственный буфер и таблицу Вх/Вых. В течении служебного цикла буферы обновляются, обмениваясь входными и выходными данными между собой.

Для более подробной информации о списке сканирования смотрите страницу 6-9.

Для настройки системы удаленных Вх/Вых выполните следующие действия:

Действие:	Смотрите:
1.Сконфигурируйте адаптеры устройств удаленных Вх/Вых	руководство пользователя для устройств
2.Проложите и подключите кабель сети удаленных Вх/Вых	<ul style="list-style-type: none"> • для проектирования см. стр.6-4 • для информации о прокладке кабеля см. главу 3 • информацию по установке вашего процессора (Для расширенных процессоров PLC-5 см. публикацию 1785-10.4; для процессоров PLC-5 Ethernet см. публикацию 1785-10.5)
3.Настройте канал сканера	стр. 6-6

Проектирование сети удаленных Вх/Вых

Для разработки сети удаленных Вх/Вых требуется соблюдать:

- требования по проектированию сети удаленных Вх/Вых;
- требования по проектированию кабелей.

Советы

Требования по проектировании сети удаленных Вх/Вых

При проектировании сети удаленных Вх/Вых необходимо выполнять следующие требования:

- все оборудование, подключенное к сети удаленных Вх/Вых, должны работать на одной скорости передачи данных (57.6, 115.2 или 230.4 кБод). Выберите скорость передачи данных, поддерживаемую всем оборудованием;
- два или более каналов одного и того же процессора, работающие в режиме сканера, не могут сканировать один и тот же неполный или полный адрес рэка. Назначьте уникальные адреса для неполного или полного рэка каждому из каналов, работающих в режиме сканера удаленных Вх/Вых;
- вы можете разделить адреса рэков между каналами сканера, однако возникнут проблемы при выполнении блок-трансферов. См. страницу 6-15;
- список сканирования может иметь максимум 16 номеров рэков или максимум 32 физических устройства, подключенных к сети при использовании терминаторов 82 Ом. Для более подробной информации о списке сканирования смотрите страницу 6-9.

Советы

Проектирование кабелей

При подключении сети по методу "ответвлений", используйте разъем 1770-SC, соблюдая следующие требования к длине кабеля:

- длина кабеля между разъемами зависит от скорости передачи данных в сети;
- длина "ответвления" - 30,4 м. (100 футов)

Для получения большей информации о подключении сети по методу «ответвлений», см. **Data Highway/Data Highway Plus/Data Highway II/Data Highway—485 Cable Installation Manual**, публикация 1770—6.2.2.

Выбирается кабель 1770-CD (Belden 9463). Подключите оборудование к сети удаленных Вх/Вых, используя метод последовательных соединений или метод «ответвлений».

Проверьте, чтобы в разработанной вами системе длина кабеля не выходила за допустимые пределы.

Важно: Максимальная длина кабеля для сети удаленных Вх/Вых зависит от скорости передачи данных. Установите для оборудования в сети удаленных Вх/Вых одинаковую скорость передачи данных.

Для определения общей длины кабеля используйте следующую таблицу:

Таблица 6.А
Выбор правильной длины кабеля

В сети Remote I/O скорость передачи:	Длина кабеля не должна превышать
57.6 кБод	3, 048 м (10, 000 футов)
115.2 кБод	1, 524м (5, 000 футов)
230.4 кБод	762 м (2, 500 футов)

Для нормальной работы установите на оба конца сети удаленных Вх/Вых внешние резисторы (терминаторы), поставляемые с программируемым контроллером. Выберите терминаторы на 150 Ом или на 82 Ом в зависимости от того, какое количество оборудования вы подключаете к сети удаленных Вх/Вых.

Если сеть Remote I/O:	Используйте терминатор	Максимальное количество оборудования, которое можно подключить к сети	Максимальное число рэков, сканируемых в сети
работает со скоростью 230.4 кБод	82 Ом	32	16
работает со скоростью 57.6 или 115.2 кБод и в сети нет оборудования, перечисленного в таблице 6.В			
содержит оборудование, перечисленное в таблице 6.В	150 Ом	16	16
работает со скоростью 57.6 или 115.2 кБод, и нет необходимости для того, чтобы сеть поддерживала более 16 единиц оборудования			

Таблица 6.В
Оборудование, для которого требуются терминаторы 150 Ом

Тип оборудования	Номер по каталогу	Серия
Сканеры	1771-SN	Все
	1772-SD, -SD2	
	1775-SR	
	1775-S4A, -S4B 6008-SQH1, -SQH2	
Адаптеры	1771-AS	A
	1771-ASB	
	1771-DCM	
Прочие	1771-AF	Все

Настройка канала процессора для работы в режиме сканера

Для помощи в выборе канала процессора, который вы можете сконфигурировать для режима сканера удаленных Вх/Вых, используйте эту таблицу:

Процессор:	Канал, поддерживающий сканер Remote I/O
PLC-5/11	1A
PLC-5/20 PLC-5/20E	1B
PLC-5/30 PLC-5/40E PLC-5/40L PLC-5/80E PLC-5/60L	1A, 1B
PLC-5/40 PLC-5/60 PLC-5/80	1A, 1B, 2A, 2B

Для конфигурации канала процессора для работы в режиме сканера:

- определите файл состояния Вх/Вых, в котором хранится информация о рэках, подключенных к процессору, используя для этого экран конфигурации процессора вашего пакета программирования;
- определите скорость передачи данных для сканера, файл диагностики и список сканирования, используя экран конфигурации канала для режима сканера вашего пакета программирования.

Определение файла состояния Вх/Вых

В файле состояния Вх/Вых содержится информация для процессорной таблицы конфигурации рэков Вх/Вых. Для хранения данных состояния каждого рэка удаленных Вх/Вых требуется 2 слова. В этих двух словах для каждого рэка помещаются биты перезапуска (**Reset**), наличия (**Present**), запрета (**Inhibit**) и ошибки (**Fault**).

Чтобы определить файл состояния Вх/Вых введите номер неиспользованного целочисленного файла (9-255) в поле файла состояния Вх/Вых (S:16) на экране конфигурации процессора. Если вы не хотите использовать таблицу конфигурации рэков Вх/Вых, введите 0. Однако, если вы хотите использовать опцию автоконфигурации для создания списка сканирования, вам нужно определить файл состояния Вх/Вых. Используйте экран конфигурации процессора вашего пакета программирования:

```

Processor Configuration
User      Control Bits      00000000 00000000  RESTART LAST  ACTIVE STEP
Fault routine prog file no.: 0          Watchdog (ms):      500
      I/O status file:  N12          Communication time slice (ms):  3
      VME Status File:      N34

```

Пакет программирования автоматически создает файл состояния Вх/Вых требуемого размера. Для определения состояния Вх/Вых Вы можете использовать отдельные слова файла состояния в вашей релейно-контактной программе.

Определение информации о конфигурации канала

Используйте экран конфигурации канала для режима сканера удаленных Вх/Вых в вашем пакете программирования для конфигурации канала в режиме сканера:

Scanner Mode				
Diag. file:	N13			Baud rate:
57.6kB				Complementary I/O:
Enabled	Rack	Starting	Rack	Range
Address	Group	Size		
2	4	1/4	024-027	
C 2	4	1/4	020-021	
3	0	1/2	030-033	
3	4	1/4	034-035	
3	6	1/4	036-037	
C 7	0	FULL	170-177	
1	0	FULL	010-007 *	

В этом поле:

Определите:

Сделайте следующее:

Diag. file	<p>Файл, содержащий информацию о состоянии канала:</p> <ul style="list-style-type: none"> •сообщений полученных; •сообщений посланных; •невозможность посылки; •сообщений, принятых с ошибкой; •сообщений, посланных с ошибкой; •повторные обращения к рэкам. 	<p>Введите номер целочисленного файла (9-999)</p> <p>ВНИМАНИЕ: Используйте отдельные файлы диагностики для каждого канала. Не назначайте файлом диагностики файл состояния входов/выходов или другой используемый целочисленный файл. Результатом может стать поломка машин и механизмов.</p>
Baud Rate	<p>Скорость передачи данных для сканера сети удаленных Вх/Вых.</p>	<p>Введите в этом поле требуемую скорость передачи данных.</p> <p>Допустимые значения: 57.6, 115.2 и 230.4 кБод.</p>
Complementary I/O	<p>Есть или нет рэки дополнительных Вх/Вых</p>	<p>В этой области введите Enable или Disable.</p>

Определение списка сканирования

Список сканирования представляет собой список оборудования Вх/Вых, опрашиваемых каналом сканера процессора. Для связи канала с подключенными к нему устройствами вы должны создать список сканирования.

Для того, чтобы:	Сделайте следующее:
Создать список сканирования	<p>Убедитесь, что процессор находится в режиме Remote Program или Program.</p> <ol style="list-style-type: none"> 1. Убедитесь, что определен файл состояния Вх/Вых в экране конфигурации процессора (см. стр. 6-7). 2. Завершите все изменения, сделанные для конфигурации канала. 3. Используйте функцию автоконфигурации. <p>Если при завершении изменений получена ошибка, сотрите список сканирования и повторно завершите все изменения.</p> <p>Если получено сообщение «Resource not Available» («Ресурсы недоступны»), значит вы не определили файл состояния Вх/Вых. Определите файл состояния Вх/Вых и проведите повторную автоконфигурацию.</p>
Вставить запись в список сканирования	<p>Убедитесь, что процессор находится в режиме Remote Program или Program.</p> <ol style="list-style-type: none"> 1. Установите курсор в ту область списка сканирования, где нужно вставить запись. 2. Введите верную запись в список сканирования. <p>Важно: Если запись содержит неправильную информацию, процессор не отобразит новую конфигурацию после сохранения.</p>

Список сканирования включает в себя следующую информацию

В этой области:	Список сканирования содержит:
Rack Adress (адрес рэка)	<p>1-3 восьмеричный (для процессоров PLC-5/11, -5/20, -5/20E) 1-7 восьмеричный (для процессоров PLC5/30) 1-17 восьмеричный (для процессоров PLC-5/40, -5/40L, -5/40E или -5/80E) 1-27 восьмеричный (для процессоров PLC-5/60, -5/60L, -5/80) если разрешены дополнительные Вх/Вых, перед адресом дополнительного рэка ставится буква С).</p>
Starting group (начальная группа)	0, 2, 4 или 6
Rack Size (размер рэка)	1/2, 1/4, 3/4 или FULL
Range	<p>Вычисляется автоматически на основании адрес рэка (Rack Adress), начальной группы модулей (Starting group) и размера рэка (Rack Size). Символ * после данного параметра означает последнюю верную запись рэка.</p>

Советы

Если Вам необходимо обновлять состояния Вх/Вых несколько раз за один скан, можно вводить логический адрес в списке сканирования тоже несколько раз. Не назначайте один и тот же адрес неполного или полного рэка более, чем одному каналу сканера. Каждый канал должен сканировать свои собственные адреса рэков.

Запомните ограничения при создании или изменении списка сканирования:

- автоматическая конфигурация всегда отображает фактическую аппаратную конфигурацию, за исключением рэков, для которых установлен бит общего запрета. В этом случае бит общего запрета рэка отменит автоматическую конфигурацию. Вы должны сначала очистить общие запреты рэков, а затем выбрать автоконфигурацию;

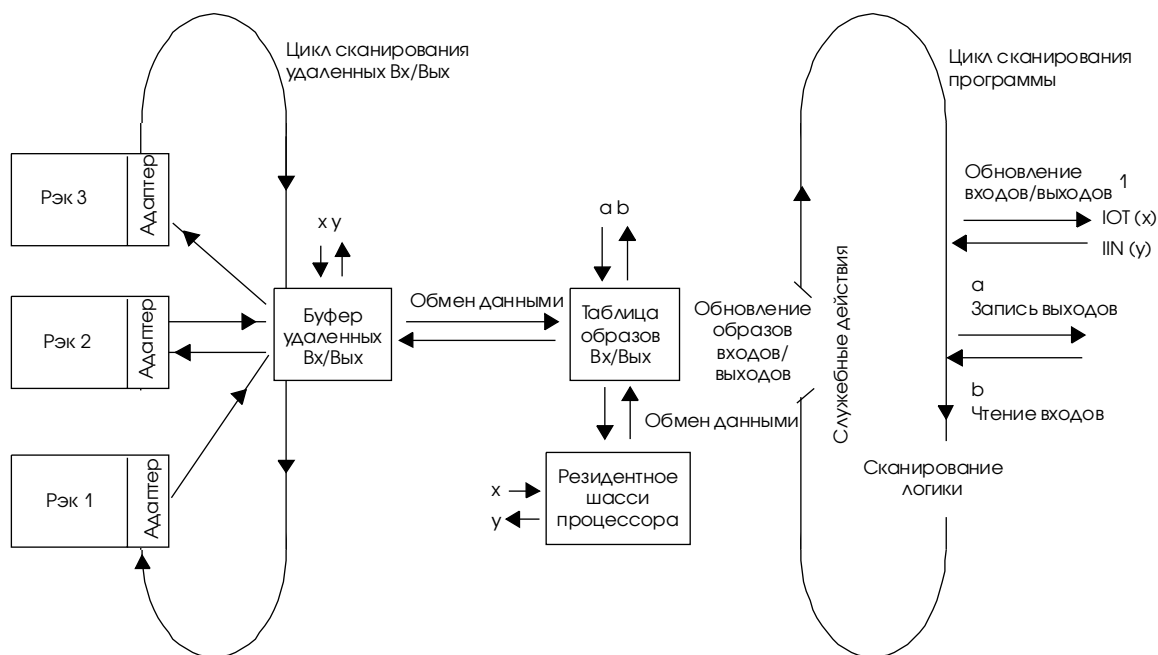
Очистите биты общего запрета рэков для канала, в котором Вы хотите продолжать сканирование. Используйте экран состояния режима сканера вашего пакета программирования;

- если Вы изменяете конфигурацию канала из режима адаптера или режима ДН + в режим сканера, используйте функцию очистки списка сканирования вашего пакета программирования. В любых других случаях очистки данных, введенных в список сканирования, используйте функцию «стереть с листа» (**delete-from-list**), для удаления данных по одному.

Связь с адаптером удаленных Вх/Вых

Канал сканера обменивается дискретными данными с адаптером удаленных Вх/Вых (1771-ASB) через буфер удаленных Вх/Вых.

Рисунок 6.1
Цикл сканирования удаленных Вх/Вых и цикл сканирования программы



Цикл сканирования удаленных Вх/Вых - это время, затрачиваемое процессором для связи со всем перечисленным в списке сканирования оборудованием. Сканирование удаленных Вх/Вых происходит независимо и асинхронно с циклом сканирования программы.

В течение служебных действий:

- происходит обмен данными между таблицей образов Вх/Вых, резидентным шасси процессора и буфером удаленных Вх/Вых;
- обновляется буфер удаленных Вх/Вых.

Запомните, что сканер Вх/Вых постоянно обновляет буфер удаленных Вх/Вых асинхронно к циклу сканирования программы.

¹ - Немедленная передача обновленных данных от удаленных рэков вызывает обновление буфера удаленных Вх/Вых.

Важно: Цикл сканирования удаленных Вх/Вых для каждого канала, сконфигурированного для режима сканера, происходит независимо и асинхронно с циклом сканирования удаленных Вх/Вых для других каналов.

Для канала сканера, связывающегося с модулем 1771-ASB, сделайте следующее:

1. Установите переключатели на монтажной плате каждого шасси Вх/Вых, в котором установлен модуль адаптера.
2. Установите переключатели на модуле адаптера.

3. Подключите кабель удаленных Вх/Вых

Для более полной информации смотрите:

глава 23

инструкция по установке вашего процессора.

Трудности поиска неисправностей при связи с удаленными Вх/Вых

Для проверки связи процессора с устройствами сети удаленных Вх/Вых выполните следующие действия.

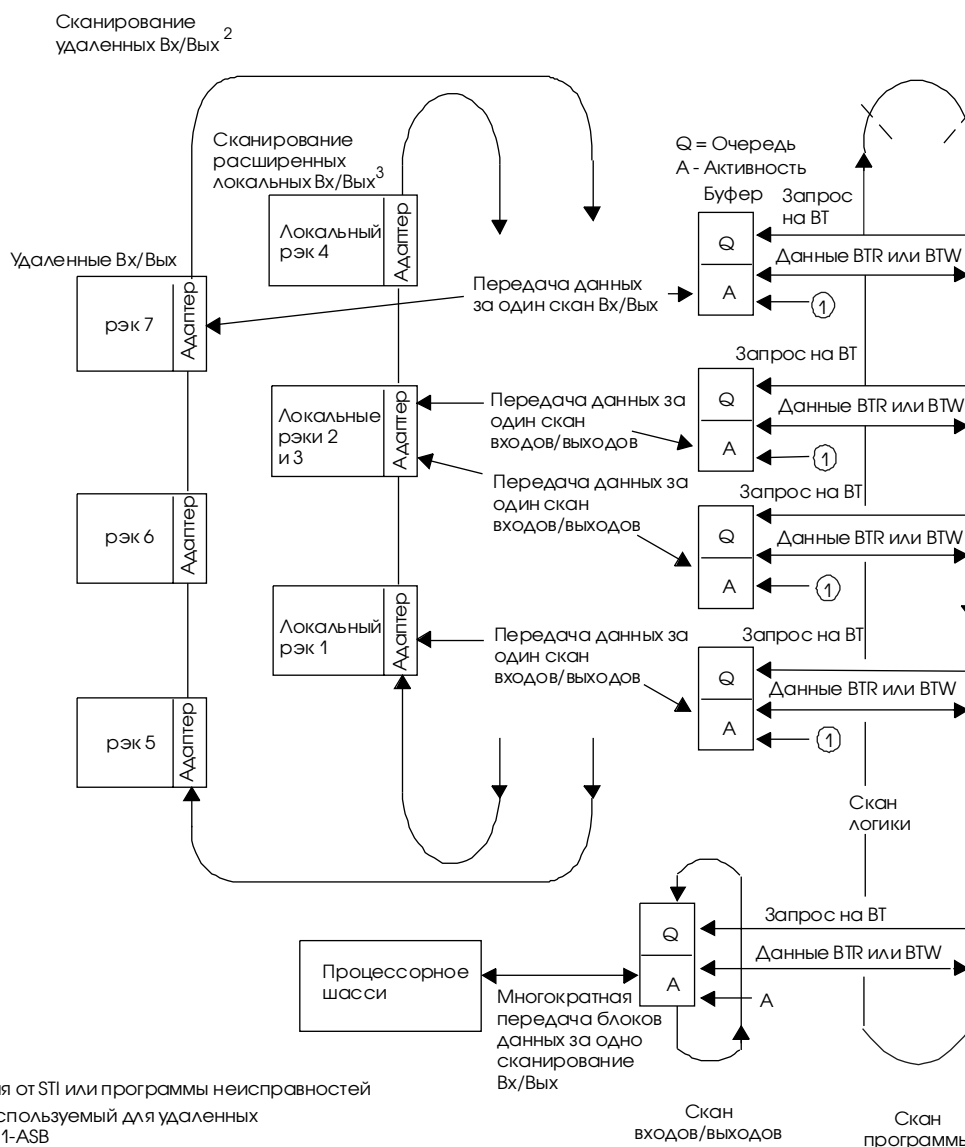
1. Переведите процессор в режим программирования. В карте памяти процессора и найдите два неиспользуемых номера файлов данных. Процессор будет использовать эти файлы. Не создавайте файлы, только запишите, какие номера файлов Вы будете использовать.
2. Перейдите к экрану состояния процессора для проверки обнуления (установки в 0) битов запрещения рэков (**Inhibit**).
3. Перейдите к экрану конфигурации процессора и назначьте один из предыдущих номеров файлов файлом состояния Вх/Вых (см. страницу 6-7).
4. Перейдите к экрану конфигурации канала для соответствующего канала и назначьте оставшийся номер файла (вышеупомянутый) файлом диагностики канала (см. страницу 6-8).
5. Выполнить автоконфигурацию и подтвердите, что все рэки были найдены и перечислены в списке сканирования Вх/Вых.
6. В экране состояния канала проверьте, нет ли повторений номеров рэков Вх/Вых, чтобы удостовериться в том, что нет никаких проблем связи.

Если Вы следовали вышеупомянутым действиям, и возникли проблемы связи с удаленными Вх/Вых, возможно нарушен файл состояния Вх/Вых. Попробуйте назначать новый файл состояния Вх/Вых и повторить шаги, перечисленные выше. Также удостоверьтесь в том, что таблицы образов Вх/Вых присутствуют для рэков, и нет проблем связи с ними.

Пересылка блоков данных

Дополнительно к передаче дискретной информации процессор может обмениваться блоками данных с удаленными Вх/Вых. Инструкции передачи блоков данных (блок-трансферы) заставляют процессор прервать нормальное сканирование Вх/Вых и передать/принять от выбранного модуля Вх/Вых до 64 слов данных. На рисунке 6.2 показано, как происходит передача блоков данных (блок-трансферов).

Рисунок 6.2
Передача блоков данных в процессорное шасси, шасси расширенных локальных Вх/Вых и удаленных Вх/Вых



- 1 Прерывания от STI или программы неисправностей
- 2 Адаптер, используемый для удаленных Вх/Вых - 1771-ASB
- 3 Адаптер, используемый для расширенных локальных Вх/Вых - 1771-ALX

Как показано на рисунке 6.2, в процессоре имеются следующие области для блок-трансферов:

Максимальное количество активных буферов для канала удаленных Вх/Вых		Область хранения	Назначение
PLC-5/60,-5/60L,-5/80,-5/80E	23	Активные буферы	Хранение инициализированных запросов на передачу блок-трансферов для канала. В таблице, приведенной слева, перечислено максимальное количество активных буферов для каждого расширенного и Ethernet процессора PLC-5. Процессор помещает запрос на передачу блок-трансферов прямо в активный буфер, только если буфер доступен и нет очередей передачи блок-трансферов в слот.
PLC-5/40,-5/40L,-5/40E	31		
PLC-5/30	39		
PLC-5/20,-5/20E	43		
PLC-5/11	43	Очереди ожидания	Хранение запросов на передачу блок-трансферов, которые не могли быть помещены в активный буфер из-за того, что: <ul style="list-style-type: none"> • все активные буферы канала используются, • слот, к которому адресован блок-трансфер, уже производит передачу блок-трансфера.

При переводе процессора в режим **PROGRAM** отменяется передача блок-трансферов в активных буферах и в очередях ожидания.

Как только завершается передача блок-трансфера в слот, процессор проверяет очередь, чтобы убедиться, есть ли ожидающий своей очереди блок-трансфер, адресованный слоту. Если он есть, процессор перемещает его в активный буфер.

Так как процессор может запрашивать передачу блок-трансферов из каждого слота в шасси одновременно, адаптер определяет порядок, в каком выполняется передача блок-трансферов в шасси. Запросы на передачу блок-трансферов обрабатываются различным образом для подпрограммы неисправностей, подпрограммы прерываний по времени (STI) и подпрограммы прерываний по входам процессора (PI); для более подробной информации см. главы 16, 18 и 19 соответственно.

Биты неосновных ошибок блок-трансфера

Бит неосновной ошибки:		Назначение:								
S:17/0	Заполнение очереди передачи блок-трансферов	Возможно, что процессор PLC-5 временно неспособен инициализировать многократные последовательные передачи блок-трансферов на удаленные Вх/Вых, запрограммированные пользователем. Для любой передачи блок-трансферов, которая временно не может быть обработана, PLC-5 процессор устанавливает бит неосновной ошибки S:17/0 и сбрасывает команду передачи блок-трансферов. Даже, если условие возникновения ошибки исчезает, бит S:17/0 остается установленным, пока Вы его не сбросите. Вы можете избежать возникновения данной ошибки, отделив цепь инструкции передачи блок-трансферов от других цепей.								
от S:17/1 до S:17/4	Переполнение очередей для канала xx	<p>Процессор PLC-5 может обрабатывать не более 64 передач блок-трансферов на пару канала (1A/1B или 2A/2B). В этот максимум входит:</p> <ul style="list-style-type: none"> • блок-трансферы, находящиеся в активном буфере, • инициализированные блок-трансферы, ожидающие выполнения в очереди. <p>Как только достигается максимум в 64 блок-трансфера, устанавливаются биты неосновной ошибки, зависящие от того, в какой паре каналов произошло переполнение:</p> <table border="1"> <thead> <tr> <th>Пара каналов:</th> <th>Устанавливается бит ошибки:</th> </tr> </thead> <tbody> <tr> <td>1A/1B</td> <td>S:17/1 и S:17/2</td> </tr> <tr> <td>2A/2B</td> <td>S:17/3 и S:17/4</td> </tr> </tbody> </table> <p>Процессор PLC-5 не может инициализировать любую инструкцию передачи блок-трансферов на удаленные Вх/Вых, превышающую максимум в 64 блока. Биты .EW, .DN и .ER перезапускаются в любом блок-трансфере, в котором превышен предел 64 блок-трансферов. Даже если условие возникновения ошибки исчезает, биты остаются установленным, пока Вы их не сбросите самостоятельно.</p>	Пара каналов:	Устанавливается бит ошибки:	1A/1B	S:17/1 и S:17/2	2A/2B	S:17/3 и S:17/4		
Пара каналов:	Устанавливается бит ошибки:									
1A/1B	S:17/1 и S:17/2									
2A/2B	S:17/3 и S:17/4									
S:10/7	Нет команд передачи	<p>Этот бит неосновной ошибки обычно связан с проблемами в прикладной программе, но он также может быть установлен при превышении максимального допустимого числа блок-трансферов, доступных в процессоре PLC-5. Блоки команд используются как в локальных, так и в удаленных блок-трансферах.</p> <table border="1"> <thead> <tr> <th>Тип PLC-5:</th> <th>Максимальное число блоков команд:</th> </tr> </thead> <tbody> <tr> <td>PLC-5/11,-5/20,-5/30</td> <td>128</td> </tr> <tr> <td>PLC-5/40</td> <td>256</td> </tr> <tr> <td>PLC-5/60,-5/80</td> <td>384</td> </tr> </tbody> </table> <p>Это происходит когда программа пытается повторно инициализировать передачу блок-трансферов, которые еще не завершились установкой битов .DN или .ER. Даже если условие возникновения ошибки исчезает, бит S:10/7 останется установленным, пока Вы его не сбросите самостоятельно.</p>	Тип PLC-5:	Максимальное число блоков команд:	PLC-5/11,-5/20,-5/30	128	PLC-5/40	256	PLC-5/60,-5/80	384
Тип PLC-5:	Максимальное число блоков команд:									
PLC-5/11,-5/20,-5/30	128									
PLC-5/40	256									
PLC-5/60,-5/80	384									

Блок-трансфер данных удаленных Вх/Вых

Передача данных блок-трансферов для удаленных Вх/Вых происходит следующим образом.

- передача блок-трансферов и сканирование программы выполняются независимо и одновременно. Как только инициализирован блок-трансфер, процессор выполняет его асинхронно с циклом программы;
- в течение каждого сканирования удаленных Вх/Вых процессор выполняет максимум один блок-трансфер, занесенный в список сканирования.

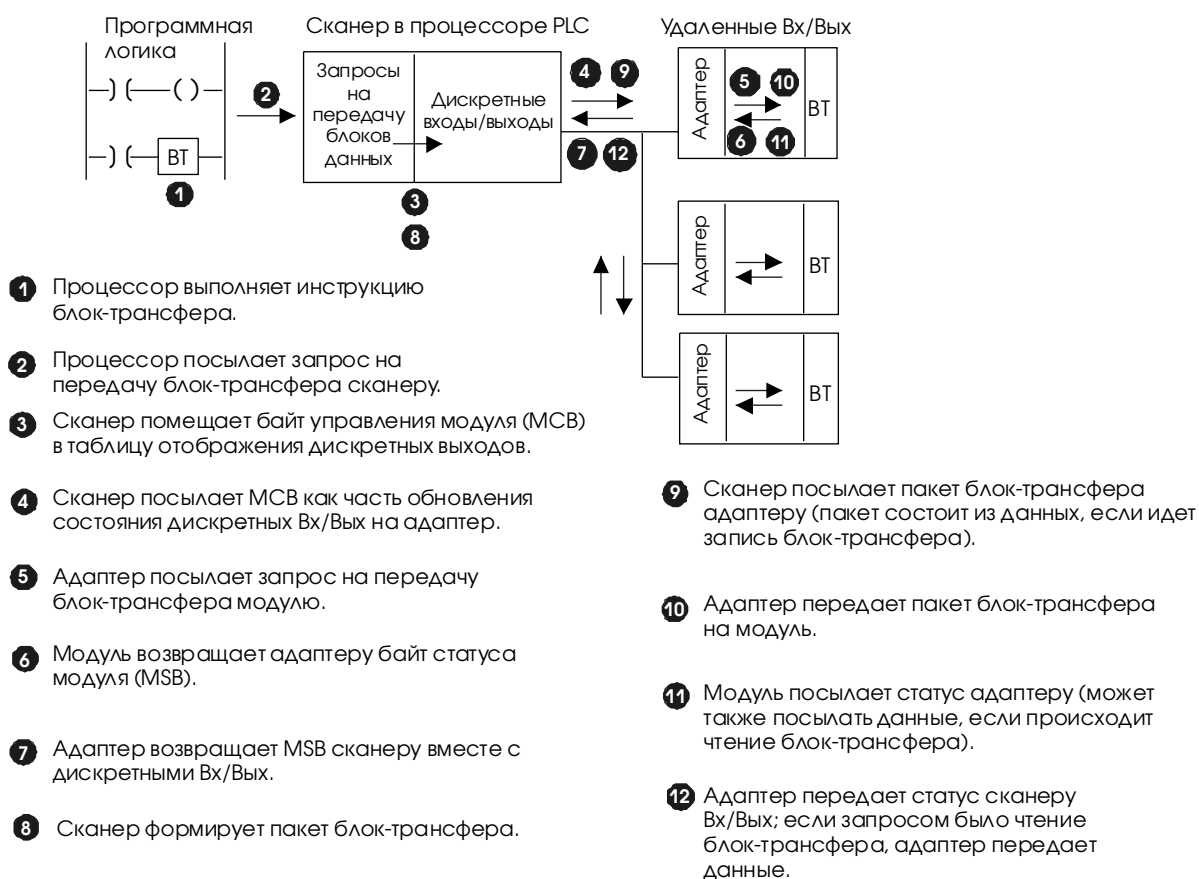
Важно: Если Вы разбиваете адреса удаленных шасси между каналами сканера, блок-трансферы на каналы сканера с более низким приоритетом не будут правильно работать. Каналы сканера имеют приоритеты в следующем порядке: 1А, 1В, 2А затем 2В.

Например: если Вы сконфигурируете каналы 1В и 2А как сканеры удаленных Вх/Вых и разобьете рэк #2 между ними, передача блок-трансферов через канал 1В, имеющий более высокий приоритет, будет завершена, но передача блок-трансферов во вторую половину рэка #2 через канал 2А, имеющий более низкий приоритет, не будет завершена.

Разбивка рэков удаленных Вх/Вых между каналами сканера не оказывает влияния на передачу дискретных данных, биты состояния входов/выходов (такие как Fault и Preset) не могут отображать истинное состояние.

На рисунке 6.3 показана последовательность передачи блок-трансфера для удаленных Вх/Вых.

Рисунок 6.3
Последовательность передачи блок-трансфера



Последовательность работы блок-трансфера и биты состояния

На рисунке 6.4 показаны значения статусных битов блок-трансфера.

Рисунок 6.4
Состояния статусных битов блок-трансфера.

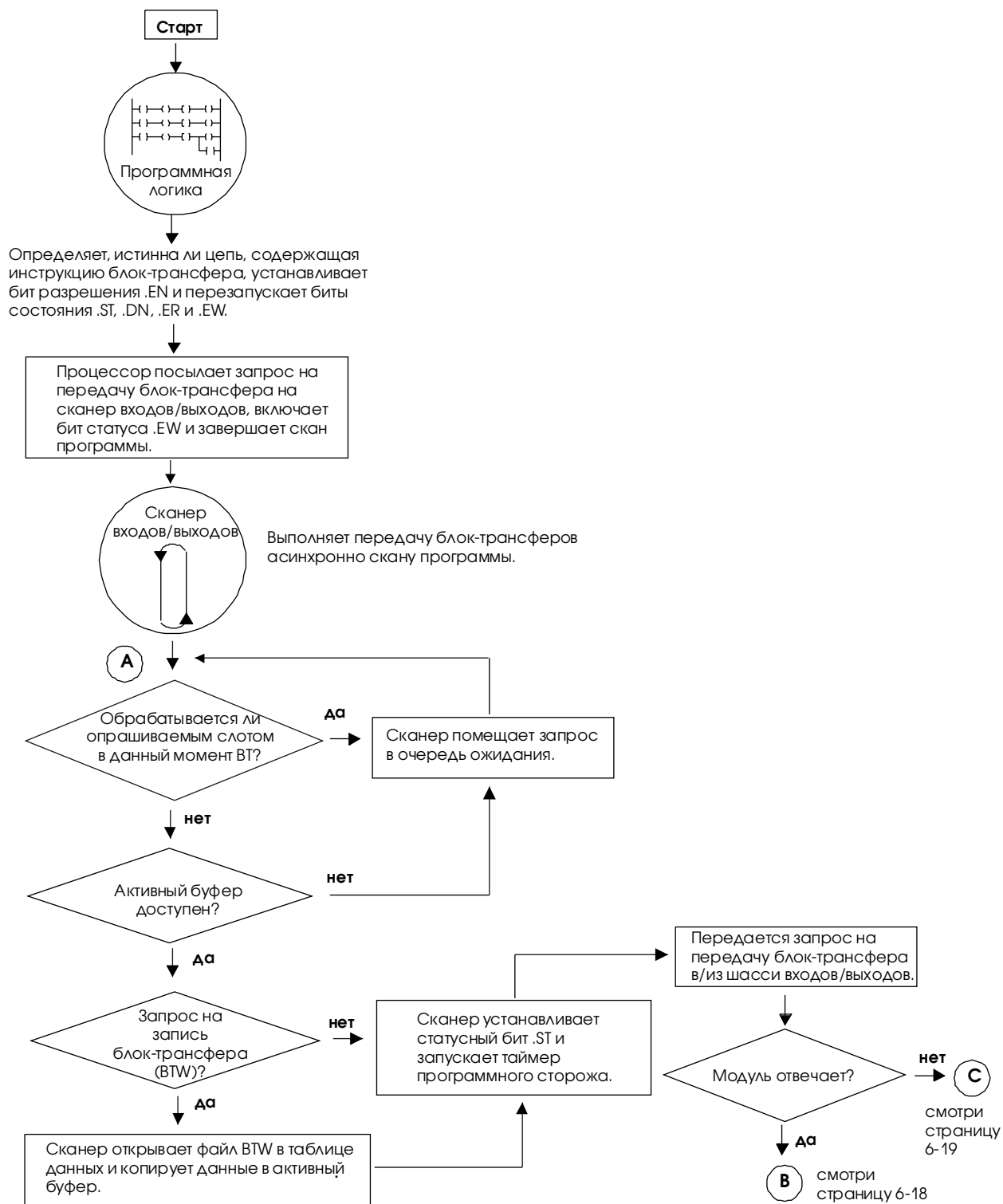


Рисунок 6.4 (продолжение)
Модуль, передающий блок-трансфер, отвечает.

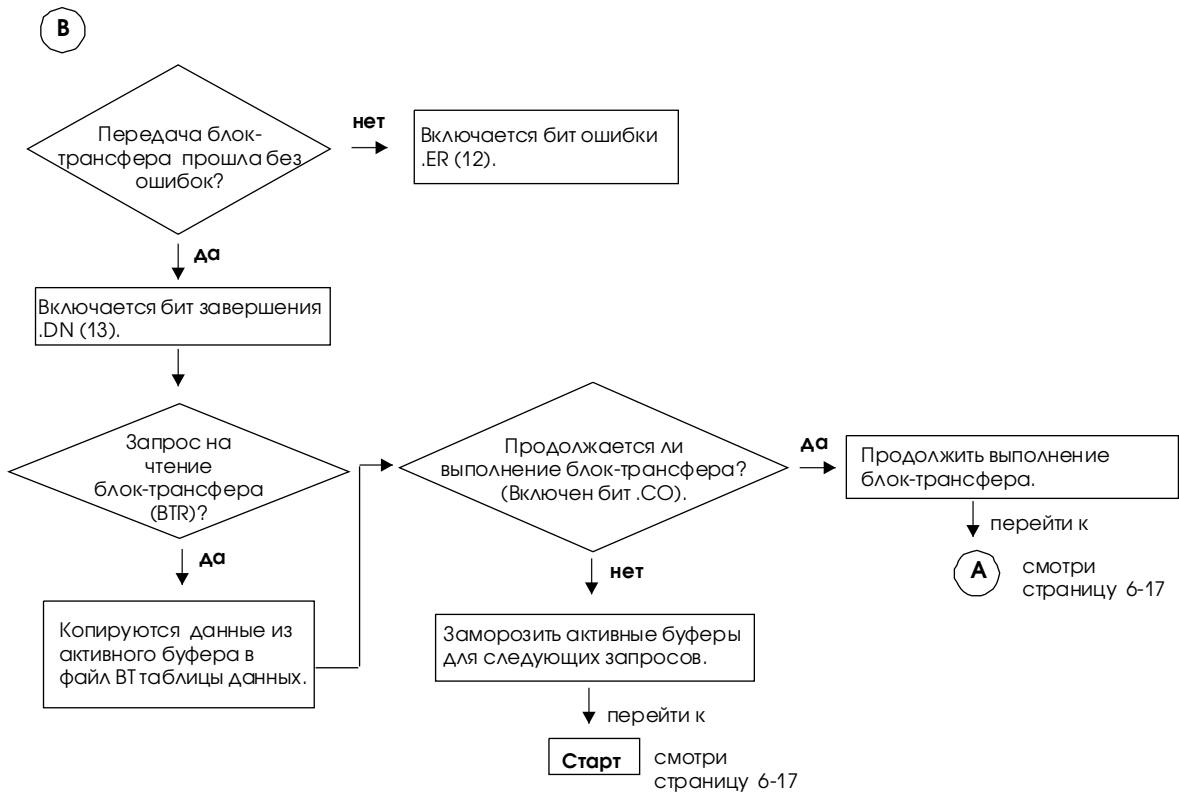
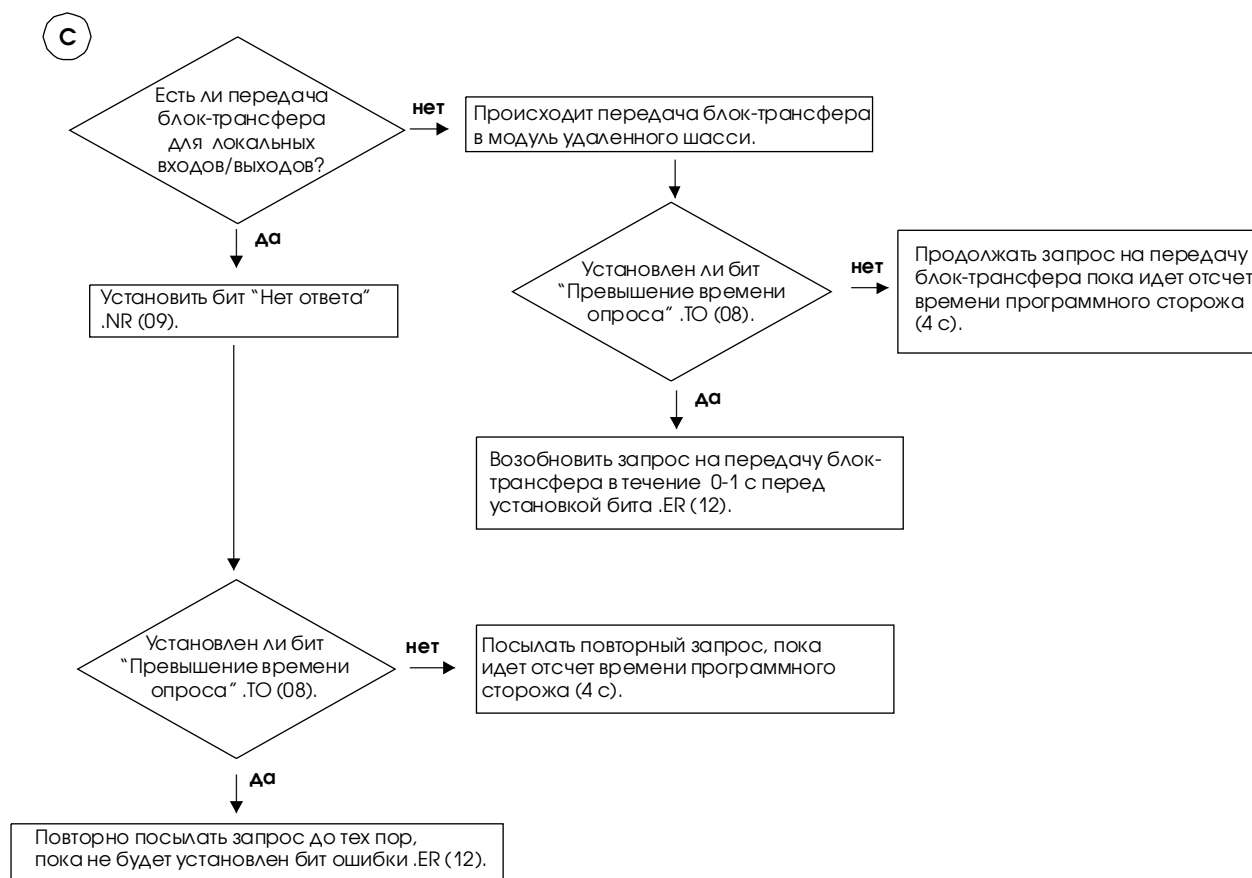


Рисунок 6.4 (продолжение)
Модуль, передающий блок-трансфер, не отвечает.



Для просмотра кодов ошибок, возникающих при передаче блок-трансферов, смотрите «**PLC-5 Programming Software Instruction Set Reference**», публикация 1785-6.1.

Требования к программированию блок-трансферов

Прочитайте этот раздел для того, чтобы понять общие требования к программированию и требования к резидентному процессорному шасси.

Советы

Основные требования

Ниже приведены основные требования при программировании блок-трансферов данных входов/выходов.

- При выполнении блок-трансферов (как для процессорного локального, так и для удаленного шасси), для любого типа процессоров PLC-5, очистите таблицу отображения выходов, соответствующую модулю, из которого происходит чтение блок-трансфера, перед тем как перевести процессор в режим **RUN**. Если Вы не очистите таблицу отображения выходов, может возникнуть ошибка передачи блок-трансфера из-за того, что незапрашиваемые блок-трансферы передаются на модуль (т. е., если модуль, из которого происходит чтение блок-трансфера, установлен в рэк 2, группа 4, сбросьте слово O:024 в 0 и не используйте эти слова для хранения данных).
- Если Вы используете инструкции блок-трансфера для удаленных Вх/Вых и бит времени ожидания (.TO) установлен в 1, процессор отключает 4-секундный таймер и продолжает запрос на передачу блок-трансферов в течение 1 секунды перед тем как установить бит ошибки (.ER).
- Процессор PLC-5, по крайней мере с одним каналом, сконфигурированным в режим адаптера, может выдать основную ошибку при переводе из режима **PROGRAM** в режим **RUN**.

Для устранения этой возможности запрограммируйте сканер на запрос только двух или трех одновременных блок-трансферов из адаптера PLC-5, поставив в условие работы команд передачи блоков данных биты выполнения/ошибки (.DN/.ER).

Для процессорного шасси

Следующие требования относятся к программированию блок-трансферов в процессорном шасси.

- В процессорном шасси число постоянно передаваемых блок-трансферов ограничивается 16 передачам по 4 слова каждый или 8 передачами по 64 слова. При превышении данных диапазонов возникает ошибка контрольной суммы (код ошибки - 5).
- Использование инструкций блок-трансферов для любого из нижеперечисленных модулей, установленных в процессорном шасси, приводит к появлению ошибок контрольной суммы:
 - 1771-OFE1, -OFE2, -OFE3 модули (все версии предшествующие серии В, ревизия В);
 - Модуль 2803-VIM (все версии предшествующие серии В, ревизия А);
 - IMC-120 (все версии).

- Для устранения ошибки контрольной суммы замените модули на модули с текущей серией и ревизией. Если замена не возможна:
 1. откройте экран конфигурации процессора в вашем пакете программирования;
 2. переведя процессор в режим программирования, установите служебный бит пользователя 4 (S:26/4) в 1 (бит совместимости локальных блок-трансферов);
 3. переведите процессор из режима **PROGRAM** в режим **RUN**.
- Не запускайте инструкции **IIN** или **IOT** для модулей, установленных в одной и той же физической группе, пока вы не убедитесь, что не происходит передача блок-трансфера. Если же Вам необходимо это сделать, используйте команду **XIO** проверки бита .EN блок-трансфера, как условие включения инструкций IIN и IOT.

Просмотр каналов сканера удаленных Вх/Вых

Для просмотра каналов, сконфигурированных для режима сканера, используйте экран состояния режима сканера вашего пакета программирования. Этот экран отображает данные диагностического файла, определяемого Вами в экране конфигурации режима сканера (страница 6-8).

Scanner Mode		Channel 1B Status	
COUNTERS LOCKED			
Messages sent:	0	Messages sent with error:	0
Messages received:	0	Messages received with error:	0
Messages unable to receive:	0		
Rack Address	Starting Group	Rack Range Size	Fault Inhibit Reset Retry
1	0	FULL 010-017	I 0 0 0
2	0	1/4 020-021	0 0 0 0
2	2	3/4 022-027	F 0 0 0
3	0	1/2 030-033	0 0 0 0
3	4	1/4 034-035	0 0 0 0
3	6	1/4 036-037	0 0 0 0
17	0	FULL 170-177	0 0 0 0
1	0	FULL 010-017 *	1 0 0 0

отображает информацию о сообщениях, хранящиеся в файле диагностики этого канала

биты Fault, Inhibit и Reset, хранящиеся в файле состояния Вх/Вых

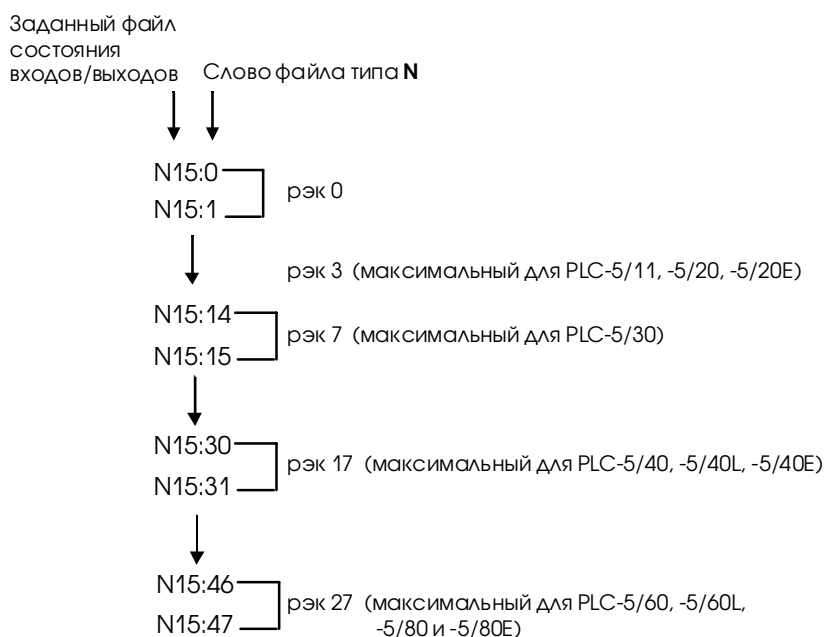
Область состояния	Расположение	Назначение
COUNTERS LOCKED		В этой области экрана отображается информация о том, что счетчики заблокированы. <ul style="list-style-type: none"> • Счетчики заблокированы: отображаются данные остановленных счетчиков, но счетчики продолжают работать. • Счетчики разблокированы: отображаются текущие величины счетчиков.
Message Counters	(Сообщения = Сообщения SDA + Сообщения SDN)	
sent	слово 1	Отображается номер сообщения, посланного каналом.
senr with error	слово 3	Отображается номер сообщения с ошибкой, посланного каналом.
received	слово 0	Отображается номер сообщения принятого каналом, в котором нет ошибки.
received with error	слово 2	Отображается номер сообщения, содержащего ошибку, и принятого каналом (например - неправильная контрольная сумма).

Область состояния	Расположение	Назначение
unable to receive	слово4	Отображает номер сообщения, в котором есть проблемы, связанные с протоколом передачи (такими, как неправильный байт состояния блок-трансфера, наряду с установленными битами чтения и записи.
Retry	от слова 5 и далее до слова 64	Показывает число попыток повторной связи с адресуемым рэком. Записи от 1 до 64.
Rack Address		В этом поле отображается номер рэка удаленного шасси, опрашиваемого каналом сканера: может сканироваться только рэк 3 (процессор PLC-5/11) 1-3 восьмеричный (процессор PLC-5/20, -5/20E) 1-7 восьмеричный (процессор PLC-5/30) 1-17 восьмеричный (процессор PLC-5/40, -5/40L, -5/40E) 1-27 восьмеричный (процессор PLC-5/60, -5/60L, -5/80, -5/80E) Если разрешено использование дополнительных (complementary) Вх/Вых, рэк дополнительных Вх/Вых обозначен символом C слева от адреса рэка.
Starting Group		В этой области показана первая сканируемая процессором группа модулей.
Rack Size		В этой области показана часть рэка Вх/Вых, адресуемая для каждого шасси. Можно сконфигурировать 1/4, 1/2, 3/4 или FULL (полный) рэк с учетом того, чтобы общая сумма рэков не превышала 8 групп Вх/Вых.
Range		Это поле отображает адрес рэка и группы модулей, сканируемых в рэке в списке сканирования. Звездочка (*) после цифры указывает, что это - последний правильно введенный адрес рэка.
Fault		Символ F в этой области указывает на сбой (ошибку) в опрашиваемом шасси. Символ F , отображаемый в этом поле, указывает, что соответствующий блок находится в режиме сбоя. Когда появляется индикация неисправности, система устанавливает связанный с ним бит неисправности в общем состоянии неисправности рэка (в экране состояния процессора вашего пакета программирования). Когда установлен глобальный бит неисправности рэка, вся информация конфигурации данного рэка теряется. При сбое рэка отображается F . Если для рэка установлены биты Fault (неисправность) и Inhibit (запрещение)для рэка, рэк отсутствует в этой группе Вх/Вых.
Inhibit		Для запрета рэка введите в область Inhibit данного рэка цифру 1. При запрете рэка процессор прекращает его сканирование. Вы можете запретить опрос текущего рэка, установив бит общего запрета для него в экране состояния процессора. Все шасси, входящие в состав данного рэка, будут запрещены и в поле Inhibit будет установлен символ I , показывающий, что для данного рэка установлен общий запрет.
Reset		Для запрета рэка введите в область Reset данного рэка цифру 1. Когда шасси перезапускается, процессор отключает выходы шасси независимо от установки переключателя последнего состояния. Вы можете перезапустить все шасси, устанавливая глобальный бит перезапуска шасси в экране состояния процессора. Все шасси, входящие в состав данного рэка будут перезапущены, и в поле Reset будет установлен символ R , показывающий, что для данного рэка установлен общий перезапуск.
Retry		В этой области отображается время повторного сканирования рэка.

Адресация файла состояния удаленных Вх/Вых

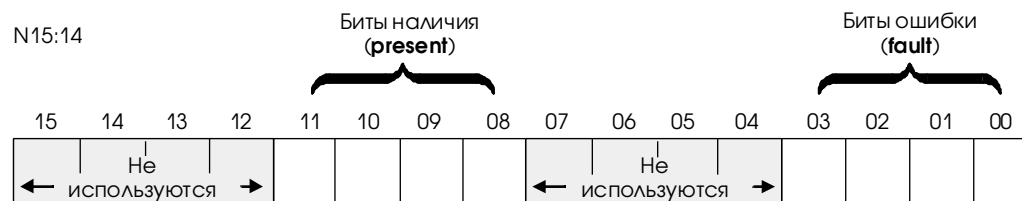
При выполнении программы Вы можете адресовать слова и биты неисправности в файле состояния входов/выходов. На рисунке 6.5 показано расположение слов в файле состояния входов/выходов для удаленного или расширенного локального рэка входов/выходов. В качестве примера приведен файл состояния N15.

Рисунок 6.5
Распределение слов в файле состояния Вх/Вых



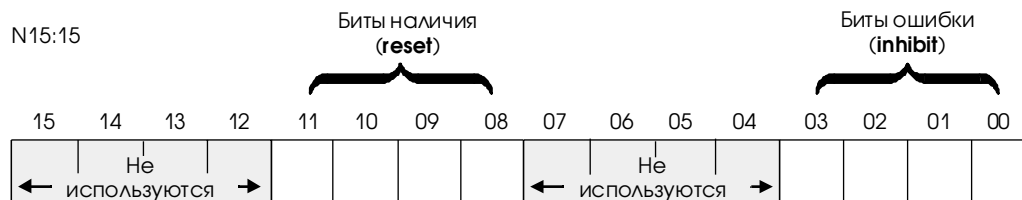
Первое слово рэка содержит биты наличия (**present**) и сбоя (**fault**), второе слово содержит биты запрета (**inhibit**) и перезапуска (**reset**). На рисунке 6.6 показано распределения битов наличия, сбоя, перезапуска и запрета для рэка 7 файла состояния входов/выходов.

Рисунок 6.6
Распределение битов первого слова выбранного рэка удаленных или локальных Вх/Вых



Бит:	Соответствует:
Биты ошибки	
00	первая 1/4 рэка начальная группа Вх/Вых 0
01	вторая 1/4 рэка начальная группа Вх/Вых 2
02	третья 1/4 рэка начальная группа Вх/Вых 4
03	четвертая 1/4 рэка начальная группа Вх/Вых 6
Биты наличия	
08	первая 1/4 рэка начальная группа Вх/Вых 0
09	вторая 1/4 рэка начальная группа Вх/Вых 2
10	третья 1/4 рэка начальная группа Вх/Вых 4
11	четвертая 1/4 рэка начальная группа Вх/Вых 6

Рисунок 6.6
Распределение битов второго слова выбранного рэка удаленных
или локальных входов/выходов



Бит: **Соответствует:**

Биты запрета

00	первая 1/4 рэка начальная группа Вх/Вых 0
01	вторая 1/4 рэка начальная группа Вх/Вых 2
02	третья 1/4 рэка начальная группа Вх/Вых 4
03	четвертая 1/4 рэка начальная группа Вх/Вых 6

Биты перезапуска

08	первая 1/4 рэка начальная группа входов/выходов 0
09	вторая 1/4 рэка начальная группа Вх/Вых 2
10	третья 1/4 рэка начальная группа Вх/Вых 4
11	четвертая 1/4 рэка начальная группа Вх/Вых 6



ВНИМАНИЕ: Когда Вы используете цикловую программу или пакет программирования для запрета и перезапуска рэка Вх/Вых, Вы должны установить или очистить биты сброса и запрета, соответствующие каждой четверти рэка в нужном шасси. Ошибка при установке всех соответствующих битов может вызывать непредсказуемую работу при сканировании только части Вх/Вых данного шасси.

Для заметок

Связь с PLC-5 через канал адаптера

Назначение этой главы

Сведения	Страница
Конфигурация канала связи адаптера к PLC-5	7-1
Программирование дискретных передач	7-8
Программирование блок-трансферов	7-8
Монитор статуса канала адаптера	7-15
Монитор статуса супервизорного процессора	7-16
Монитор удаленных Вх/Вых канала адаптера	7-17

Конфигурация канала связи адаптера к PLC-5

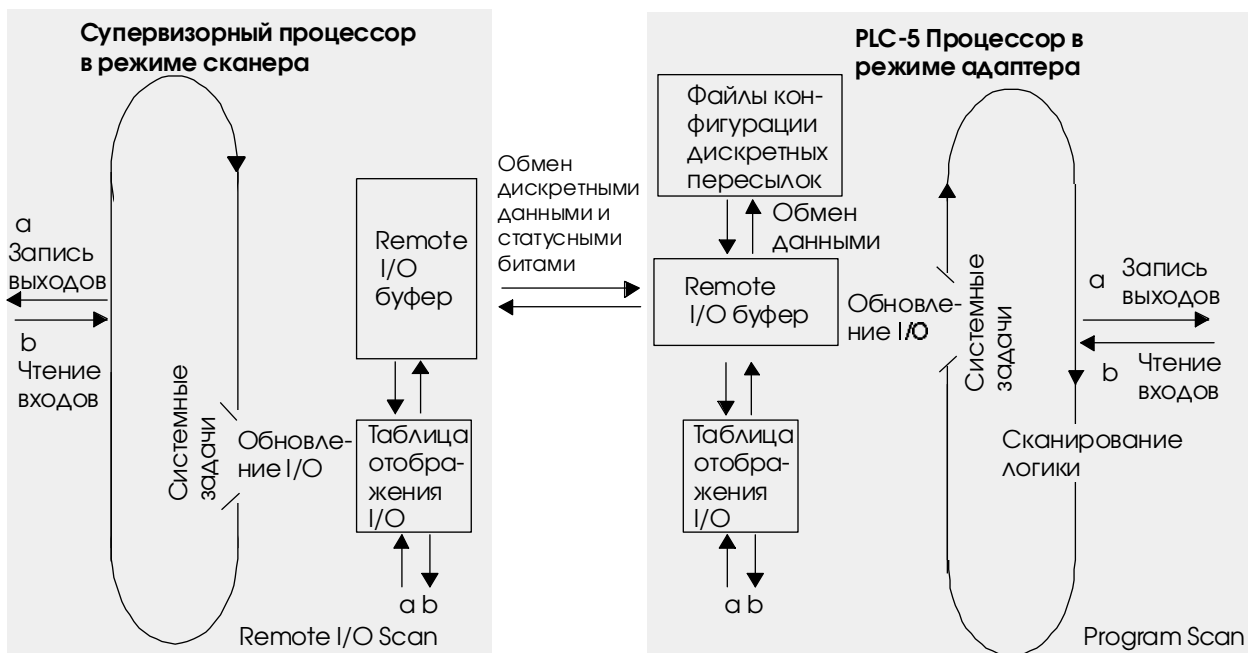
Эта глава объясняет, как сконфигурировать процессор для связи через канал адаптера:

1. Выбрать, какой канал сконфигурировать как адаптер;
2. Выбрать файл диагностики. Используйте отдельный неиспользуемый целочисленный файл;
3. Определить адрес рэка, стартовую группу и размер рэка. Адрес рэка по умолчанию 3;
4. Определить дискретные файлы передачи (вводится как десятичное число). Источник входов, то куда поступают выходные дискретные биты супервизорного процессора (по умолчанию 001-024 – десятичное представление рэка 3). Источник выходов, то куда поступают входы супервизорного процессора (по умолчанию 000-023);
5. Создать необходимые файлы управления блок-трансферами (один ВТР и один ВТW);
6. Сконфигурировать блок-трансферы так, чтобы супервизорный процессор знал, куда их адресовать.

Конфигурирование канала связи адаптера к PLC-5

Так как канал адаптера процессора PLC-5 более интеллектуален, чем модуль 1771-ASB, передача данных и задачи конфигурации для каналов адаптера обрабатываются по другому.

Супервизорный процессор или канал сканера и канал процессора в режиме адаптера автоматически обмениваются дискретными и статусными данными через удаленные Вх/Вых супервизорного процессора.



Во время каждого сканирования удаленных Вх/Вых, супервизорный процессор передает 2, 4, 6, или 8 слов в зависимости от конфигурации процессора в режиме адаптера 1/4, 1/2, 3/4, или полный рэк.

Процессор в режиме адаптера передает 2, 4, 6, или 8 слов в зависимости от конфигурации 1/4, 1/2, 3/4, или полный рэк.

Дискретные данные и блок-трансферы статусных битов передаются между таблицей отображения удаленных Вх/Вых сканера и каналом адаптера через файл конфигурации дискретных пересылок адаптера, который Вы определяете в экране конфигурации Adapter Channel PLC-5.

Для обеспечения связи канала сканера с каналом адаптера процессора PLC-5 необходимо выполнить следующее:

1. Задать скорость связи, адрес и размер рэка (количество слов передачи)
2. Сконфигурировать файл пересылки дискретных данных, через которые процессор получает и передает данные для супервизорного процессора
3. Если Вы планируете передачу данных на канал адаптера через блок-трансферы, назначьте и сконфигурируйте файл блок-трансферов
4. Подключите кабель удаленных Вх/Вых

Для более подробной информации смотри:

Стр. 7-3

Стр. 7-4

Стр. 7-8

Инструкция по установке
вашего процессора

Задание для канала адаптера скорости связи, адреса и размера рэка

Использование этой таблицы поможет вам определить какой канал процессора можно сконфигурировать как удаленный адаптер Вх/Вых:

Процессор	Каналы которые поддерживают режим адаптера удаленных Вх/Вых
PLC-5/111A	1A
PLC-5/20, PLC-5/20E	1B
PLC-5/30, PLC-5/40E, PLC-5/40L, PLC-5/80E, PLC-5/60L	1A, 1B
PLC-5/40, PLC-5/80, PLC-5/60	1A, 2A, 1B, 2B

Для определения канала в качестве адаптера, используйте экран конфигурации адаптера в вашем программном обеспечении:

```

Adapter Mode
Channel 1B Configuration

Diagnostics file:      N12
Baud rate:            57.6kB

Rack address:         3
Starting group:       0
Rack size:            FULL
Last Rack:           YES
Discrete Transfer Configuration:
  Input source:       001:024
  Output source:     000:024
} См. стр. 7-4

Group  Module      BTW control  BTR control
  0      0          BT000:000   BT000:000
  0      1          BT000:000   BT000:000
} См. стр. 7-8

```

Для поля	Назначить	Выполнить следующее
Diagnostics file (файл диагностики)	Файл который содержит статусную информацию канала	Установить курсор на поле и ввести номер целочисленного файла (9-999). ВНИМАНИЕ: Назначьте уникальный номер файла диагностики для каждого канала. Не назначайте в качестве файла диагностики файл статуса Вх/Вых или другой используемый целочисленный файл. В противном случае это может привести к повреждению машин. Важно: Вы должны назначить файл диагностики для канала независимо от использования, (даже если Вы не используете канал), если Вы хотите иметь статусную информацию этого канала.
Baud rate	Скорость связи для удаленных входов/выходов.	Установить курсор на поле и ввести желательную скорость. Допустимые скорости передачи: 57.6, 115.2 или 230.4 кбит/с.

Для поля	Назначить	Выполнить следующее
Rack address	Адрес рэка который относится к сканеру	Установить курсор на поле и ввести адрес. Допустимые адреса: <ul style="list-style-type: none"> • 3 восьмеричное (процессор PLC-5/11) • 1-3 восьмеричное (процессора PLC-5/20, -5/20E) • 1-7 восьмеричное (процессор PLC-5/30) • 1-17 восьмеричное (процессора PLC-5/40, -5/40E, -5/40L) • 1-27 восьмеричное (процессор PLC-5/60, -5/60L, -5/80, -5/80E) По умолчанию рэк 3.
Starting group	Номер рэка стартовой группы	Установить курсор на поле и ввести номер. Допустимые значения: 0, 2, 4 или 6
Rack size	Количество слов Вых./Вх. для обмена с управляющим процессором.	Установить курсор на поле и выбрать размер рэка, который зависит от того, какую стартовую группу Вы выбрали: Если Вы хотите использовать для связи: <ul style="list-style-type: none"> • 2 слова – выберите 1/4 (стартовая группа 6) • 4 слова – выберите 1/2 (стартовая группа 4) • 6 слов – выберите 3/4 (стартовая группа 2) • 8 слов – выберите FULL (стартовая группа 0) Например, если Вы выбрали 6 стартовую группу, Вы сможете передать только 2 слова. Если Вы выбрали 4 стартовую группу, Вы сможете передать 4 или 2 слова.
Last rack	Указывает управляющему процессору, что это последнее шасси. Эта информация важна, когда супервизорный процессор PLC-2	Установить курсор на поле и выбрать YES (ДА) или NO (НЕТ).

Задание файла конфигурации дискретных пересылок

Discrete Transfer Configuration:

Input source: 010:015

Output source: 011:020

↑
Файл
Данных 11

↑
Стартовое
слово 0
(десятичное)

Файл конфигурации дискретных пересылок (файл источника выходов и файл назначения входов) главное приспособляемое средство для передачи дискретных данных и блок-трансферов статусных битов между каналом адаптера PLC-5 и каналом сканера или супервизорным процессором (см. рис 7.1).

Файлы дискретных пересылок могут быть целочисленными десятичными или двоичными файлами. Убедитесь, что файлы источника входов и выходов созданы прежде, чем определять их. Если Вы не выполните своевременно конфигурацию, то при попытке принять изменения получите ошибку.

Примечание: Если созданы N11 и N10, файлы конфигурации дискретных пересылок будет целочисленными; если созданы D10 и D11 файлы конфигурации дискретных пересылок будет BCD.

Совет

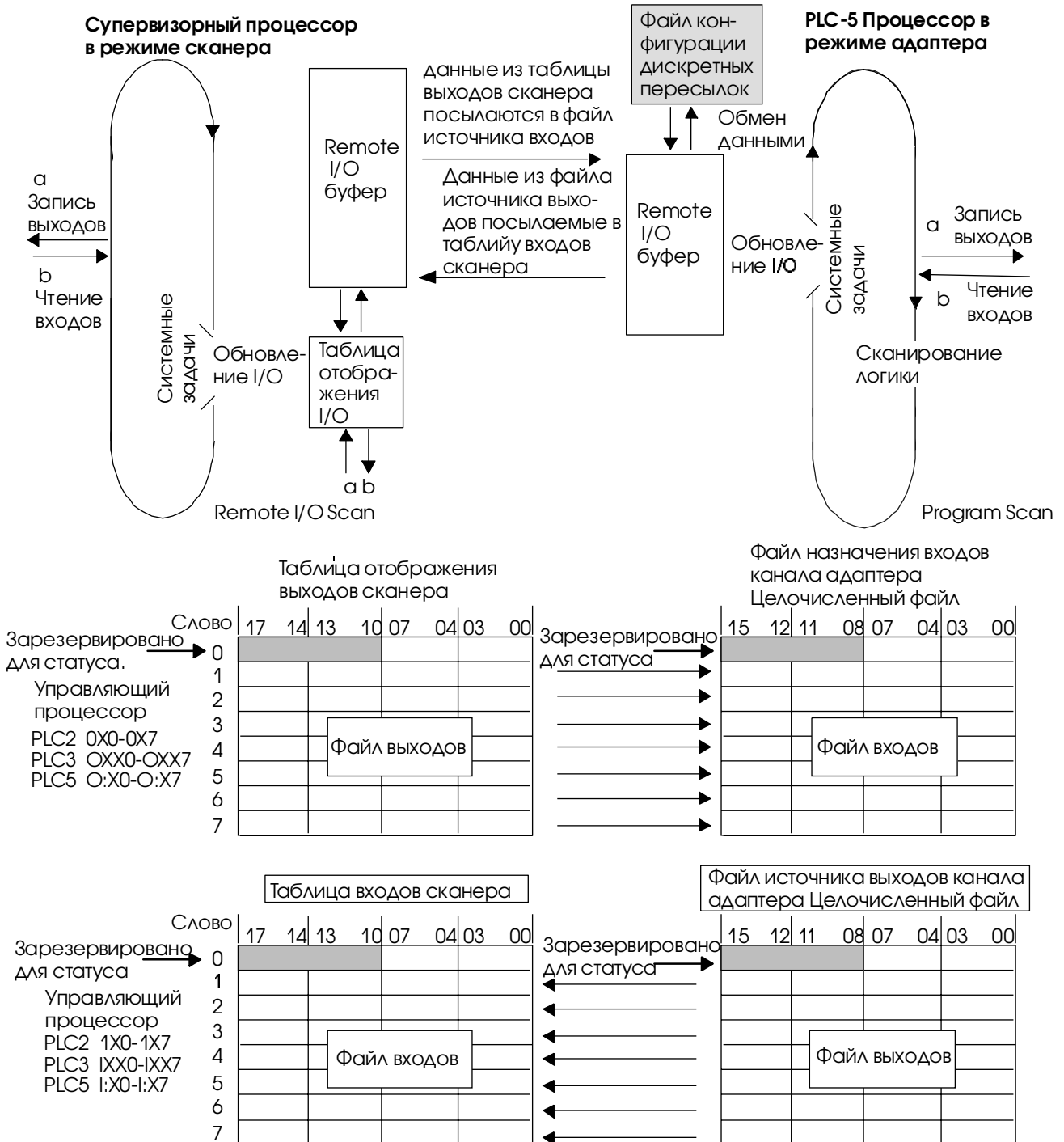
Конфигурируйте файлы дискретных пересылок как целочисленные файлы. Хотя процессор PLC-5 позволяет Вам использовать область входов или выходов, зарезервируйте ее для реальных Вх/Вых канала сканера. При этом, Вы избегаете возможность конфликта в дальнейшем при добавлении рэка, использующего ту же область Вх/Вых.

Важно:

Не делайте при конфигурации каналов адаптера, файл передачи дискретных входов как таблицу отображения входов. Вы рискуете очистить входы при выполнении автоконфигурации канала сканера на том же процессоре.

Входы адаптера не будут обновляться до тех пор пока обнаруженные изменения входных данных не будут посланы процессору.

Рис. 7.1
Обмен дискретными данными и блок-трансферами статуса между сканером и каналом адаптера удаленных Вх/Вых через файлы конфигурации дискретных пересылок.



Два, четыре, шесть или восемь слов данных могут передаваться между сканером и каналом адаптера. Количество передаваемых слов определяется размером рэка заданным в экране конфигурации канала адаптера.

Если данные с супервизорного процессора предназначены для управления выходами процессора в режиме адаптера, напишите релейно-контактную логику перемещения данных из файла входов в таблицу выходов в процессоре в режиме адаптера. Используйте для битовых данных инструкции XIC и OTE, а для слов – инструкции перемещения и копирования.

Если Вы хотите, чтобы супервизорный процессор читал данные из файла данных процессора в режиме адаптера, напишите релейно-контактную логику в процессоре в режиме адаптера для перемещения данных из своего выходного файла источника в таблицу входов супервизорного процессора.

Для создания Файлов конфигурации дискретных передач используйте экран конфигурации Adapter Mode Вашего программного обеспечения.

```

Adapter Mode
Channel 1B Configuration

Diagnostics file:      N12
Baud rate:            57.6kB

Rack address:         3
Starting group:       0
Rack size:            FULL
Last Rack:            NO
Discrete Transfer Configuration:
  Input source:        000:000
  Output source:       000:000

```

} Введите номер слова и
файла назначения входов
} Введите номер слова и
файла назначения выходов

Group	Module	BTW control	BTR control
0	0	BT000:000	BT000:000
0	1	BT000:000	BT000:000

Важно: Процессор определяет количество используемых слов согласно указанному Вами размеру рэка.

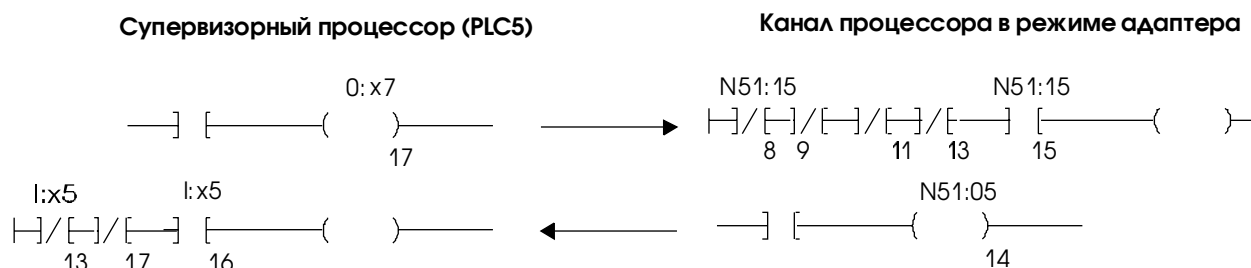
Для поля	Назначить	Выполнить следующее
Input destination	Место куда сканер помещает выходные слова во входной файл адаптера)	<ol style="list-style-type: none"> 1. Введите номер файла (десятичный) источника данных. 2. Введите количество слов (десятичное) источника данных. Задайте таблицу входов, таблицу выходов, как целочисленные десятичные или шестнадцатичные файлы. <p>Например: Если Вы используете файл N7:0 и полный рэк, сканер займет 8 слов в файле N7:0 слова с 0 по 7 (старшие байты первых слов под статусные данные).</p>
Output source	Место куда адаптер помещает слова дискретных выходов из файла дискретных входов сканера.	<ol style="list-style-type: none"> 1. Введите номер файла (десятичный) источника данных. 2. Введите количество слов (десятичное) для данных. Задайте таблицу входов, таблицу выходов, как целочисленные десятичные или шестнадцатичные файлы. <p>Например: Если Вы используете файл N7:10 и размер рэка полный, канал адаптера займет 8 слов в файле N7 слова с 10 по 17 (старшие байты первых слов под статусные данные)</p>



Для более подробной информации по конфигурации этого файла, смотри раздел конфигурации каналов, вашего программного обеспечения.

Программирование дискретных пересылок для режима адаптера

Обычно, каждая выходная инструкция в одном процессоре должна иметь соответствующую входную инструкцию в другом процессоре. Номер рэка канала процессора в режиме адаптера определяет адрес который Вы используете.



- N51 файл конфигурации дискретных пересылок процессора в режиме адаптера. Назначение входов и источники выходов определяются входными и выходными словами.
- Релейно-контактная логика супервизорного процессора использует номер рэка канала процессора в режиме адаптера.
- Создают релейно-контактную логику в процессоре адаптера с битами состояния (страница 7-15).

Программирование блок-трансферов для канала адаптера

Блок-трансферы в режиме адаптер по существу непрерывны. Как только передача завершена, другой блок-трансфер немедленно выполняется в процессоре в режиме адаптера; этот затем ждет (в буфере данных динамической выборки) выполнения другого запроса блок-трансфера. Следовательно, данные, которые перемещены после запроса - данные из предыдущей передачи. Если супервизорный процессор выполняет запрос блок-трансфера из процессора в режиме адаптера, каждые 500 мс, например, данные - по крайней мере 500 мс старые.

Супервизорный процессор содержит инструкции релейно-контактной логики, которые управляют фактической передачей связи. Однако, канал процессора в режиме адаптера управляет:

- фактическим числом слов данных, которые передаются;
- расположением в таблице данных, из которого данные передаются.

Важно: Не используйте инструкции блок-трансферов в релейно-контактной логике для канала процессора в режиме адаптера; конфигурируйте блок-трансферы из экранов конфигурации канала и монитора данных.

Конфигурирование запросов блок-трансферов

Чтобы сконфигурировать блок-трансферы на канал процессора в режиме адаптера, используйте экран конфигурации режима адаптера в вашего программного обеспечения.

		Adapter Mode													
		Channel 1B Configuration													
		Diagnostics file:	N12												
		Baud rate:	57.6kB												
		Rack address:	3												
		Starting group:	0												
		Rack size:	FULL												
		Last Rack:	NO												
		Discrete Transfer Configuration:													
		Input source:	000:000												
		Output source:	000:000												
Номер файла блок-трансфера	Номер элемента блок-трансфера	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: left;">Group</th> <th style="text-align: left;">Module</th> <th style="text-align: left;">BTR control</th> <th style="text-align: left;">BTR control</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>BT000:000</td> <td>BT000:000</td> </tr> <tr> <td>0</td> <td>1</td> <td>BT001:000</td> <td>BT000:000</td> </tr> </tbody> </table>	Group	Module	BTR control	BTR control	0	0	BT000:000	BT000:000	0	1	BT001:000	BT000:000	
Group	Module	BTR control	BTR control												
0	0	BT000:000	BT000:000												
0	1	BT001:000	BT000:000												

1. Определите необходимые Вам файлы управления BTR и BTR. Эти файлы управления должны уже существовать (должны быть в таблице памяти), иначе редактирование по окончании выдаст ошибку. Каждое слово управления должно содержать уникальный адрес блок-трансфера, чтобы правильно передать блок-трансфер.

A. Ввести номер файла блок-трансфера.

B. Ввести количество элементов.

C. Запишите на бумаге BT файлы, которые Вы определили. Это поможет Вам при конфигурировании BT в экране монитора данных.

2. Так как канал в режиме адаптера управляет расположением перемещенных данных а также количеством данных, загружайте данные в файлы блок-трансфера, используя данные экрана Монитор Данных Вашего программного обеспечения.

A. Определить файл управления BT, который Вы определили.

B. Ввести длину передачи в .RLEN.

C. Ввести файл и число элементов, из которых данные должны быть перемещены в .ФАЙЛ и .ELEM соответственно.

Пример:

Блок-трансфер, Записывающий 10 слов из файла 24, элемент 10 с файлом управления ВТ для группы 0, модуль, 0 из ВТ12:000, выглядит как:

Adapter Mode Configuration screen

Group	Module	BTW control	BTR control
0	0	BT02:000	BT000:000

Data Monitor screen

Address	EN	ST	DN	ER	CO	EW	NR	TO	RW	RLEN	DLEN	FILE	ELEM	R	G	M
BT12:000	0	0	0	0	0	0	0	0	0	10	0	24	10	0	0	0

Программируйте множество блок-трансферов в процессоре в режиме адаптера и соответствующие блок-трансферам в супервизорном процессоре согласно файлу контроля адаптера.

В этом примере, первый блок передачи в супервизорном процессоре использует слово управления ВТR перечисленное в группе 0 модуль 0, которая выглядит ВТ010:000.

ВТ10:0 указывает на файл 24 и элемент 10 и имеет длину 64 слова.

Принято, что файл 24 был создан как целочисленный файл, данные, записанные из первого блок-трансфера будут находится с N24:10 по N24:73. Второй блок-трансфер в супервизорном процессоре пишет данные в файл, на который указывает ВТ11:0, и третий блок-трансфер пишет данные в файл, на который ВТ11:40 указывает.

Вы можете иметь до 15 записей и 15 чтений. Каждый блок-трансфер к специфическому расположению группы/модуля использует адреса Вх/Вых битов состояния для того рэка /группы . Эти области не боступны для дискретных передач. Следовательно, если Вы конфигурируете все доступные 15 блоков чтения - записи блок-трансферами, никакие биты не будут доступны для дискретной передачи. Для более подробной информации см. страницу 7-14.

Важно: В режиме адаптера поблочная пересылка чтения, и поблочная пересылка записи для той же группы / модуля, должен иметь ту же самую длину.

Если Вы хотите передать локальные данные Вх/Вых резидентного процессора, канал которого находится в режиме адаптера на супервизорный процессор, или если Вы хотите передать локальные данные Вх/Вых от супервизорного процессора до резидентного процессора, канал которого находится в режиме адаптера, Вы должны использовать команды MOV или COP внутри процессора, канал которого находится в режиме адаптера, чтобы переместить данные в или из файла данных. Используйте файл управления блок-трансфера адаптера.

Пример блок-трансфера релейно-контактной логики

Для блок-трансферов релейно контактной логики в	См.
PLC-5 супервизорный процессор	Рис. 7.2
PLC-5/250 супервизорный процессор	Рис. 7.3

Рис. 7.2
Пример двунаправленного повторяющегося блок-трансфера в супервизорном процессоре PLC-5

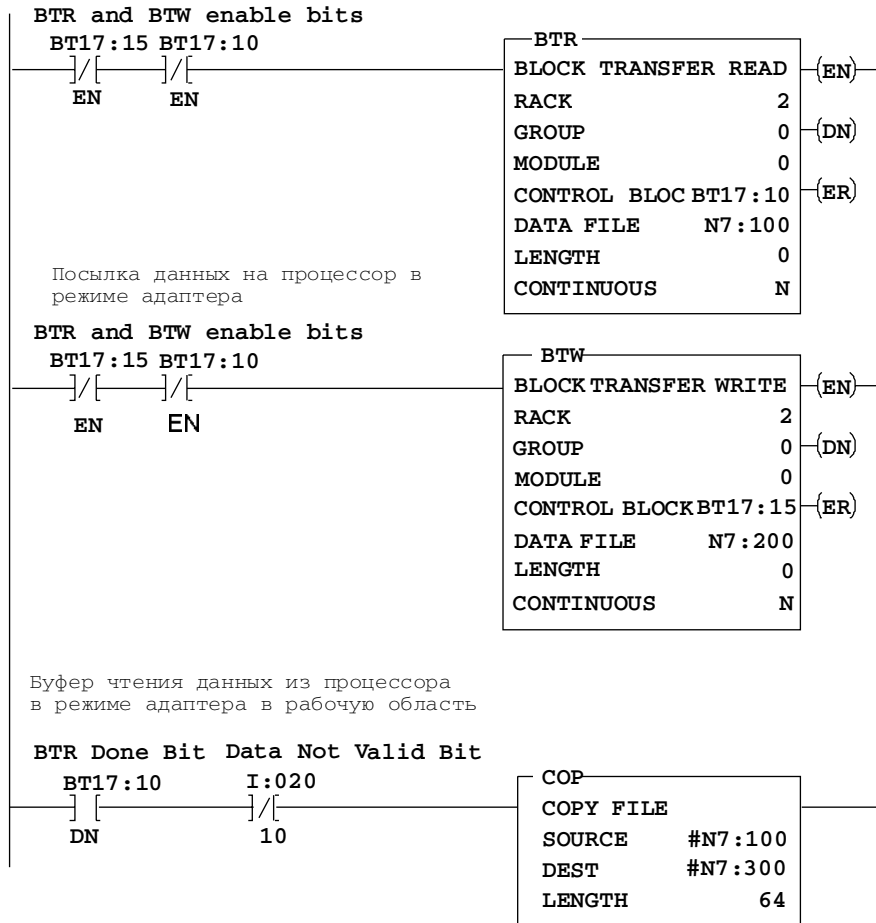
Чтение данных с процессора в режиме адаптера

Введите следующие параметры для инструкции блок-трансфера в супервизорном процессоре.

- Установите длину 0.
- Используйте номер удаленного рэка Вх/Вых, для которого Вы сконфигурировали процессор в режиме адаптера.
- Используйте номер группы и модуля, для которого Вы сконфигурировали процессор в режиме адаптера.
- Используйте в условии работы инструкции BTR бит разрешения чтения данных.

Все комментарии адреса для контактов, показанных в следующих примерах представляют состояние бита (1) в процессоре PLC-5.

Вам, вероятно, придется выполнять BTR в PLC-5 канала сканера дважды, если задержка времени BTR больше чем 2-3 скана программы. Если Вы не выполняете BTR дважды, BTR будет читать старые данные из процессора адаптера.



Канал процессора в режиме адаптера сконфигурирован как рэк 2.

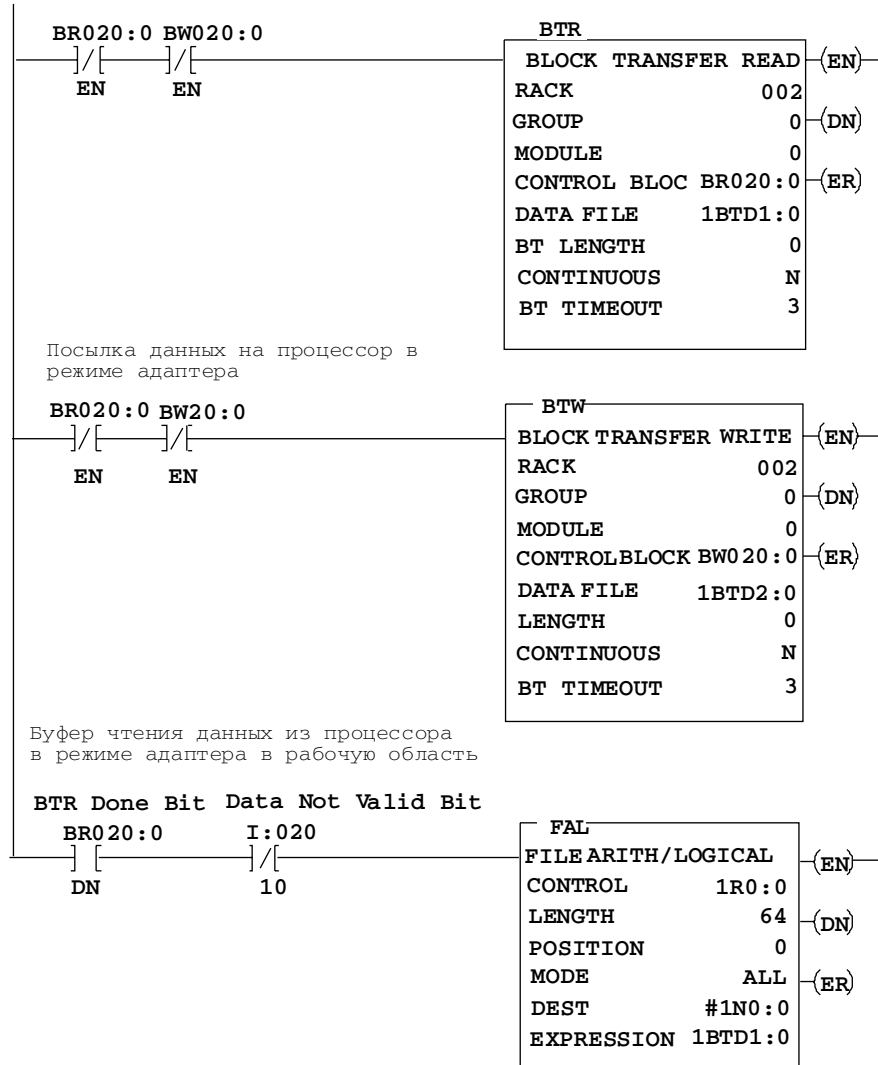
Рис. 7.3
Пример двунаправленного повторяющегося блок-трансфера в супервизорном процессоре PLC-5/250

Чтение данных с процессора в режиме адаптера

Введите следующие параметры в инструкции блок-трансфера в супервизорном процессоре.

- Установите длину 0.
- Используйте номер удаленного рэка Вх/Вых, для которого Вы сконфигурировали процессор в режиме адаптера.
- Используйте номер группы и модуля, для которого Вы сконфигурировали процессор в режиме адаптера.
- Используйте в условии работы инструкции BTR бит разрешения чтения данных.

Все комментарии адреса для контактов, показанных в следующих примерах представляют состояние бита (1) в процессоре PLC-5.



Канал процессора в режиме адаптера сконфигурирован как рэк 2.

Эффект программирования блок-трансферов на канал процессора в режиме адаптера для дискретных пересылок

Так как файлы конфигурации дискретных пересылок используются для передачи дискретных данных, также как для обмена статусными блок-трансферами между супервизорным процессором и каналом процессора в режиме адаптера, то выполняются многократные блок-трансферы пересылки данных к и из канала процессора в режиме адаптера.

Каждая группа/модуль, которая программируется как блок-трансфер канала адаптера, использует один байт в файле назначения входов канала адаптера.

Например:

Блок-трансфер запрос для группы 3, модуль 0 использует эти байты в файле. Этот байт теперь недоступен для дискретной передачи данных.

Будьте осторожны при планировании блок-трансферов и дискретных пересылок данных на канал процессора в режиме адаптера.



ВНИМАНИЕ: Используйте предостережение при выполнении передачи данных. Дискретные выходные данные записываются поверх блок-трансфера для группы/модуля. Если Вы пишете оба типа передачи к тому же самому слоту группы, возможны непредсказуемые действия машин, которые могут привести к повреждению оборудования или травмированию персонала.

Совет

Не программируйте блок-трансферы для группы 0, модуль 1, так как эта область файла конфигурации дискретных передач используется для обмена статусными данными между супервизорным процессором и каналом процессора в режиме адаптера. Например:

Монитор статуса канала адаптера

Супервизорный процессор получает биты состояния от процессора в режиме адаптера в слове 0 из таблицы отображения входов для рэка, которой процессор в режиме адаптера эмулирует.

Когда этот бит:	Установлен:	Это отображает состояние:
10 восьмеричное (8 десятичное)	1	данные не доступны
15 восьмеричное (13 десятичное)	1	процессор в режиме адаптера находится в программировании или в режиме теста
10 восьмеричное (8 десятичное) и 15 восьмеричное (13 десятичное)	0	Процессор в режиме адаптера находится в работе
10 восьмеричное (8 десятичное) и 15 восьмеричное (13 десятичное)	1	Процессор в режиме адаптера находится в программировании или в режиме тест

Пишите релейно-контактную логику в супервизорном процессоре, чтобы контролировать биты неисправности рэка для рэка канала процессора в режиме, эмулирующего состояние удаленной связи Vх/Вых.

Монитор статуса супервизорного процессора

Канал процессора в режиме адаптера резервирует биты 10-17 первого слова файла назначения входов для статусных данных. Эти биты сообщают процессору в режиме адаптера состояние супервизорного процессора и целостность линии связи удаленных Вх/Вых.



Когда этот бит:	Установлен:	Это отображает состояние:
10 восьмеричное (8 десятичное)	1	обнаружена авария связи или получена команда сброс из супервизорного процессора; будет установлен или бит 11 восьмеричный (9 десятичный) или бит 15 восьмеричный (13 десятичный).
11 восьмеричное (9 десятичное)	1	получена команда сброс из супервизорного процессора (процессор в режиме программирования или тест)
13 восьмеричное (11 десятичное)	1	обнаружено, что на супервизорном процессоре отключено питание, этот бит сбрасывается при первой связи с супервизорным процессором
15 восьмеричное (13 десятичное)	1	обнаружена авария связи (т.е. на сети удаленных Вх/Вых не обнаружено сетевой активности за последние 100 мсек

Монитор удаленных Вх/Вых канала адаптера

Чтобы контролировать каналы, которые сконфигурированы, чтобы поддерживать режим адаптера, используют экран состояния режима адаптера. Отображаемые данные сохранены в диагностическом файле, который Вы определили в экране конфигурации режима адаптера вашего программного обеспечения .

```

Adapter Mode
Channel 1B Status
COUNTERS LOCKED
Messages sent:          0      Messages sent with error:    0
Messages received:     0      Messages received with error: 0
Messages unable to receive: 0

Link timeout:          0      No scans received:          0
Mode changed:          0      Protocol fault:              0
Missed turn-around time: 0

```

Статусное поле:	Расположение:	Описание:
Counters Locked		<p>Это поле отображается в негативном виде, если счетчики заблокированы.</p> <ul style="list-style-type: none"> Счетчики заблокированы: дисплей счетчиков останавливает изменение, но счетчики продолжают выполняться в фоновом режиме. Счетчики разблокированы: система отображает текущее значение.
Message Counters		(СООБЩЕНИЕ = СООБЩЕНИЕ SDA + сообщения SDN)
sent	Слово 1	Отображает количество сообщений, посланных каналом.
sent with error	Слово 3	Отображает количество сообщений, содержащих ошибки, посланные каналом.
received	Слово 0	Отображает количество сообщений, не содержащих ошибки, полученные каналом.
received with error	Слово 2	Отображает количество сообщений, содержащих ошибки, полученные каналом.
unable to receive	Слово 4	Отображает количество сообщений, которые содержали ошибки протокола или, пакеты которые были искажены адаптером.
Adapter Mode Status Diagnostic Counters		
Link timeout	Слово 5	Отображает число тайм-аутов наступивших на удаленной связи Вх/Вых.
No scans received	Слово 6	Отображает сколько раз канала адаптера не получал пакет, адресованный себе.
Mode changed	Слово 7	Отображает количество включений канал адаптера в режим онлайн.
Protocol fault	Слово 8	Отображает число недопустимых сообщений Вх/Вых полученных каналом адаптера.
Missed turn-around time	Слово 9	Отображает сколько раз канал адаптера использовал более чем 2 мс, чтобы обработать пакет сообщения. Время цикла передачи получения сообщения порядка - 2 мс.

Для заметок: _____

Подключение локальных расширенных Вх/Вых

Назначение этой главы

Сведения	Страница
Выбор устройств для подключения	8-1
Кабельные соединения	8-2
Размещение и адресация Вх/Вых	8-2
Передача данных	8-4
Конфигурирование процессора как сканера локальных расширенных Вх/Вых	8-9
Мониторинг статуса локальных расширенных Вх/Вых	8-13

В этом разделе объясняется, как конфигурировать процессор для связи с локальными расширенными Вх/Вых:

1. Выбор канала для конфигурации локальных расширенных Вх/Вых.
2. Определение файла диагностики.

Используйте уникальный неиспользуемый номер файла.

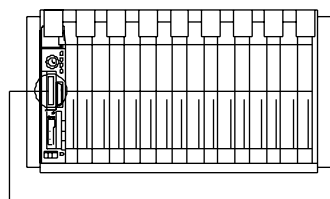
3. Определение списка сканирования.

Выбор устройств для подключения

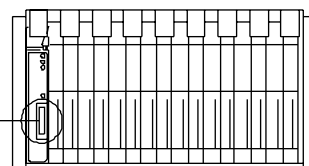
Единственными устройствами, с помощью которых можно сформировать локальные расширенные Вх/Вых, являются процессоры PLC-5/40L и PLC-5/60L с модулем адаптера локальных расширенных Вх/Вых.

Процессор локальных расширенных Вх/Вых не может быть адаптером локальных расширенных Вх/Вых.

Процессор PLC-5/40L или PLC-5/60L



Адаптер локальных расширенных Вх/Вых 1771-ALX



Связь локальных расширенных Вх/Вых

Кабельные соединения

Совет

Максимальная длина кабеля для подключения систем локальных расширенных Вх/Вых составляет 30.5 м (100 ft). Подключение адаптеров локальных-расширенных Вх/Вых производится одним из следующих кабелей:

Длина кабеля	Номер по каталогу
1 м (3.3 ft)	1771-СХ1
2 м (6.6 ft)	1771-СХ2
5 м (16.5 ft)	1771-СХ3

Важно : Недопустимо соединение или сращивание кабеля для получения желаемой длины. Например, при расстоянии 4 м между двумя локальными адаптерами расширенных Вх/Вых или между процессором и адаптером расширенных Вх/Вых, недопустимо использование 2-х кабелей по 2 м каждый. Вам следует применить кабель длиной 5 м, при этом 1 м будет запасом.

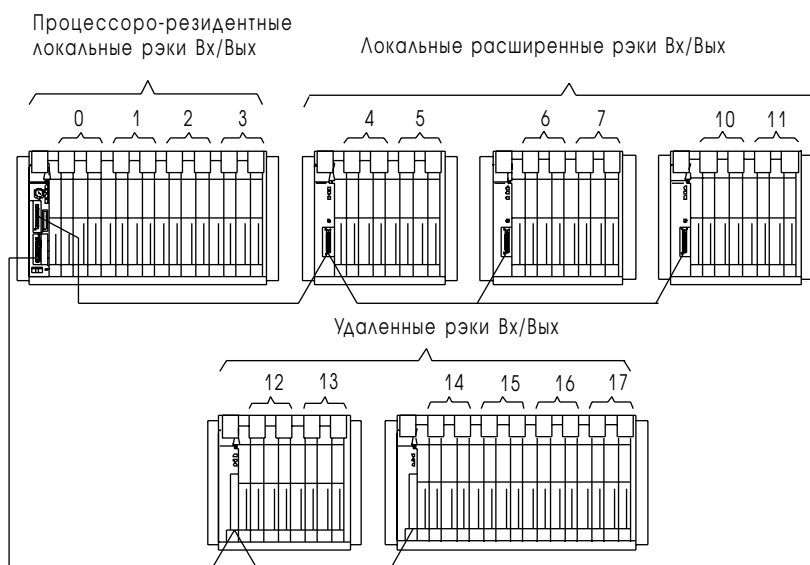
На последнем модуле адаптера установите локальный терминатор (1771-СХТ), без которого система не будет работоспособной. Терминатор поставляется вместе с процессором.

Размещение и адресация Вх/Вых

Совет

В случае, когда процессор PLC-5/40L или -5/60L используется для сканирования как локальных расширенных, так и удаленных рэков Вх/Вых, суммарное число удаленных и локальных расширенных рэков не должно превышать максимально допустимое число для данного процессора (16 рэков для PLC-5/40L и 24 рэка для PLC-5/60L). На рис. 8.1 показан процессор PLC-5/40L, контролирующий локальные рэки расширения и удаленные рэки Вх/Вых.

Рис. 8.1
Процессор PLC-5/40L, способный адресовать 16 рэков
(разделенных между локальными расширенными и удаленными Вх/Вых)



Замечание : Нумерация рэков не обязательно должна быть последовательной. Например, удаленные рэки Вх/Вых могут иметь номера 6, 7, 14, 15, 16 и 17, в то время как, локальные расширенные рэки могут нумероваться так: 4, 5, 10, 11, 12 и 13.

8584

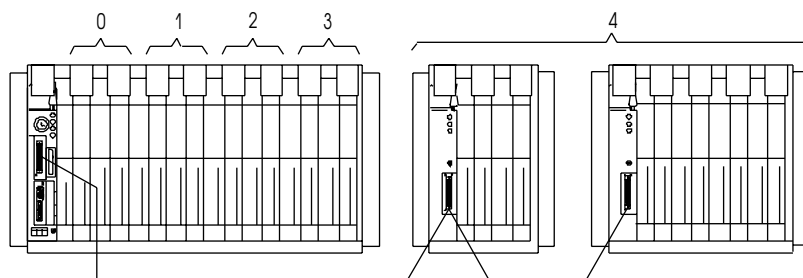
Процессор PLC-5 и модуль адаптера автоматически предоставляет следующий по счету номер рэка оставшимся группам Вх/Вых шасси. Например, при использовании в 16-слотовом шасси (1771-A4В) 1/2-слотовой адресации в резидентном шасси процессора, процессор назначит этому шасси номера рэков 0, 1, 2 и 3.

Совет

При назначении номеров рэкам локальных расширенных Вх/Вых следуйте рекомендациям:

- не используйте рэк одновременно для локальных расширенных и удаленных Вх/Вых. Например, если вы используете часть рэка для удаленных Вх/Вых, оставшуюся часть рэка нельзя использовать для локальных расширенных Вх/Вых. См. рис. 8.2;
- возможно распределение локальных расширенных рэков по нескольким шасси, находящимся на шине локальных расширенных Вх/Вых. См. рис. 8.2;

Рис. 8.2
Назначение номера рэка локальных расширенных
Вх/Вых нескольким шасси Вх/Вых



- допустимо использовать различные методы аппаратной адресации для каждого из локальных расширенных шасси Вх/Вых;
- нельзя сконфигурировать более одного рэка с одним и тем же начальным номером рэка и группой модуля, т.е. вы не можете использовать комплементарные шасси Вх/Вых.

При планировании использовать локальные расширенные Вх/Вых, следуйте данным указаниям.

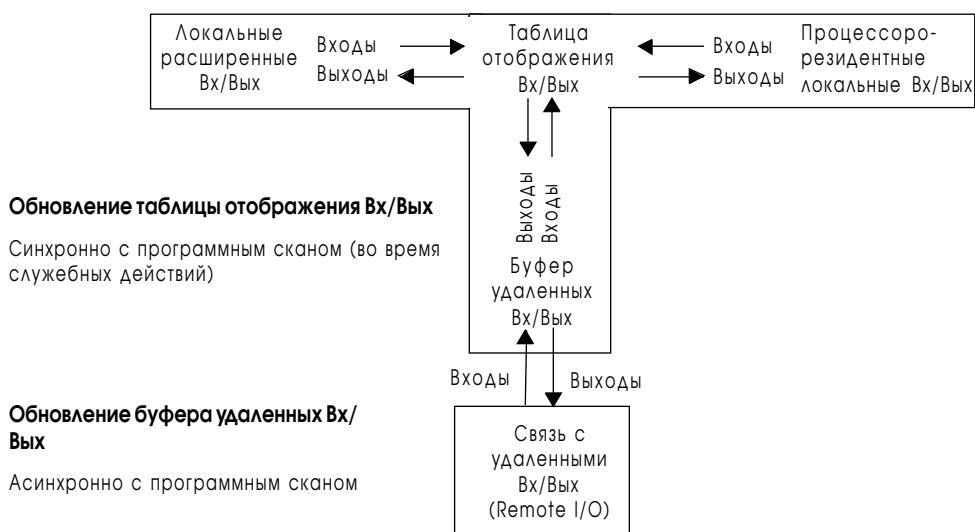
- Не конфигурируйте входы локального расширенного шасси в качестве входов прерывания процессора (processor input interrupts - PI). Последние могут находиться только в резидентном локальном рэке процессора.
- В локальном рэке расширения Вх/Вых допустимо использование 32-точечных Вх/Вых модулей, либо модулей 1771-IX или -IY, с применением любого из методов адресации. Тип используемых модулей должен быть указан путем установки конфигурационной перемычки на адаптере локальных расширенных Вх/Вых.
- При необходимости совместного использования в одном шасси термопарного модуля и модулей Вх/Вых на 32 точки, применяйте модуль 1771-IXE.

Передача данных

Процессоры PLC-5/40L или -5/60L могут сканировать резидентные Вх/Вых процессора, локальные расширенные Вх/Вых и удаленные Вх/Вых.

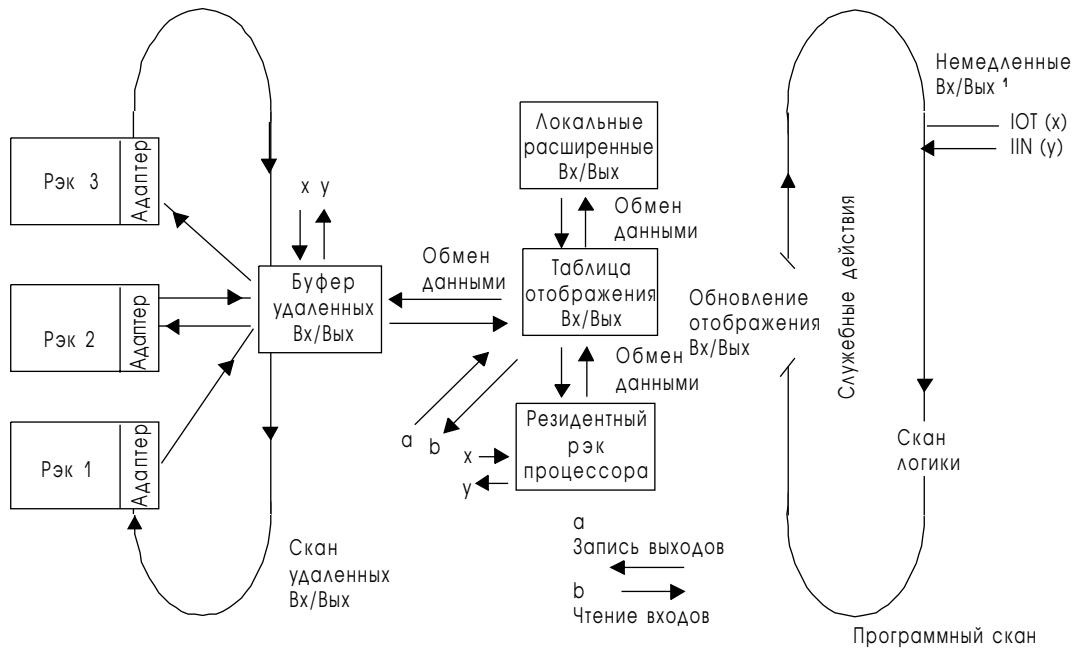
На рис. 8.3 показан процесс выполнения сканирования и обновления Вх/Вых процессорами PLC-5/40L и -5/60L.

Рис. 8.3
Сканирование и обновление данных PLC-5/40L и -5/60L



Передача дискретных данных

Процессоры сканируют локальные расширенные Вх/Вых во время служебной части программного скана.
Дискретные данные локальных расширенных Вх/Вых передаются между таблицей отображения процессора и входами/выходами локальных расширенных шасси.

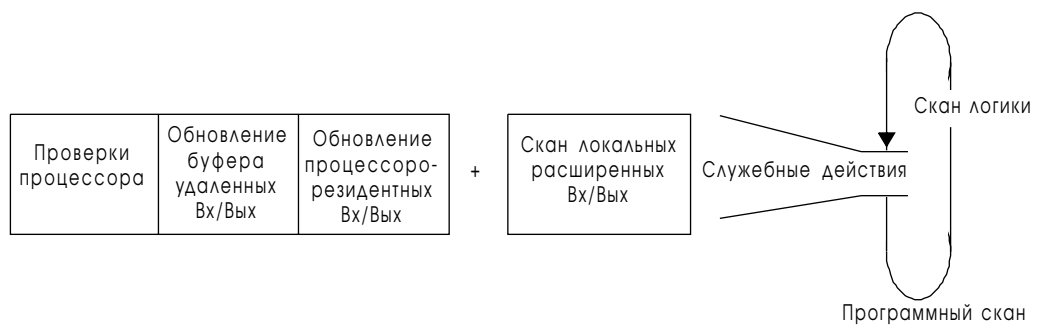


Обмен данными происходит во время служебных действий.
Во время скана логики происходит запись выходов в таблицу отображения и чтение входов из таблицы отображения.

¹ IIN и I/OТ передают данные непосредственно в модули выходов и из модулей входов локальных расширенных Вх/Вых шасси.

Время, затрачиваемое на сканирование локальных расширенных Вх/Вых шасси, добавляется ко времени служебных действий. См. рис. 8.4.

Рис. 8.4
Время сканирования локальных расширенных Вх/Вых PLC-5/40L и -5/60L



Время, затрачиваемое на сканирование локального расширенного шасси Вх/Вых зависит от числа модулей адаптеров 1771-ALX и числа расширенных рэков Вх/Вых. Вычисление суммарного времени сканирования локальных расширенных шасси Вх/Вых производится по формуле:

Время сканирования локальных расширенных Вх/Вых= $(0.32 \text{ мс} \times A) + (0.13 \text{ мс} \times L)$,

где :

A = число модулей 1771-ALX ;

L = число рэков в системе локальных расширенных Вх/Вых

Пример : Если вы имеете три модуля 1771-ALX, установленных в трех шасси, и в сумме 4 рэка, то полное время сканирования вычисляется следующим образом:

Время сканирования локальных расширенных Вх/Вых= $(0.32 \text{ мс} \times 3) + (0.13 \text{ мс} \times 4)=1.48 \text{ мс}$

Время, затрачиваемое на служебные операции= 1.48 мс (локальные расширенные Вх/Вых) + 4.50 мс (др.служебные операции)= 5.98 мс

Передача блоков данных

Запросы на передачу данных блок-трансферов выставляются в течение скана логики. Одновременно с выполнением логики программы, запросы на передачу данных блок-трансферов направляются соответствующему модулю(модулям) адаптера 1771-ALX, которые и передают данные. Модуль адаптера 1771-ALX может начать операции по передаче блоков данных с несколькими слотами и производит эти операции параллельно с работой шасси Вх/Вых.

Продолжительность процесса блок-трансферов не влияет на время скана логики. Такая передача данных происходит одновременно с выполнением логики программы.

Продолжительность операций блок-трансферов - это интервал времени между активизацией инструкции блоковой передачи и получением бита о выполнении.

Расчет времени выполнения блок-трансферов

Существует 2 характеристики времени выполнения блок-трансферов:

- оценка наихудшего времени выполнения всех блок-трансферов системы;
- время исполнения блок-трансфера для любого модуля блок-трансфера в системе.

Оценка наилучшего времени выполнения

Продолжительность block-transfer (мс) = D x R

$$D = 2E \times L + (0.1 W) \quad R = \frac{\text{скан логики+служебные действия}}{\text{скан логики}}$$

В формуле предполагается:

- инструкции блок-трансферов расположены последовательно в логике программы;

- модули блок-трансферов имеют готовность для выполнения запроса;

или

R=1 (если D < времени скана логики)

где :

E = число локальных расширенных шасси Вх/Вых, содержащих модули блок-трансферов;

L = наибольшее число модулей блок-трансферов в любом из локальных расширенных шасси;

W = количество слов в самом длинном запросе блок-трансфера.

Время исполнения для любого модуля блок-трансфера

Продолжительность блок-трансфера (мс) = D x R

$$D = [2E \times M + (0.1 W)] \quad R = \frac{\text{скан логики+служебные действия}}{\text{скан логики}}$$

В формуле предполагается:

- инструкции блок-трансферов расположены последовательно в логике программы;

- модули блок-трансферов имеют готовность для выполнения запроса;

или

R=1 (если D < времени скана логики)

где :

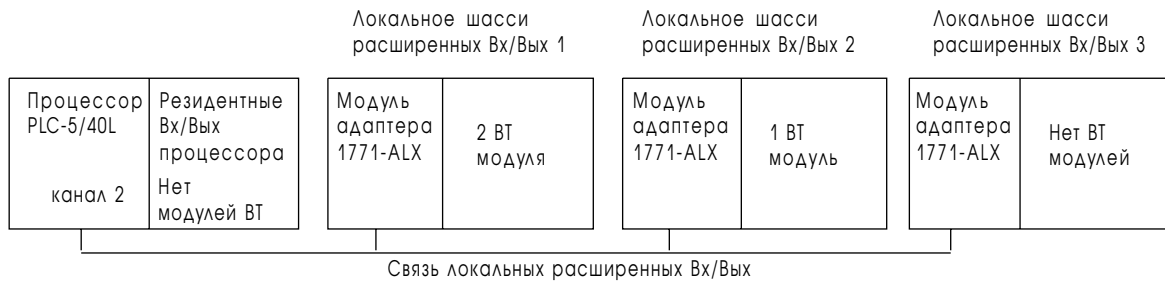
E = число локальных расширенных шасси Вх/Вых, содержащих адаптеры 1771-ALX и модули блок-трансферов;

M = количество модулей блок-трансферов в шасси, для которого производится расчет;

W = количество слов в запросе на передачу блока.

Пример вычислений :

Ниже рассмотрена система, для которой рассчитаны оценка наихудшего времени выполнения и время выполнения для модулей из шасси 2.



Скан логики выполняется за 15 мс. Служебные действия занимают приблизительно 6 мс (как посчитано по формуле на стр. 8-6).
Наибольший запрос на блоковую передачу требует 20 слов.

ВТ-блок-трансфер

Оценка наихудшего времени выполнения (T)=D x R

$D = 2E \times L + (0.1W)$ и $R = 1$, т.к. $10 < 15$ (время скана логики)

D (мс) = $(2 \times 2) \times (2) + (0.1 \times 20)$

$D = 10$ мс

$T = 10 \times 1$

$T = 10$ мс

Время выполнения (T) для модуля из шасси 2: D x R

Длина передаваемого блока=20

$D = 2E \times M + (0.1W)$ и $R = 1$, т.к. $6 < 15$ (время скана логики)

D (мс) = $(2 \times 2) \times (1) + (0.1 \times 20)$

$D = 6$ мс

$T = 6 \times 1$

$T = 6$ мс

Совет

Требования для локальных расширенных рэков

Здесь даны некоторые соображения по программированию передачи блоков данных в локальный рэк расширения :

- ниже дан перечень модулей, которые будучи размещены в локальном расширенном рэке вызовут часто повторяющуюся ошибку контрольной суммы при выполнении инструкции блок-трансфера :
 - модули 1771-OFE1, -OFE2, -OFE3 всех версий, предшествующих серии В, ревизии В;
 - все версии модулей IMC-120 и IMC-123.
- при использовании блок-трансферов к расположенному в локальном расширенном шасси модулю 2760-RB, убедитесь , что **бит тайм-аута** в файле управления блок-трансфера **не установлен**.

Конфигурирование процессора в качестве сканера локальных расширенных Вх/Вых

Для конфигурирования локальных расширенных Вх/Вых (канал 2) используйте экран конфигурации локальных Вх/Вых расширения.

Extended Local I/O

Channel 2 Configuration

Diag. file N100

Rack	Starting	Chassis	Backplane	Range
Address	Group	Size	Addressing	
1	0	16-SLOT	1-SLOT	010-027
3	0	4-SLOT	2-SLOT	030-031
3	2	12-SLOT	1-SLOT	032-045 *

Это поле : **Описывает :** **Конфигурируется выполнением следующих действий :**

Diag. file Файл, содержащий информацию о статусе канала Укажите курсором на это поле и введите номер файла (9-999).

Внимание : Для каждого канала следует назначать отдельный файл диагностики. Не назначайте файл, назначенный вами в качестве файла статуса Вх/Вых или уже используемый вами целочисленный файл. Это может вызвать непредсказуемую поломку оборудования.

Важно : Даже в случае неиспользования канала (например, если вы хотите получать только статусную информацию), все равно, вы должны определить файл диагностики.

Scan list Конфигурация канала Вх/Вых Как создавать и модифицировать список сканирования см. следующий параграф.

Список сканирования локальных Вх/Вых расширения подобен списку сканирования удаленных Вх/Вых. Различия заключаются в следующем :

- список сканирования удаленных Вх/Вых показывает размер рэка. Размер рэка определяется числом слотов и методом адресации соединительной платы шасси. Таблица 8.А объясняет отношения между размером шасси, методом адресации и размером рэка.

Таблица 8.А

Как размер шасси и метод адресации определяют число рэков Вх/Вых

Если вы используете этот размер шасси :	И 2-слотовую адресацию (одиночной емкости)	Или 1-слотовую адресацию (двойной емкости)	Или 1/2-слотовую адресацию (четверной емкости)
4-слота	1/4 логического рэка	1/2 логического рэка	1 логический рэк
8-слот	1/2 логического рэка	1 логический рэк	2 логических рэка
12-слот	3/4 логического рэка	1+1/2 логического рэка	3 логических рэка
16-слот	1 логический рэк	2 логических рэка	4 логических рэка

Список сканирования локальных расширенных Вх/Вых показывает не размер рэка, а актуальный размер шасси и метод адресации.

- список сканирования локальных расширенных Вх/Вых имеет отдельное значение для каждого из физических шасси канала.

Если адаптер канала удаленных Вх/Вых размещен в физическом шасси, содержащем более одного рэка Вх/Вых, список сканирования удаленных Вх/Вых этого шасси будет содержать более одной позиции.

Рис. 8.5 показывает список сканирования для удаленных Вх/Вых и локальных расширенных Вх/Вых. Для каждого канала показано 16-слотовое шасси с использованием 1-слотовой адресации со стартовым адресом рэка 4, группой модулей 0.

Данное шасси содержит логические рэки 4-5.

Рис. 8.5

Список сканирования удаленных Вх/Вых и локальных расширенных Вх/Вых

Удаленные Вх/Вых (Remote I/O)				Локальные расширенные Вх/Вых (extended-loc I/O)				
Rack ##	Starting Group	Rack Size	Range	Rack Address	Starting Group	Chassis Size	Backplane Addressing	Range
4	0	FULL	040-047	4	0	16-slot	1-slot	040-057
5	0	FULL	050-057					

Список сканирования состоит из следующего :

Для этого поля :	Список сканирования содержит :
Scan rack address (Адрес сканируемого рэка)	1-17 в восьмеричном коде (процессоры PLC-5/40L) 1-27 в восьмеричном коде (процессоры PLC-5/60L)
Starting group number (Начальный номер группы)	0, 2, 4 или 6
Chassis size (Размер шасси)	4-слота, 8-слот, 12-слот, 16-слот
Backplane addressing (Метод адресации)	1-слотовая, 2-слотовая или 1/2-слотовая
Range (Диапазон)	Автоматически вычисляется на основе адреса рэка, начального номера модульной группы и размера шасси. Звездочка (*) после диапазона означает последнее допустимое значение рэка.

Совет

При создании/модификации вашего списка сканирования имейте в виду следующие ограничения :

- список сканирования может содержать только 16 значений, т.к. на канале 2 может находиться только 16 адаптеров;
- автоматическая конфигурация всегда показывает текущую аппаратную конфигурацию, за исключением шасси, имеющим установленный глобальный бит блокировки (global inhibit bit). В этом случае, указанный глобальный бит перекрывает установки автоматической конфигурации. Для начала, вы должны очистить глобальные биты блокировки всех шасси канала и, затем, использовать функцию автоконфигурации;
Очистка глобального бита блокировки производится на экране статуса процессора;
- список сканирования может содержать максимум 16 шасси. Значения списка сканирования не могут дублироваться.

Для создания/модификации вашего списка сканирования используйте следующую таблицу.

Чтобы :	Делайте следующее :
Создать список сканирования	<p>Удостоверьтесь, что процессор находится в режиме Remote Program или Program.</p> <ol style="list-style-type: none"> 1. Проверьте создание статусного файла на экране конфигурации процессора. 2. Введите любые изменения, сделанные в конфигурации канала. 3. Используйте функцию автоконфигурации. <p>Если во время ввода изменений возникли ошибки, очистите список сканирования и введите изменения снова.</p> <p>Если некоторые из адаптеров не видны в списке сканирования, проверьте, что они включены и что каналы подсоединены соответствующим образом. Проверьте, также, правильную установку всех переключателей на адаптерах.</p>
Вставить позицию (entry) в список сканирования	<p>Удостоверьтесь, что процессор находится в режимах Remote Program, Program или Remote Run.</p> <ol style="list-style-type: none"> 1. Позиционируйте курсор на месте списка сканирования, куда вы хотите вставить позицию (entry). 2. Вставьте в список сканирования позицию (entry) и заполните лист соответствующими значениями. <p>Важно : При вводе некорректной информации, после процесса сохранения информации, процессор не покажет новую конфигурацию.</p>
Удалить позицию (entry) из списка сканирования	<p>Удостоверьтесь, что процессор находится в режимах Remote Program, Program или Remote Run.</p> <ol style="list-style-type: none"> 1. Позиционируйте курсор на месте списка сканирования, где вы хотите удалить позицию (entry). 2. Удалите позицию (entry). <p>Важно : При вводе некорректной информации, после процесса сохранения информации, процессор не покажет новую конфигурацию.</p>

Мониторинг статуса локальных расширенных Вх/Вых

Для наблюдения за статусом локальных Вх/Вых расширения процессоров PLC-5/40L и PLC-5/60L используйте экран статуса локальных расширенных Вх/Вых вашего программного обеспечения.

Extended Local I/O
Channel 2 Status

COUNTERS LOCKED

Channel retry counter : 999

Rack Address	Starting Group	Chassis Size	Backplane Addressing	Range	Fault	Inhibit	Reset	Retry
3	2	4-SLOT	1-SLOT	032-035 *	F	1	1	10000
0				000-000	0	0	0	0
0				000-000	0	0	0	0
0				000-000	0	0	0	0
0				000-000	0	0	0	0
0				000-000	0	0	0	0

Состояние счетчиков сохраняется в определенном для этого канала файле диагностики

Таблицы рэков Вх/Вых сохраняются в файле статуса Вх/Вых

Поле статуса	Место	Описание
Counters Locked		Если счетчики заблокированы, это поле показано в негативе. •Счетчики заблокированы : индикация состояния счетчиков остановлена, но счетчики продолжают фоновый счет. •Счетчики разблокированы : система показывает текущие значения.
Channel retry	слово 0	Показывает число неудачных попыток связи сканера локальных Вх/Вых расширения со всеми адаптерами канала.
Retry	слово 10 слово 20 слово 30 и т.д. слово 160	Показывает число попыток связи с соответствующим рэком (номера слов умножаются на 10). Позиция рэка (entry) 1 Позиция рэка (entry) 2 Позиция рэка (entry) 3 и т.д. Позиция рэка (entry) 16
Rack Address		Поле показывает адреса рэков локальных расширенных Вх/Вых в скан листе.
Starting Group		Показывает начальную группу модулей в сканируемом рэке.
Chassis Size		Показывает размер шасси Вх/Вых первой позиции (entry) списка сканирования.
Backplane Addressing		Поле показывает положения переключателей 5 и 6 соединительной платы шасси (с помощью которых устанавливается метод адресации шасси).
Range		Показывает адрес просканированного рэка и номера группы. Звездочка (*) говорит, что это последняя допустимая позиция рэка (rack entry).
Fault		Символ F указывает на ошибку соответствующего шасси. При появлении символа ошибки, система устанавливает бит ошибки в глобальном статусе рэка на экране статуса процессора. Когда глобальный бит ошибки рэка установлен, теряется вся информация о конфигурации части оборудования содержащей ошибку. Ошибка рэка индицируется символом F. Если установлены оба бита FAULT и INHIBIT, в этой группе Вх/Вых не существует рэка.

Поле статуса	Место	Описание
Inhibit		<p>Для блокировки рэка позиционируйте курсор на поле INHIBIT и введите "1".</p> <p>Когда шасси заблокировано, процессор прекращает его сканирование. Вы можете заблокировать рэк целиком путем установки в "1" глобального бита блокировки этого рэка на экране статуса процессора. Все шасси этого рэка будут заблокированы, что индицируется появлением "I" в поле INHIBIT (рэк глобально заблокирован).</p>
Reset		<p>Для перезапуска рэка позиционируйте курсор на поле RESET и введите "1".</p> <p>При этом процессор сбрасывает все выходы шасси независимо от положения переключателя последнего состояния. Вы можете сбросить весь рэк путем установки глобального бита сброса рэка в "1" на экране статуса процессора. При этом, все шасси этого рэка будут сброшены. О чем будет свидетельствовать появление символа "R" в поле RESET (глобальный перезапуск рэка).</p>
Retry		<p>Поле индицирует число раз, которое рэк был пересканирован заново.</p>

Оптимизация производительности системы

Назначение этой главы

Сведения	Страница
Сканирование программы	9-1
Расчет производительности	9-5
Задержка входных и выходных модулей	9-5
Пересылка через монтажную плату Вх/Вых	9-5
Время сканирования удаленных Вх/Вых	9-6
Время процессора	9-10
Пример расчета	9-11
Влияние онлайн операций на производительность	9-11
Вставка релейно-контактных цепей около границы в 56 К слов	9-12

Для информации о процессорном времени, занимаемом выполнением специфической инструкцией, смотри главу 22.

Сканирование программы

Так как сканирование программы состоит из сканирования логики и служебных действий, любое событие, которое влияет на время одного из этих сегментов влияет на сканирование программы.

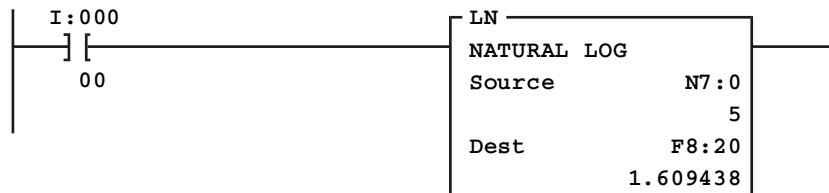
Вы можете проверить время сканирования, используя экран статуса процессора в вашем программном обеспечении.

Если не происходит никаких изменений в статусе входов, и процессор продолжает выполнять те же инструкции релейной логики, программный цикл сканирования постоянен. В реальных системах, тем не менее, программный цикл сканирования колеблется вследствие следующих факторов:

- ложная логика выполняет быстрее чем истинная логика;
- различные инструкции выполняются с различной скоростью;
- различное состояние входов приводит к выполнению различных разделов логики;
- программы прерывания влияют на время сканирования программы;
- редактирование программ в режиме онлайн влияет на служебное время;

Эффекты от ложной и истинной логики на время сканирования логики

В нижележащей цепи изменение состояния от одного сканирования к другому изменит время сканирования приблизительно на 400 мкс.



Если I:000/00	То цепь
On	Истинна, и процессор вычисляет натуральный логарифм. Выполнение инструкции натурального логарифма занимает 409 мкс.
Off	Ложна, и процессор сканирует цепь, но не выполняет ее. Это занимает только 1.4 мкс, чтобы только просканировать цепь.

Другие инструкции могут иметь больший или меньший эффект.

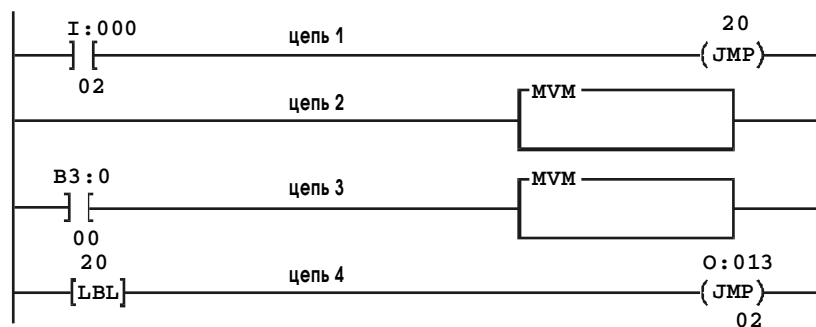
Эффекты от различных инструкций на время сканирования логики

Некоторые инструкции имеют значительно больший эффект на время сканирования логики, чем другие, базирясь на времени, занимаемом выполнением каждой инструкции.

Время сканирования программы также зависит от построения ваших релейно-контактных цепей. Размер цепи и количество ветвей могут привести к существенным колебаниям времени сканирования.

Эффекты от различных состояний входов на время сканирования логики

Вы можете написать вашу логику так, что она выполняет различные цепи за разное время, базирясь на входных условиях. Сумма логики, выполненной при сканировании, приводит к различию во времени сканирования программы. Например, простые различия в выполнении цепей в следующем примере изменяют сканирование программы.



Если I:000/02	Цепи 2 и 3
On	Пропускаются
Off	Выполняются

Если вы используете подпрограммы, время сканирования программы может изменяться на время сканирования целых логических файлов.

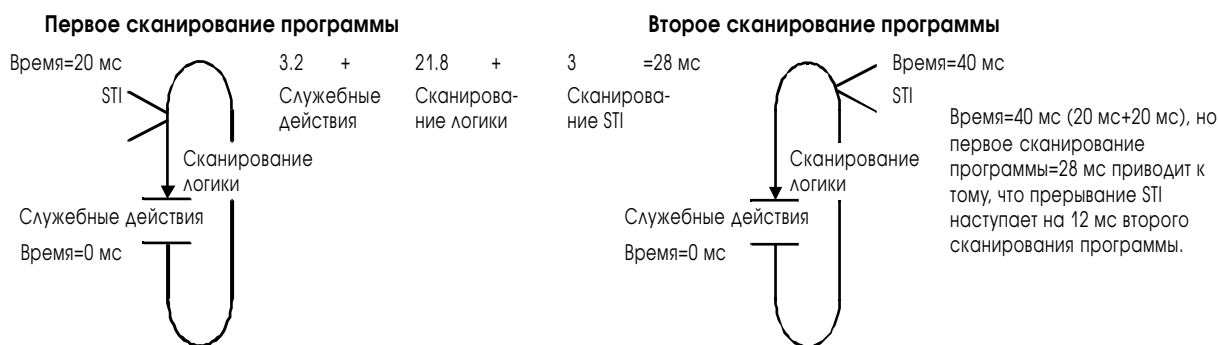
Эффекты от использования прерываний на время сканирования логики

Время сканирования программы также зависит от программ прерывания. Прерывание является специальной ситуацией, которая заставляет выполняться отдельную программу независимо от нормального сканирования программы. Вы определяете специальное событие и тип прерывания, которое должно произойти. Для дальнейшей информации о программах прерывания, смотри главы 18 и 19.

Например, избирательное синхронизированное прерывание (STI) является программным файлом, который вы определяете, для выполнения один раз за каждый период времени. Пример, приведенный ниже, имеет следующие параметры:

- вы конфигурируете STI на выполнение каждые 20 мс;
- программа STI занимает 3 мс на выполнение;
- сканирование логики - 21.8 мс;
- служебные действия занимают 3.2 мс.

Первое сканирование программы в этом примере продолжается в общей сложности 28 мс. Сканирование программы выглядит похожим на следующее:



Поскольку первое сканирование программы занимает 28 мс, STI действительно происходит на 12 мс во втором сканировании программы (28 + 12 = 40, что является временем возникновения второго STI). Этот пример указывает на то, что, если период времени STI другой, чем время сканирования программы, STI происходит в различных местах сканирования программы. Также имейте в виду, что, из-за колебаний во времени сканирования программы, неоднократные STI могут выполняться в течение одного сканирования и не выполняться вообще в течение другого.

Эффекты служебного времени

В процессорах PLC-5, основные служебные действия занимают 3.5 мс. Если выполнение релейно-контактной программы занимает 21.8 мс процессорного времени, общее время сканирования программы составит 25.3 мс. Любое увеличение служебных действий влияет на сканирование вашей программы.

Следующие действия могут увеличить служебное время:

- редактирование в режиме REMOTE RUN;
- размещение блок-трансфер модулей в резидентном шасси процессора;
- использование глобальных файлов флагов состояния.

Редактирование в режиме REMOTE RUN

Время для онлайн редактирования релейно-контактных программ нижеследующее:

Важно: Онлайн редактирования программ также задерживает выполнение ПИ и STI.

Операция редактирования	Тип программы	Время
Принять (ассерт) цепь (после вставки, изменения или удаления редактируемой цепи)	кроме отредактированного файла	0.35 мс на 1000 слов
	без меток	3 мс + 0.35 мс на 1000 слов
	с метками	3.5 мс + 0.35 мс на 1000 слов
Испытать изменения (test edits) программы (влияет на одно сканирование программы)		0.2 мс для изменения состояния изменений от TEST к UNTEST или от UNTEST к TEST
Собрать изменения (assemble edits)	без раздвижки цепей	0.35 мс на 1000 слов
	с раздвижкой цепей, без меток	2.0 мс + 1.5 мс на 1000 слов
	с раздвижкой цепей, с метками	2.0 мс + 1.9 мс на 1000 слов

Размещение блок-трансфер модулей в резидентном шасси процессора

Поскольку резидентные по отношению к процессору рэки не могут быть обновлены до завершения активных блок-трансферов, то размещение блок-трансфер модулей в резидентном шасси процессора может увеличить время служебных действий, в самом неблагоприятном случае, приблизительно на *100 мкс на одно слово данных блок-трансфера*. Имейте в виду, что эта оценка базируется на самом неблагоприятном сценарии. Обычно, этот эффект на служебных действиях, если он и имеется, будет минимальным.

Использование файлов глобальных флагов состояния

Файлы глобальных флагов состояния обновляются в течение служебных действий. Это увеличивает служебное время следующим образом:

- каждый файл глобальных флагов состояния в канале (например, канал 1A или 1B) добавляет 3 мс;
- служебное время не увеличивается более, чем на 6 мс, даже если есть более, чем два файла глобальных флагов состояния.

Совет

Если вам нужно два файла глобальных флагов состояния, разделите их между двумя каналами.

Расчет производительности

Производительность – это время, требующееся на то, чтобы был активизирован выход после того, как связанный с ним вход был активизирован. Вам нужно рассмотреть следующие компоненты при оценке производительности:

- задержка входного и выходного модулей;
- пересылка через монтажную плату Вх/Вых;
- время сканирования удаленных Вх/Вых;
- время процессора.

Для того, чтобы вычислить производительность, используйте следующее уравнение:

$$\begin{array}{cccccccc} \text{Задержка} & & \text{Пересылка по} & & \text{Наихудшее время} & & \text{Наихудшее} & & \text{Наихудшее время} & & \text{Пересылка по} & & \text{Задержка} \\ \text{входного} & + & \text{монтажной} & + & \text{сканирования} & + & \text{время} & + & \text{сканирования} & + & \text{монтажной} & + & \text{выходного} \\ \text{модуля} & & \text{плате Вх/Вых} & & \text{удаленных Вх/Вых} & & \text{процессора} & & \text{удаленных Вх/Вых} & & \text{плате Вх/Вых} & & \text{модуля} \end{array}$$

Задержка входных и выходных модулей

Все входные и выходные модули имеют «время задержки» – это время, требующееся выходному модулю, чтобы передать информацию с монтажной платы Вх/Вых на внешние устройства (и в обратном направлении для входного модуля).

В зависимости от типа модулей, которые вы используете, это время задержки изменяется; но, время должно быть принято во внимание при расчете системной производительности. Выберите модули, которые выполняют нужную вам функцию, с самым низким возможным временем задержки.

Пересылка через монтажную плату Вх/Вых

Время передачи через монтажную плату Вх/Вых – это время, требующееся модулю адаптера 1771-ASB, чтобы обменяться данными с модулями Вх/Вых в шасси, обычно это 1-2 мс для полного рэка Вх/Вых.

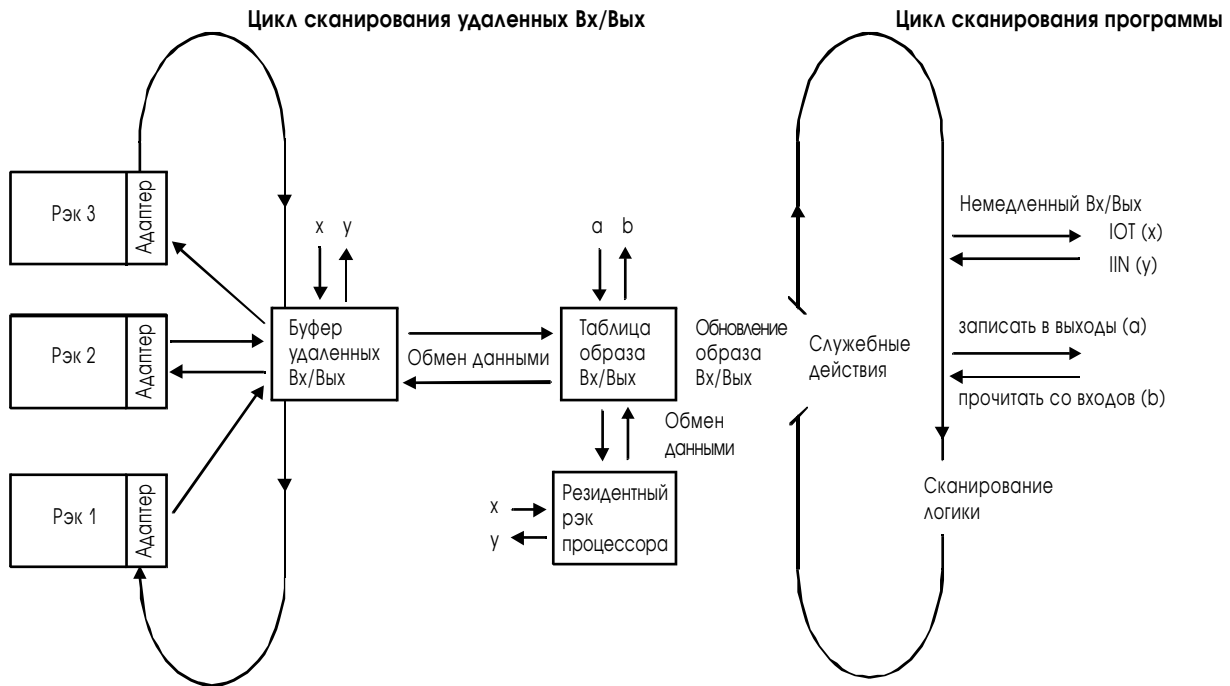
Это время довольно незначительное по сравнению с общей системной производительностью, но может быть оптимизировано в ситуациях, когда в шасси существуют пустые слоты или модули, которые используют только мощность монтажной платы. Например, если последние четыре слота шасси содержат модуль 1785-КА и блок питания (с двумя пустыми слотами), 1771-ASB может быть сконфигурирован на игнорирование последних четырех слотов.



Для дальнейшей информации о выборе конфигурации модулей адаптера, обратитесь к Руководству пользователя модуля удаленного адаптера 1771, публикация 1771-6.5.83.

Время сканирования удаленных Вх/Вых

Время сканирования удаленных Вх/Вых – время, требующееся сканеру, чтобы произвести обмен с каждым устройством в системе с удаленными Вх/Вых.



Эти три показателя влияют на время сканирования удаленных Вх/Вых:

- скорость связи;
- количество рэков;
- блок-трансферы.

Скорость связи

Скорость связи определяется временем, требуемом сканеру, чтобы обменяться данными с каждым из элементов в своем списке сканирования. Таблица 9А указывает время, требующееся, чтобы обменяться данными с устройством, при различной скорости связи.

Таблица 9А
Времена обмена при различной скорости связи

Скорость связи (кбит/с)	Время (мс)
57.6	10
115.2	7
230.4	3

Учтите, что это время для полных рэков. Меньшие по заполнению рэки уменьшат это время.

Если в списке сканирования четыре полных рэка, то сканирование Вх/Вых для этого канала при 57.6 кбит/с – $4 \times 10 = 40$ мс. Если вы измените скорость связи на 230.4 кбит/с, то время сканирования Вх/Вых уменьшится до $4 \times 3 = 12$ мс.

Количество рэков

Вы определяете общее время сканирования удаленных Вх/Вых в системе с удаленными Вх/Вых по этой формуле:

общее время сканирования удаленных Вх/Вых = количество рэков X время для рэка в списке сканирования (смотри таблицу 9А на странице 9–6)

Например, если один канал имеет в два раза больше рэков, чем другой, то время сканирования для первого канала – в два раза больше.

Для того, чтобы оптимизировать это время сканирования, разделите ваши рэки Вх/Вых между несколькими каналами. Установите ваши наиболее критичные по времени Вх/Вых в одном канале, и некритичные Вх/Вых в другом канале. Поскольку все каналы Вх/Вых независимы, длительное сканирование удаленных Вх/Вых в одном канале не повлияет на сканирование удаленных Вх/Вых в другом канале.

Блок-трансферы

Блок-трансфер представляет собой прерывание нормального порядка сканирования удаленных Вх/Вых, чтобы передать блок данных в специфический модуль Вх/Вых. Большую часть времени, которое процессор тратит в выполнение блок-трансфера, расходуется для процедуры квитирования, которое происходит между процессором и блок-трансфер модулем. Это квитирование встроено в пересылку дискретных Вх/Вых и не оказывает влияния на сканирование удаленных Вх/Вых. Время сканирования удаленных Вх/Вых изменяется, когда происходит фактическая пересылка данных.

Время, на которое блок-трансфер прерывает сканирование удаленных Вх/Вых, зависит от количества передаваемых слов, скорости связи и сопутствующих потерь.

Используйте эту формулу и нижеуказанную таблицу, чтобы вычислить время выполнения блок-трансфера:

время выполнения блок-трансфера = (количество пересылаемых слов X мс/слово, основанное на скорости связи) + потери при данной скорости связи

Скорость связи (кбит/с)	мс/слово	Потери (мс)
57.6	0.28	3
115.2	0.14	2.5
230.4	0.07	2

Например, если скорость связи – 115.2 кбит/с, и вы хотите выполнить блок-трансфер 10 слов, прерывание сканирования удаленных Вх/Вых составит:

$$(10 \times 0.14) + 2.5 = 1.4 + 2.5 = 3.9 \text{ мс}$$

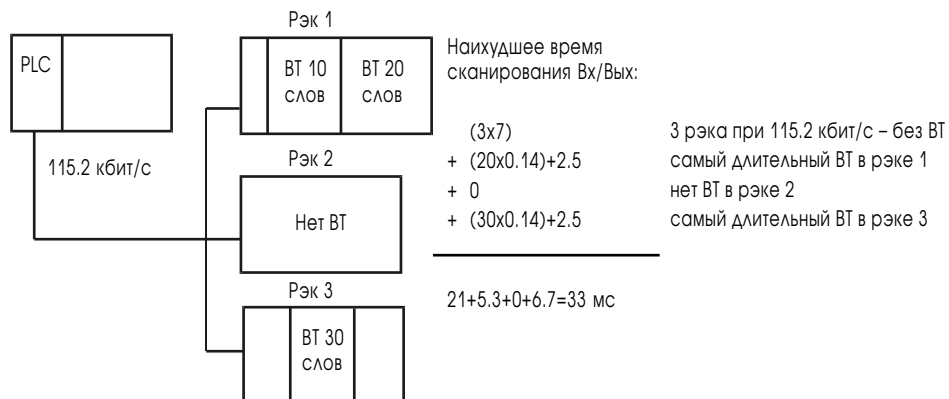
Для конкретного удаленного Вх/Вых, в котором происходит блок-трансфер, будет добавлено 3.9 мс ко времени сканирования удаленных Вх/Вых.

Вычисление наихудшего времени сканирования удаленных Вх/Вых

Поскольку невозможно предсказать, в пределах какого сканирования удаленных Вх/Вых произойдет блок-трансфер, вы можете вычислить только наихудшее время сканирования удаленных Вх/Вых. Чтобы вычислить наихудшее время:

1. Определите нормальное время сканирования Вх/Вых (без блок-трансферов)
2. Добавьте время самого длительного блок-трансфера для каждого элемента в списке сканирования. (Процессор может выполнить только один блок-трансфер для каждого элемента в списке сканирования за одно сканирование Вх/Вых.)

Например, если ваша система такова:



Оптимизация времени сканирования удаленных Вх/Вых

Наилучший путь, чтобы оптимизировать время сканирования – это установить наиболее критичные по времени Вх/Вых в канал, отдельный от некритичных Вх/Вых. Если вы имеете только один доступный канал для Вх/Вых, тем не менее, вы все-таки можете оптимизировать сканирование, используя конфигурируемый список сканирования процессора.

В нормальной системе с 4 рэками, список сканирования должен быть таким:

- рэк 1
- рэк 2
- рэк 3
- рэк 4

Если вы используете 57.6 кбит/с, нормальное сканирование Вх/Вых составит 4 рэка x 10 мс = 40 мс. Каждый элемент имеет равный приоритет, так что каждый рэк сканируется каждые 40 мс.

Однако, если рэк 2 имеет наиболее критичные по времени Вх/Вых, используйте конфигурируемый список сканирования, чтобы определить следующий порядок сканирования:

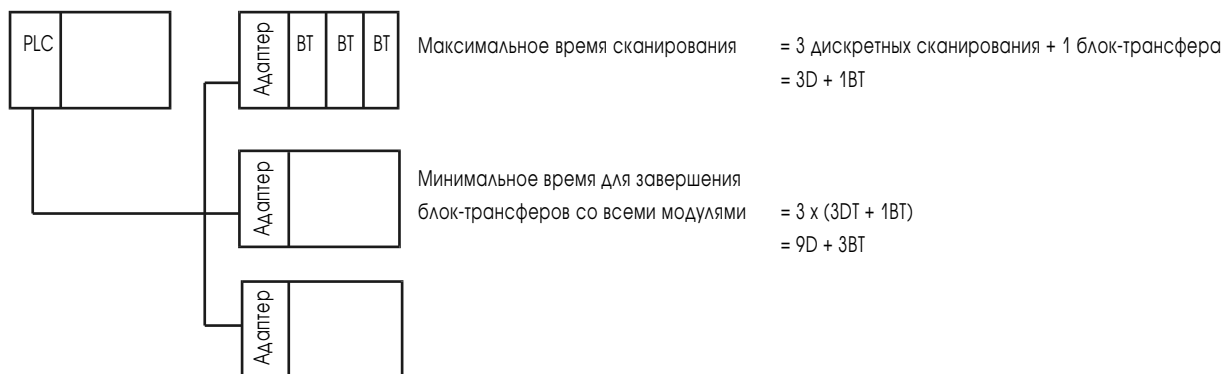
- рэк 1
- рэк 2
- рэк 3
- рэк 2
- рэк 4
- рэк 2

При использовании этого списка сканирования, рэк 2 сканируется после каждого другого рэка. Список имеет 6 элементов, таким образом нормальное время сканирования Вх/Вых – 6 x 10 мс = 60 мс. Так как рэк 2 сканируется после каждого другого рэка, тем не менее, **эффективное** время сканирования рэка 2 составит 2 x 10 мс = 20 мс. Остальные рэки сканируются каждые 60 мс. Поэтому, ценой за более быстрое сканирование рэка 2 (каждые 20 мс) будет то, что другие рэки сканируются только каждые 60 мс.

Вы можете также оптимизировать блок-трансферы в пределах канала. Вы осуществляете блок-трансфер только с одним блок-трансфер модулем из элементов в списке сканирования за одно сканирование Вх/Вых. Если у вас есть три блок-трансфер модуля в одном рэке Вх/Вых, то завершение блок-трансферов со всеми модулями займет минимум три сканирования Вх/Вых:

Система, оптимизированная для пересылки дискретных данных

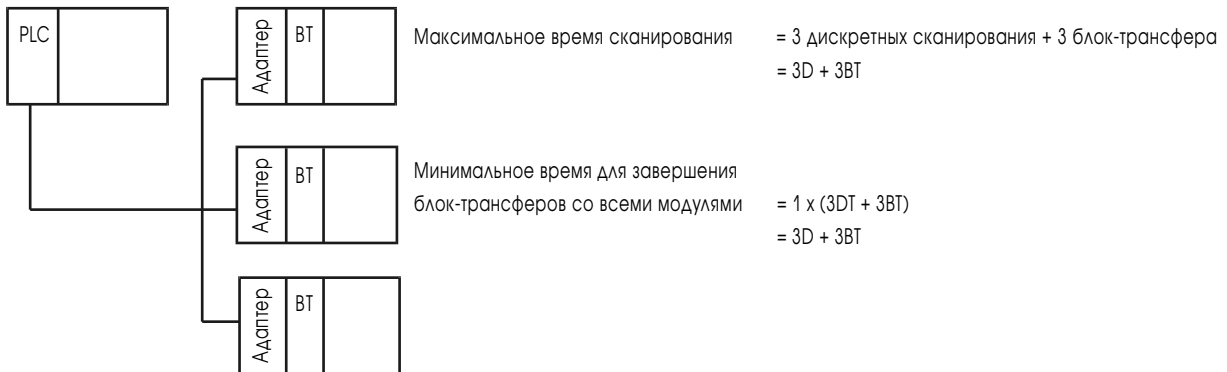
С таким расположением за каждые 3 сканирования дискретных Вх/Вых может наступить только один блок -трансфер к каждому из ВТ модулей.



Однако, если вы разместите три блок-трансфер модуля в разных стойках, вы сможете осуществить блок-трансферы со всеми тремя модулями за одно сканирование Вх/Вых. Для того, чтобы оптимизировать вашу систему для передачи данных блок-трансферов, используйте размещение подобное следующему:

Система, оптимизированная для пересылки данных блок-трансферов

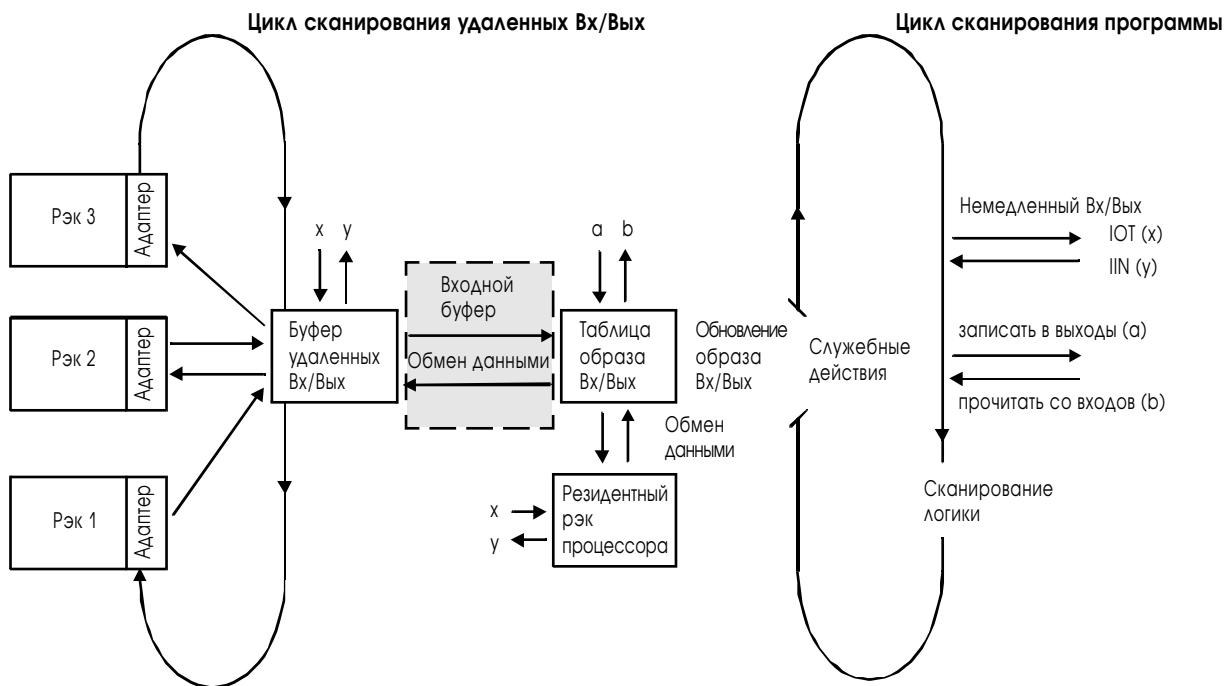
С таким расположением, блок -трансфер к каждому из ВТ модулей может наступить при каждом сканировании.



Время процессора

Время процессора – это время необходимое для того, чтобы обработать входы и установить соответствующие выходы. Время процессора различно для разных процессоров и базируется на буферизации входов, сканировании программы, и т.п.

В системе PLC-5, входы буферизуются между таблицей образа Вх/Вых и буфером удаленного Вх/Вых. Перемещение входов из буфера удаленного Вх/Вых во входной буфер асинхронно по отношению к перемещению данных из входного буфера в таблицу образа Вх/Вых.



Наихудшее время процессора:

Периодическое обновление входного буфера из буфера удаленных Вх/Вых	= 10 мс
Одно сканирование программы для гарантии чтения входов	= xx мс
Одно сканирование программы для гарантии записи выходов	= xx мс
0.18 мс умноженные на число рэков	= xx мс

Для системы с тремя рэками и временем сканирования программы 20 мс
наихудшее время процессора: $10 + 20 + 20 + (0.18 \times 3) = 50.54$ мс.

Пример расчета

Ниже приведен пример вычисления наихудшего времени обновления, базирующийся на результатах вычисления каждого из компонентов производительности, представленных в данной главе:

Задержка входного модуля	= 10 мс (типичная)
Монтажная плата Вх/Вых	= 1 мс
Наихудшее время сканирования удаленных Вх/Вых	= 33 мс
Наихудшее время процессора	= 50.54 мс
Наихудшее время сканирования удаленных Вх/Вых	= 33 мс
Монтажная плата Вх/Вых	= 1 мс
Задержка выходного модуля	= 1 мс (типичная)
<hr/>	
Итого:	129.54

Влияние онлайн операций на производительность

Когда вы выполняете онлайн операции через связь ДН+ с вашими программными файлами в режиме RUN, это оказывает влияние на производительность процессора PLC-5. Влияющие воздействия:

- сообщения ДН+;
- сообщения по последовательному порту;
- сообщения по каналу ЗА;
- удаленные блок-трансферы.

Время, на которое обмен сообщениями и блок-трансферы могут быть задержаны, **пропорционально размеру релейно-контактного файла**. Таблица 9Б перечисляет эффекты, влияющие на производительность (при использовании любого из пакетов программирования PLC-5 серии 6200, который поддерживает используемый вами процессор).

Таблица 9Б
Наихудшее влияние онлайн операций на производительность в режиме RUN

Пересылка данных, на которую оказывается влияние:	Онлайн операции через какой-либо канал DH+:	
	Выполнение Page Up/Page Down в конце программного файла	Вставка/удаление цепей релейной логики
удаленные блок-трансферы	20 мс/кслово	50 мс/кслово
сообщения по DH+	20 мс/кслово	50 мс/кслово
сообщения по последовательному порту	200 мс/кслово	50 мс/кслово
сообщения по каналу 3А	не влияет	50 мс/кслово

Вы должны перепроектировать ваши программы, чтобы избежать возможных пауз связи, если вы к настоящему времени:

- используете большие релейно-контактные программы;
- имеете критичные по времени удаленные блок-трансферы и/или сообщения по последовательному порту, порту DH+ и каналу 3А;
- должны редактировать программу в режиме RUN онлайн.

Для наилучшей производительности процессора, сегментируйте ваши программные файлы, используя модульные методы программирования, как, например, главные управляющие программы (MCP), последовательные функциональные диаграммы (SFC) и инструкции перехода на подпрограммы (JSR).

Эффект вставки релейно-контактных цепей около предела в 56 кслов

Выполнение редактирования в режиме REMOTE RUN или PROGRAM файлов релейно-контактной логики, которые достигают максимального размера программного файла в 57,344 слов, может:

- привести к невозможности вставки цепи;
- привести к зависанию пакета программирования PLC-5 серии 6200 (релиз 4.3 и более поздний).

Для того, чтобы избежать или скорректировать эту проблему, сегментируйте ваш программный файл, используя модульные методы программирования, как, например, главные управляющие программы (MCP), последовательные функциональные диаграммы (SFC) и инструкции перехода на подпрограммы (JSR).

Если вы не можете сегментировать ваш программный файл, чаще сохраняйте файл во время редактирования.

Если вы встретите ошибку `Memory Unavailable for Attempted Operation`, то очистите память процессора.

Это соображение относится к процессорам PLC-5/60, -5/60L, -5/80 и -5/80E, когда вы редактируете программный файл, который достигает максимального файлового предела в 57,344 слов.

Связь с оборудованием по сети DH+

Назначение этой главы

Сведения	Страница
Выбор устройств, которые вы хотите подключить	10-1
Проектирование сети	10-2
Конфигурация канала для связи по DH+	10-2
Использования файла основных флагов состояния	10-4
Просмотр каналов связи DH+	10-5
Оценка работы сети DH+	10-9
Требования к приложениям	10-14

Выбор устройств, которые вы хотите подключить

Вы можете использовать связь по DH + для передачи данных другим процессорам PLC-5 или компьютерам верхнего уровня, а так же как связь для программирования нескольких процессоров PLC-5. Процессор PLC-5 может связываться по сети DH + как с другими процессорами, так и с терминалом программирования. Вы можете присоединять к одной сети DH + максимум 64 станции.

Таблица 10.A
Оборудование, которое вы можете подключать

Наименование изделия:	Номер по каталогу:	Наименование изделия:	Требуемый кабель:
Интерфейсный модуль Data HighWay или Data HighWay Plus (RS-232C или RS-422-A)	1770-KF2	Подключение асинхронного (RS-232C) к сети Data HighWay или DH+	1770-CD
Data HighWay/ Data HighWay Plus на broadband	1771-KRF	Сетевой мост, соединяющий до 18 сетей Data HighWay для связи по кабельной системе facility-wide broadband	
Интерфейсная карта связи	1784-KL	Подключение программатора T47 к сети DH+	1784-CP
Интерфейсный модуль XT/AT Data HighWay Plus	1784-KT	Подключение IBM XT или AT совместимых компьютеров к сети DH+	1784-CP2 1784-CP3
Интерфейсный модуль PS/2 Data HighWay Plus	1784-KT2	Подключение IBM PS/2 совместимых компьютеров к сети DH+	1784-CP5 1784-CP6
Интерфейсный модуль связи Data HighWay с Data HighWay Plus	1785-KA	Соединение сети Data HighWay и сети DH+	1770-CD
Интерфейсный модуль связи DH+ с DH-485	1785-KA5	Соединение сети DH-485 и сети DH+	
Интерфейсный модуль связи Data HighWay Plus с RS-232C	1785-KE	Соединение асинхронных (RS-232C) устройств к DH+	
Карта PCMCIA	1784-PCMK	Соединение шины PCMCIA ноутбука с DH+	1784-PCM5

Проектирование сети

При подключении сети по методу "ответвлений", используйте разъем 1770-SC, соблюдая следующие требования к длине кабеля:

- длина кабеля между разъемами зависит от скорости передачи данных в сети;
- длина "ответвления" - 30.4 м. (100 футов)

Для получения большей информации о подключении сети по методу «ответвлений», см. **Data Highway/Data Highway Plus/Data Highway II/Data Highway—485 Cable Installation Manual**, публикация 1770—6.2.2.

Выберите кабель 1770-CD (Belden 9463). Подключитесь к сети DH+, используя последовательное подключение или метод «ответвлений».

Удостоверьтесь, что запроецированная вами длина кабеля соответствует допустимым размерам.

Важно: Максимальная длина кабеля для сети DH+ зависит от скорости передачи данных. Сконфигурируйте все оборудование на сети DH+ для связи на одной и той же скорости.

Для конфигурации последовательных подключений используйте таблицу, определяющую общую длину используемого кабеля.

Таблица 10.В
Выбор правильной длины кабеля

Скорость передачи данных в сети DH+	Длина кабеля не должна превышать:
57.6 кБод	3 048 м (10 000 футов)
115.2 кБод	1 524 м (5 000 футов)
230.4 кБод	762 м (2 500 футов)

Для нормальной работы установите на **оба** конца сети DH+ внешние резисторы, прилагаемые к программируемым контроллерам. Выберите терминатор на 150 Ом или на 82 Ом, в зависимости от того, какое количество оборудования подключается к сети DH+.

Если сеть DH+ работает на скорости: Используйте терминатор:

230.4 кБод	82 Ом
57.6 кБод	150 Ом

Конфигурация каналов для связи по DH+

В зависимости от того, какой процессор используется, можно конфигурировать следующие каналы:

Процессор:	Каналы, поддерживающие DH+:
PLC-5/11	1A
PLC-5/20	1A (фиксированный DH+), 1B
PLC-5/30	1A, 1B
PLC-5/40L	
PLC-5/60L	
PLC-5/40	1A, 2A, 1B, 2B
PLC-5/60	
PLC-5/80	

Важно: Для определения адреса в сети DH+ и скорости передачи канала 1A вы должны соответствующим образом установить переключатель SW1 на процессоре; адрес узла не может быть установлен через пакет программирования

Для конфигурации каналов, поддерживающих сеть DN+, используйте экран конфигурации сети DN+ пакета программирования.

```

Data Highway Plus
Channel 1A Configuration

Diagnostics file:          N14
Baud rate:                57.6KB
Global Status flag file:  N12

Node address:             36
Link ID:                  0
  
```

Поле экрана:	Предназначено для:	Сконфигурируйте, следующим образом:
Diagnostics file	Файл, содержащий информацию о состоянии канала	<p>Введите номер целочисленного файла (10-999). Система создаст целочисленный файл длиной 40 слов.</p> <p>ВНИМАНИЕ: Используйте отдельный файл диагностики для каждого канала. Не используйте в качестве файла диагностики файл, назначенный для контроля состояния Вх/Вых или другой используемый файл. Это может привести к непредсказуемым повреждениям машин и механизмов.</p> <p>Важно: Вы должны определить диагностический файл для любого сконфигурированного канала, даже и не используемого (т.е. если вы не используете канал), если вы хотите иметь информацию о статусе этого канала.</p>
Baud rate	Скорость передачи данных канала	<p>Если каналом DN+ является:</p> <ul style="list-style-type: none"> • канал 1A, определите скорость передачи установкой переключателя SW1 (смотри главу 23); • любой другой канал, выберите 57,6 кБод, 115,2 кБод или 230,4 кБод с помощью пакета программирования.
Global Status flag file	Это файл, где вы хотите хранить передаваемые данные	<p>В этой области введите номер целочисленного файла (10-999) и нажмите (Enter)</p> <p>Будет создан целочисленный файл длиной 64 слова.</p> <p>ВНИМАНИЕ: Когда вы меняете режим процессора из Run или Test в Program, процессор пишет нули во флаги основного файла состояния. Любая предыдущая информация в этом файле теряется. Для более подробной информации о флагов основного файла состояния смотрите ниже.</p>
Node address	Адрес узла сети для вашего процессора	<p>Если каналом DN+ является:</p> <ul style="list-style-type: none"> • канал 1A - выберите номер узла сети DN+ установкой переключателя SW1 на процессоре (смотри главу 23); • любой другой канал - в поле Node address введите величину 0-77 (восьмеричную) и нажмите (Enter). <p>Каждая станция в сети DN+ должна иметь свой, отличный от прочих, адрес.</p>
Link ID	Локальная сеть, к которой подключен канал	<p>Если ваша сеть DN+ подключена через мост (bridge) к другой сети Data Highway, в этом поле введите десятичную цифру, определяющую протокол сети, к которой подключен канал, и нажмите (Enter).</p>

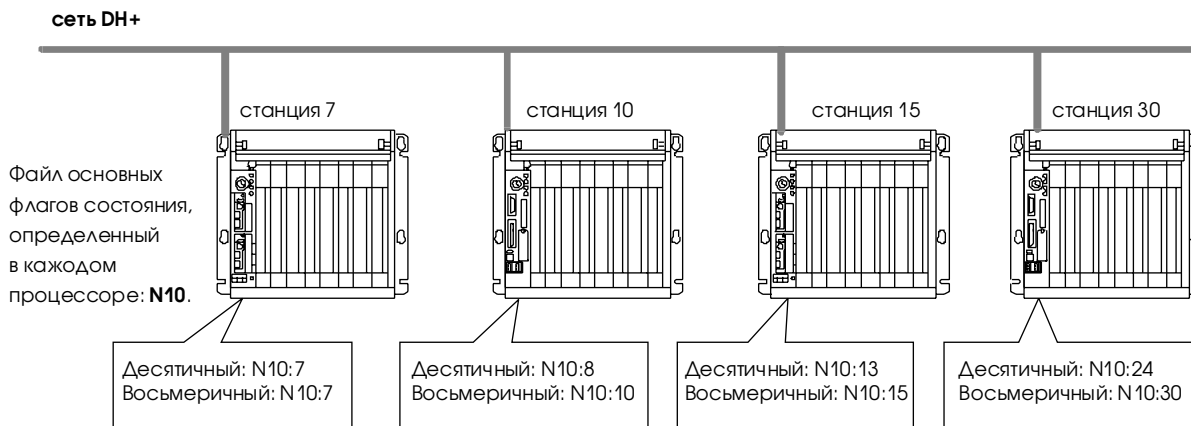
Использования файла основных флагов состояния

Используйте файл основных флагов состояния для хранения передаваемых данных. Этот файл хранит 16-битное слово данных для каждой станции сети DH+. Станции используют этот файл для автоматического предоставления данных другим станциям, не требуя специального программирования для пользователя.

Когда станция посылает сообщение на другую станцию, она, на самом деле, выдает сообщение, содержащее 1 слово информации о своем адресе в собственном файле основных флагов состояния. Пересылаемые данные берутся из слова в файле основных флагов состояния, соответствующего собственному адресу станции. Сообщения "видят" все станции. Каждая станция в сети проверяет сообщение и помещает слово из файла основных флагов состояния, от станции-источника в слово, соответствующее адресу станции-источника.

Этот процесс позволяет каждой станции автоматически видеть самое последнее обновление данных. Вы можете создать цикловую логику для наблюдения и объяснения этих данных в своем приложении.

Данные в файле основных флагов состояния для каждого адреса узла сети DH+ помещаются в слово, адрес которого соответствует восьмеричному адресу узла. Например, если в сети DH+ имеются процессоры с адресами узлов 7, 10, 15 и 30 и файл основных флагов состояния для каждого процессора N10, данные в файле основных флагов состояния будут помещены следующим образом:



Вы можете назначить в качестве файла основных флагов состояния любой целочисленный файл вашего процессоре; однако для идентичности выберите один и тот же номер файла для всех процессоров PLC-5 в вашей сети DH+.

Удостоверьтесь, что размер файла основных флагов состояния во всех процессорах вашей сети DN+ превышает наибольший адрес узла; это нужно для того, чтобы все процессоры могли связываться между собой. Например, если станция с номером 30 является наибольшим адресом узла, файл основных флагов состояния в каждом процессоре должен быть длиной 24 слова (восьмеричное 30 = десятичному 24). В конфигурации по умолчанию размер файла основных флагов состояния автоматически устанавливается в 64 слова.

Важно: Не разрешайте запись внутренних или внешних сообщений в файл основных флагов состояния. Запись в файл основных флагов состояния вызывает ошибку процессора.

Вы можете изменить основание системы счисления при просмотре данных на восьмеричную таким образом, что бы вы могли видеть восьмеричный адрес, соответствующий номеру узла.

Просмотр каналов связи DN+

Используйте экран состояния (**status screen**) DN+ вашего пакета программирования для просмотра состояния каналов, сконфигурированных для сети DN+. Показываемые данные хранятся в диагностическом файле, определенном в экране конфигурации канала (**configuration screen**) DN+ вашего пакета программирования. Помните, что этот экран не показывает таблицу активных узлов, хранящуюся в файле диагностики.

```

Data Highway Plus
Channel 1A Status

Messages
COUNTERS LOCKED
sent: 0 sent with error: 0
received: 0 received with error: 0 unable to receive: 0
Send data acknowledged
received: 0 received but full: 0 received retrans.: 0
received SAP off: 0 received with error: 0 transmit NAK misc.: 0
transmit failed: 0 transmit confirm: 0 transmit not ACKed: 0
transmit timeout: 0 transmit NAK full: 0 transmit NAKed SAP: 0
Send data no acknowledged
received: 0 transmit failed: 0 transmit confirm: 0
SDA/SDN retrans.: 0
Duplicate node: 0 Claims lost: 0 Network dead: 0
Claims won: 0 Dropped token: 0 Linear scan failed: 0
Token retry: 0 Solicit rotations: 0 Started linear scan: 0
New successor: 0 Token failed: 0

```

Поле экрана:	Слово (а):	Назначение:															
Counters Locked	-	Эта область показывается в режиме <i>reverse video</i> , если счетчики заблокированы. <ul style="list-style-type: none"> счетчики заблокированы: на экране изменений не происходит, но счетчики продолжают работать; счетчики разблокированы: система показывает текущую величину. 															
Active Node Table (не показывается)	0-3	Хранение таблицы активных узлов для станций, сконфигурированных на этот канал DN+. Эта информация не отображается на экране. Таблица активных узлов сети (Active Node Table) <table border="1"> <thead> <tr> <th>Слово</th> <th>Биты</th> <th>Номер станции сети DN+</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0-15</td> <td>00-17</td> </tr> <tr> <td>1</td> <td>0-15</td> <td>20-37</td> </tr> <tr> <td>2</td> <td>0-15</td> <td>40-57</td> </tr> <tr> <td>3</td> <td>0-15</td> <td>60-77</td> </tr> </tbody> </table> Таблица активных узлов канала 1A так же размещается в словах S:3 - S:6 файла состояния.	Слово	Биты	Номер станции сети DN+	0	0-15	00-17	1	0-15	20-37	2	0-15	40-57	3	0-15	60-77
Слово	Биты	Номер станции сети DN+															
0	0-15	00-17															
1	0-15	20-37															
2	0-15	40-57															
3	0-15	60-77															

Поле экрана:	Слово: Бит:	Назначение:
Messages		Охватывает все данные посылаемые и принимаемые портом
sent	5	Общее число сообщений, посланных станцией. Эта цифра является суммой счетчиков данных посланных с подтверждением (SDA) и счетчиков данных посланных без подтверждения (SDN).
Sent with error	7	Количество посланных сообщений, которые не были подтверждены. Эта цифра является суммой: <ul style="list-style-type: none"> • SDA transmit NAK misc • SDA/SDN retrans • Transmit NAK full • dropped token • SDA transmit NAKed
received	4	Количество сообщений, принятых станцией без ошибок. Эта цифра является суммой счетчиков полученных SDA и SDN.
Received with error	6	Число сообщений с ошибками, принятых станцией. Эта цифра является суммой счетчиков SDA, полученных с ошибками, и SDA, полученных SAP off.
Unable to receive	8	Общее время, пока станция посылала NAK (<i>no acknowledge</i>) на приходящие сообщения из-за переполнения буфера. Эта цифра будет соответствовать посланным SDA, при заполненном счетчике.
Send Data Acknowledge Counters		Эти счетчики предназначены для сообщений с SDA. Отправитель, посылающий SDA сообщения требует, чтобы приемник подтвердил получение сообщения немедленно после получения. Приемник подтверждает сообщения, посылая: <ul style="list-style-type: none"> • ACK (<i>acknowledge</i>), который сообщает отправителю, что сообщение принято и будет обработано; • NAK (<i>no acknowledge</i>), который сообщает отправителю, что сообщение принято, но не может быть обработано. <p>Оборудование, отправившее сообщение, автоматически отправляет сообщение повторно.</p>
received	19	Количество сообщений SDA, принятых станцией без ошибок.
Received but full	22	Количество сообщений SDA, принятых станцией, но ответа (<i>acknowledge</i>) не было из-за отсутствия доступного буфера для обработки сообщений. Это означает, что сообщения приходят на процессор PLC-5 быстрее, чем он может их обработать. Убедитесь, что вы: <ul style="list-style-type: none"> • не выполняете ненужной пересылки информации (например, вы посылаете сообщения постоянно, тогда как вам нужно только изменять их раз в секунду); • Посылаете сообщения только при изменении данных.
Received retransmission	21	Количество времени, затраченного передающей станцией для повторной передачи сообщений SDA, которые были приняты с ACK или NAK. Если узел посылает сообщение, но не получает ACK или NAK ответ, узел повторно передаст сообщение. Если узел повторно передал сообщение, потому что ответ подтверждения первого сообщения был потерян, узел, получающий сообщение обнаруживает повторную передачу и посылает подтверждающий ответ. Но узел получения не принимает двойного сообщения. Большая цифра указывает на проблемы с кабелем или помехи; проверьте целостность кабеля и его защиту от помех.

Поле экрана:	Слово (а):	Назначение:
received SAP off	23	Количество сообщений SDA, принятых станцией, но которые не смог обработать процессор из-за отключенного SAP (<i>service access point</i>). Этот счетчик может всегда быть 0.
received with error	20	Количество ошибочных сообщений SDA, принятых станцией. Это может быть вызвано: <ul style="list-style-type: none"> плохой CRC; в сообщении указан неверный адрес источника; сообщение имеет нераспознаваемый управляющий байт; связь прервана. Этот счетчик показывает помехи; улучшите экранирование кабеля от помех.
transmit NAK misc.	25	Количество пришедших сообщений SDA, которые не были подтверждены из-за того, что счетчик NAK переполнен или неактивен (например из-за плохой CRC).
transmit failed	29	Количество сообщений SDA, посланных станцией, в которых определена ошибка. Этот счетчик - сумма переданных SDA, на которых не был получен ACK, и числа превышений времени передачи SDA.
transmit confirm	24	Количество сообщений SDA, успешно посланных и подтвержденных адресуемой станцией.
transmit not ACKed	27	Количество посланных сообщений SDA не подтвержденных адресуемой станцией. Это может произойти из-за <ul style="list-style-type: none"> сообщение могло быть NAK; вернулось неправильное подтверждение (ACK); ответа не было. Счетчик может показывать: <ul style="list-style-type: none"> помехи или проблемы с кабелем; станция-приемник отключена от сети; станция-приемник не осуществляет связь.
transmit timeout	26	Количество посланных сообщений SDA, которые были посланы, но ни ACK, ни NAK не были получены от станции-приемника. Этот счетчик увеличивается, как только сообщение не проходит повторно, и если станция-приемник не может осуществить связь. Этот счетчик показывает помехи или проблемы с кабелем (станция-приемник не "видит" сообщение).
transmit NAK full	30	Время, в течение которого станция принимает NAK на сообщение, так как станция-приемник заполнена. Это означает, что сообщения посылаются на станцию-приемник быстрее, чем процессор PLC-5 может их обработать. Вероятнее всего, на сети DH+ на одну и ту же станцию посылаются сообщения с нескольких станций. Проверьте, что вы: <ul style="list-style-type: none"> не выполняете ненужной пересылки информации (например, вы посылаете сообщения постоянно, тогда как вам нужно только изменять их раз в секунду); посылаете сообщения только при изменении данных.
transmit NAKed SAP	31	Количество успешно посланных сообщений SDA, не подтвержденных станцией-приемником из-за того, что SAP, определенный для этих сообщений был неверен. Этот счетчик может быть равен 0.
Send Data No Acknowledge Counters		Эти счетчики относятся к сообщениям SDN (посылка данных без подтверждения), означающим, что отправитель не требует, чтобы приемник подтвердил получение сообщения немедленно после получения. Сообщения SDN используются только для пересылки.
transmit received	35	Количество успешно полученных сообщений SDN.
transmit failed	33	Количество сообщений SDN, посланных станцией, имеющих ошибку. Эта ошибка может не отображаться.
transmit confirm	32	Количество успешно посланных сообщений SDN.
SDA or SDN transmit retry	28	Общее количество сообщений SDA или SDN, переданных повторно. Некоторыми причинами, почему станция может повторно передавать сообщения, являются: <ul style="list-style-type: none"> в сообщении SDA потерян или разрушен ACK, что указывает на помехи; первое сообщение не было подтверждено.

Поле экрана:	Слово (а):	Назначение:
Network Link Maintenance Counters		Этот счетчик показывает события в сети, как их "видит" станция.
Duplicate node	17	Время, в течение которого станцией в сети была обнаружена другая станция с таким же сетевым адресом. В результате этого станция переходит в оффлайн.
Claims lost	11	Время, в течение которого станция не получала право на передачу. Смотри ниже для более полной информации Claims won.
Network dead	9	Время, в течение которого станция определила отсутствие передачи данных по сети. Это обычно вызвано тем, что с передающей станции снято питание или она отключается от сети. Прочие станции ожидают, когда право на передачу придет к ним. Обычно в такой ситуации объявляется об отключении сети и выполняется последовательный переход передачи данных к другой станции. (Смотри ниже для более полной информации Claims won)
Claims won	10	Время, в течение которого станция пробивала (won) требования о переключении передачи. Все станции начинают выполнение требований о переключении передачи, при снятии питания на оборудование сети, при первом включении, если станции на сети обнаруживают, что никто не ведет передачу, или когда передающая станция отключается или удалена из сети. Требования о переключении передачи - это когда все станции на сети пытаются взять передачу на себя. Когда несколько станций пытаются взять передачу на себя, приоритет отдается станции с наименьшим номером.
Dropped token	18	Время, в течение которого станция обнаруживает, что на сети находится дублирующий узел и отключает себя от сети. Станция обнаруживает дублирующий узел, когда она определяет, что ответ на сообщение или приемник является некорректным. Например, если ответ приходит от станции, до этого не связывавшейся с ней, передающая станция решает, что ответ послан на другую станцию с таким же номером узла. Станция отключается от сети и ожидает некоторое время для повторного входа в сеть. Она только тогда подключится обратно к сети, когда станция с таким же номером узла будет отключена от сети. , так как станции с идентичными номерами в сети не могут совместно работать.
Linear scan failed	16	Время, в течение которого станция запрашивает каждый номер станций без получения ответа. Для более полной информации смотри ниже Started linear scan.
Token retry	13	Время, в течение которого станция повторно передает запрос на передачу (token pass). Станция повторно передает запрос на передачу (token pass), если обнаруживает, что станция приемник не отвечает на передачу. Это может быть вызвано помехами.
Solicit rotations	34	Время завершения опроса следующей передающей станции не на сети. Запрос на следующую передающую станцию происходит во время прохождения передачи по сети. Станция, которая не была в сети, запрашивается для того чтобы всем станциям было "видно", что она добавлена в сеть. Во время прохождения каждой передачи запрашиваются различные номера станций; запрос происходит последовательно. Станция может войти в сеть только если на нее пришел запрос из сети.
Started linear scan	15	Время, в течение которого станция пыталась осуществить передачу к каждому активному узлу, определенному в ее таблице, и не получила ответ В этом случае станция начнет последовательное сканирование всех номеров станций до их ответа.
New successor	12	Время, в течение которого станция обнаружила новую станцию, готовую к передаче. Это происходит, когда станция обнаруживает, что к сети подключена новая станция с номером узла большим чем у нее самой, но меньшим, чем у той, к которой шла передача..В этом случае станция должна осуществлять передачу к вновь добавленной станции.
Token failed	14	Время, в течение которого станция не могла передать очередь на передачу данных следующей по порядку станции, адрес которой у нее имелся. Это может быть вызвано тем, что: <ul style="list-style-type: none"> • станция отключена от сети; • помехи или проблемы с кабелем.

Оценка работы сети DH+

На работу сети DH+ влияют многие факторы, включая:

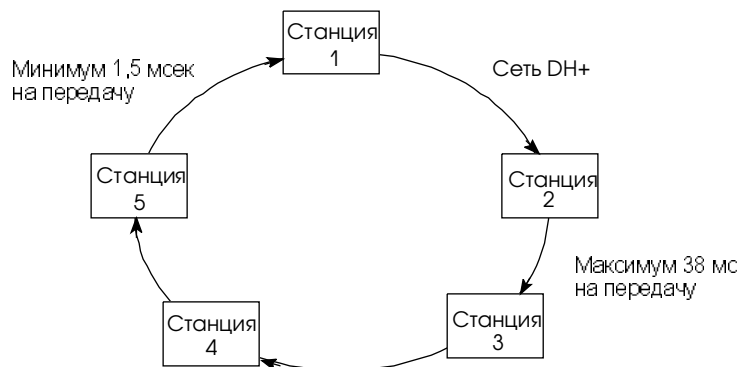
- узлы сети;
- размер и число сообщений;
- кому предназначено сообщение;
- внутреннее время обработки.

Узлы

Узлы влияют на время передачи следующим образом:

- в течение одного цикла передачи каждый узел в сети DH+ последовательно получает передаваемую информацию, вне зависимости от того, имеется у него или нет информация для передачи;
- каждый узел тратит от 1,5 мс (если он не имеет сообщений для передачи) до 38 мс (максимальное затрачиваемое время) на передачу, если нет повторных попыток передачи информации (смотри Рис. 10.1).

Рисунок 10.1
Прохождение передачи



Размер и число сообщений

Процессоры PLC-5 кодируют сообщения в пакеты для передачи по сети DH+. Максимальное число слов данных в пакете зависит от передающей станции и типа команд как показано в таблице, приведенной ниже.

Передающая станция	Тип команды	Максимальный размер пакета (слов данных)
PLC-5	Типа ЧТЕНИЕ/ЗАПИСЬ (READ/WRITE)	114
PLC-5	Диапазон слов ЧТЕНИЕ/ЗАПИСЬ	117
PLC-2	Незащищенная ЧТЕНИЕ/ЗАПИСЬ	121

Эти ограничения вызваны протоколом сети, который ограничивает передачу станцией максимум 271 байт за один цикл прохождения передачи. Станция может посылать более одного сообщения за весь цикл прохождения передачи, с учетом того, что общее количество команд и байтов данных не будет превышать 271.

Если размер сообщения будет превышать максимальный разрешенный размер пакета данных, передающей станции потребуется для завершения сообщения более одного цикла прохождения передачи. Например, если процессору PLC-5 необходимо послать сообщение размером 150 слов, ему нужно будет организовать два сообщения, потребоующих нескольких циклов прохождения передачи.

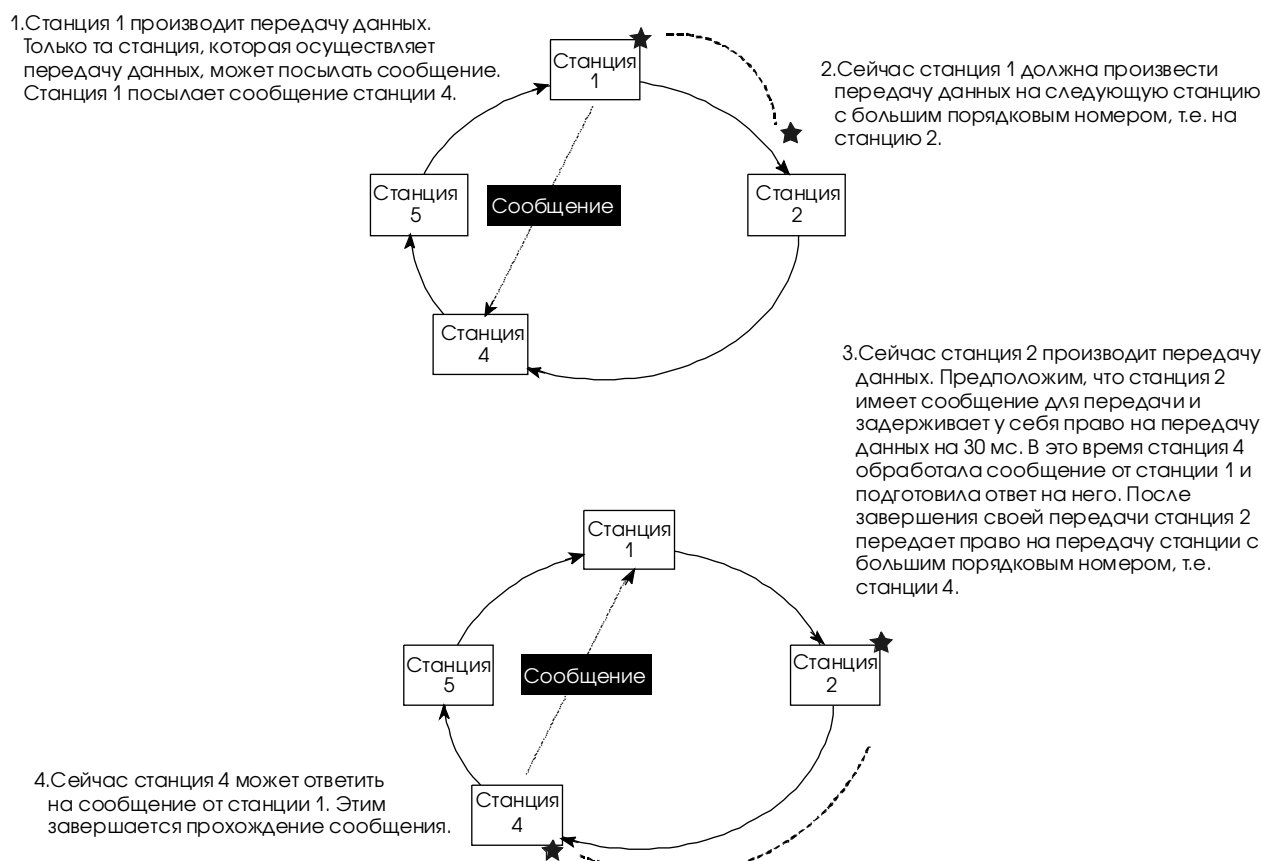
Число сообщений, посылаемых станцией, воздействует на время прохождения передачи. Например, если станция имеет в очереди на передачу три сообщения и разрешено четвертое, четвертому сообщению, возможно, придется ожидать до обработки предыдущих трех.

Кому предназначено сообщение

Время прохождения данных сильно зависит от того, может ли принимающая станция обработать сообщение и выдать ответ до того, как получит право на передачу. На рисунке 10.2 показано, как проходит сообщение от станции 1 на станцию 4.

Рисунок 10.2

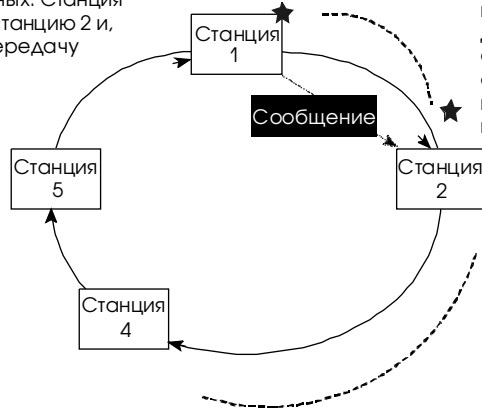
Станция, которой предназначено сообщение, имеет достаточно времени для выдачи ответа на сообщение



На рисунке 10.2 станция 4 имеет достаточно времени для обработки сообщения и выдачи ответа на него. На рисунке 10.3 станция 2 не имеет достаточно времени для выдачи ответа на сообщение.

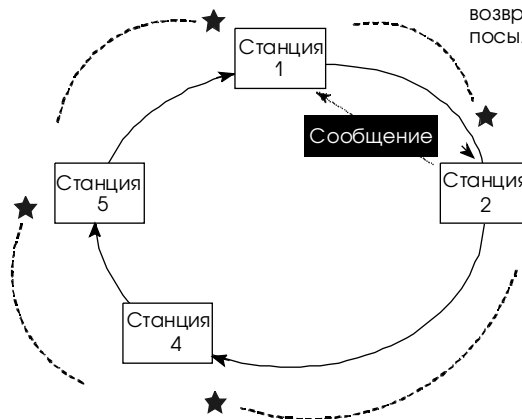
Рисунок 10.3
Станция, которой предназначено сообщение, не имеет достаточно времени для выдачи ответа на сообщение

1.Предположим, что на этом рисунке станция 1 собирается передать идентичное сообщение, как на рисунке 10.2, но только на станцию 2. Станция 1 производит передачу данных. Станция 1 посылает сообщение на станцию 2 и, затем передает право на передачу данных станции 2.



2.Сейчас станция 2 имеет право на передачу данных, но у нее нет достаточно времени для выдачи ответа на станцию 1. Таким образом, станция 2 посылает любое другое имеющееся сообщение и передает право на передачу данных станции 4.

3.Станции 4, 5 и 1 по порядку осуществляют свое право на передачу данных и посылают любые имеющиеся сообщения.



4.Право на передачу данных теперь возвращается к станции 2, которая посылает ответ на станцию 1.

В этом примере требуется дополнительный проход передачи вокруг сети для завершения прохождения сообщения, даже при том, что сообщение было идентично приведенному на рисунке 10.2

Внутреннее время обработки

Внутреннее время обработки зависит от того, насколько загружен процессор в сети при посылке или приеме сообщения.

Например, процессор А получает запрос ЧТЕНИЕ (READ) от процессора В в сети. Если процессор А уже имеет три собственных сообщения для передачи, передача ответа на запрос ЧТЕНИЕ от процессора В будет задержана до выполнения имеющихся до этого сообщений.

Начальные уставки тестирования

Один из 22 процессоров PLC-5 использовался с одним терминалом программирования в режиме онлайн. В каждом процессоре PLC-5 выполнялась цикловая программа размером 1К.

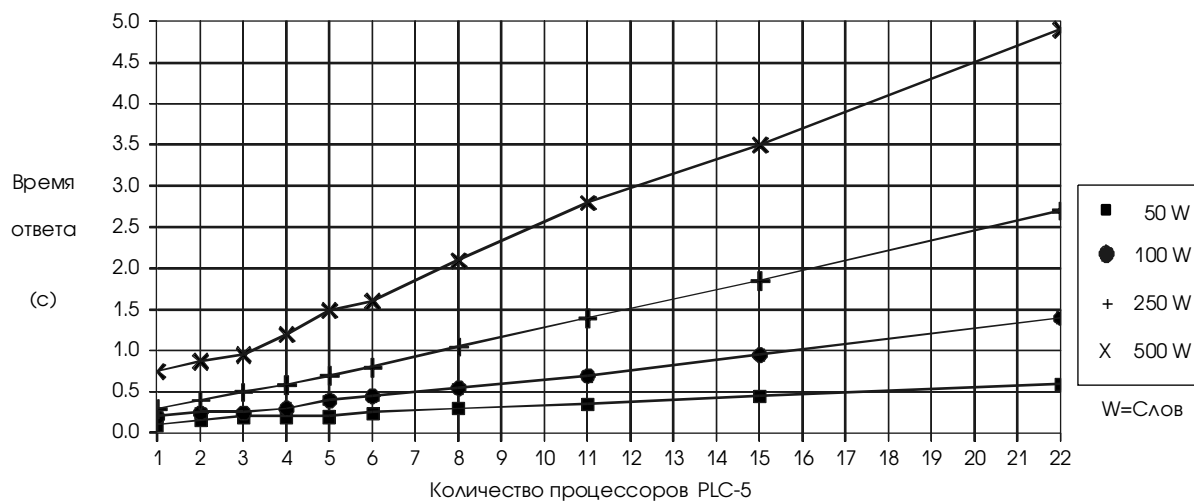
Первоначальное тестирование было произведено с одним процессором PLC-5, записывающим данные в другой процессор PLC-5. Время ответа записывалось. В сеть добавлялись дополнительные процессоры PLC-5, записывающие такое же количество данных в следующие процессоры с большим номером станции. Было выполнено четыре отдельных тестирования с использованием объема передаваемой информации в 50, 100, 250 и 500 слов.

Результаты тестирования лучшего времени ответа сети DH+

В этом разделе графиками представлены результаты тестирования сети DH+, где изменялось число станций и число передаваемых слов в сообщениях.

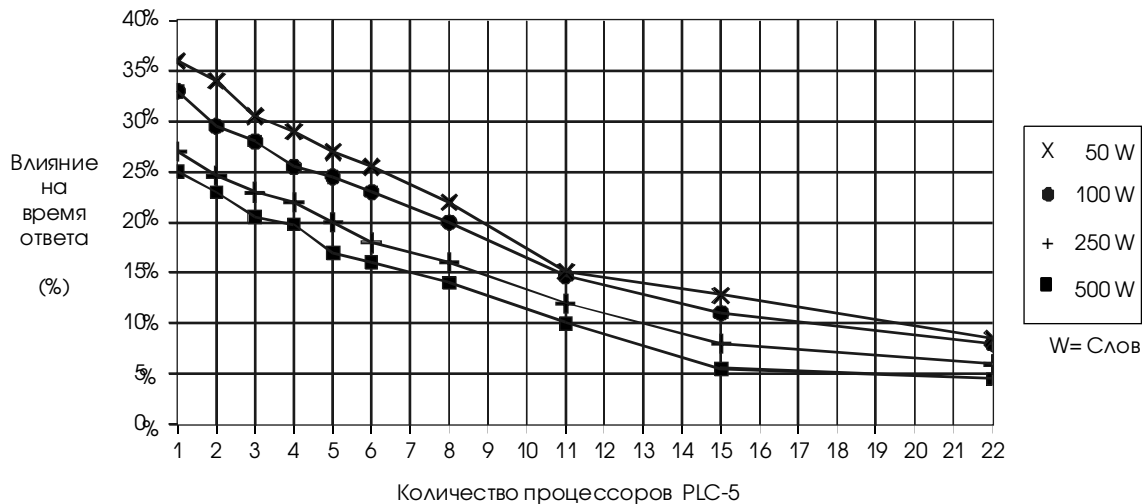
На рисунке 10.4 показано лучшее время ответа в сети DH+ для сообщений различной величины для разного количества станций. Здесь вы можете определить типовое время ответа в вашей сети DH+.

Рисунок 10.4
Лучшее время ответа для всех процессоров PLC-5



На рисунке 10.5 приведено влияние терминала программирования на время ответа на сообщение при различных конфигурациях.

Рисунок 10.5
Увеличение времени ответа (%) из-за влияния терминала программирования



Требования к приложениям

Советы

Выполните следующие требования при проектировании сети DN+ для вашей системы:

- уменьшите число узлов сети DN+ для получения приемлемого времени ответа. Учитывайте частоту и размер сообщений между оборудованием;
- ограничьте число узлов вашей сети, если вы хотите получить самый быстрый ответ. Вы можете использовать отдельные сети DN+, соединенные через "мосты" дополнительных станций. Используйте мосты для соединения сетей DN+;
- когда вы подключаете к сети компьютер как интерфейс оператора или какое-то третье оборудование, выберите самую быструю скорость передачи последовательного интерфейса;
- не добавляйте и не убирайте узлы из сети во время работы процесса и механизмов. Если передача данных в сети производится оборудованием, которое убирается, порядок передачи данных может быть утерян для всей оставшейся части сети. Сеть автоматически восстановится, но это займет несколько секунд. Управление в это время будет ненадежным или прерванным;
- сеть DN+ имеет таймаут в 90 сек, однако, вы можете ввести предохранительные таймеры в программе логики для передачи данных по сети DN+ (чтобы при сбое обеспечить организованное отключение);
- когда это возможно, не программируйте процессор в режиме онлайн во время работы процесса и механизмов. Это может привести к увеличению размеров пакетов данных, передаваемых по сети DN+, и сильно увеличить время ответа. Для большей информации смотрите главу 9;
- когда это возможно, добавьте выделенную сеть DN+ для программируемых процессоров для снижения влияния терминала программирования на работу сети DN+.

Связь с оборудованием по последовательному порту

Назначение этой главы

Сведения	Страница
Выбор между RS-232C, RS-422A и RS-423	11-1
Настройка последовательного порта процессора	11-2
Использование канала 0	11-2
Подключение	11-5
Конфигурация канала 0	11-5
Просмотр состояния канала 0	11-18



Если ваш процессор PLC-5 больше используется для задач супервизорного контроля и сбора данных (SCADA) смотрите:

- **SCADA System Selection Guide**, публикация AG-2.1
- **SCADA System Application Guide**, публикация AG-6.5.8

Выбор между RS-232C, RS-422A и RS-423

В таблице, приведенной ниже, показаны различия между протоколами связи RS-232C, RS-422A и RS-423:

Протокол:	Используется, если:
RS-232C	длина трассы передачи данных до 50 футов (15,2 м); Приложения, где расстояния превышают указанную величину, могут использовать модемы или line drives; используйте протокол RS-232C для полу- или дуплексной связи. Например, связь с процессорами или модемами в приложениях SCADA .
RS-422A	вы хотите передавать данные на RS-422A совместимое оборудование на расстояния, превышающие возможности RS-232C. Смотрите таблицу 11.A на странице 11-5; используйте связь RS-422A типа «точка-точка» для одной единицы оборудования, связывающегося не менее чем с 16 единицами оборудования.
RS-423	вы хотите передавать данные на RS-423 совместимое оборудование на расстояния, превышающие возможности RS-232C. Смотрите таблицу 11.A на странице 11-5; используйте связь RS-423 типа «точка-точка» для одной единицы оборудования, связывающегося не менее чем с 16 единицами оборудования.

Настройка последовательного порта процессора

Канал 0 является последовательным портом и настраивается для RS-232C, RS-423 или RS-422A совместимых протоколов. Используйте переключатель SW2 для определения конфигурации последовательного порта.

Для получения информации об установке переключателей на процессоре смотрите главу 23 или же посмотрите в таблице, расположенной на боковой стороне процессора, назначение каждого переключателя в блоке переключателей SW2.

Использование канала 0

Вы можете использовать последовательный порт процессора (канал 0) для связи процессора с оборудованием, которое:

- может посылать и принимать символы ASCII в режиме **User** («Пользователь»);
- связываться по протоколу DF1, используя один из трех режимов **System** («Системный»)..

Режим **User** («Пользователь»)

В режиме **User** все данные принимаются и посылаются через буфер. Для обработки или отправки этих данных используйте ASCII-инструкции в вашей цикловой программе. Данные типа ASCII, посылаемые процессором, не содержат никаких дополнительных характеристик протокола связи.

В режиме **User** могут использоваться только ASCII-инструкции. Если попытаетесь использовать инструкцию сообщения (MSG), адресованную на последовательный порт, будет установлен бит ошибки (.ER).

Примером оборудования, использующим ASCII данные, являются:

- терминалы ASCII;
- датчики чтения штрихового кода;
- Allen-Bradley Dataliner;
- weigh scales;
- принтеры.

Режим System («Системный»)

В режиме **System** процессор обрабатывает все данные от другого оборудования. Используйте режим **System** для связи с другим оборудованием по сети. Режим **System**, с протоколом DF1, является отдельным и отличающимся сетевым протоколом от сети DH+.

В режиме **System** вы можете передавать данные на оборудование, используя:

- инструкцию связи (MSG); или
- инструкцию записи ASCII (ASCII write) (посылаемую как строка ASCII).

Все данные помещены внутри пакета протокола связи DF1; поэтому процессор может связываться только с оборудованием, поддерживающим протокол DF1.

Примером оборудования, поддерживающего протокол DF1 являются:

- терминалы программирования;
- такие модули связи, как 1771-KF2 серии C, 1771-KE, 1771-KF и 1785-KE;
- модемы.

Используйте режим: Для того, чтобы выполнить:

«Точка-точка»	Связь между процессором PLC-5 и одной единицей оборудования, поддерживающим протокол DF1. В режиме «точка-точка» процессор PLC-5 использует дуплексный протокол.
DF1 режим Master	Управление посылкой и прохождением сообщений между управляющей (Master) и любой удаленной станцией. В режиме Master процессор PLC-5 использует полу-дуплексный режим посылки данных. Сеть в данном режиме включает в себя один процессор PLC-5, сконфигурированный для режима Master и до 254 удаленных узлов. Удаленное оборудование использует модемы или line drivers. В сети могут быть узлы с номерами 0-376 (восьмеричные). Узел 377 зарезервирован для пересылки. Каждый узел должен иметь свой собственный, отличный от других, номер. Также для создания данной сети должно присутствовать не менее 2 узлов (1 станция Master и одна - удаленная как раз составляют две станции).
DF1 режим Slave	Использование процессора как удаленной станции в последовательной сети типа Master/Slave. Когда в сети есть несколько удаленных станций, вы связываете удаленное оборудование, используя модемы или line drivers. Если в вашей сети всего одна удаленная станция, вам не нужен модем для связи удаленной станции со станцией Master; вы можете конфигурировать управляемые параметры без подтверждения связи. Вы можете подключить от 2 до 255 единиц оборудования к одной сети. В режиме Slave процессор «следует» за полудуплексным протоколом DF1. Один из узлов объявляется как Master и он управляет доступом к сети. Например, Master может быть как процессор PLC-5/250 или PLC-5/40, так и компьютер, на котором запущено программное обеспечение ControlView с опцией SCADA. Все другое оборудование будет являться удаленными станциями и должно ожидать разрешения на передачу данных от станции Master. Управляющая станция (Master) (кроме PLC-5/250) может посылать или принимать сообщения от всех узлов сети и к другим узлам сетей Data Highway, присоединенных multidrop; поэтому удаленная станция может только отвечать станции Master.

Метод связи станции Master с удаленной станцией

Станция PLC-5 Master может связаться с удаленной станцией двумя методами:

Метод связи:	Наименование опции:	Основные преимущества:
<p>Инициализация пакета отправки на удаленную станцию в соответствии с ее положением в списке опроса</p> <p>Посылаемые пакеты формируются независимо от программы пользователя.</p>	Стандартный способ связи	<p>Этот метод связи должен использоваться в конфигурации point-to-multipoint.</p> <p>Имеются возможности:</p> <ul style="list-style-type: none"> удаленная станция может посылать сообщения на станцию Master (отправка ответа на запрос); удаленная станция может посылать сообщения к удаленным станциям; позволяет станции Master обслуживать таблицу активных узлов. <p>Пакет отправки включает в себя адрес назначения и файл данных целочисленного типа. Вы можете:</p> <ul style="list-style-type: none"> включать в список опроса станцию Master; конфигурировать станцию Master для пересылки данных между станциями (Master передает любое необходимое сообщение перед отправкой данных на другую удаленную станцию); иметь Master станцию как в списке отправок так и сконфигурированную для запросов между станциями.
<p>Инициализация пакета отправки на удаленную станцию, с использованием инструкции сообщений (MSG).</p> <p>Каждый запрос данных от удаленной станции должен программироваться в инструкции сообщения.</p> <p>Станция Master запрашивает от удаленной станции ответ на сообщение после указанного пользователем периода времени. Время ожидания предоставляется удаленной станции для подготовки ответа и его передачи. После передачи всех сообщений станцией Master разрешается передача сообщений между удаленными станциями.</p>	Метод связи, основанный на инструкциях сообщений	<p>Если в вашем приложении используется передача информации через спутники связи или через коммутируемые телефонные сети, выберите данный метод. Связь с удаленной станцией инициируется по необходимости.</p> <p>Выберите данный метод, если вам нужно связываться с удаленными неинтеллектуальными терминалами (RTUs).</p>

Изменение режима

Для изменения режима связи сконфигурируйте канал 0 процессора используя:

- экран конфигурации канала в пакете программирования;
- параметры изменения режима связи (управляющие параметры ASCII), посылаемые в удаленный процессор, который сам переключает режим.

Подключение

В таблице 11.A показана максимальная длина кабеля, которую можно применять с каналом 0.

Таблица 11.A
Длина кабеля для связи через RS-порт

Порт	Скорость передачи	Максимальная длина кабеля
RS-232C	Все	15 м (50 футов)
RS-422A (совместимый)	Все	61 м (200 футов)
RS-423	Все	61 м (200 футов)

Важно: Руководствуйтесь следующими принципами:

- когда канал 0 сконфигурирован как RS-422A совместимый, не используйте в сети терминаторы;
- когда канал 0 сконфигурирован как RS-422A совместимый, не превышайте длину 61 м (200 футов). Это ограничение расстояния не зависит от скорости передачи данных.

Перечень кабелей программирования через последовательный порт и описание разъемов смотрите в главе 25.

Конфигурация канала 0

Используйте блок переключателей SW2 на процессоре для определения вида связи канала 0 - RS-232C, RS-422A (совместимый) или RS-423.

Вы можете сконфигурировать канал 0 для связи, используя следующие протоколы:

Если вы хотите использовать:		Смотрите страницу:
Режим System	DF1 «точка-точка»	11-6
	DF1 slave	11-8
	DF1 master	11-10
Режим User	ASCII	11-15

Конфигурация канала 0 для связи DF1 «точка-точка»

Чтобы сконфигурировать канал 0 для режима связи DF1 «Точка-точка», используйте экран конфигурации системного режима (system mode configuration screen) вашего пакета программирования.

```

System Mode (DF1 Point-to-Point)
Channel 0 Configuration

Diag. file: 20

Remote mode change:      DISABLED          System mode char.:      S
Mode attention char.:    \0x1b             User mode char.:       U

Baud rate:               2400              Parity:                 NONE
Stop bits:               1

Control line:            NO HANDSHAKING

Duplicate detect:        ON                  Error detect:          BCC
ACK timeout (20 ms):    100              NAK receive:           3
Msg appl timeout (30sec): 0              DF1 ENQS:              3
  
```

Установите оборудование как Serial to PLC, метод обнаружения ошибки должен быть идентичен в PLC и в терминале.

Поле экрана:	Предназначено для:	Сконфигурируйте следующим образом:
Diag. file	Файл, содержащий информацию о состоянии канала	Введите номер целочисленного файла (10-999). ВНИМАНИЕ: Используйте отдельный файл диагностики для каждого канала. Не используйте в качестве файла диагностики файл, предназначенный для контроля состояния входов/выходов или другой используемый файл. Это может привести к непредсказуемым повреждениям машин и механизмов. Важно: Вы должны определить диагностический файл для любого сконфигурированного канала, даже и не используемого (т.е. если вы не используете канал), если вы хотите иметь информацию о статусе этого канала.
Remote mode change	Разрешение удаленного изменения режима	Выберите РАЗРЕШЕНО (ENABLED) или ЗАПРЕЩЕНО (DISABLE)
Mode attention char.	Параметры разрешения для удаленного изменения режима System или User	Введите параметр . Здесь вы должны использовать параметры управления, соответствующие ASCII.
System mode char.	Параметры, используемые с mode attention character (смотри выше)	Введите параметр. Здесь вы должны использовать параметры управления, соответствующие ASCII. Когда процессор обрабатывает параметр внимание (attention) и параметр "режим System", он устанавливает связь по каналу 0 в режим System. Опция Remote mode change должна быть РАЗРЕШЕНА (ENABLED).
User mode char.	Параметры, используемые с mode attention character (смотри выше)	Введите параметр "Режим User". Здесь вы должны использовать параметры управления, соответствующие ASCII. Когда процессор обрабатывает параметр внимание (attention) и параметр "режим User", он устанавливает связь по каналу 0 в режим User. Опция Remote mode change должна быть РАЗРЕШЕНА (ENABLED).
Baud rate	Скорость передачи по каналу 0. Установите для всех единиц оборудования системы одну скорость передачи.	Выберите 110, 300, 600, 1200, 2400, 4800, 9600 или 19.2 00 кБод.
Stop bits	Введите количество стоповых битов (stop bits) для оборудования, с которым вы связываетесь	Выберите 1, 1.5 или 2

Поле экрана:	Предназначено для:	Сконфигурируйте следующим образом:
Parity	Установка паритета (parity) для канала 0. Паритет осуществляет дополнительную проверку на наличие ошибки в пакете сообщения	Выберите НЕТ (NONE) или ВСЕГДА (EVEN)
Control line	Режим, в котором работает драйвер	Выберите метод в соответствии с конфигурацией вашей системы: <ul style="list-style-type: none"> если вы не используете модем - выберите БЕЗ ПОДТВЕРЖДЕНИЯ (NO HANDSHAKING); если вы используете дуплексный модем - выберите ДУПЛЕКСНЫЙ (FULL-DUPLEX).
Duplicate detect	Здесь вы разрешаете процессору обнаруживать и игнорировать дублирующиеся сообщения.	Выберите желаемую установку.
Error detect	Здесь вы устанавливаете наличие ошибки по BCC или CRC.	Выберите одно из: <ul style="list-style-type: none"> BCC: процессор посылает и обрабатывает сообщения, заканчивающиеся байтом контроля ошибки BCC. BCC легче и быстрее для обработки драйверами компьютера. CRC: процессор посылает и обрабатывает сообщения с 2-я байтами CRC контроля ошибки. CRC осуществляет более полную проверку. CRC осуществляет более полную проверку. Установите на обеих станциях один метод контроля ошибки.
ACK timeout	Время, в течение которого процессор ожидает подтверждения о приеме переданного сообщения.	Введите цифру 0-65535. Ограничение вводится с шагом в 20 мсек. Например: для времени ожидания 40 мсек введите 2. Рекомендуемое время ожидания - 1 сек. Для определения 1 сек введите 50.
MSG appl timeout	Время (секунды), в течение которого должно быть получено повторное сообщение перед установкой бита ошибки в сообщении. Таймер начинает работать при первом запуске инструкции сообщения цикловой программой и перезапускается, если/ когда получен ACK.	Введите одну из цифр: <ul style="list-style-type: none"> 1: 30-60 сек 2: 60-90 сек 3: 90-120 сек 4: 120-150 сек 5: 150-180 сек 6: 180-210 сек 7: 210-240 сек
NAK receive	Количество NAK, посылаемое процессором на переданное сообщение	Введите цифру 0-255. Рекомендуемая уставка - 3.
DF1 ENQS	Количество запросов, посылаемых процессором после таймаута ACK	Введите цифру 0-255. Рекомендуемая уставка - 3.

Конфигурация канала 0 как станции Slave

Чтобы сконфигурировать канал 0 для режима связи DF1 Slave, используйте экран конфигурации системного режима (system mode configuration screen) вашего пакета программирования.

```

System Mode (DF1 Slave)
Channel 0 Configuration

Diag. file:          10
Remote mode change: DISABLED
Mode attention char.: \0x1b
Baud rate:          2400
Stop bits:          1
Control line:       NO HANDSHAKING
Duplicate detect:   ON
ACK timeout (20 ms): 100
DF1 retries:        3
Msg appl timeout (30sec): 0

System mode char.:  S
User mode char.:   U
Parity:            NONE
Station address:   1
Error detect:      BCC
RTS off delay (20 ms): 0
RTS send delay (20 ms): 0

```

Поле экрана:	Предназначено для:	Сконфигурируйте следующим образом:
Diag. file	Файл, содержащий информацию о состоянии канала	Введите номер целочисленного файла (10-999). ВНИМАНИЕ: Используйте отдельный файл диагностики для каждого канала. Не используйте в качестве файла диагностики файл, назначенный для контроля состояния входов/выходов или другой используемый файл. Это может привести к непредсказуемым повреждениям машин и механизмов. Важно: Вы должны определить диагностический файл для любого сконфигурированного канала, даже и неиспользуемого (т.е. если вы не используете канал), если вы хотите иметь информацию о статусе этого канала.
Remote mode change	Разрешено ли удаленное изменение режима	Выберите РАЗРЕШЕНО (ENABLED) или ЗАПРЕЩЕНО (DISABLE)
Mode attention char.	Параметры разрешения для удаленного изменения режима System или User	Введите параметр Attention. Должны использоваться параметры управления, соответствующие ASCII.
System mode char.	Параметры, используемые с mode attention character (смотри выше)	Введите параметр "System". Должны использоваться параметры управления, соответствующие ASCII. Когда процессор обрабатывает параметр «внимание» (attention) и параметр "режим System", он устанавливает связь по каналу 0 в режим System. Опция Remote mode change должна быть РАЗРЕШЕНА (ENABLED).
User mode char.	Параметры, используемые с mode attention character (смотри выше)	Введите параметр "User". Должны использоваться параметры управления, соответствующие ASCII. Когда процессор обрабатывает параметр внимание (attention) и параметр "режим User", он устанавливает связь по каналу 0 в режим User. Опция Remote mode change должна быть РАЗРЕШЕНА (ENABLED).
Baud rate	Скорость передачи по каналу 0. Установите для всех единиц оборудования системы одну скорость передачи.	Выберите 110, 300, 600, 1200, 2400, 4800, 9600 или 19.200 кБод.
Stop bits	Введите количество стоповых битов (stop bits) для оборудования, с которым вы связываетесь	Выберите 1, 1.5 или 2
Parity	Установка паритета (parity) для канала 0.	Выберите НЕТ (NONE) или ВСЕГДА (EVEN)
Station address	Адрес станции для канала 0 DF1 полудуплексной сети	Введите правильный адрес сети DF1 (0-376 восьмеричный)

Поле экрана:	Предназначено для:	Сконфигурируйте следующим образом:
Control line	Режим, в котором работает драйвер	<p>Выберите метод в соответствии с конфигурацией вашей системы:</p> <ul style="list-style-type: none"> · если вы не используете модем - выберите БЕЗ ПОДТВЕРЖДЕНИЯ (NO HANDSHAKING); · если Master модем - дуплексный с постоянной несущей частотой и удаленный модем полу-дуплексный, выберите ПОЛУДУПЛЕКСНЫЙ МОДЕМ С ПОСТОЯННОЙ НЕСУЩЕЙ (HALF-DUPLEX MODEM WITH CONTINUOUS CARRIER); · если все модемы в системе - полу-дуплексные, выберите ПОЛУДУПЛЕКСНЫЙ МОДЕМ БЕЗ ПОСТОЯННОЙ НЕСУЩЕЙ (HALF-DUPLEX MODEM WITHOUT CONTINUOUS CARRIER).
Duplicate detect	Здесь вы разрешаете процессору обнаруживать и игнорировать дублирующиеся сообщения.	Выберите желаемую установку.
Error detect	Здесь вы устанавливаете наличие ошибки по BCC или CRC.	<p>Выберите одно из:</p> <ul style="list-style-type: none"> · BCC: процессор посылает и обрабатывает сообщения, заканчивающиеся байтом контроля ошибки BCC. BCC легче и быстрее для обработки драйверами компьютера; · CRC: процессор посылает и обрабатывает сообщения с 2-я байтами CRC контроля ошибки. CRC осуществляет более полную проверку.
ACK timeout	Время, в течение которого процессор ожидает подтверждения о приеме переданного сообщения.	Введите цифру 0-65535. Ограничение вводится с шагом в 20 мсек. Например: для времени ожидания 40 мсек введите 2. Рекомендуемое время ожидания - 1 сек. Для определения 1 сек введите 50.
RTS off delay	Время между окончанием передачи сообщения и подтверждением сигнала RTS.	Введите цифру 0-255. Ограничение вводится с шагом в 20 мсек. Например: для времени ожидания 40 мсек введите 2.
DF1 retries	Время, в течение которого удаленная станция повторяет сообщение, перед тем, как объявить его "нежизнеспособным"	Введите цифру 0-255. Рекомендуемая уставка - 3.
RTS send delay	Время между установкой сигнала RTS и началом передачи сообщения. Это время дает возможность модему подготовиться к передаче сообщения. Сигнал CTS должен быть высоким для прохождения сообщения.	Введите цифру 0-255. Ограничение вводится с шагом в 20 мсек. Например: для времени ожидания 40 мс введите 2. Рекомендуется устанавливать время 0, если не применяется модем, автоматически возвращающий CTS как только получен RTS. В этом случае введите уставку времени для того, чтобы модем был готов к передаче перед тем, как он попытается передать сообщение.
MSG appl timeout	Время (секунды), в течение которого должно быть получено повторное сообщение перед установкой бита ошибки в сообщении. Таймер начинает работать при первом запуске инструкции сообщения цикловой программой и перезапускается, если/ когда получен ACK.	<p>В поле ввода введите величину от 1 до 7, и нажмите (Enter).</p> <p>Разрешено ввести:</p> <ul style="list-style-type: none"> • 1: 30-60 с; • 2: 60-90 с; • 3: 90-120 с; • 4: 120-150 с; • 5: 150-180 с; • 6: 180-210 с; • 7: 210-240 с.

Конфигурация канала 0 как станции Master

Чтобы сконфигурировать канал 0 для режима связи DF1 Master, используйте экран конфигурации системного режим (system mode configuration screen) вашего пакета программирования.

```

System Mode (DF1 Master)
Channel 0 Configuration

Diag. file: 20
Remote mode change:      DISABLED
Mode attention char.:    \0x1b
Baud rate:               2400
Stop bits:               1
Control line:            NO HANDSHAKING
Reply msg wait (20 ms):  1792
ACK Timeout (20 ms):    100
DF1 retries:             3
Msg appl timeout (30sec): 0

System mode char.:      S
User mode char.:        U
Parity:                 NONE
Station address:        1
Error detect:           BCC
RTS off delay (20 ms):  0
RTS send delay (20 ms): 0

Polling Mode:          STANDARD (MULTIPLE MESSAGE TRANSFER PER NODE FILE SCAN)

```

Поле экрана:	Предназначено для:	Сконфигурируйте следующим образом:
Diag. file	Файл, содержащий информацию о состоянии канала	Введите номер целочисленного файла (10-999). ВНИМАНИЕ: Используйте отдельный файл диагностики для каждого канала. Не используйте в качестве файла диагностики файл, назначенный для контроля состояния входов/выходов или другой используемый файл. Это может привести к непредсказуемым повреждениям машин и механизмов. Важно: Вы должны определить диагностический файл для любого сконфигурированного канала, даже и неиспользуемого (т.е. если вы не используете канал), если вы хотите иметь информацию о статусе этого канала.
Remote mode change	Разрешено ли удаленное изменение режима	Выберите РАЗРЕШЕНО (ENABLED) или ЗАПРЕЩЕНО (DISABLE)
Mode attention char.	Параметры разрешения для удаленного изменения режима System или User	Введите параметр «Внимание» (Attention). Здесь вы должны использовать параметры управления, соответствующие ASCII.
System mode char.	Параметры, используемые с mode attention character (смотри выше)	Введите параметр «Режим System». Здесь вы должны использовать параметры управления, соответствующие ASCII. Когда процессор обрабатывает параметр «внимание» (attention) и параметр «режим System», он устанавливает связь по каналу 0 в режим System. Опция Remote mode change должна быть РАЗРЕШЕНА (ENABLED).
User mode char.	Параметры, используемые с mode attention character (смотри выше)	Введите параметр «Режим User». Здесь вы должны использовать параметры управления, соответствующие ASCII. Когда процессор обрабатывает параметр «внимание» (attention) и параметр «режим User», он устанавливает связь по каналу 0 в режим User. Опция Remote mode change должна быть РАЗРЕШЕНА (ENABLED).
Baud rate	Скорость передачи по каналу 0. Установите для всех единиц оборудования системы одну скорость передачи.	Выберите 110, 300, 600, 1200, 2400, 4800, 9600 или 19.200 кБод.
Stop bits	Введите количество стоповых битов (stop bits) для оборудования, с которым вы связываетесь	Выберите 1, 1.5 или 2

Поле экрана:	Предназначено для:	Сконфигурируйте следующим образом:
Control line	Режим, в котором работает драйвер	<p>Выберите метод в соответствии с конфигурацией вашей системы:</p> <ul style="list-style-type: none"> • если вы не используете модем - выберите БЕЗ ПОДТВЕРЖДЕНИЯ (NO HANDSHAKING); • если Master модем - дуплексный с постоянной несущей частотой и удаленный модем полудуплексный, выберите ПОЛУДУПЛЕКСНЫЙ МОДЕМ С ПОСТОЯННОЙ НЕСУЩЕЙ (HALF-DUPLEX MODEM WITH CONTINUOUS CARRIER); • если все модемы в системе - полудуплексные, выберите ПОЛУДУПЛЕКСНЫЙ МОДЕМ БЕЗ ПОСТОЯННОЙ НЕСУЩЕЙ (HALF-DUPLEX MODEM WITHOUT CONTINUOUS CARRIER).
Parity	Установка паритета (parity) для канала 0.	Выберите НЕТ (NONE) или ВСЕГДА (EVEN)
Station address	Адрес станции для канала 0 сети DF1	Введите правильный адрес сети DF1 (0-376 восьмеричный)
Reply msg wait	Время, в течение которого станция Master после получения ACK будет в режиме ожидания до выдачи станции slave запроса на повторную передачу.	Определяется только для метода связи, основанного на инструкциях сообщений. Введите величину 0-65535 (с шагом 20 мс).
Error detect	Здесь вы устанавливаете наличие ошибки по ВСС или CRC.	<p>Выберите одно из:</p> <ul style="list-style-type: none"> • ВСС: процессор посылает и обрабатывает сообщения, заканчивающиеся байтом контроля ошибки ВСС. ВСС легче и быстрее для обработки драйверами компьютера; • CRC: процессор посылает и обрабатывает сообщения с 2-я байтами CRC контроля ошибки. CRC осуществляет более полную проверку.
ACK timeout	Время, в течение которого процессор ожидает от slave станции подтверждения о приема сообщения перед повторной передачей или установкой ошибки передачи сообщения.	Введите цифру 0-65535. Ограничение вводится с шагом в 20 мс. Например: для времени ожидания 40 мс введите 2. Рекомендуемое время ожидания - 1 с. Для определения 1 с введите 50.
DF1 retries	Время, в течение которого slave станция повторяет сообщение перед тем, как объявить его "нежизнеспособным"	Введите цифру 0-255.
RTS send delay	Время между установкой сигнала RTS и началом передачи сообщения.	Введите цифру 0-255. Ограничение вводится с шагом в 20 мс. Например: для времени ожидания 40 мс введите 2. Рекомендуется устанавливать время 0 если не применяется модем, автоматически возвращающий CTS как только получен RTS. В этом случае введите уставку времени для того, чтобы модем был готов к передаче перед тем, как он попытается передать сообщение.
RTS off delay	<p>Это время дает возможность модему подготовиться к передаче сообщения.</p> <p>Время между окончанием передачи сообщения и подтверждением сигнала RTS.</p>	Введите цифру 0-255. Ограничение вводится с шагом в 20 мс. Например: для времени ожидания 40 мс введите 2. Рекомендуется устанавливать время 0, если не применяется модем, автоматически возвращающий CTS, как только получен RTS. В этом случае введите уставку времени для того, чтобы модем был готов к передаче перед тем, как он попытается передать сообщение.
MSG appl timeout	Время (секунды), в течение которого должно быть получено повторное сообщение перед установкой бита ошибки в сообщении. Таймер включается при первом запуске инструкции сообщения и перезапускается, если/когда получен ACK.	<p>Введите:</p> <ul style="list-style-type: none"> • 1: 30-60 с; • 2: 60-90 с; • 3: 90-120 с; • 4: 120-150 с; • 5: 150-180 с; • 6: 180-210 с; • 7: 210-240 с.

Поле экрана:	Предназначено для:	Сконфигурируйте следующим образом:
Polling mode	Текущий метод связи	<p>Выберите метод в соответствии с конфигурацией вашей системы:</p> <ul style="list-style-type: none"> • ОСНОВАННЫЙ НА СООБЩЕНИЯХ (ПОЗВОЛИТЬ ВЕДОМОЙ СТАНЦИИ ИНИЦИИРОВАТЬ СООБЩЕНИЯ) -эта опция, установленная по умолчанию позволяет удаленной станции выдавать сообщения, которые будут обработаны после окончания передачи сообщений Master-станцией. При этом подтверждается, что сообщения будут обработаны; • ОСНОВАННЫЙ НА СООБЩЕНИЯХ (НЕ ПОЗВОЛЯТЬ ВЕДОМОЙ СТАНЦИИ ИНИЦИИРОВАТЬ СООБЩЕНИЯ) - удаленная станции выдает сообщения, которые будут подтверждены, но не будут обработаны; • СТАНДАРТНЫЙ (НЕСКОЛЬКО СООБЩЕНИЙ ПЕРЕДАЮТСЯ ЗА ОДИН ЦИКЛ) - основная передающая станция выбирается из списка; каждая станция может передавать несколько сообщений за один цикл сети; • СТАНДАРТНЫЙ (ОДНО СООБЩЕНИЕ ПЕРЕДАЁТСЯ ЗА ОДИН ЦИКЛ) - основная передающая станция выбирается из списка; каждая станция может передавать одно сообщение за один цикл сети.

Если выбран стандартный режим передачи:

Поле экрана:	Предназначено для:	Сконфигурируйте следующим образом:
Master message transmit	Передаются текущие данные канала 0 основной передающей станции	<p>Если вы хотите, чтобы Master-станция:</p> <ul style="list-style-type: none"> • передала все инструкции MSG, инициируемые Master-станцией, на удаленные станции перед разрешением передачи данных следующей удаленной станции в лист опроса, выберите «Пересылка Между Станциями» (Between Station Polls).. <p>Этот метод делает возможным регулярную посылку сообщений, инициируемых Master-станцией (после посылки сообщений каждой удаленной станцией).</p> <ul style="list-style-type: none"> • только посылаемые инструкции MSG, инициируемые Master-станцией, после ее выбора в Polling Sequence, выберите Poll Sequence. <p>В этом методе сообщения, посылаемые Master-станцией, зависят от расположения Master-станции в списке опроса. Для достижения таких же целей, как в методе Пересылки Между Станциями (Between Station Polls), адрес Master-станции должен располагаться после адреса каждой удаленной станции.</p> <p>Процессор устанавливает неосновную ошибку, если вы используете IN POLL SEQUENCE и Master-станции нет ни в обычном списке опроса ни в проритетном списке опроса.</p>
Normal poll node file	Целочисленный файл, содержащий адреса удаленных станций, которые вы хоткли бы видеть в нормальном списке опроса.	Введите номер целочисленного файла в диапазоне 10-999.
Normal poll group size	Количество активных станций, расположенных в нормальном списке опроса, которым разрешена передача сообщений во время скана перед возвращением в проритетный список опроса.	Введите число в диапазоне 10-999.
Priority poll node file	Файл тпа N, содержащий адреса удаленных станций, которые вы хоткли бы видеть в проритетном спискеопроса.	Введите номер целочисленного файла в диапазоне 10-999.

Для определения схемы передачи данных с использованием стандартного режима вы должны определить в основном экране конфигурации DF1 (DF1 master configuration screen) вашего программного обеспечения:

Конфигурируемый параметр	Назначение
Polling mode	Каким образом Master-станция должна отправлять списки станций.
Master message transmit	Когда Master-станция должна посылать сообщения.
Normal poll file	Целочисленный файл, где вы помещаете номера удаленных станций. Размер по умолчанию - 64 слова.
Priority poll file	Целочисленный файл, где вы помещаете номера станций, с которых вам нужно чаще собирать информацию. Размер по умолчанию - 64 слова.
Normal poll group size	Количество станций, опрашиваемых Master-станцией прежде, чем будут опрошены станции в приоритетном списке опроса.
Active station file	Битовый файл, в котором хранятся адреса станций для всех активных станций в сети. Размер по умолчанию - 18 слов. Как нормальный, так и приоритетный список опроса может иметь активные и неактивные станции. Станция становится неактивной, когда она не отвечает на запрос Master-станции о передаче данных.

Master-станция опрашивает Slave-станцию в следующей последовательности.

1. Все станции в активном приоритетном файле опроса.
2. Указанные станции в активном нормальном файле опроса. Число станций, опрошенных в этом файле, определено нормальным размером группы опроса, указанным на экране конфигурации. Если размер группы, например, был 3, то три станции будут опрошены в нормальном файле перед тем, как ведущий перейдет к следующему шагу в последовательности.
3. Опрашивается одна станция в неактивном файле опроса после всех активных станций в нормальном файле опроса.

Для создания списка станций поместите адрес каждой станции в отдельном слове файла опроса (нормальном и/или приоритетном), начиная со слова 2. Назначение слов файла опроса следующее:

Слово файла опроса:	Содержит информацию:
Слово 0	Общее количество станции в списке.
Слово 1	Размещение адреса станции, опрашиваемой в настоящий момент. Например: цифра 1 означает, что опрашивается станция, адрес которой записан в слове 2, цифра 2 означает, что опрашивается станция, адрес которой записан в слове 3 и т.д. Это слово меняется автоматически при опросе Master-станцией новой удаленной станции.
Слова со 2 до xx	Адреса удаленных станций, в том порядке, в котором они опрашиваются.

Для того, чтобы поместить адрес станции в файл опроса сделайте следующее:

1. разрешите мониторинг данных в пакете программирования;
2. определите адрес целочисленного файла, который будет либо нормальным, либо приоритетным файлом опроса.(т.е. если нормальный файл опроса - N11, вы определяете N11:0);
3. введите адреса для удаленных станций в файл опроса, начиная со слова 2. Разместите их в том порядке, в котором они будут опрашиваться.

Важно: Адреса станций - восьмеричного типа. Файл опроса - целочисленного типа. По умолчанию основание - десятичное. Для того чтобы ввести адрес станции в списке опроса без ошибки, вы должны сделать одно из двух:

- изменить основание файла в восьмеричное;
- преобразовать восьмеричный адрес станции в десятичный перед вводом адреса.

Ниже приведен пример с тремя станциями с восьмеричными адресами 10, 11 и 12. Опрашивается станция 12 (в десятичном 10).

Файл опроса	Слово 0	Слово 1	Слово 2	Слово 3	Слово 4
N:11	3	3	08	09	10
N:xx	Общее количество станций	Показан адрес слова опрашиваемой станции	Адрес первой станции в списке опроса	Адрес второй станции в списке опроса	Адрес третьей станции в списке опроса

Конфигурация канала 0 для режима User (протокол ASCII)

Чтобы сконфигурировать канал 0 для режима User, воспользуйтесь экраном конфигурации режима User (user mode configuration screen) вашего пакета программирования.

```

User Mode
Channel 0 Configuration
Diag. file:          20                                XON/XOFF          DISABLED
Remote mode change: DISABLED                          System mode char.: S
Mode attention char.: \0x1b                           User mode char.:  U
Baud rate:          9600                               Parity:           NONE
Stop bits:          1                                 Bits per character: 8
Control line:       NO HANDSHAKING
Echo/delete mode:   CRT                               RTS send delay (20 ms): 0
RTS off delay (20 ms):0
Termination 1:     \0xd                               Append 1:         \0xd
Termination 2:     \0xff                              Append 2:         \0xa

```

Поле экрана:	Предназначено для:	Сконфигурируйте следующим образом:
Diag. file	Файл, содержащий информацию о состоянии канала	<p>Введите номер целочисленного файла (10-999).</p> <p>ВНИМАНИЕ: Используйте отдельный файл диагностики для каждого канала. Не используйте в качестве файла диагностики файл, назначенный для контроля состояния входов/выходов или другой используемый файл. Это может привести к непредсказуемым повреждениям машин и механизмов.</p> <p>Важно: Вы должны определить диагностический файл для любого сконфигурированного канала, даже и не используемого (т.е. если вы не используете канал), если вы хотите иметь информацию о статусе этого канала.</p>
XON/XOFF	Хотите вы или нет разрешить XON/XOFF	При получении данных процессор определяет, сколько еще он может принять данных без потери. Когда XON/XOFF разрешен, процессор посылает "остановить передачу данных" (XOFF). Если в оборудование-источник имеет возможность XON/XOFF, то оно останавливает передачу данных. Когда в процессоре память освободится, он пошлет "начать передачу данных" (XON). Выберите РАЗРЕШЕНО (ENABLED) или ЗАПРЕЩЕНО (DISABLE)
Remote mode change	Разрешено ли удаленное изменение режима	Выберите РАЗРЕШЕНО (ENABLED) или ЗАПРЕЩЕНО (DISABLE)
Mode attention char.	Параметры разрешения для удаленного изменения режима System или User	Введите параметр. Здесь вы должны использовать параметры управления, соответствующие ASCII.
System mode char.	Параметры, используемые с mode attention character (смотри выше)	Введите параметр. Здесь вы должны использовать параметры управления, соответствующие ASCII. Когда процессор обрабатывает параметр attention и параметр System, он устанавливает связь по каналу 0 в режим System. Опция Remote mode change должна быть РАЗРЕШЕНА (ENABLED).
User mode char.	Параметры, используемые с mode attention character (смотри выше)	Введите параметр. Здесь вы должны использовать параметры управления, соответствующие ASCII. Когда процессор обрабатывает параметр внимание (attention) и параметр "режим User", он устанавливает связь по каналу 0 в режим User. Опция Remote mode change должна быть РАЗРЕШЕНА (ENABLED).
Baud rate	Скорость передачи по каналу 0. Установите для всех единиц оборудования системы одну скорость передачи.	Выберите 110, 300, 600, 1200, 2400, 4800, 9600 или 19.200 кБод.

Поле экрана:	Предназначено для:	Сконфигурируйте следующим образом:
Stop bits	Введите количество стоповых битов (stop bits) для оборудования, с которым вы связываетесь	Выберите 1, 1.5 или 2
Parity	Установка паритета (parity) для канала 0.	Выберите НЕТ (NONE) или ВСЕГДА (EVEN)
Bits per character	Количество бит в блоке передаваемых данных	Выберите 7 или 8
Control line	Режим, в котором работает драйвер	Выберите одно из: · ПОЛУДУПЛЕКСНЫЙ МОДЕМ С ПОСТОЯННОЙ НЕСУЩЕЙ (HALF-DUPLEX MODEM WITH CONTINUOUS CARRIER) - если передающее оборудование и процессор имеет полудуплексный модем; · ПОЛУДУПЛЕКСНЫЙ МОДЕМ БЕЗ ПОСТОЯННОЙ НЕСУЩЕЙ (HALF-DUPLEX MODEM WITHOUT CONTINUOUS CARRIER) - если передающее оборудование имеет дуплексный модем, а процессор имеет полудуплексный модем; · БЕЗ ПОДТВЕРЖДЕНИЯ (NO HANDSHAKING) - если вы не используете модем.
Echo/delete mode	Что будет делать процессор при получении параметра ASCII "delete"	Выберите одно из: · DISABLE - данные принимаются процессором и посылаются только на счетчик Echo а не на выходное оборудование; · CRT - процессор по переднему фронту параметра "delete" игнорирует данные и посылает сигнал CRT для того, чтобы стереть данные. · PRINTER - процессор по переднему фронту параметра "delete" игнорирует данные и посылает сигнал на принтер для того, чтобы стереть данные.
RTS send delay	Время между установкой сигнала RTS и началом передачи сообщения.	Введите цифру 0-255. Ограничение вводится с шагом в 20 мс. Например: для времени ожидания 40 мс введите 2. Рекомендуется устанавливать время 0, если не применяется модем, автоматически возвращающий CTS, как только получен RTS. В этом случае введите уставку времени для того, чтобы модем был готов к передаче перед тем, как он попытается передать сообщение.
RTS off delay	Время между окончанием передачи сообщения и подтверждением сигнала RTS.	Введите цифру 0-255. Ограничение вводится с шагом в 20 мс. Например: для времени ожидания 40 мс введите 2. Рекомендуется устанавливать время 0 если не применяется модем, автоматически возвращающий CTS, как только получен RTS. В этом случае введите уставку времени для того, что бы модем был готов к передаче перед тем, как он попытается передать сообщение.
Termination 1 Termination 2	Определяемые вами последние символы.	Введите не более двух символов. Используйте символы завершения с инструкцией «Читать строку ASCII» (ASCII Read Line) или «Проверка Строки в Буфере» (Test Buffer for Line) (ABL) для индикации ввода строки. Символ по умолчанию соответствует параметру ASCII (RETURN), 0x0D. Вы также можете использовать символ ASCII, соответствующий LINE FEED (0x0A). Для отказа от использования характеристик введите \FF.
Append 1 Append 2	Определяемые вами дополнительные символы.	Введите не более двух символов. Используйте дополнительные символы с инструкцией «Писать ASCII с дополнением» (ASCII Write with Append) (AWA) для обозначения конца строки. Дополнительные символы являются последними символами, посылаемыми после данных строки. Символ по умолчанию соответствует параметру ASCII (RETURN) (/OD). Вы также можете использовать символ ASCII, соответствующий LINE FEED (/OA). Для отказа от использования характеристик введите \FF.

Конфигурация канала 0 для изменения метода связи

Вы можете сконфигурировать канал 0 таким образом, чтобы он мог переключаться из одного режима связи в другой при получении команды управления. Вы определяете символ `mode attention` и символ режима `user` или `system`.

Параметр:	Приказывает процессору:	Символ по умолчанию:
<code>mode attention</code>	Ожидать команду изменения режима связи	(Esc)
Режим System	Переключить режим связи в System	S
Режим User	Переключить режим связи в User	U

Каждый раз, когда процессор получает символ `mode attention` и либо символ `System` либо `User`, режим связи канала 0 будет переключен в новый режим.

Что бы сконфигурировать канал 0 для удаленного изменения режима связи, выполните действие, указанное ниже:

Если вы хотите:	Выберите:
Разрешить удаленное изменение режима связи канала 0	Разрешить (ENABLE)
Запретить удаленное изменение режима связи канала 0	Запретить (DISABLE)

Важно: Если вы не хотите допустить удаленное изменение режима связи канала 0, убедитесь в том, что опция удаленного изменения режима связи запрещена. Запрет изменения режима связи предотвращает его неожиданное переключение.

Символ `Mode Attention` говорит процессору об ожидаемом изменении режима связи. Если вы используете управляющие символы, введите символ, соответствующий символу ASCII в шестнадцатеричном виде. Для других символов просто введите нужный символ. Сделайте следующее:

Введите символ, который вы хотите использовать для того, чтобы приказать процессору изменить режим связи канала 0. Если вы используете символ управления, используйте символ ASCII в шестнадцатеричном виде. Для других символов просто введите нужный символ.

Просмотр состояния канала 0

Экраны состояния канала 0 (channel 0 status screens) показывают информацию, хранящуюся в файле диагностики, определяемом при конфигурации канала 0.

Если канал 0 в режиме:	Смотрите:
Режим System (DF1 "точка-точка")	Рисунок 11.1
Режим System (DF1 slave)	Рисунок 11.2
Режим System (DF1 master)	Рисунок 11.3
Режим User (ASCII)	Рисунок 11.4

Использование Экрана Состояния Режимы System

В этом разделе объясняются данные состояния канала, показываемые в пакете программирования для режима System:

Рисунок 11.1
Экран состояния режима System (DF1 "точка-точка")

```

System Mode (DF1 Point-to-Point)
Channel 0 Status
COUNTERS LOCKED

DCD recover:          0          Lost modem:          0
Messages sent:       0          Undelivered messages: 0
Messages received:  0
Inquiry received:   0          Inquiry sent:        0
Received NAK:       0          Bad packet/no ACK:   0
Lack of memory/sent NAK: 0          Duplicate messages received: 0

Modem lines
DTR   DCD   DSR   RTS   CTS
OFF   OFF   OFF   OFF   OFF

```

Рисунок 11.2
Экран состояния режима System (DF1 slave)

```

System Mode (DF1 Slave)
Channel 0 Status
COUNTERS LOCKED

DCD recover:          0          Lost modem:          0
Messages sent:       0          Undelivered messages: 0
Messages received:  0          Messages retry:      0
Polling received:   0
Received NAK:       0          Bad packet/no ACK:   0
Lack of memory/no ACK sent: 0          Duplicate messages received: 0

Modem lines
DTR   DCD   DSR   RTS   CTS
OFF   OFF   OFF   OFF   OFF

```

Рисунок 11.3
Экран состояния режима System (DF1 master)

```

System Mode (DF1 Master)
Channel 0 Status
DCD recover:          0          Lost modem:          0
Messages sent:        0          Undelivered messages: 0
Messages received:    0          Messages retry:      0
EOT received on first poll: 0          Bad packet/no ACK:   0
                                                Duplicate messages received: 0

Normal poll list scan [100 ms] last: 0      max: 0
Priority poll list scan [100 ms] last: 0      max: 0

Modem lines
DTR   DCD   DSR   RTS   CTS
OFF   OFF   OFF   OFF   OFF

```

Поле экрана: **Слово: Бит:** **Сконфигурируйте следующим образом:**

Counters Locked	-	Эта область показывается в режиме reverse video, если счетчики заблокированы. Нажав (F2) - Lock (Unlock) Counter , вы блокируете или разблокируете счетчики. <ul style="list-style-type: none"> Счетчики заблокированы: на экране изменений не происходит, но счетчики продолжают работать. Счетчики разблокированы: система показывает текущую величину.
DCD recover	11	Показывает время, в течение которого процессор определил, что DCD handshaking line перешла из состояния низкого в высокое.
Lost modem	12	Показывает время, в течение которого модем был отключен.
Messages sent	1	Показывает общее количество сообщений DF1, посланных процессором (включая повторные).
Messages received	2	Показывает количество сообщений, полученных процессором без ошибки.
Undelivered messages	3	Показывает количество сообщений, посланных процессором, но не принятых оборудованием-адресатом.
Messages retry	4	Для режимов slave и master показывает количество повторно посланных сообщений.
Inquiry received	6	Для режима "точка-точка" показывает количество запросов, сделанных оборудованием-адресатом.
Inquiry sent	4	Для режима "точка-точка" показывает количество запросов, сделанных процессором.
Polling received	6	Для режима slave показывает время, которое было предоставлено процессору для передачи оборудованием DF1 master.
Duplicate messages received	9	Показывает время, в течение которого процессор принимал пакет сообщений, идентичный предыдущему пакету.
Received NAK	5	Для режимов "точка-точка" и slave показывает число сообщений с NAK, принятых процессором.
EOT End of Transmission	8	Показывает время, в течение которого master принимал EOT в ответ на первую посылку данных станцией.
Bad packet/no ACK	7	Показывает количество пакетов с некорректными данными, принятых процессором.
Lack of memory/sent NAK	8	Для режимов "точка-точка" и slave показывает время, в течение которого процессор не мог принять сообщение из-за недостатка памяти.
Lack of memory/no ACK sent		

Поле экрана:	Слово: Бит:	Сконфигурируйте следующим образом:
Normal Poll List scan last (100 ms)	5	Время, затраченное на завершение предыдущего цикла нормального списка передачи.
Normal Poll List scan maximum (100 ms)	6	Наибольшее время, затраченное на завершение цикла нормального списка передачи.
Priority Poll List scan last (100 ms)	10	Время, затраченное на завершение предыдущего цикла приоритетного списка передачи.
Priority Poll List scan maximum (100 ms)	13	Наибольшее время, затраченное на завершение цикла приоритетного списка передачи.
DTR Data Terminal Read	0: 4	Показывает состояние DTR handshaking line (подтверждаемый процессором).
DCD Carrier Detect	0: 3	Показывает состояние DCD handshaking line (принятый процессором).
DSR Data Set Ready	0: 2	Показывает состояние DSR handshaking line (принятый процессором).
RTS Request to Send	0: 1	Показывает состояние RTS handshaking line (подтверждаемый процессором).
CTS Clear to Send	0: 0	Показывает состояние CTS handshaking line (принятый процессором).

Использование экрана состояния режима User (ASCII)

В этом разделе объясняются данные состояния канала, показываемые в пакете программирования для режима User (ASCII):

Рисунок 11.4
Экран состояния режима User

```

User Mode
Channel 0 Status
COUNTERS LOCKED

DCD recover:                0                Lost modem:                0

Chars. received with error: 0

Modem lines
DTR   DCD   DSR   RTS   CTS
OFF   OFF   OFF   OFF   OFF

```

Поле экрана:	Слово: Бит:	Сконфигурируйте следующим образом:
Counters Locked		Эта область показывается , если счетчики заблокированы. <ul style="list-style-type: none">Счетчики заблокированы: на экране изменений не происходит, но счетчики продолжают работать.Счетчики разблокированы: система показывает текущую величину.
DCD recover	11	Показывает время, в течение которого процессор определил, что DCD-handshaking line перешла из нуля в единицу.
Lost modem	12	Показывает время, в течение которого модем был отключен.
DTR Data Terminal Read	0: 4	Показывает состояние DTR handshaking line (подтверждаемый процессором).
DCD Carrier Detect	0: 3	Показывает состояние DCD handshaking line line (принятый процессором).
DSR Data Set Ready	0: 2	Показывает состояние DSR handshaking line (принятый процессором).
RTS Request to Send	0: 1	Показывает состояние RTS handshaking line (подтверждаемый процессором).
CTS Clear to Send	0: 0	Показывает состояние CTS handshaking line (принятый процессором).
Character received with error	10	Показывает количество отвергнутых символов, принятых процессором с паритетом или с ошибкой.

Для заметок

Связь с устройствами в сети Ethernet

Назначение этой главы

Сведения	Страница
Средства информации и носители	12-1
Установка и назначение IP адреса	12-2
Адресация сети	12-2
Конфигурация канала 2 для сети Ethernet	12-3
Использование DOS/Windows BOOTP	12-6
Использование расширенных функций Ethernet	12-11
Расшифровка кодов ошибок	12-16
Расшифровка данных состояния сети Ethernet	12-17
Характеристики Ethernet PLC-5	12-18

Средства информации и носители

Ethernet - это локальная сеть со скоростью связи 10М бит/с. Физические средства связи могут быть любые стандарта 802.3, включая:

- толстый коаксиальный кабель (10Base5);
- тонкий коаксиальный кабель (10Base2);
- витая пара (10Base-T);
- оптоволокно;
- радиочастотный сигнал.

Ethernet порт (канал 2) подключается к сети тонкий-коаксиал, толстый-коаксиал, витая пара через 15 контактный трансивер или через устройство доступа к носителю (Medium Access Unit – MAU). Для более детальной информации по подключению Ethernet см. раздел 25.

Установка и назначение IP адреса



Способ назначения **IP адреса** зависит от того, насколько вы знакомы с адресацией в сети Internet.

Если вы не знакомы с адресацией Internet, см. Comer, Douglas E; *Internetworking with TCP-IP, Volume 1: Protocols and Architecture*; Englewood Cliffs, N.J.: Prentice-Hall, 1990. Используйте уникальные адреса протокола TCP/IP.

Если вы знакомы с адресацией Internet, обратитесь к администратору вашей сети или Network Information Center для назначения адреса IP для вашего процессора PLC-5/20E, -5/40E или PLC-5/80E.

Адресация сети

Так как Ethernet PLC-5 процессоры используют протокол TCP/IP, каждому процессору в сети требуется уникальный адрес IP. Адрес IP устанавливается программно с использованием любого программного обеспечения использующего протокол BOOTP.

Если вы используете BOOTP протокол, Вы должны также использовать аппаратные средства адресации Ethernet. Allen-Bradley назначает каждому Ethernet PLC-5 процессору адрес, устанавливаемый на заводе. Смотрите адрес:

- на задней части, верхний угол модуля; или
- на экране конфигурации канала 2 программного обеспечения.

Конфигурирование канала 2 для сети Ethernet

После назначения адреса IP вашему PLC-5E процессору, Вы должны сконфигурировать канал 2 так чтобы процессор был опознан сетью. Конфигурацию канала можно выполнить одним из двух методов:

- используя программное обеспечение через DH+ или последовательный порт; или
- динамическая конфигурация с использованием утилит BOOTP.

Если Вы:	То:
имеете BOOTP сервер	используйте этот сервер; или используйте ручную конфигурацию с помощью программного обеспечения
не имеете BOOTP сервер в вашей сети	динамическая конфигурация с использованием утилиты BOOTP на диске который идет с процессором; или используйте ручную конфигурацию с помощью программного обеспечения

Ручная конфигурация канала 2

Вы можете вручную сконфигурировать канал 2 для Ethernet сети с использованием программного обеспечения через DH+ или последовательный порт.

Ethernet Channel 2 Configuration	
	Diagnostics file: N10
	Ethernet address: 00:00:BC:1C:00:00
	IP address: 130.151.133.222
BOOTP поле разрешения →	BOOTP enable: NO
	Message connect timeout (msec): 15000
	Message reply timeout (msec): 3000
	Inactivity timeout (minutes):
	Advanced Functions
	Broadcast address: 255.255.255.255
	Subnet mask: Default
Если используете сегменты и шлюзы См. страницу 12-12 →	Gateway address: 130.151.133.1

Введите IP адрес и установите в поле BOOTP enable – NO. Введите информацию в остальные поля конфигурации. См. таблицу 12.A на следующей странице.

Вы не сможете изменить Ethernet конфигурацию если связаны с процессором через Ethernet.

Важно: По умолчанию BOOTP enable (разрешен). Вы не сможете изменить IP адрес пока BOOTP разрешен (enable).

Таблица 12.А
Ethernet Канал 2, поля конфигурации

Наименование поля	Назначение	Для конфигурации выполнить следующее
Diagnostics file	Файл содержит статусную информацию о канале	<p>Введите целочисленный файл с номером (10-999). Система создаст целочисленный файл длиной 44 слова.</p> <p>ВНИМАНИЕ: Создайте свой отдельный диагностический файл для каждого канала. Не используйте в качестве файла диагностики файлы, которые используются в программе. Иначе вы получите непредсказуемые результаты.</p> <p>Важно: Вы должны в любом случае (даже если не используете канал) назначить файл диагностики для конфигурации канала, если вы хотите иметь статусную информацию для этого канала.</p>
Ethernet Address	Аппаратный адрес Ethernet процессора Только отображение	<p>Установлено Allen-Bradley и не может быть изменено. Отображение как 6 шестнадцатеричных слов, разделенных двоеточием.</p>
IP Address	Адрес Internet процессора	<p>Установить BOOTP – disable (запрещено). Вы не сможете изменить IP адрес из программного обеспечения 6200 если BOOTP – enabled (разрешен).</p> <p>Введите адрес в виде:</p> <p>a.b.c.d. Где: a,b,c,d десятичное число от 1 до 254.</p> <p>Вы должны установить IP адрес если хотите подключить процессор к сети TCP/IP. Не используйте для адресации IP числа 0 и 255.</p>
BOOTP Enable	Разрешение/запрет BOOTP	<p>Выбрано NO.</p> <p>Перед тем как ввести NO, убедитесь, что набран адрес IP. При установке BOOTP – NO, процессор использует введенные вами параметры.</p> <p>Если BOOTP – разрешен (enable), смотри страницу 12-5.</p>
MSG Conn Timeout	Количество миллисекунд отводимых для установления связи с устройствами с использованием MSG инструкции.	<p>Введите время тайм-аута в миллисекундах. (Среднее время цикла составляет 250мс.) Допустимый диапазон превышения времени связи 0-65535 мс.</p> <p>Значение по умолчанию 15000 мсек.</p>
MSG Reply Timeout	Количество миллисекунд которые Ethernet ожидает ответа. (для команды MSG)	<p>Введите время тайм-аута в миллисекундах. (Среднее время цикла составляет 250 мс.). Допустимый диапазон превышения времени связи 0-65535 мс.</p> <p>Значение по умолчанию 3000мсек.</p>
Inactivity Timeout	Время в мин для перевода в неактивное состояние после потери связи	<p>Введите время тайм-аута в минутах. Допустимый диапазон времени связи 0-65535 минут.</p> <p>Значение по умолчанию 30 минут.</p>
Advanced Functions	(дополнительные функции)	
Broadcast Address	Широковещательный адрес на который процессор должен ответить	<p>Информацию о дополнительных функциях сети см. на странице 12-11, включая использования широковещательной адресации.</p> <p>Эта функция не позволяет послать одно сообщение нескольким процессорам PLC-5E.</p>

Наименование поля	Назначение	Для конфигурации выполнить следующее
Subnet Mask	Маска процессора для сегмента сети. Маска используется когда сеть разделена на сегменты.	См. страницу 12-12 по использованию сегментов и шлюзов.
Gateway Address	IP адрес шлюза который подключает сегмент IP сети.	См. страницу 12-12 по использованию сегментов и шлюзов.

Динамическая конфигурация Канала 2 с помощью BOOTP

Вы можете также использовать BOOTP для получения маски сегмента и адреса шлюза.
Смотри стр. 12-12.

BOOTP – это протокол, который обеспечивает процессор информацией конфигурации при включении питания. BOOTP позволяет Вам динамически распределять IP адреса в процессорах по сети Ethernet.

При использовании BOOTP, в локальной сети Ethernet должен находиться BOOTP сервер. Сервер – это компьютер (персональный компьютер с операционной системой VAX или UNIX) с установленным программным обеспечением BOOTP-server и читающий с устройств в сети текстовые файлы содержащие информацию о сети.

Если в Вашей сети нет загрузочного сервера, а Вы хотите динамически конфигурировать канал 2, воспользуйтесь утилитой BOOTP на дискете, которая поставляется вместе с процессором PLC-5E.

Для разрешения утилиты BOOTP, воспользуйтесь окном конфигурации канала 2 программного обеспечения. Установите в поле BOOTP – YES, для разрешения BOOTP.

Важно: Если вы измените это поле с No на Yes, изменение не вступит в силу до переключения питания.

```

                                Ethernet
                                Channel 2 Configuration

Diagnostics file:                N10
Ethernet address:                00:00:BC:1C:00:00
IP address:                      130.151.133.222
BOOTP enable:                   NO
Message connect timeout (msec):  15000
Message reply timeout (msec):    3000
Inactivity timeout (minutes):

Advanced Functions
Broadcast address:               255.255.255.255
Subnet mask:                     Default
Gateway address:                 130.151.133.1

```

поле разрешения BOOTP →

См. страницу 12-12 если используете сегменты и шлюзы →

Укажите дальнейшую конфигурационную информацию используя этот экран. Руководствуйтесь табл. 12.A на странице 12-4 для описания других полей.

Когда BOOTP разрешен, после включения питания происходит следующее:

- Процессор передает по широковещательному адресу BOOTP сообщение, содержащее его аппаратный адрес для локальной сети или подсети.
- BOOTP сервер сравнивает аппаратный адрес с адресом в файле своей таблицы BOOTPTAB.
- BOOTP сервер посылает обратно процессору сообщение, которое заменяет IP адрес и другую сетевую информацию в процессоре.

Вы можете легко из одного места поменять адреса IP с помощью конфигурационного файла BOOTP.

Если у Вас BOOTP разрешен, и поступило сообщение “BOOTP response not received”, проверьте подключение кабеля и BOOTP сервер.

Важно: Если BOOTP запрещен, или в сети не установлен BOOTP сервер, Вы должны использовать программное обеспечение PLC-5 для назначения IP адреса для каждого процессора.

Allen-Bradley обеспечивает поддержку BOOTP в среде DOS и Windows. Версии DOS/Windows BOOTP утилит поставляются на дискете с процессором PLC-5E.

Использование DOS/Windows BOOTP

Диск который Вы получили с вашим процессором содержит DOS и Windows утилиты для BOOTP сервера. Обе утилиты поддерживают BOOTP Ethernet для процессоров PLC-5. Независимо от установленной у Вас операционной системы, Вы должны:

- установить утилиту boot-server
- отредактировать конфигурационный файл boot-server
- запустить boot-server утилиту

Важно: Не используйте BOOTP утилиту, если у вас уже установлен INTERCHANGE. Вместо этого используйте boot-server свойства, которые пришли с программным обеспечением INTERCHANGE.

Инсталляция DOS/Windows BOOTP сервера

Инсталляция DOS BOOTP server:

1. Установите в дисковод дискету с утилитами, которая пришла с вашим процессором.
2. Сделайте дисковод "а:" диском по умолчанию.
3. Введите **install**, нажмите **[Enter]**.
4. Программное обеспечение установится на диске C:\ABIC\BIN. Укажите эту директорию в пути вашего файла AUTOEXEC.BAT.

Редактирование файла конфигурации DOS/Windows BOOTP

Файл конфигурации boot-server, BOOTPTAB, находится в директории C:\ABIC\BIN. Этот файл содержит информацию, необходимую для загрузки Ethernet процессоров PLC-5.

Вы должны отредактировать файл BOOTPTAB, представляющий собой ASCII текст, содержащий имя, IP (Internet Protocol) адрес и аппаратный адрес для каждого PLC-5 процессора. Отредактируйте этот файл:

1. Откройте текстовым редактором файл BOOTPTAB.

- Файл состоит из строк, которые выглядят как:

```
#Default string for each type of Ethernet client
defaults5E: ht=1 :vm=rfc1048
```

- Это параметры по умолчанию для Ethernet процессора PLC-5, эти строки должны всегда предшествовать строке пользователя в файле BOOTPTAB.

- Файл также содержит строку, которая выглядит как:

```
plc5name: tc-defaults5E:ip=aa.bb.cc.dd:ha=000BC1ххуу
```

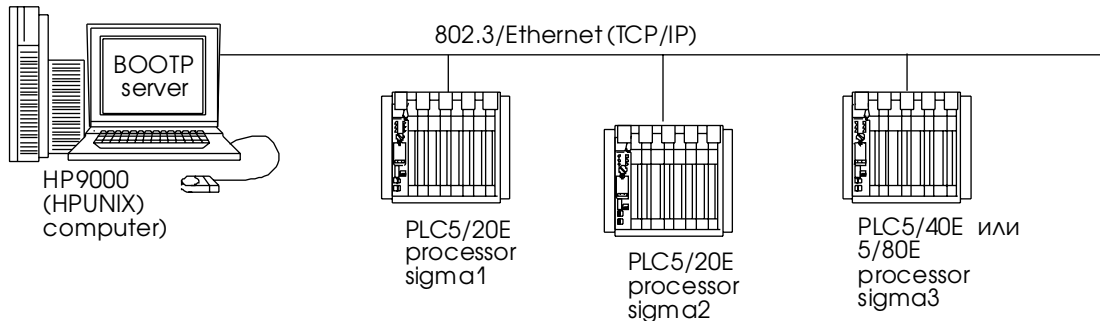
Важно: Используйте эту строку как шаблон для процессоров Ethernet PLC-5.

2. Сделайте копию шаблона для каждого процессора Ethernet PLC-5 вашей системы.
3. Отредактируйте каждую копию шаблона следующим образом:
 - A. Замените имя `plc5name` каждому процессору PLC-5 Ethernet. Используйте только латинские буквы и цифры; не используйте спецсимволы.
 - B. Замените `aa.dd.cc.dd` на адрес IP назначенный процессору.
 - C. Замените `xxuu` на последние четыре цифры аппаратного адреса. Используйте только шестнадцатичные числа (0-9, A-F); не вводите дефисы раздела чисел. (Аппаратный адрес напечатан на боковой стенке процессора PLC-5 Ethernet.)
4. Сохраните изменения в файле, закройте файл и сделайте резервную копию этого файла.

Пример:

В этом примере показаны три процессора Ethernet PLC-5 и программируемый терминал HP 9000. Имена и адреса устройств приведены в списке:

Устройство	Имя	IP адрес	Аппаратный адрес
Ethernet PLC-5	sigma1	12.34.56.1	00-00-BC-1C-12-34
Ethernet PLC-5	Sigma2	12.34.56.2	00-00-BC-1C-56-78
Ethernet PLC-5	Sigma3	12.34.56.3	00-00-BC-1C-90-12



На основании данной конфигурации, файл BOOTPTAB будет выглядеть как:

```
#      Legend:      gw-gateways
#                      Ha-hardware address
#                      Ht-hardware type 1
#                      Ip-host IP address
#                      sm-subnet mask
#                      vm-BOOTP vendor extensions format 2
#                      Tc-template host

#Default string for Ethernet client
Defaults5E:  ht=1:vm=rfc1048

#Entries for Ethernet PLC-5 processors:
sigma1:tc=defaults5E:ip=12.34.56.1:ha=0000BC1C1234
sigma2:tc=defaults5E:ip=12.34.56.2:ha=0000BC1C5678
sigma3:tc=defaults5E:ip=12.34.56.3:ha=0000BC1C9012
```

¹ 1 = 10MB Ethernet

² Use rfc1048

Запуск утилиты Boot Server

Вы можете запустить на выполнение BOOT утилиту для операционной среды DOS или Windows, но не обе одновременно.

Если ваша операционная система	Исполняемый файл	Запустить из	См. стр.
DOS	DTLBOOTD.EXE	Из командной строки DOS (при необходимости с дополнительными параметрами)	12-10
Windows	DTLBOOTW.EXE	Windows Program Manager	12-10

Обе утилиты расположены в директории C:\ABIC\BIN и используют данные содержащиеся в файле BOOTPTAB.

Совет

Убедитесь, что файл BOOTPTAB находится в директории, из которой Вы запускаете утилиту BOOTP. Если файл не будет найден в этой директории, утилита будет искать файл в директориях указанных в системной переменной ABIC_CONFIG.

Запуск DOS утилиты

Для запуска DOS утилиты, DTLBOOTD.EXE, выполните следующее:

1. В командной строке DOS введите:

```
DTLBOOTD [-D] [-T <timeout>] [-B <numboots>] [-F
<numfiles>] [ configfile] [ logfile]
```

Параметр	Назначение
-D	дополнительная информация для отладки
-T <timeout>	выход через <timeout> секунд в неактивное состояние
-B <numboots>	выход после ответа <numboots> количеств boot запросов
-F <numfiles>	Выход после ответа <numfiles> количеств file запросов
configfile	наименование файла конфигурации boot сервера. По умолчанию файл конфигурации %ABIC_CONFIG%\BOOTPTAB.
logfile	Имя используемого log файла. По умолчанию файл %ABIC_CONFIG%\DTLBOOTD.LOG.

Для выхода, нажмите [Ctrl-C] или [Esc].

После запуска утилиты, она будет выполняться пока не выполнится один из указанных параметров выхода.

2. Подать питание на все шасси с процессорами Ethernet PLC-5.

При включении питания каждый Ethernet PLC-5 процессор посылает широковещательный запрос, если в экране конфигурации канала 2 BOOTP- разрешен. Ethernet boot сервер сравнивает аппаратный адрес с адресом в таблице файла BOOTPTAB и посылает установленный IP адрес и другие конфигурационные данные, которые переписываются через BOOTP.

Запуск Windows утилиты

Для запуска Windows утилиты boot сервера, DTLBOOTW.EXE, выполните следующее:

1. Запустите Microsoft Windows 3.x, если он не запущен.
2. Откройте Program Manager, если он не открыт.
3. Выберите в меню **File** и выбрать **Run** в меню.

4. В диалоговом окне выберите, C:\ABIC\BIN\DTLBOOTW, затем выберите **ОК** или введите [Enter].

После запуска утилиты, она будет выполняться пока Вы не закроете DTLBOOTW.EXE или не выйдете из Windows.

5. Подать питание на все шасси с процессорами Ethernet PLC-5

При включении питания каждый Ethernet PLC-5 процессор посылает широковещательный запрос, если в экране конфигурации канала 2 BOOTP- разрешен. Ethernet boot сервер сравнивает аппаратный адрес с адресом в таблице файла BOOTPTAB и посылает установленный IP адрес и другие конфигурационные данные, которые переписываются через BOOTP.

Использование дополнительных функций Ethernet

Используя окно конфигурации Ethernet канала 2, сконфигурируйте следующие расширенные функции сети:

- широковещательный адрес;
- маска сегмента;
- адрес шлюза.

Если вы используете	См стр.
Широковещательный адрес	12-11
Маску сегмента и шлюз	12-12

Использование широковещательной адресации

Широковещательный адрес это часть IP протокола, используемая для посылки сообщений хостом по каждому IP адресу на шине. Это поле в экране конфигурации канала 2 определяет адрес, по которому модуль будет получать широковещательные сообщения посланные хостом.

Важно: Широковещательный адрес используется только для приема сообщений. Когда используется контекстовая адресация Ethernet, широковещательная функция не относится к сообщениям в программе релейно-контактной логики.

Таким образом, эта функция не позволяет послать одно сообщение на несколько процессоров PLC-5 одновременно.

В большинстве случаев Вы можете оставить установку широковещательного адреса по умолчанию.

```

Ethernet
Channel 2 Configuration

Diagnostics file:          N10
Ethernet address:         00:00:BC:1C:00:00
IP address:               130.151.133.222
BOOTP enable:            YES
Message connect timeout (msec): 15000
Message reply timeout (msec): 3000
Inactivity timeout (minutes):

Advanced Functions
Broadcast address:       Default
Subnet mask:            Default
Gateway address:        130.151.133.1

```

Поле конфигурации	Выполнить следующее
-------------------	---------------------

Broadcast Address	<p>Установите курсор на поле и введите адрес в формате: a.b.c.d. Где: a,b,c,d десятичные числа 0-255.</p> <p>Перед заменой адреса по умолчанию, его необходимо обнулить, т.е. вписать 0.0.0.0.</p>
-------------------	--

Использование маски сегмента и шлюзов

Если ваша сеть разделена на подсети (сегменты), которые используют шлюзы или маршрутизаторы, Вы должны указать следующую информацию при конфигурировании канала 2:

- маска сегмента;
- адрес шлюза.

Маска сегмента, это фильтр который используется для определения нахождения адреса IP в локальной сети или в сегменте. Если адрес расположен на другом сегменте, сообщения передаются через шлюз или маршрутизатор.



Для более подробной информации по использованию маски сегмента и шлюзов, смотри Comer, Douglas E; *Internetworking with TCP-IP, Volume 1: Protocols and Architecture*; Englewood Cliffs, N.J.: Prentice-Hall, 1990.

Если ваша сеть не разделена на сегменты, Вы можете установить поле маски сегмента по умолчанию.

Если Вы	Тогда	См. стр
Вручную конфигурируете канал 2 и сеть разбита на сегменты	<ul style="list-style-type: none"> • убедитесь, что в поле BOOTP enable стоит NO • Используйте программное обеспечение 6200 для маски сегмента и установки адреса маски шлюза; см Таблицу 12.B. 	12-13
Используете BOOTP для конфигурации канала 2 и сеть разбита на сегменты	<ul style="list-style-type: none"> • убедитесь, что в поле BOOTP enable стоит YES • сконфигурируйте файл BOOTPTAB, включая маски сегментов и адреса шлюзов. 	12-13

Ручная конфигурация канала 2 для процессора сегмента

Если вы вручную конфигурируете канал 2 для процессора, расположенного на сегменте сети, установите для каждого процессора маску сегмента и адрес шлюза с помощью программного обеспечения б200 согласно таблице 12.В.

Ethernet
Channel 2 Configuration

```

Diagnostics file:          N10
Ethernet address:         00:00:BC:1C:00:00
IP address:               130.151.133.222
BOOTP enable:            NO
Message connect timeout (msec): 15000
Message reply timeout (msec): 3000
Inactivity timeout (minutes):

Advanced Functions
Broadcast address:       255.255.255.255
Subnet mask:            255.255.255.0
Gateway address:        130.151.194.1
  
```

поле разрешения
BOOTP →

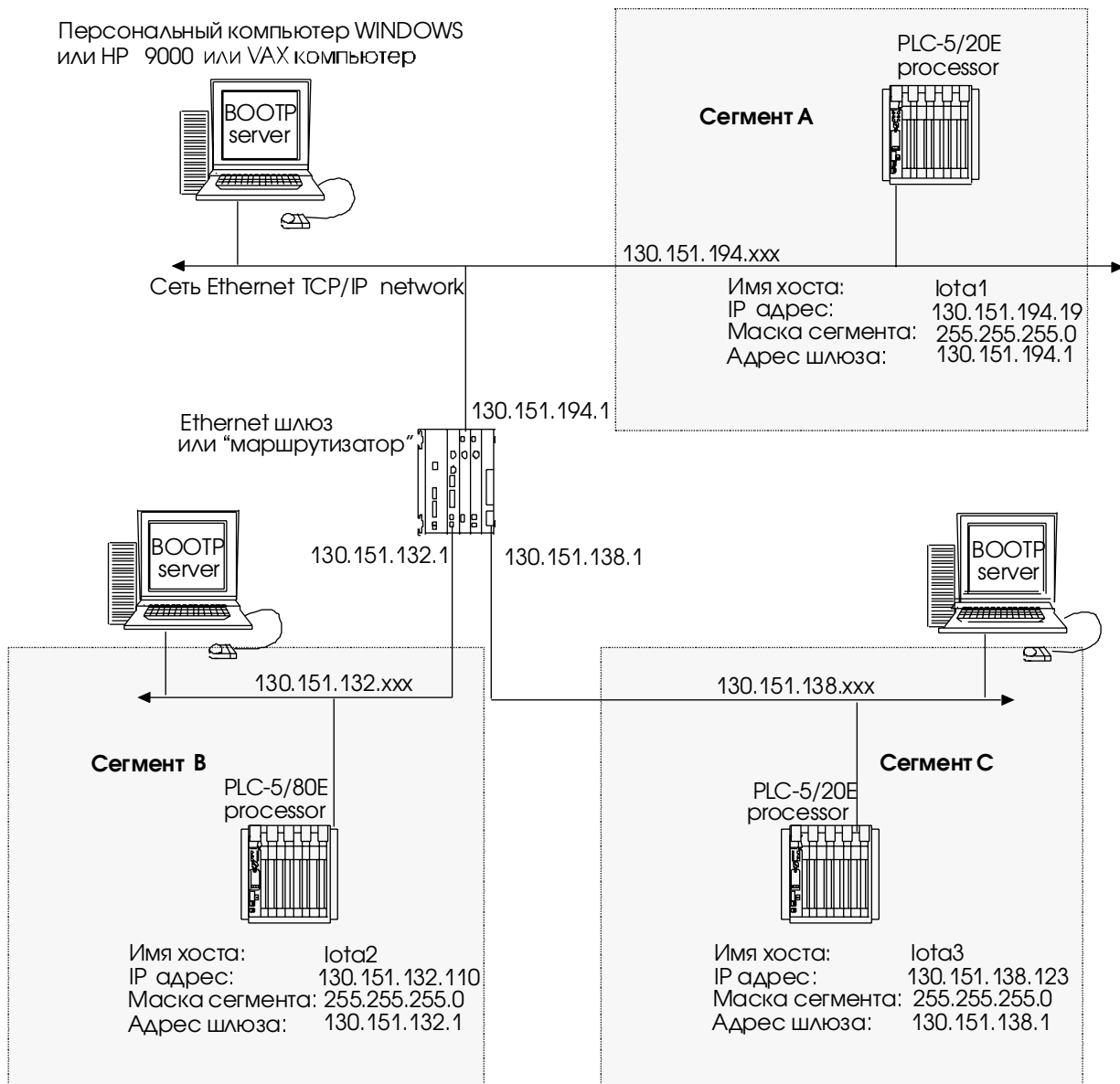
Таблица 12.В
Ethernet Канал 2 Конфигурация расширенных функций

Поле	Описание	Для конфигурации выполнить следующее
Subnet Mask (маска сегмента)	Маска сегмента для процессора. Маска сегмента используется для передачи IP адреса когда сеть разбита на сегменты	<p>ввести адрес в формате: a.b.c.d. Где: a,b,c,d десятичные числа 0-255</p> <p>Если ваша сеть не разбита на сегменты, маску сегмента можно установить по умолчанию. Перед изменением значения по умолчанию маску необходимо обнулить, т.е. ввести 0.0.0.0.</p>
Gateway Address (адрес шлюза)	IP адрес шлюза, который обеспечивает связь с другой IP сетью. Это поле используется, когда вы связываетесь с устройствами не в локальной сети.	<p>ввести адрес в формате: a.b.c.d. Где: a,b,c,d десятичные числа 0-255</p> <p>Адрес по умолчанию – NO Gateway</p>

Использование BOOTP для конфигурации канала 2 процессора сегмента

Сконфигурируйте файл BOOTPTAB, включая маски сегментов и адреса шлюзов, для каждого процессора PLC-5E. На следующей странице приведен пример конфигурации файла BOOTPTAB.

Важно: Так как запросы BOOTP возможны только в локальной сети, поэтому на каждом сегменте должен быть свой BOOTP сервер и BOOTPTAB файл.



Файл BOOTPTAB представленный в этом примере будет выглядеть так:

```
# Legend: gw -- gateways
#          ha -- hardware address
#          ht -- hardware type
#          ip -- host IP address
#          sm -- subnet mask
#          vm -- BOOTP vendor extensions format
#          tc -- template host
#Default string for each type of Ethernet client
defaults5E: ht=1:vm=rfc1048:sm=255.255.255.0
#Entries for Ethernet PLC-5 processors:
lota1:\
    tc=defaults5E:\
    gw=130.151.194.1:\
    ha=0000BC1C1234:/
    ip=130.151.194.19
```

```
# Legend: gw -- gateways
#          ha -- hardware address
#          ht -- hardware type
#          ip -- host IP address
#          sm -- subnet mask
#          vm -- BOOTP vendor extensions format
#          tc -- template host
#Default string for each type of Ethernet client
defaults5E: ht=1:vm=rfc1048:sm=255.255.255.0
#Entries for Ethernet PLC-5 processors:
lota2:\
    tc=defaults5E:\
    gw=130.151.132.1:\
    ha=0000BC1C5678:/
    ip=130.151.132.110
```

```
# Legend: gw -- gateways
#          ha -- hardware address
#          ht -- hardware type
#          ip -- host IP address
#          sm -- subnet mask
#          vm -- BOOTP vendor extensions format
#          tc -- template host
#Default string for each type of Ethernet client
defaults5E: ht=1:vm=rfc1048:sm=255.255.255.0
#Entries for Ethernet PLC-5 processors:
lota3:\
    tc=defaults5E:\
    gw=130.151.138.1:\
    ha=0000BC1C9012:/
    ip=130.151.138.123
```

Расшифровка кодов ошибок

Если процессор обнаруживает ошибку при передаче сообщений он выставляет бит .ER и записывает код ошибки, который Вы можете просмотреть с помощью программного обеспечения.

Код ошибки (шестнадцатеричное слово1 блока управления)	Описание ошибки (Выдается в экране монитора данных)
0010	No IP address configured for the network (не сконфигурирован IP адрес для сети)
0011	Already at maximum number of connections (максимальное число связей)
0012	Invalid internet address or host name (ошибка адреса сети или имени хоста)
0013	No such host (нет такого хоста)
0014	Cannot communicate with the name server (нет связи с названным сервером)
0015	Connection not completed before user_specified timeout (связь не окончена до наступления тайм-аута установленного пользователем)
0016	Connection timed out by the network (сетевой тайм-аут)
0017	Connection refused by destination host (связь отвергнута хостом назначения)
0018	Connection was broken (связь прервана)
0019	Reply not received before user_specified timeout (ответ не получен до наступления тайм-аута установленного пользователем)
001A	No network buffer space available (нет места в буфере сети)
0037	Message timed out in local processor (таймаут сообщения в локальном процессоре)
0083	Processor is disconnected (процессор отключен)
0089	Processor's message buffer is full (буфер сообщений полон)
0092	No response (regardless of station type) (нет ответа от станции не зависимо от типа станции)
00D3	You formatted the control block incorrectly (некорректно создан контрольный блок)
00D5	Incorrect address for the local data table (неправильный адрес для таблицы локальных данных)
1000	Illegal command from local processor (недопустимая команда локального процессора)
2000	Communication module not working (коммуникационный модуль не работает)
4000	Processor connected but faulted (hardware) (ошибка связи, аппаратная, процессор подключен но в режиме ошибка)
5000	You used the wrong station number (использование ошибочного номера станции)
6000	Requested function is not available (требуемая функция недоступна)
7000	Processor is in program mode (процессор в режиме программирования)
8000	Processor's compatibility file does not exist (нет файла совместимости процессора)
B000	Processor is downloading so it is inaccessible (процессор в режиме загрузки поэтому недоступен)
F001	Processor incorrectly converted the address (Процессор неправильно преобразовал адрес)
F002	Incomplete address (неполный адрес)
F003	Incorrect address (неправильный адрес)
F006	Addressed file does not exist in target processor (обращение к несуществующему файлу процессора назначения)
F007	Destination file is too small for number of words requested (размер файла назначения меньше количества запрошенных слов посылки)
F00A	Target processor cannot put requested information in packets (Процессор назначения не может поместить требуемую информацию в пакетах)
F00B	Privilege error, access denied (ошибка привилегии, доступ отвергнут)
F00C	Requested function is not available (требуемая функция недоступна)
F00D	Request is redundant (запрос изменен)
F011	Data type requested does not match data available (запрошенный тип данных не соответствует доступным данным)
F012	Incorrect command parameters (неправильные параметры команды)

Расшифровка данных состояния сети Ethernet

Экран статуса Ethernet PLC-5 процессора вызывается из экрана конфигурации канала 2.

```

Ethernet
Channel 2 Status

Commands
  sent:
  received:
Replies
  sent:
  sent with error:
  timed out:
  received
  received with error:
Ethernet
  In Octets:
  In Packets:
  alignment errors:
  carrier sense errors
  excessive deferrals
  MAC transmit errors
  multiple collisions
  late collisions
  Out Octets:
  Out Packets:
  FCS errors:
  excessive collisions:
  MAC receive errors:
  single collisions
  deferred transmission:

```

Диагностические данные находятся в диагностическом файле, указанном в экране конфигурации Канала 2.

Поле статуса	Байты	Показывает количество
Commands	посылка	0-3
	прием	4-7
Replies	посылка	8-11
	прием	12-15
	посылка с ошибкой	16-19
	прием с ошибкой	20-23
	тайм-аут	24-27

Поле статуса	Байты	Показывает количество	
Ethernet	In Octets	28-31	Октетов, полученных на канале
	Out Octets	32-35	Октетов, посланных каналом
	In Packets	36-39	Пакетов, полученных каналом, включая широковещательные пакеты
	Out Packets	40-43	Пакетов, посланных каналом, включая широковещательные пакеты
	alignment errors	44-47	Фреймов, полученных каналом, размер которых не равен суммарному числу октетов по длине
	FCS errors	48-51	Фреймов, полученных каналом, которые не прошли проверку FCS
	carrier sense errors	52-55	Время в течении которого несущая отсутствовала или вообще не появлялась при попытке передачи фрейма
	excessive collisions	56-59	Фреймов, для которых передача потерпела неудачу из-за чрезмерных коллизий
	excessive deferrals	60-63	Фреймов, для которых передача отсрочена на слишком большой период времени
	MAC receive errors	64-67	Фреймов, для которых прием на интерфейсе терпит неудачу из-за внутренней ошибки принимающего MAC подслюя
	MAC transmit errors	68-71	Фреймов, для которых прием на интерфейсе терпит неудачу из-за внутренней ошибки передающего MAC подслюя
	single collisions	72-75	Успешно переданных фреймов, для которых передача была задержана из-за коллизий
	multiple collisions	76-79	Успешно переданных фреймов, для которых передача была задержана больше чем один раз из-за коллизий
	deferred	80-83	Фреймов, для которых первая попытка передачи была задержана потому что носитель занят
late collisions	84-87	Количество раз когда коллизия замечена позднее чем 512 бит пакета было уже передано	

Характеристики Ethernet PLC-5

Характеристики Ethernet процессоров PLC-5 зависят от:

- размера Ethernet сообщений;
- частоты Ethernet сообщений;
- загрузки сети;
- выполнения вашей прикладной программы в процессоре.

Таблица 12.С

Оптимальная работа : Связь в Ethernet процессоров PLC-5 – PLC-5 (два устройства в сети)

Операция	Слов	MSG в с	мс на MSG	Слов в с
Одна запись	1	32	31.3	32
	20	32	31.7	632
	100	31	32.7	3,058
	1000	14	73.4	13,625

Защита ваших программ

Использование этой главы

Сведения	Страница
Относительно паролей и привилегий	13-1
Определение класса привилегии	13-3
Назначение класса привилегии на канал или файл офлайн	13-4
Назначение класса привилегии к узлу	13-4
Назначение привилегии на чтение/запись программного файла	13-5
Назначение привилегии на чтение/запись файла данных	13-5
Использование защиты процессора	13-5

Читайте эту главу для краткого обзора:

- определение классов привилегий;
- назначение класса привилегии на канал или файл офлайн;
- назначение класса привилегии к узлу;
- назначение привилегии на чтение/запись программного файла;
- назначение привилегии на чтение/запись файла данных.



Внимание: При установке программного обеспечения выберите опцию полной установки паролей и привилегий.

Для более подробной информации относительно конфигурации привилегий, см. документацию по программному обеспечению.

Если ваше приложение требует дополнительных привилегий по сравнению с привилегиями для расширенных и Ethernet PLC-5 процессоров, см. публикацию 1785-2.28. “PLC-5 Protected Processor Product Data for 1785-5/26, -5/46, and -5/86 processors”.

Относительно паролей и привилегий

Функции паролей и привилегий для усовершенствованных и Ethernet PLC-5 процессоров помогут Вам защитить ваши программы и ограничить доступ к файлам и функциям процессора.

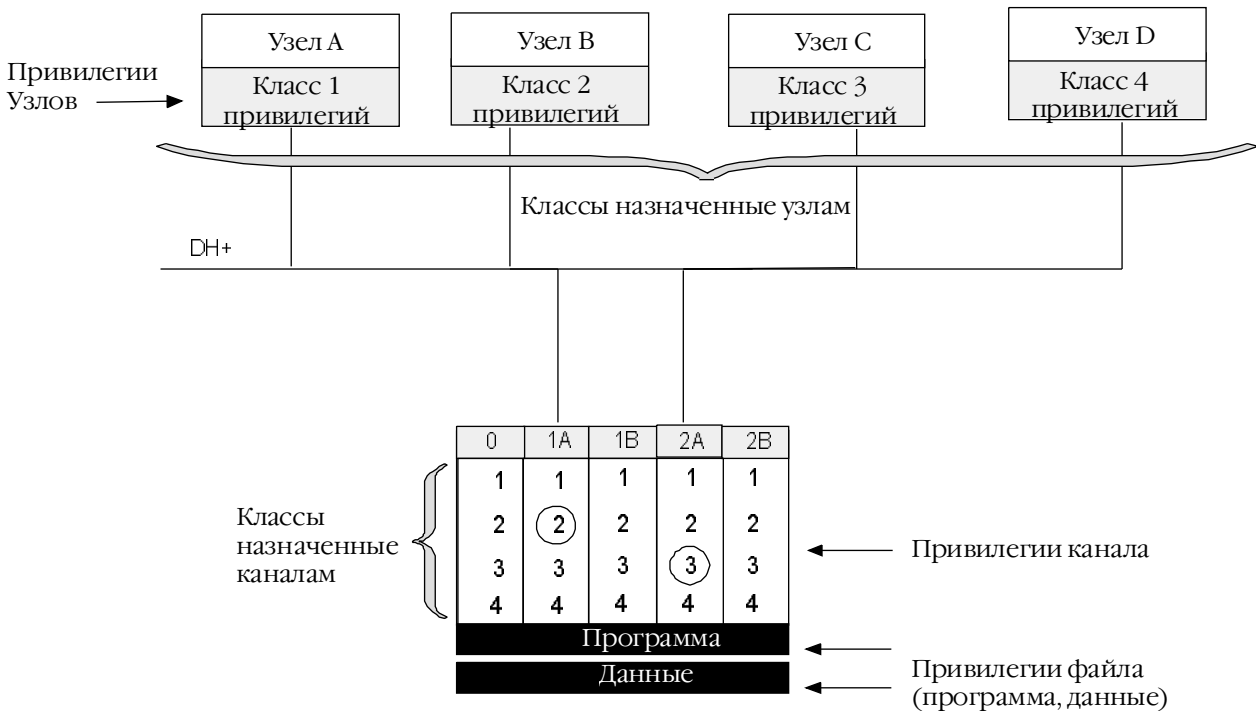
Вы можете назначить **класс привилегии** на узел, канал или файл. Класс привилегии определяет уровень доступа (чтение или запись) или тип функций (форсирование входов/выходов, очистка памяти) для всех процессоров PLC-5.

Привилегия	Ограничивает доступ
Узел	к отдельному узлу процессора
Канал	к отдельному каналу процессора
Файл	к просмотру или изменению файла

Важно: Класс привилегий узла подавляет класс привилегий канала.

Рис. 13.1

Привилегии поддерживаемые расширенными и Ethernet PLC-5 процессорами.



На рис. 13.1, Вы можете видеть, что классы привилегий назначенные каждому узлу управляют доступом к устройствам процессора. Например:

- узел В имеет Класс 2 доступа на канал 1А, основанный на привилегии установленной на узел процессора;
- узел С имеет Класс 3 доступа на канал 2А, основанный на привилегии установленной на узел процессора.

Важно: Если привилегии узла не были назначены в этом примере, узел будет имел тот же самый класс привилегий, какой назначен на канал.

Совет

При использовании функции привилегий и паролей пользуйтесь следующими руководящими принципами:

- Вы должны определить пароли и информацию привилегий для каждого процессора в вашей системе.
- Вы **не можете** назначать привилегии на каналы сконфигурированные как сканер или адаптер. Привилегии чтение/запись, которые Вы видите на экране привилегий канала, устанавливаются на этот канал соответствующие привилегии чтения/записи из экрана конфигурации канала. Привилегии чтение/запись для каждого файла диагностики канала (экран статуса канала) должны быть заданы через экран привилегий данных (Data Table Privileges screen). Все станции/узлы, подключенные к каналу, имеют тот же класс привилегий, что этот канал.
- Сообщите всем пользователям вашего программного обеспечения какой класс привилегией они могут использовать и соответствующий пароль. Если они хотят изменить класс привилегий (другой, чем тот, для которого терминал программирования сконфигурирован), они должны ввести новый класс и пароль.
- Пароли и привилегии помогут Вам защитить от несанкционированного или случайного изменения в системе. Однако, пароли и привилегии имеют ограничения; это не защитит от действия злонамеренного вмешательства, и при этом это не может гарантировать, что изменения, сделанные лицом с паролем будут соответствовать практическому приложению.

Определение классов привилегий

Вы можете определять четыре класса привилегии (класс 1-4), каждый с собственным паролем. В пределах каждого класса, Вы можете назначать доступ на некоторые действия в программном обеспечении (как, изменения программы или файла данных, или конфигураций канала). Эти классы привилегий – являются верхним уровнем организации структуры паролей.

Например, Вы можете определить для Класса 1 все привилегии, это будет эквивалентно системному администратору. Для остальных трех классов можно назначить меньше привилегий.

Например, установите классы привилегий следующим образом (X указывает, что привилегия есть):

Привилегии \ Наименование класса	Класс1	Класс2	Класс3	Класс4
Modify Privileges (Изменение привилегий)	X			
Data Table File Create/Delete (Файл данных Создание/Удаление)	X	X	X	
Program File Create/Delete (Программный файл Создание/Удаление)	X	X	X	
Logical Write (Восстанавливать файл памяти процессора, используя логическую адресацию)	X	X	X	X
Physical Write (Восстанавливать файл памяти процессора, используя физическую адресацию)	X	X	X	X
Physical Read (Чтение из процессора, используя физическую адресацию)	X	X	X	X
Logical Read (Чтение из процессора, используя логическую адресацию)	X	X	X	X
Mode Change (Изменение режима работы)	X	X	X	X
I/O Force (Форсирование Вх\Вых)	X	X		
SFC Force (Форсирование ПФС)	X	X		
Clear Memory (Очистка памяти)	X			
Restore (Восстановление программ)	X			
On-line Editing (Онлайн редактирование)	X			
Modify passwords (Изменение пароля)	X			

Назначение класса привилегии на канал или файл офлайн

Вы можете установить класс привилегий на все каналы (исключая remote I/O сканер или канал адаптера) и файлы офлайн. По умолчанию любому каналу назначен класс привилегий 1.

Экран привилегий канала показывает класс привилегий чтения/записи установленный для канала. По умолчанию все каналы и файлы имеют 1 Класс привилегий.

Экран привилегий канала показывает привилегии чтения/записи, установите в экране конфигурации канала доступ чтение/запись для канала. В экране привилегий таблицы данных установите привилегии чтение/запись для диагностических файлов каналов. По умолчанию, все станции узлы, подключенные к каналу, имеют тот же класс привилегий назначенный для этого канала.

Важно: Вы не можете назначить привилегии по умолчанию для канала, работающего в режиме сканера или адаптера.

Назначение класса привилегий для узлов

Все станции/узлы имеют по умолчанию тот же класс привилегий, что и канал через который они работают. Если необходимо Вы можете назначить узлу свой класс привилегий, отличный от привилегии, назначенной каналу.

Важно: Класс привилегий узла подавляет класс привилегий канала.

Назначение привилегий чтения/записи для программных файлов

Вы можете назначить привилегии чтения или записи для каждого файла процессора. Эти привилегии ограничивают доступ пользователей к просмотру или изменению ваших программных файлов. Две различные привилегии определяют, сможет ли пользователь читать программный файл или писать в него:

- класс привилегий пользователя;
- назначены ли привилегии чтения и записи самому файлу программы.

Назначение привилегии чтения/записи файлу данных

Вы можете назначить привилегии чтения и записи для каждого файла данных процессора. Эти привилегии ограничивают доступ пользователя к просмотру или изменению значений файлов данных. Две различные привилегии определяют, сможет ли пользователь читать файл данных или писать в него:

- класс привилегий пользователя;
- назначены ли привилегии чтения и записи самому файлу данных.

Важно: Удаление доступа на чтение и запись файла таблицы данных не позволит вам вызвать этот файл.

Использование защиты процессоров

Чтобы избежать компромиссов безопасности при импорте и экспорте файлов из(в) процессоров PLC-5/26, -5/86 серии С ревизии Н и более поздних, используйте программное обеспечение 6200 версии 5.0 и более поздней.

Более ранние версии программного обеспечения 6200 не поддерживают связь с защищенными процессорами ревизии Н и более поздних, и могут неправильно определить их.

Аналогично, более раннее программное обеспечение других производителей может неправильно распознать процессоры PLC-5 ревизии Н и более поздних.



Для более подробной информации по программированию защиты процессоров, смотрите публикацию 1785-6.5.13. "PLC-5 Protected Processor Supplement".

Для заметок

Соглашения по программированию**Назначение этой главы**

Сведения	Страница
Форсирование	14-1
Использование специальных подпрограмм	14-2
Задание приоритетов для прерываний и МСР	14-4
Определение и программирование подпрограмм прерываний	14-6

Форсирование

Форсирование Вх/Вых позволяет Вам изменять состояние битов входов и выходов во включенное или выключенное состояние для производства наладочных работ. Форсирование битов вкл. или откл. и переходов ПФС позволяет Вам имитировать работу или управлять устройствами.

Важно: Форсирование входов дает возможность форсировать биты в таблице данных входов. Форсирование выходов позволяет форсировать реальные выходы модуля, не изменяя состояния файла таблицы данных выходов.

Важно: Форсирования осуществляются процессором (а не программным терминалом). Форсирования остаются, даже при отключенном программном терминале.



ВНИМАНИЕ: Когда Вы что-либо форсируете (включить или выключить), удалите персонал из района работы механизма. Форсирование может привести к неожиданным движениям механизма, которые могут травмировать персонал.

Форсирование входов и выходов

Вы можете форсировать биты непосредственно из редактора релейно-контактной логики или из экрана монитора форсажа. Если Вы определились какой бит форсировать, то до форсирования Вы должны сначала разрешить форсажи. Вы можете форсировать только те точки Вх/Вых, которые физически присутствуют и сконфигурированы в вашей системе.

В локальном процессоре с адресацией 1/2-слота Вы не можете форсировать входные биты для пустого слота или для следующего слова слота с 8-ми или 16-ти точечным модулем. Например, если 8-ми или 16-ти точечный модуль установлен в первом слоте вашего локального шасси (слово 0 и 1 таблицы Вх/Вых, 1/2-слота адресация), Вы не можете форсировать входные биты слова 1 (I:001).

Вы не можете форсировать:

- выходной адрес входной инструкции;
- входной адрес выходной инструкции;
- другие адреса битов отличные от входов и выходов, такие как N, B, T, C, и др. адреса.

Форсирование переходов ПФС

При контроле работы ПФС можно форсированием включать и выключать переходы. Это дает возможность изменять ход выполнения вашей ПФС, для поиска неисправностей.

Использование специальных подпрограмм

При необходимости используйте в Вашем проекте одну или несколько следующих специальных подпрограмм:

- подпрограмма включения питания;
- подпрограмма обработки ошибок (необходима для исправления ошибок оборудования);
- программа прерывания по времени (программируемое время прерывания);
- подпрограмма прерывания по событию (прерывание по входу процессора).

Таблица 14.А поясняет когда использовать соответствующие программы.

Таблица 14.A
Рекомендации по использованию специальных подпрограмм

Если логика выполнила	Пример	Используйте	Необходимо выполнить следующее
Как только обнаружены условия требующие старта	Перезапуск системы после того, как она была отключена	Подпрограмму включения питания (Power-UP Routine)	Создать отдельный файл для процедуры запуска, который будет выполняться первым после старта программы после выполнения системных процедур. Процессор выполняет подпрограмму подачи питания .
Как только обнаружена главная ошибка	Отключение устройств, как только обнаружена главная ошибка или посылка статуса головному процессору	Подпрограмму ошибок (Fault Routine)	Создать специальный файл, который будет обрабатывать главную ошибку. Первая обнаруженная ошибка определяет, какая программа будет выполнена. Процессор выполнит подпрограмму обработки ошибки. Если подпрограмма очистит ошибку процессор продолжит выполнение программы с того места, где она была прервана. Если нет, то по ошибке процессор будет переведен в режим программирования
Через определенный интервал времени	Отображение положения механизма каждые 250 мс и вычисление среднего значения перемещения или Сравнение измеренного значения с заданным каждую 1 сек.	Выбор прерывания по времени (STI)	Создать специальный программный файл и задать интервал времени. Процессор будет прерывать выполнение главной программы через указанные интервалы и выполнять программу прерывания по времени, после выполнения главная программа будет продолжена с того места где, было вызвано прерывание. Процессор прерывает выполнение главной программы через заданный промежуток времени и запускает программу STI. Когда инструкция блок-трансферов к удаленным I/O сталкивается с STI, процессор переходит к выполнению программы с более низким приоритетом (главной логической программы) пока блок-трансфер не выполнится. Если это происходит и Вы хотите, чтобы ваш STI запускался до возвращения к главной логической программе, используйте в вашем программном файле STI инструкции UID и UIE.
Сразу после наступления события	Удалить бракованную бутылку с линии разлива	Подпрограмму прерывания процессора по входу (PII)	Создать отдельный программный файл и задать 16 входов в слове Вх/Вых. Когда наступит событие, процессор прервет выполнение главной программы, и запустит программу PII, и по завершению продолжит с того места, где было вызвано прерывание. Когда инструкция блок-трансфера к удаленным Вх/Вых сталкивается с PII, процессор переходит к выполнению программы с более низким приоритетом (главной логической программы) пока блок-трансфер не выполнится. Если это происходит и Вы хотите, чтобы ваш PII запускался до возвращения к главной логической программе, используйте в вашем программном файле STI инструкции UID и UIE.

Задание приоритетов для прерываний и МСР

Процессоры PLC-5 выполняют подпрограммы ошибок, прерываний, и главные программы управления по приоритетам. Эти приоритеты называются “планирование”. Процессор PLC-5 выполняет в первую очередь наиболее важные задачи планирования. Приоритет планирования каждой задачи следующий (от высшего к низшему):

1. подпрограмма ошибок;
2. прерывания процессора по входу (PII);
3. прерывания по времени (STI);
4. главная программа управления (МСР).

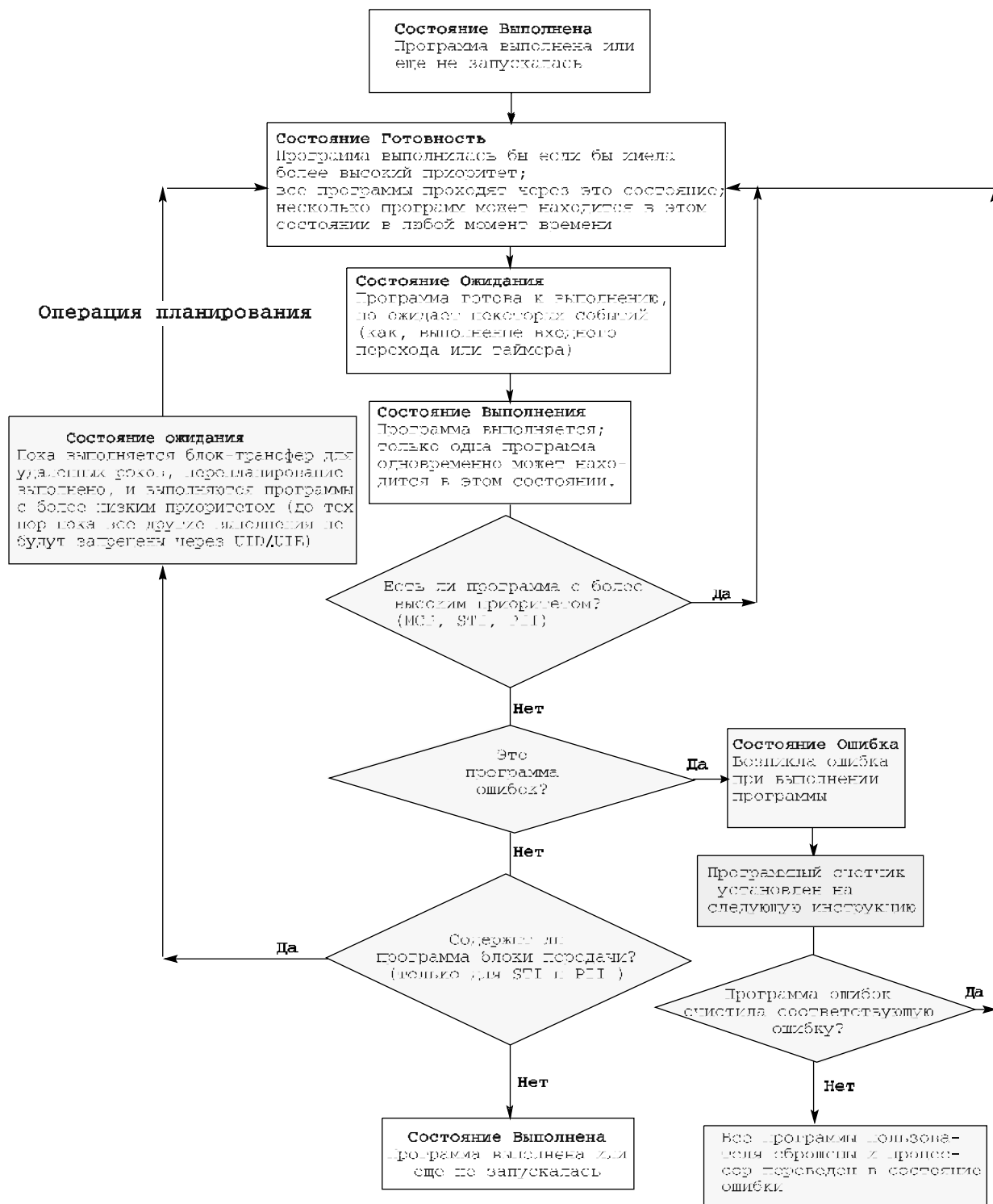
Это планирование определяет, какой части программы сейчас выполняться. Например, если выполняется PII, не может быть вызвано прерывание STI, пока PII не закончится (так как PII имеет приоритет планирования выше, чем STI). Если во время выполнения МСР вызывается программа ошибок, выполнение МСР будет прервано так, как программа ошибок имеет приоритет выше, чем МСР.

Важно: Вы можете временно изменить приоритет планирования путем применения инструкций UID и UIE. Таким образом, с помощью этих инструкций можно прервать выполнение подпрограммы ошибок (см. стр. 14-4).

Программа ошибок, PII, STI управляют прерываниями. Они могут выполняться в любое время за исключением режима редактирования. МСР выполняются от начала до конца пользовательской программы.

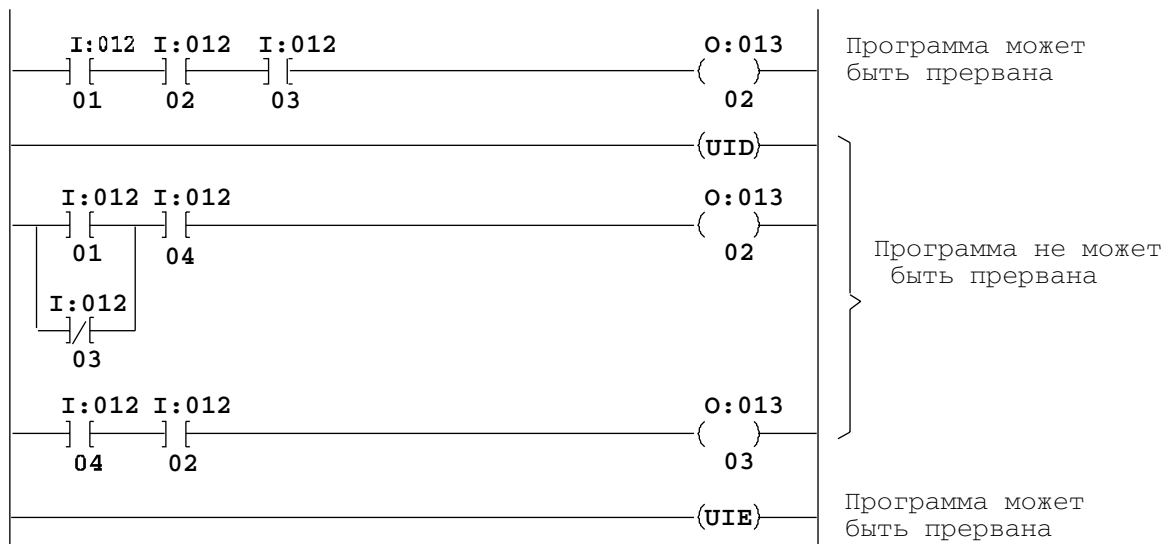
Состояние выполняемой программы

Программа пользователя в PLC-5 процессоре может находиться в одном из пяти состояний: выполнена, готова, выполняется, ожидание или ошибка.



Влияние на планирование Приоритетов

Используйте UID (запрещение прерывания пользователя) и UIE (разрешение прерывания пользователя) инструкции для изменения планирования в пользовательской программе. Они могут быть использованы для защиты важных участков программы, которые должны выполняться до завершения. Инструкции UID/UIE разработаны, чтобы использоваться в парах. Например:



После того, как выполнилась инструкция UID, прерывания откладываются. Программы прерывания переводятся в состояние готовности. После выполнения UIE инструкции, программы пользователя, находящиеся в состоянии готовности, начинают выполняться по их приоритетам. Если программа в состоянии "готовность" имеет приоритет выше, чем исполняемая программа, исполняемая программа переводится в состояние готовности, и начнется выполнение программы по прерыванию. Пока процессор исполняет программу в пределах зоны инструкций UID/UIE, исполняемая программа не может быть прервана за исключением программы ошибки.



Для более подробной информации по программированию инструкций UID и UIE см. публикацию 1785-6.1 "PLC-5 Programming Software Instruction Set Reference".

Определение и программирование подпрограмм прерывания

Для информации по конфигурированию и программированию этих подпрограмм, см. соответствующий раздел:

Для информации о:	См. главу:
Подпрограмма подачи питания	15
Программа ошибок	16
Главные программы управления (MCPs)	17
Прерывание по времени (STIs)	18
Прерывания процессора по входу (PIIs)	19

Установка подпрограмм подачи питания

Назначение этой главы

Сведения	Страница
Установка защиты подачи питания	15-1
Разрешение или запрещение старта	15-1
Определение процедуры подачи питания	15-2

Установка защиты подачи питания

Вы можете сконфигурировать ваш процессор так, что при потере питания во время работы, процессор не вернется обратно в режим работа после подачи питания. Бит контроля S:26/1 определяет, как работать защите по питанию при подаче питания (например, выполнить программу ошибок).

Если S:26/1 После потери питания процессор выполнит

Установлен (1) Сканирование программы ошибок перед возвратом к сканированию рабочей программы. Когда установлен бит, процессор однократно сканирует программу ошибок при включении после потери питания. Вы можете в программе ошибок определить состояние статусных слов процессора и в зависимости от логики программы запретить или разрешить старт процессора.

Установлен (0) При подаче питания всегда начинать с первой цепи первого файла программы.

Установите S:26/1 вручную в экране Статуса Процессора или в вашей релейно контактной программе.

Разрешение или запрещение старта

Основное повреждение, бит S:11/5, контролирует, когда после подачи питания на процессор его можно перевести в режим работа. Не путайте этот бит с битом контроля S:26/1.

Этот бит	Указывает процессору
Бит контроля S:26/1	Сканировать или нет программу ошибок при подаче питания до возврата к сканированию рабочей программы.
Основное повреждение S:11/5	Есть или нет повреждение в конце сканирования программы ошибок.

После потери питания, когда процессор находился в режиме работа, процессор автоматически устанавливает основное повреждение бит S:11/5, если установлен бит контроля S:26/1.

**Если программа
ошибок обнаружит
S:11/5:**

Тогда процессор:	Если программа ошибок обнаружит S:11/5:
Установлен (1)	Устанавливает ошибку в конце сканирования программы ошибок. Оставить бит установленным, чтобы запретить старт.
Установлен (0)	Помещает результат сканирования в файл памяти процессора. Сбросить бит, чтобы разрешить старт.

Важно: Вы можете использовать инструкции JMP и LBL для сканирования части программы ошибок связанной со спецификой ошибки подачи питания.

Определение процедуры подачи питания

Биты контроля S:26/0 и S:26/1 определяют, как перейти процессору в режим работа после потери питания, или когда Вы перевели ключ в режим работа из режима программирования или тестирования.

```

Processor Configuration
User Control Bits 00000000 00000000 RESTART FIRST STEP
Fault routine prog file no.: 0 Watchdog (ms): 500
I/O status file: 0 Communication time slice (ms): 3
VME Status File: N34

```

Для установки и сброса бита:

1. установить курсор на поле бита;
2. для установки ввести 1; для сброса бита ввести 0.

**Используйте Для:
этот бит:**

0	Контроля процессоров использующих ПФС. Этот бит определяет перезапуск ПФС или сообщает о последнем активном шаге после потери питания.
1	Выбор защиты от потери питания. Если этот бит установлен и была потеря питания, процессор устанавливает основное повреждение бит 5 и выполняет программу ошибок перед возвратом к сканированию рабочей программы.

В таблице 15.А даны возможные варианты программ подачи питания. Для более подробной информации о программе ошибок, см. главу 16, "Установка программ ошибок".

См. главу 21 для назначения битов контроля (S:26/0-6).

Таблица 15.А
Возможные варианты программ подачи питания

Если Вы:	Совместно:	И Вы хотите:	Установите биты 0 и 1: 15..... 0
Используете ПФС	Нет программы ошибок	Перезапуск с первого шага	xxxxxxx xxxxx00
		Перезапуск с последнего активного шага	xxxxxxx xxxxx01
Не используете ПФС	Программа ошибок	Запуск с первого файла	xxxxxxx xxxxx0x
		Перезапуск с использованием файла ошибок	xxxxxxx xxxxx1x
Используете файл ошибок	ПФС	Перезапуск с использованием файла ошибок и затем с первого шага	xxxxxxx xxxxx10
		Перезапуск с использованием файла ошибок и затем с последнего активного шага	xxxxxxx xxxxx11
Не используется файл ошибок	Не используется ПФС	Старт с первого файла памяти процессора	xxxxxxx xxxxx00

Каждый x показывает, что бит может быть установлен в 0 или 1

Для заметок

Установка подпрограмм ошибок

Назначение этой главы

Сведения	Страница
Понятие концепции подпрограммы ошибок	16-1
Понятие обнаружения процессором основных повреждений	16-2
Описание программы ошибок	16-4
Описание программного сторожа	16-5
Программирование программы ошибок	16-6
Монитор ошибок	16-10

Понятие концепции подпрограммы ошибок

Подпрограмма ошибок выполняется когда процессор PLC-5 во время режима работа обнаруживает ошибку (основное повреждение). Если ваш процессор обнаружил ошибку во время выполнения программы, Вы можете задать процессору прервать выполнение программы, запустить вашу подпрограмму ошибок, а затем продолжить выполнение программы.

Подпрограмма обработки основных повреждений, находит в S:11 установленный бит и на основании установленного бита определяет направление дальнейшей работы программы. Программа ошибок выполняет, например, следующее:

- систематически останавливает процесс или операцию контроля;
- регистрирует и очищает ошибку и продолжает нормальную работу.

Для детального списка статусных слов процессора см. главу 21 “Файл состояния процессора”.

Реакция на основное повреждение

Когда процессор обнаруживает основную ошибку, он немедленно прерывает выполнение текущей программы. Если есть подпрограмма обработки ошибок (т.е. в слове S:29 назначен файл обработки ошибок), процессор запускает эту подпрограмму ошибки для удаления неисправности. В зависимости от типа ошибки процессор затем:

- возвратится к выполнению текущей программы, если сможет сбросить ошибку
- переведет процессор в режим “авария”, если не сможет сбросить ошибку

Например, эта цепь содержит инструкцию определения причины основной ошибки:



В этом примере процессор выполняет программу ошибок, после обнаружения ошибки. Если программа ошибок сбрасывает бит ошибки, процессор возвращается к выполнению следующей цепи перед которой произошла ошибка.

Если Вы не запрограммировали цепь для сброса ошибки В, процессор переходит в состояние ошибка.

Понятие обнаружения процессором основных повреждений

В основном.

Если процессор обнаруживает:	Он устанавливает:
Основную ошибку	Устанавливает бит основной ошибки и сбрасывает Вх/Вых
Ошибка выполнения	Устанавливает бит основной ошибки и устанавливает выходы на 1771-ASB и 1771-ALX в соответствии с установленными на них переключателями. Выходы остаются в последнем состоянии или обесточиваются в зависимости от того, как установлен переключатель на шасси.

Важно: В резидентном шасси процессора PLC-5 выходы сбрасываются независимо от установки переключателей при следующих условиях:

- процессор обнаружил ошибку выполнения;
- вы установили бит статусного слова “обнулить локальный рЭК”;
- вы выбрали режим программирования или тестирования.

Для установки переключателей Вы должны определиться, как будут действовать механизмы в системе при возникновении ошибки. Например:

- как будут реагировать машины, на выходы остающиеся в их последнем состоянии или на отключение выходов?
- что подключено к каждому выходу?
- будет ли машина продолжать движение?
- может ли это привести к нестабильности процесса?

По вопросу установки переключателей см. главу 23 “Руководство по установке переключателей”.

Ошибка процессора в резидентном или расширенном локальном рэке Вх/Вых

Если возникает проблема с объединительной платой шасси, процессор устанавливает соответствующую не основную ошибку, биты S:7/0-7, и продолжает сканирование программы и Вх/Вых. Как только устанавливается этот бит, выходы для этого рэка сбрасываются. Однако процессор продолжает нормальную работу.

Выходы будут разрешены к сканированию только после того как будут устранены условия вызвавшие ошибку рэка.

Ваша релейно-контактная программа должна контролировать биты ошибок Вх/Вых (S:7/0-7) и принимать соответствующие восстановительные действия.



ВНИМАНИЕ: Если произошла ошибка локального рэка Вх/Вых, и Вы не имеете метода восстановления ошибки, таблица состояния Вх/Ввы рэка с ошибкой остается в своем последнем состоянии. Результатом этого может быть причинение потенциального ущерба персоналу и повреждение машин.

Ошибка удаленного шасси Вх/Вых

Как правило, когда происходит ошибка шасси удаленных Вх/Вых, процессор устанавливает бит ошибки рэка и затем продолжает сканирование программы и контроль Вх/Вых.

Выходы рэка с ошибкой остаются в их последнем состоянии или обесточиваются, это зависит от того, как на шасси Вы установили переключатель последнего состояния Вх/Вых.



ВНИМАНИЕ: Если состояние выходов контролируются через входы другого рэка и в удаленном рэке произошла ошибка (входной рэк), входы останутся в их последнем состоянии, а не в ошибке. Контролируемое состояние выходов может не соответствовать их истинному положению, результатом этого может быть причинение потенциального ущерба персоналу и повреждение машин. Убедитесь, что Вы имеете метод восстановления ошибки.

Выходы процессора резидентного шасси и удаленного шасси, не имеющие ошибок, будут оставаться активными, если ошибка обнаружена в удаленном шасси. Выполните вашу программу таким образом, чтобы система знала состояние выходов в резидентном шасси и удаленном шасси не имеющим ошибок, даже если контролируемые входы расположены в удаленном шасси с ошибкой. Ваша программа должна сохранить последнее состояние входов, или индексировать ошибку рэка и установить таблицу входов неисправного удаленного шасси в неактивное состояние.

Возможны два метода программирования:

- В первой выполняемой инструкции, программа контролирует биты ошибок. Если устанавливается любой бит ошибки, программа копирует нули (0) в таблицу входов рэка с ошибкой. Программа должна постоянно в начале сканирования записывать нули в таблицу входов пока присутствует ошибка рэка, так как процессор в конце сканирования программы установит входы в их последнее состояние перед ошибкой.
- В первой выполняемой инструкции, программа контролирует биты ошибок. Если устанавливается любой бит ошибки, процессор устанавливает бит запрета для рэка с ошибкой. Программа должна однократно записать нули в таблицу входов рэка с ошибкой.

Описание программы ошибок

Вы можете создать несколько подпрограмм ошибок и хранить их в нескольких файлах, но процессор запускает только одну программу ошибок когда PLC-5 обнаруживает основную ошибку. Вы можете однако заменить программу ошибок из вашей релейно-контактной логики. Если Вы не назначите номер файла ошибок, то после обнаружения ошибки процессор переключится в режим ошибка.

Для назначения в процессоре программы ошибок:

```

Processor Configuration
User Control Bits      00000000 00000000      RESTART LAST ACTIVE STEP
Fault routine prog file no.: 0                Watchdog (ms):      500
I/O status file:      0                      Communication time slice (ms): 3
VME Status File:      N34
  
```

↑
Введите в поле S:29 номер программного файла, который будет выполняться при обнаружении ошибки. Установите в этом поле 0 если не используете файл ошибок.

Для более подробной информации по кодам ошибок, см. документацию по вашему программному обеспечению.

Описание программного сторожа

Программный сторож (S:28) контролирует время сканирования программы. Если время сканирования больше, чем значение уставки таймера программного сторожа, то запускается подпрограмма ошибок.

Таймер опеределеляет максимальное время сканирования (мс) для программного сторожа; или если Вы используете ПФС, максимальное время одного прохода всех активных шагов.

Для определения значения отличного от установленного по умолчанию:

```

Processor Configuration
User Control Bits 00000000 00000000
Fault routine prog file no.: 0
I/O status file: 0
VME Status File: N34
RESTART LAST ACTIVE STEP
Watchdog (ms): 500
Communication time slice (ms): 3

```

↑
Введите новое значение 10-32767 мс
Избегайте произвольного изменения значения по умолчанию.

Важно: Программный сторож может быть установлен не менее, чем 10 мс, даже если в программе всего один вход.

Уход от многочисленных ошибок программного сторожа

Если вы столкнулись с аппаратной ошибкой или основной ошибкой программного сторожа, это может быть потому что многочисленные ошибки программного сторожа появляются, когда процессор занят обслуживанием основных ошибок цикловой логики. Аппаратная ошибка появляется когда возникает заполнение очереди ошибок, которая хранит максимум шесть ошибок, и негде хранить очередную ошибку.

Если у Вас возникает аппаратная ошибка или многочисленные ошибки программного сторожа, перед тем как обратится к представителю сервисного обслуживания проверьте:

Если вы столкнулись:	Тогда:
Ошибка программного сторожа и бит ошибки	Установите величину программного сторожа такой, чтобы ошибка исчезла. Проверьте биты основных ошибок. Игнорируйте бит ошибки программного сторожа, используйте другие биты ошибок для определения источника ошибки процессора.
Аппаратная ошибка	<ol style="list-style-type: none"> 1. Отключите и снова подайте питание на процессор. 2. Перезагрузите программу. 3. Установите значение программного сторожа = 10 x нормальная уставка 4. Запустите снова программу.

Если у Вас продолжает появляться аппаратная ошибка, обратитесь к представителю ALLEN-BRADLEY



Создание подпрограммы ошибок

Создайте вашу подпрограмму ошибок, которая будет определять тип основной ошибки процессора PLC-5 и затем решать, что делать дальше до того, как процессор автоматически перейдет в состояние ошибки:

- установить тип аварии;
- очистить ошибку;
- выполнить соответствующую программу ошибок через релейно-контактную логику;
- выполнить соответствующую релейно-контактную логику, что бы избежать ошибку.

Важно: Если процессор PLC-5 обнаруживает ошибку в подпрограмме ошибок (условие двойной ошибки), процессор PLC-5 переходит в состояние ошибки без вызова подпрограммы ошибок.

Установка аварии

Если Вам необходимо подать сигнал аварии при возникновении основной ошибки, поместите в первой цепи вашей программы ошибок цепь:



и объедините это со счетчиком. Вы можете также установить сигнал аварии в вашей программе аварий, когда очищаете основную ошибку.

Очистка главной ошибки

Вы можете очистить главную ошибку одним из следующих методов:

- поверните ключ на процессоре PLC-5 из **REM** в **PROG** и **RUN**;
- используйте программное обеспечение для очистки основной ошибки (если возможно).

В статусном экране процессора, нажмите [F10] – **Clear Major Fault**



ВНИМАНИЕ: Очистка главной ошибки, не всегда корректное устранение ошибки. Убедитесь, что исследовали бит ошибки и устранили причину ошибки перед ее очисткой.

Например, если установлен бит основной ошибки S:11/2, индицирующий ошибку программы, **не используйте** программу ошибок для очистки ошибки пока Вы не исправите вашу программу.

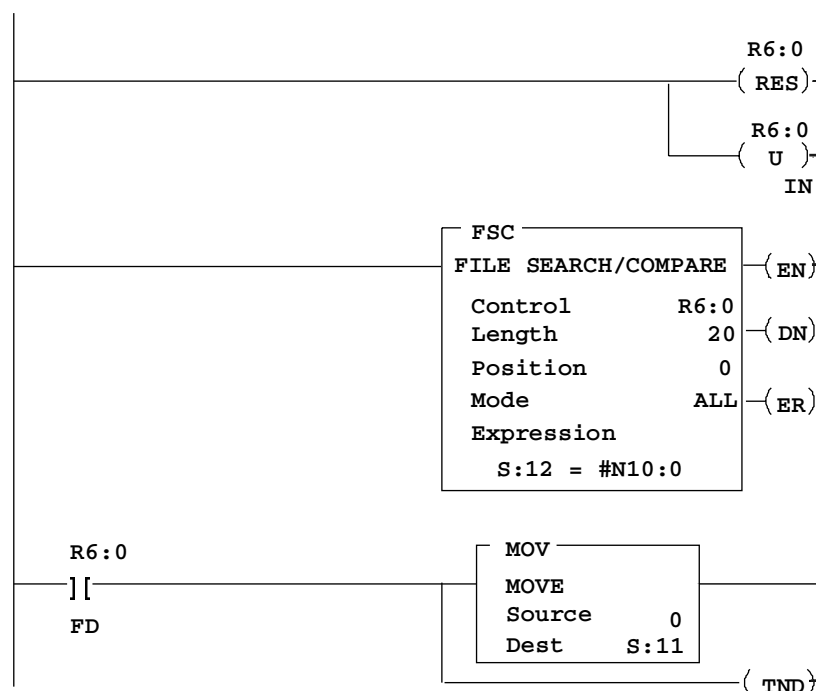
Если вы решили очистить ошибку в программе ошибок, выполните следующее:

1. поместите релейно-контактную логику для очистки ошибки в начале программы ошибок;
2. определите тип основной ошибки;
3. Используйте только те приложения, которые позволяют очистить вашу ошибку. Используйте ваши ссылки на коды ошибок;
4. определите по коду ошибки процессора S:12 причину основного повреждения;
5. Используйте инструкцию Сравнения Файлов (FSC), для сравнения кода ошибки с файлом содержащим допустимые коды ошибок. Если процессор находит совпадение, FSC инструкция устанавливает бит обнаружения (.FD) структуры управления;
6. используйте инструкцию MOV для очистки ошибки в слове S:11.

На рис 16.1, #N10:0 – файл ссылок на ошибки.

Рисунок 16.1

Пример сравнения кода основной ошибки с файлом ссылок.



Далее продолжение программы ошибок.

Если программа ошибок	Тогда процессор
Очистила S:11	Возвращается к программному файлу и продолжает выполнение программы
Не очистила S:11	Прекращает выполнение программы ошибок и переходит в режим ошибка

Важно: Если программа ошибок очистила основную ошибку, процессор прерывает выполнение программы ошибок и возвращается к выполнению инструкции основной программы, которая следует за инструкцией вызвавшей ошибку.

Выполняются оставшиеся цепи, и кажется будто ошибки никогда не было. Программа ошибок будет выполняться до тех пор пока, вы не устраните причину ошибки.

При создании программы ошибок пользуйтесь следующими принципами:

- запоминайте возникающие события и сбрасывайте другие данные для достижения нормальной работы в дальнейшем;
- Контролируйте отключение критических выходов. Используйте циклы, если это необходимо для увеличения времени сканирования подпрограммы ошибок больше времени уставки программного сторожа, что бы ваша программа могла подтвердить наличие критических событий.

Выбор программы ошибок из релейно-контактной логики

Вы можете заменить файл ошибок из вашей релейно-контактной логики путем копирования нового номера файла ошибок в слово статусного файла процессора S:29.

На рисунке 16.2 показан пример цепи для изменения номера файла ошибок.

Рисунок 16.2

Пример изменения номера файла ошибок.



ВНИМАНИЕ: Не разрушайте номер программного файла ошибок и не используйте этот файл в других целях. Если номер файла, указанного вами, не содержит подпрограммы ошибок, процессор немедленно перейдет в режим ошибка после обнаружения ошибки. Неожиданные действия машин могут привести к поломке оборудования и/или травмированию персонала.

Использование релейно-контактной логики для устранения причин ошибок

Если Вы имеете соответствующую программу ошибок и релейно-контактную логику для соответствующего отключения вашей системы, вы можете определить ошибку Вх/Вых как неосновное повреждение. Вы можете запрограммировать релейно-контактную логику для выбора нескольких способов устранения ошибки рэка входов/выходов.

Таблица 16.А Способы программирования релейно-контактной логики для устранения ошибки рэка

Метод	Описание
Формирование пользователем основной ошибки	<p>Когда возникает ошибка удаленного рэка Вх/Вых процессор переходит к программе ошибок. Другими словами, если статусный бит индицирует ошибку, ваш процессор воспринимает это, как основную ошибку (т.е. переходит к программе ошибок). Вы затем программируете в программе ошибок остановить процессор или надлежащим образом закрываете вашу систему. Когда процессор выполнит последнюю инструкцию программы ошибок, пользователь формирует основную ошибку.</p>
Сброс таблицы данных входов	<p>Вы контролируете статусные биты и если возникает ошибка, ваша программа определяет это как не основное повреждение. После того, как статусный бит покажет ошибку, переведите неисправный рэк в неактивное состояние с помощью статусного экрана Вх/Вых вашего программного обеспечения. Затем с помощью релейно-контактной логики установите или сбросьте биты критических входов в таблице данных входа согласно состоянию выходов рабочего рэка.</p> <p>Если вы сбросили биты таблицы входов, то во время следующего обновления таблицы Вх/Вых, входные биты неисправного рэка снова установятся в их последнее состояние. Для предотвращения этого, ваша программа должна установить запрет для неисправного рэка. Глобальное запрещение битов контроля таблицы входов на базе полного рэка; запрет части рэка для контроля таблицы входов на базе 1/4 рэка. Для более подробной информации по глобальным статусным битам см. ваше программное обеспечение.</p> <p>Этот метод требует детального и осторожного подхода к восстановлению операций Вашей системы. Для дальнейшей информации по запрещению рэков Вх/Вых, смотри информацию по Вашему программному обеспечению</p>
Метод программирования зоны ошибок	<p>Используя метод программирования зоны ошибок, Вы запрещаете часть вашей программы с MCR зонами. Используя статусные биты, Вы контролируете рэки; когда обнаруживается ошибка, в вашей программе становится активна цепь с MCR инструкцией. Этот метод позволяет сделать неактивными и обесточить выходы внутри зон MCR для рэков с ошибками.</p> <p>Для более подробной информации по программированию MCR зон смотри документацию по программному обеспечению.</p>

Блок-трансфер в программе ошибок

Если процессор запускает программу ошибок из инструкции содержащей блок-трансфер, процессор помещает этот блок-трансфер в начало буфера очереди блок-трансферов, и этот блок-трансфер ожидает вызова во главе буфера.

В программе ошибок или программе прерывания STI может находиться блок-трансфер только для локальных входов/выходов.



ВНИМАНИЕ: Если инструкция блок-трансфера для удаленного шасси находится в программе ошибок или в программе STI, знайте, что MCP будет ожидать окончания передачи блок-трансфера, если вы не используете пару инструкций UIE/UID.

Проверка программы ошибок

Для проверки программы ошибок используйте инструкцию перехода JSR к программе ошибок. Установите код ошибки, как входной параметр инструкции JSR. Процессор устанавливает код ошибки в S:12 и устанавливает бит ошибки в S:11.

Вы можете обнаружить и послать свою ошибку, используя код ошибки 0-9 или используя процессором обнаруженный код ошибки 10-87.

Монитор ошибок

Контролируйте ошибки процессора с помощью Статусного Экрана Процессора.

Вы можете контролировать	Описание	Смотри страницу
Основные и неосновные ошибки	Процессор разделяет ошибки на основные и неосновные. Процессор заряжает свой бит каждой ошибки и показывает текст, описывающий ошибку.	16-11 для
Код ошибки	Коды ошибок содержат информацию об обнаруженных процессором ошибках.	21-5
Глобальный статусный бит	Глобальный статусный бит устанавливается, если обнаружена ошибка любого логического река.	16-11
Биты состояния множеств шасси	Биты состояния множеств шасси используются для контроля состояния реков Вх./Вых. вашей системы.	16-11

Монитор основных и не основных ошибок и кодов ошибок

Когда происходит ошибка, на экране Статуса Процессора указывается номер программного файла и номер цепи вызвавшие ошибку.

Processor Status	
Fault code	0
Where faulted:	0:0
Major fault:	00000000 00000000
Minor fault	1: 00000000 00000000
	2: 00000000 00000000

← программный файл (S:13), номер цепи (S:14)

Интерпретация основной ошибки

Выдаваемое описание основной ошибки

- В статусном поле текста появляется сообщение о наиболее существенной ошибке, если курсор не находится на статусном поле основной ошибки.
- Если курсор указывает на установленный бит статусного слова ошибки, выдается текстовое сообщение для этого бита ошибки.
- Если биты не установлены, текстовое поле не заполнено.

Для очистки ошибки выполнить

- Для сброса всех основных ошибок нажмите (F10) в экране Статуса Процессора. Если Вы очистили основную ошибку, код ошибки, номер программного файла и номер цепи, вызвавшие ошибку тоже очистятся.
- Сбросьте конкретные биты. Если вы имеете более, чем одну основную ошибку, и вы сбросили бит, в статусном поле текста появится следующее сообщение об ошибке.

Для расшифровки битов основных ошибок смотри раздел 21.

Интерпретация неосновной ошибки

Выдаваемое описание неосновной ошибки:

- В статусном поле текста появляется сообщение о наиболее существенной ошибке, если курсор не находится на статусном поле основной ошибки.
- Если курсор указывает на установленный бит статусного слова ошибки, выдается текстовое сообщение для этого бита ошибки.
- Если биты не установлены, текстовое поле не заполнено.

Для очистки ошибки выполнить:

- Для сброса всех неосновных ошибок нажмите (F9) в экране Статуса Процессора.
- Сбросьте конкретные биты. Если вы имеете более, чем одну неосновную ошибку, и вы сбросили бит, в статусном поле текста появится следующее сообщение об неосновной ошибке.

Для расшифровки битов неосновных ошибок слово 1 (S:10) и слово 2 (S:17), смотри главу 21

Монитор статусных битов

Два типа статусных битов выдают информацию о вашей системе: глобальные статусные биты и статусные биты для множеств шасси.

Каждый бит относится к своему рэку, независимо от того, сколько шасси используется в рэке. (Помните, что Вы можете сконфигурировать максимум 4 шасси по четверть рэка, занимающие один полный рэк Вх/Вых.) Это биты младших восьми разрядов слов S:7, S:32, и S:34.

Глобальные статусные биты устанавливаются, если происходит ошибка в любом рэке. См. таблицу определяющую количество битов.

Процессор	Возможное количество битов рэков Вх/Вых
PLC-5/11, -5/20, -5/20E	4
PLC-5/30	8
PLC-5/40, -5/40L, -5/40E	16
PLC-5/60, -5/60L, -5/80, -5/80E	24

Статусные **биты для множеств шасси** используются для контроля рэков входов/выходов вашей системы. Эта информация сохраняется в статусном файле Вх/Вых (S:16, младший байт), который вы задали в экране Конфигурации Процессора. Программное обеспечение для хранения статусных битов автоматически создает два целочисленных слова для каждого рэка.



Для более подробной информации по глобальным статусным битам и битам контроля множеств шасси см. ваше программное обеспечение.

Использование главных программ управления

Назначение этой главы

Сведения	Страница
Выбор главных программ управления	17-1
Понимание обработки процессором МСР	17-1
Конфигурирование МСР	17-3
Монитор МСР	17-4

Выбор главных программ управления

Вы можете иметь одновременно 16 активных программ. Каждая из этих программ называется “главная программа управления” (МСР). Вы можете выбрать одну МСР для каждого механизма или функции процесса. Это позволит Вам разделить последовательные функциональные схемы (ПФС), релейно-контактную логику, структурированный текст, разбить ваш процесс на модули и облегчит поиск неисправностей.

Рассмотрите использование этой технологии:	Если Вы:
ПФС	имеете процесс с определенным порядком выполнения
Релейно контактная логика	<ul style="list-style-type: none"> • более знакомы с программированием релейно-контактной логики, чем с языком программирования типа БЕЙСИК; • выполняете диагностику.
Структурированный текст	<ul style="list-style-type: none"> • более знакомы с языками программирования типа БЕЙСИК, чем с релейно-контактной логикой; • использование сложных математических алгоритмов; • использование программ состоящих из повторов или циклов; • создание пользовательских мониторов данных.

Главная программа управления может быть ПФС с номером 1-999; это может быть также любой программный файл, релейно-контактная логика или структурированный текст, с номером 2-999. Вы можете также использовать любое сочетание ПФС, релейно-контактной логики или структурированного текста в 16 главных программах управления. Для всех МСР используется один экран данных (т.е., Вы не имеете экрана данных для каждого МСР).

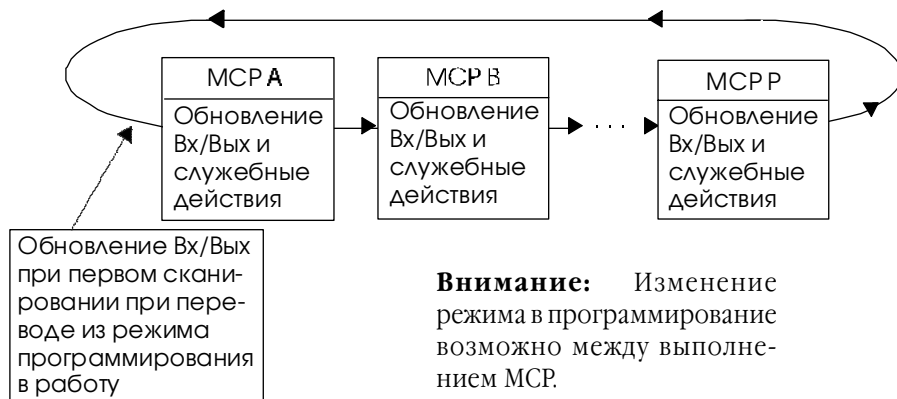
Понимание обработки процессором МСР

МСР исполняются в порядке указанном Вами в экране Конфигурации Процессора. Вы можете сконфигурировать:

- обновление таблицы Вх/Вых и служебные действия после выполнения каждой МСР (параметр по умолчанию), или;
- процессор пропускает сканирование Вх/Вых и переходит к выполнению следующей МСР.

После выполнения последнего МСР, все МСР повторяются в том же порядке. Знайте, что программный сторож контролирует одно сканирование всех МСР. На рис. 17.1 показано как процессор обрабатывает МСР, когда обновление Вх/Вых указано после выполнения каждого МСР.

Рисунок 17.1
Выполнение МСР с обновлением Вх/Вых после каждой МСР

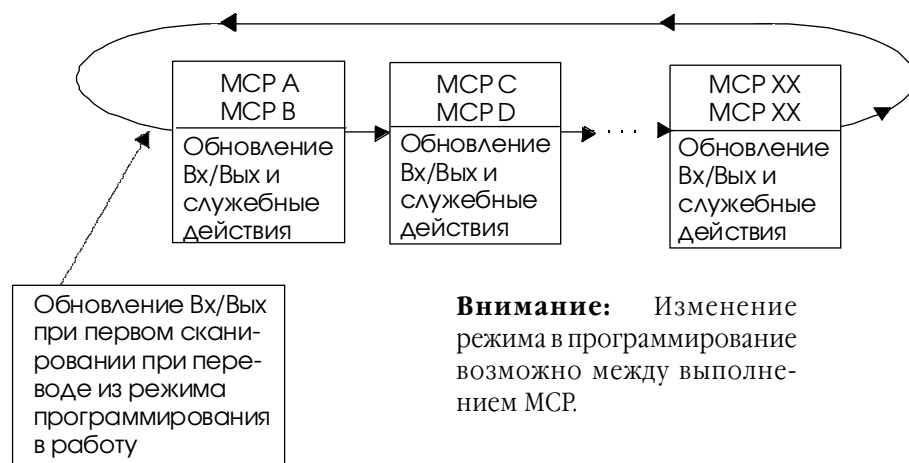


Запрещая сканирование Вх/Вых между МСР, Вы можете уменьшить время сканирования программы на 2-3 мс для каждого запрета обновления Вх/Вых. Процессор обновит ваши Вх/Вых когда встретит следующую команду сканирования Вх/Вых, которая может быть:

- разрешено сканирование Вх/Вых между МСР и/или;
- в конце прохода всего списка МСР.

Процессор всегда обновляет Вх/Вых после сканирования всего списка МСР.

Рисунок 17.2
Выполнение МСР с обновлением Вх/Вых не после каждого МСР



Если МСР:	Происходит следующее:
Релейно-контактная программа	<ol style="list-style-type: none"> 1. Все цепи выполняются с первой до последней, все счетчики, таймеры, переходы и подпрограммы активны. 2. После инструкции END релейно-контактной программы, процессор обновляет Вх/Вых – читает локальные входы, записывает локальные выходы, читает удаленные буферы и записывает удаленные буферы и выходы. 3. Процессор запускает следующее МСР.
Структурированный текст программа	<ol style="list-style-type: none"> 1. Выполняет коды, как обычно. 2. После выполнения последней инструкции программы, процессор обновляет Вх/Вых. 3. Процессор запускает следующую МСР.
ПФС	<ol style="list-style-type: none"> 1. Только активный шаг сканируется, и проверяются переходы для этого активного шага. 2. После прохода всех активных шагов, процессор обновляет Вх/Вых. 3. Процессор запускает следующую МСР.

Конфигурация МСР

Вы конфигурируете МСР в экране Конфигурации Процессора вашего программного обеспечения.

```

Processor Configuration

User control bits:      00000000 00000000      RESTART FIRST STEP
Fault routine prog file no.:  0              Watchdog (ms): 500
I/O status file:       0              Communication time slice (ms): 3
Processor input interrupt bit mask:  00000000 00000000
                        compare value: 00000000 00000000
                        down count:      0
                        prog file no.:   0  module group: 0
Selectable timed interrupt prog file no.: 0  setpoint: 0

Main control program A:  prog file no.:  0      disable: 0      I/O update: 0
                        B:      prog file no.:  0      disable: 0      I/O update: 0
                        C:      prog file no.:  0      disable: 0      I/O update: 0
                        D:      prog file no.:  0      disable: 0      I/O update: 0
                        E:      prog file no.:  0      disable: 0      I/O update: 0

```


Для поля:	Выполнить следующее:	Файл статуса:
Prog file no. (программный файл)	<p>Ввести номер программного файла для МСР А-Р в порядке, как они будут выполняться. Эта конфигурация читается перед выполнением МСР; если Вы делаете изменения в экране конфигурации, касающиеся МСР, эти изменения вступят в силу при следующем выполнении МСР. Вы можете изменить информацию МСР из экрана Конфигурации Процессора или из вашей релейно-контактной логики.</p> <p>Если Вы задали файл МСР, который не является релейно-контактной программой, структурированным текстом или ПФС, в статусном файле появится основная ошибка. Не основная ошибка появится также если все файлы МСР будут установлены в ноль.</p> <p>Вы можете задать один номер файла для нескольких МСР. Например, Вы хотите чтобы программа выполнялась чаще и имела выше приоритет чем другие программы.</p> <p>Если Вы не желаете использовать множество МСР, создайте ПФС (файл 1), релейно-контактную программу (файл 2), или структурированный текст (файл 2) и процессор будет выполнять вашу главную программу. Вам не надо делать какие либо вхождения в экране Конфигурации Процессора (процессор автоматически сконфигурирует первый программный файл, как первую МСР).</p>	S:80-S:127
disable (запрещение)	<p>Путем установки и сброса бита в этом поле Вы указываете процессору пропустить МСР пока бит сброшен. Если программный файл МСР запрещен, процессор пропускает файл и переходит к выполнению следующего файла.</p> <p>ВНИМАНИЕ: Если Вы запретили МСР, выходы останутся в состоянии в каком они были при последнем сканировании (т.е., все действия останутся активными). Удостоверьтесь, что вы контролируете любые выходы управляемые внутри этой МСР пред запрещением ее. Иначе, это может привести к травмированию персонала или повреждению оборудования.</p> <p>Запретите МСР, если Вы хотите временно удерживать состояние механизма, независимо от состояния перехода (например, механизм в состоянии ошибки). Запрещение МСР также позволит вам уменьшить время сканирования; если Вам нет необходимости запускать одну МСР каждый скан, Вы можете запретить его на необходимое время.</p> <p>Для установки и сброса битов МСР А-Р, установите курсор на соответствующее поле и введите 1 для запрета (пропуска) МСР или 0 для разрешения (сканирования) этого МСР.</p> <p>Если бит запрещения установлен для всех МСР программных файлов (это говорит о том, что все файлы пропускаются), не основная ошибка выдается в статусном файле процессора.</p>	S:79
I/O update (обновление Вх/Вых)	<p>1 установленная в этом поле указывает процессору пропустить сканирование Вх/Вых после этой МСР. Установка 0 указывает процессору на обновление Вх/Вых после этой МСР.</p> <p>Для задания бита обновления Вх/Вых, установите курсор на соответствующее поле и введите 0 или 1.</p>	S:78

Важно: Если Вы планируете использовать ПФС подпрограмму, удостоверитесь, что Вы задали какую либо МСР - достаточно даже пустого файла релейно-контактной логики. Если МСР не определена, процессор выдаст ошибку при повторном сканировании ПФС с кодом неисправности 71, подпрограмма ПФС уже выполняется.

Монитор МСР

Время сканирования для каждой МСР отображается в статусном файле процессора, включая текущее и максимальное время сканирования. В статусном файле также отображается общее время сканирования всех МСР, S:8 (время сканирования одного прохода всех МСР) и максимальное время сканирования, S:9.

Использование прерываний по времени

Назначение этой главы

Сведения	Страница
Использование прерываний по времени	18-1
Назначение прерываний по времени	18-3
Монитор прерываний по времени	18-4

Использование прерываний по времени

Программируемое прерывание по времени (STI) указывает процессору прервать выполнение программы (через указанное время) и выполнить однократно программу STI. Затем процессор продолжит выполнение программы, из которой было вызвано прерывание. Например, Вы хотите использовать STI для периодического обновления аналоговых данных для контура регулирования процесса или посылать данные машине через заданные интервалы.

Задание STI в релейно-контактной логике

Совет

Следуйте этим рекомендациям для записи релейно-контактной логики для STI.

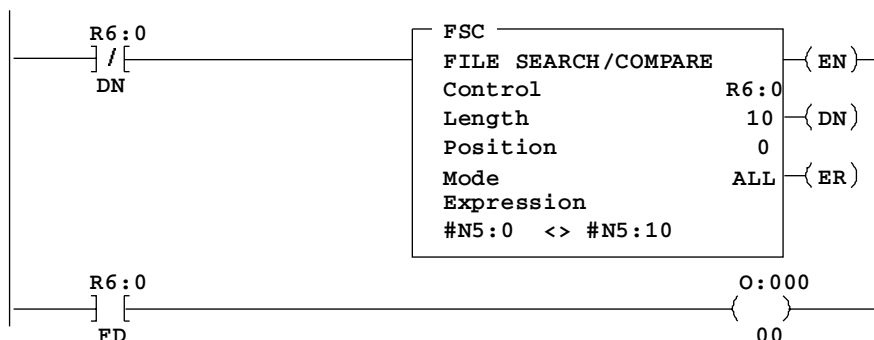
- Задайте STI файл в вашей цикловой программе.
- Убедитесь, что интервал прерывания (статусное слово S:30) больше чем время выполнения программы STI. Если это не так, STI накладываются и это вызывает неосновную ошибку процессора S:10/2.
- Убедитесь, что программный сторож процессора продолжает работать при выполнении программы STI.

Важно: Если прерывание происходит во время выполнения инструкции, процессор прерывает выполнение инструкции, однократно сканирует файл прерывания и затем продолжает выполнение инструкции. В действительности STI прозрачно к выполнению программы, если Вы не задали слишком короткое время STI. Слишком короткое время STI может привести к ошибке программного сторожа или значительному увеличению времени сканирования программы.

Онлайн редактирование воздействует на выполнение программы STI. STI не может прервать процессор пока он находится в режиме онлайн редактирования. STI вход должен быть немного больше по времени, чем требуется для выполнения онлайн редактирования. Если это не так, STI не выполнится.

Пример выполнения STI

Периодически осуществляется проверка статуса сети DN+ процессора PLC-5. Статусное слово каждого процессора сравнивается с файлом данных (смотри цепь ниже). Если найдены совпадения, устанавливается бит. Выполняйте это сравнение каждые 800 мс. Имейте в виду, что другой активный шаг восстанавливает статусные данные от PLC-5 процессоров через инструкцию MSG и загружает их во временный файл (N5:10).



Блок-трансферы в файле прерывания по времени

Если процессор запускает файл прерывания по времени, содержащий инструкцию блок-трансфера, процессор выполняет этот блок-трансфер немедленно по завершению передачи текущего блок-трансфера перед запросами блок-трансферов ожидающих в очереди.

Вы можете программировать “немедленный” блок-трансфер для локального шасси Вх/Вых, используя STI программу (т.е., вызывается STI, и блок-трансфер немедленно передается). Процессор немедленно выполняет блок-трансфер, затем заканчивает выполнение остальных цепей STI и возвращается к выполнению релейно-контактной логики программы.

Совет

Установите бит .TO для инструкции блок-трансфера, предназначенный для того же слота, что и блок-трансфер в STI. Этот блок-трансфер выполнится однократно, так что это не задержит завершения программы STI.

Блок-трансфер в программе ошибок или STI возможен только для локальных Вх/Вых. Использование блок-трансферов для удаленных Вх/Вых в программе STI, приведет к ожиданию пользовательской программы завершения выполнения блок-трансфера. Если Вы хотите, чтобы STI завершался до возвращения к вашей главной логической программе, включите пару инструкций UID и UIE в Вашу STI программу.



ВНИМАНИЕ: Когда процессор запускает программу ошибок или STI с инструкцией передачи блока данных для удаленных шасси, MCP продолжает выполняться пока PII или STI ожидают завершения выполнения блока передачи, если используются инструкции UID/UIE.

Назначение прерываний по времени

Для конфигурации прерывания по времени, Вы должны задать:

```

Processor Configuration
User Control Bits      00000000 00000000      RESTART LAST ACTIVE STEP
Fault routine prog file no.: 0              Watchdog (ms): 500
I/O status file:      0              Communication time slice (ms):3

VME Status File:      N34

Processor input interrupt  bit mask: 0
                        compare value: 0      down count: 0
                        prog file no.: 0      module group: 0

Selectable timed interrupt  prog file no.: 0      setpoint (ms): 0
  
```

Для поля:	Выполнить:	Статусный файл:
-----------	------------	-----------------

program file no.	Введите номер программного файла, который будет содержать программу STI.	S:31
------------------	--	------

Если вы не используете STI введите ноль.

set point	Введите временной интервал между прерываниями (1-32767 мсек).	S:30
-----------	---	------

Если Вы не используете STI введите ноль.

Важно: Помните, что время прерывания должно быть больше, чем время выполнения программы STI. Если это не так процессор установит неосновную ошибку (S:10, бит 2).

Например, Вы ввели 7 в S:31 и 15 в S:30. В этом случае процессор будет выполнять релейно-контактный файл 7 каждые 15 миллисекунд.

Вы можете использовать только одну STI одновременно. Однако, Вы можете разрешить или запретить прерывание, изменить файл прерывания или изменить время прерывания. Если необходимо используйте файл релейно-контактной логики для изменения значений слова S:30 и S:31.



ВНИМАНИЕ: STI программа увеличивает время сканирования программы на время равное времени выполнения файла прерывания, умноженного на количество вызовов программы прерывания в течении одного сканирования программы.

Важно: Если Вы запрещаете STI из вашей релейно-контактной программы (записывая 0 в S:30), процессору может потребоваться до 100 миллисекунд для изменения статуса STI. Если Вы запрещаете STI, записывая 0 в S:31, процессор использует значение S:30 для определения, как часто вызывать ненулевое значение S:31.

Монитор прерываний по времени

Используйте вторую страницу экрана Статуса Процессора для контроля STI.

```

Processor Status

Processor input interrupt scan [ms]      last:      0      maximum: 0
events since last interrupt:          0
changed bits                          00000000 00000000

Selectable timed interrupt scan [ms]    last:      0      maximum: 0

Overall program scan [ms]              last:      0      maximum: 0

Main control program
A scan [ms]      last:      0      maximum: 0
B scan [ms]      last:      0      maximum: 0
C scan [ms]      last:      0      maximum: 0
D scan [ms]      last:      0      maximum: 0
E scan [ms]      last:      0      maximum: 0
F scan [ms]      last:      0      maximum: 0
G scan [ms]      last:      0      maximum: 0
H scan [ms]      last:      0      maximum: 0
I scan [ms]      last:      0      maximum: 0

```

Наименование поля	Хранится в:	Отображает:
Last	S:53	постоянное или последнее время сканирования STI
Maximum	S:54	максимальное время сканирования STI

Применение прерываний процессора по входу

Назначение этой главы

Сведения	Страница
Использование прерываний по входу процессора	19-1
Установка прерываний по входу процессора	19-5
Контроль прерываний по входу процессора	19-6

Использование прерывания процессора по входу

Прерывание процессора по входу (processor input interrupt - PI) указывает, когда управляемый событием вход вызывает прерывание выполнения программы процессором и однократный запуск выполнения файла программы PI. Затем, процессор возобновляет выполнение файла программы с того места, где программа была прервана. Используйте прерывания по входу процессора только для входов расположенных в локальном шасси процессора.

Вы можете использовать прерывание процессора по входу (PI) как управляемое событием прерывание или использовать его в приложениях высокоскоростной обработки данных. Например, вам может понадобиться быстро подсчитывать входные данные, чтобы следить за таким производством, как, например, линия консервирования. Или же используйте PI, если ваше приложение должно вызвать немедленное обновление входных данных тогда, когда определено наличие заготовки на конвейере, и вам нужно быстрое обновления выходов для выполнения последующих операций. Например: вам нужно обнаружить и остановить движущуюся по конвейеру деталь, чтобы смонтировать недостающую часть.

Ваша программа PI может содержать инструкции немедленного обновления для завершения высокоскоростных функций управления. Когда выполняется ваша релейная программа и появляются соответствующие входные данные, процессор прерывает выполнение программы и запускает файл программы PI. Затем процессор продолжает выполнение программного файла с того места, где она была прервана.

Совет**Написание релейно-контактной логики ПИ.**

Следуйте этим правилам, когда вы пишете релейную логику для ПИ.

- Храните программу для ПИ как релейный файл.
- Убедитесь, что входное условие (вызывающее прерывание) не наступает чаще, чем время выполнения подпрограммы ПИ. Если появляется второе идентичное условие входных данных до завершения выполнения программы по обработке первого условия входных данных, то возникает перекрытие ПИ, и процессор устанавливает бит неосновной ошибки S:10/12.

Распределение времени для ПИ следующее:

- 1 мс для переключения на приложение ПИ;
- время выполнения релейной логики ПИ;
- 1 мс для возврата к выполнению программы управления.

Так как вам надо выделить не менее 1 мс для выполнения логики ПИ, определите время ПИ по крайней мере 3 мс для исключения перекрытия ПИ.

- Таймер программного сторожа продолжает идти во время выполнения программы ПИ.
- ПИ может обнаружить событие в пределах 100 мкс, но все же вы должны иметь по крайней мере 3 мс между последовательными событиями ПИ.

Примеры ПИ приложения.

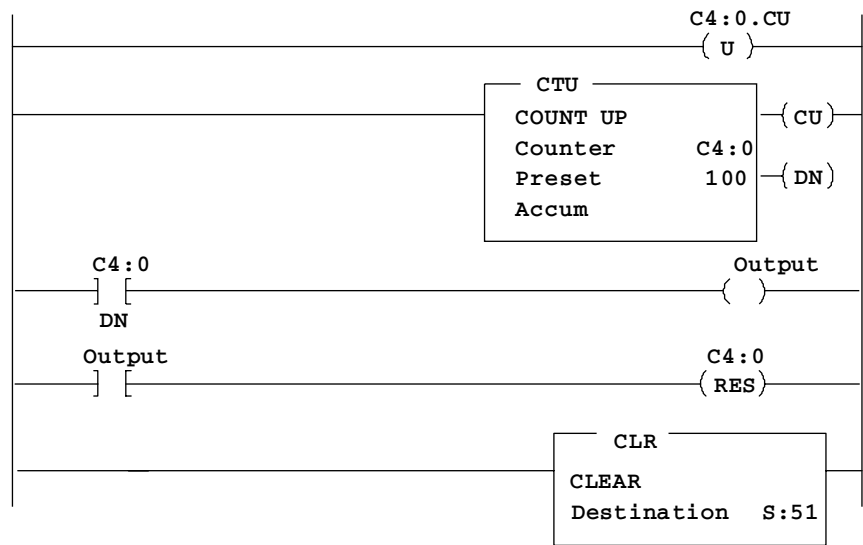
Два способа, которыми вы можете использовать программу ПИ:

Режим	Описание
Счетчик	Применяя режим счетчика, вы используете внутренний счетчик процессора. Вы конфигурируете ПИ предустановленным значением, так что аппаратно подсчитываются входные условия, и, затем, когда накопленное значение будет равняться предустановленному, запускается ПИ. Релейная логика ПИ содержит только необходимые вам выходы.
Изменение бита	Применяя режим по изменению бита, вы настраиваете наступление ПИ каждый раз, когда условие ввода верно. Например, вы хотите вести счет таблеткам, сходящим с конвейера со скоростью 100 таблеток в секунду. Механизм упаковывает по 100 таблеток в пачку. Предполагается, что фотоэлемент определяет проход каждой таблетки.

Программа ПИ (Рис. 19.1) должна:

- считать таблетки группами по 100;
- устанавливать выход на сотой таблетке;
- сбрасывать счетчик для следующей группы.

Рис 19.1
Пример PII программы



Выходной бит остается установленным до наступления следующего отсчета.

Блок-трансфер в прерываниях по вводу (PII)

Если процессор выполняет PII, содержащие инструкции блок-трансфера, то он немедленно выполняет их после завершения каких-либо блок-трансферов в активном на данный момент буфере, но перед остальными запросами на блок-трансферы, стоящими в очереди.

Вы можете запрограммировать «немедленные» блок-трансферы в локальном шасси Вх/Вых применяя программу PII (т.е. PII вызван и блок-трансфер наступает немедленно). Процессор исполняет блок-трансфер немедленно, завершает остальные цепи в PII и возобновляет выполнение релейной программы.

Вы можете использовать PII для блок-трансфера к удаленному Вх/Вых. Инструкции удаленного блок-трансфера в PII вызывают продолжение выполнения программ пользователя процессором, включая STI, во время ожидания завершения блок-трансфера. Если вы хотите, чтобы PII выполнилось полностью до возвращения к вашей основной программе, включите в ваш файл программы PII пару инструкций UID и UIE. Поставьте инструкцию блок-трансфера внутри UID/UIE пары.

Важно: Если прерывания возникают во время выполнения инструкции, процессор останавливает выполнение инструкции, просматривает файл прерывания до конца, затем продолжает выполнение этой инструкции. Фактически, выполнение ПП прозрачно во время выполнения программы, если вы не программируете слишком частые и слишком длительные ПП. Слишком многочисленные ПП могут вызывать срабатывание таймера программного сторожа или вызвать избыточно длительное сканирование программы.

Изменения конфигурации ПП не принимаются до тех пор, пока процессор не перейдет от режима PROGRAM к режиму RUN или TEST.

Совет**Требования при проектировании**

Рассмотрите следующие указания, когда планируете ПП.

- Не используйте 2-ухслотовую адресацию, когда используете ПП.
- Не используйте 1771-IG или -IGD, 8- и 16-точечные TTL модули для ПП. Взамен используйте модуль входа 1771-IQ16. Так как задержка входного фильтра модуля изменяема, вы можете установить задержку в 0 или около 200 мкс.
- Избегайте использования модулей, использующих блок-трансферы, в локальном рэке вместе со сконфигурированным ПП, потому что вы можете пропустить импульс на входе во время работы блок-трансфера. Тем не менее, если вам надо использовать блок-трансферы, убедитесь, что входной импульс ПП не менее 400мкс, так что блок-трансфер не будет оказывать влияния на ПП.
- Редактирование во время выполнения программы влияет на производительность программы ПП. ПП не может вызвать прерывания процессора, пока он распределяет память в связи с редактированием. Входные данные ПП должны быть включены на время, чуть большее чем действительное время, требуемое на завершение редактирования.. Без этого ПП не будет выполняться.
- Очищайте S:51 одним из двух путей:
 - используя CLR инструкцию (см. рисунок 19.1)
 - помещая инструкцию MOV в последней цепи файла ПП. Поместите 0 в S:51 чтобы сбросить биты ПП до завершения файла ПП.

Важно: Если S:51 не очищен, на странице состояния устанавливается бит перекрытия ПП, вызывая неосновную ошибку.

Установка прерывания процессора по входу

Чтобы определить ПИ, используйте экран конфигурации процессора в вашем пакете программирования.

```

Processor Configuration
User Control Bits      00000000 00000000      RESTART LAST ACTIVE STEP
Fault routine prog no.: 0                      Watchdog (ms): 500
I/O status file:      0                      Communication time slice (ms): 3
VME Status File:      N34

Processor input interrupt  prog file no.: 0      module group: 0
                          down count: 0
                          bit mask: 00000000 00000000
                          compare value: 00000000 00000000
    
```

В этом поле конфигурации ПИ:	Сделайте следующее:	Адрес в файле состояния:
prog file no.	Введите номер программного файла, содержащего программу ПИ. Это единственный параметр ПИ, который вы можете изменить во время того, когда процессор находится в режиме RUN.	S:46
module group	Введите номер назначенного рэка и номер группы входа для контроля (например 21 для рэка 2, группы 1). Не вводите адрес. (Только для входов на локальном шасси процессора). Если указанный номер слова находится не в локальном рэке или если в адресованном слоте нет модуля входа, то при смене режима будет установлен бит неосновной ошибки (S:10/11).	S:47
bit mask	Каждая группа модуля (указанная в S:47) содержит бит управления, служащий для отображения входного бита. <ul style="list-style-type: none"> • Чтобы контролировать бит, введите 1. • Чтобы проигнорировать бит, введите 0. 	S:48
compare value	Каждая группа модуля (указанная в S:47) содержит бит, который используется при управлении ПИ через битовый переход. <ul style="list-style-type: none"> • Для счета переходов от ложного значения к истинному (триггер бита), введите 1. • Для счета переходов от истинного значения к ложному (триггер события), введите 0. 	S:49
down count	Введите предустановленное значение для определения количества наступления условий до возникновения прерывания. Возможные значения 0 - 32767. Если вы хотите чтобы прерывание возникало каждый раз, введите 0 или 1.	S:50

Важно: Если вы изменяете конфигурацию ПИ находясь в режиме RUN, то вы должны переключить режим в режим PROGRAM и затем обратно в режим RUN, для того чтобы изменения вошли в силу.

Контроль прерываний процессора по входу

Используйте экран состояния процессора в вашем пакете программирования для отображения PPIs.

Processor Status					
Processor input interrupt scan [ms]	last:	0	maximum:	0	
events since last interrupt:	0				
changed bits	00000000 00000000				
Selectable timed interrupt scan [ms]	last:	0	maximum:	0	
Overall program scan [ms]	last:	0	maximum:	0	
Main control program	A scan [ms]	last:	0	maximum:	0
	B scan [ms]	last:	0	maximum:	0
	C scan [ms]	last:	0	maximum:	0
	D scan [ms]	last:	0	maximum:	0
	E scan [ms]	last:	0	maximum:	0
	F scan [ms]	last:	0	maximum:	0
	G scan [ms]	last:	0	maximum:	0
	H scan [ms]	last:	0	maximum:	0
	I scan [ms]	last:	0	maximum:	0

Этот адрес:	Хранит:	Описание:
S:51	Возвращаемую маску (измененные биты)	<p>Отображает битовые переходы, которые вызвали прерывание. Если бит равен 1, то со времени последнего прерывания бит уже сменился. Если бит равен 0, то он не менялся со времени последнего прерывания. Вы можете использовать эту информацию для обуславливания других цепей в вашей релейной программе.</p> <p>Важно: если один из этих битов уже установлен (т.е. предыдущее прерывание установило бит), то процессор устанавливает неосновную ошибку (S:10/12), чтобы отобразить возможное перекрытие PPI.</p> <p>Важно: если вы хотите контролировать это перекрытие, убедитесь, что последняя цепь в вашей программе PPI очищает маску возврата в файле состояния.</p>
S:52	Накопитель (событий)	<p>Отображает количество условий наступивших до прерывания. Это значение должно совпадать со значением в поле счетчика в вашей конфигурационной установке. Если они не совпадают:</p> <ul style="list-style-type: none"> • прерывания не выполняются когда должны • вы пытаетесь сосчитать слишком много событий • события случаются слишком быстро, чтобы сосчитать
S:55	Время сканирования (последнее)	Отображает текущее или последнее время сканирования подпрограммы PPI.
S:56	Время сканирования (максимальное)	Отображает максимальное из значений, которые были отображены в предыдущем поле просмотра.

Используйте S:51/0-15 внутри файла PPI потому, что эти биты:

- отображают действительное состояние входов модуля, которое было использовано для PPI;
- надежно сохраняют состояние.

Чтобы подпрограмма PPI выполнялась правильно, не используйте адреса входных битов модуля внутри этой программы.

Характеристики системы

Характеристики процессора

Потребляемый ток	PLC-5/11, -5/20, -5/26, -5/30	2.3 А
	PLC-5/40, -5/46, -5/40L, -5/60, -5/60L, -5/80, -5/86	3.3 А
	PLC-5/20E, -5/40E, -5/80E	3.6 А
Рассеиваемая теплота	PLC-5/11, -5/20, -5/26, -5/30	12.10 Вт
	PLC-5/40, -5/46, -5/40L, -5/60, -5/60L, -5/80, -5/86	17.30 Вт
	PLC-5/20E, -5/40E, -5/80E	18.00 Вт
Условия окружающей среды	Рабочая температура	0 ... 60° С
	Температура хранения	-40 ... 85° С
	Относительная влажность	5 ... 95% (без конденсации)
Удар	Работающий	пиковое ускорение 30 g в течение 11±1 мс
	Неработающий	пиковое ускорение 50 g в течение 11±1 мс
Вибрация (работающий и неработающий)	1 g при частоте 10 ... 500 Hz амплитуда 0.3 мм	
Часы и календарь	Максимальная погрешность при 60° С	±5 мин в месяц
	Типичная погрешность при 20° С	±20 с в месяц
	Точность отсчета выдержки времени	1 сканирование программы
Батарея	1770-ХУС	
Модули памяти	1785-МЕ16	1785-МЕ64
	1785-МЕ32	1785-М100
Типичное время сканирования дискретных Вх/Вых	<ul style="list-style-type: none"> • 0.5 мс / расширенный локальный Вх/Вых • 10 мс / удаленный Вх/Вых, при скорости связи адаптера 57.6 кбит/с • 7 мс / удаленный Вх/Вых, при скорости связи адаптера 115.2 кбит/с • 3 мс / удаленный Вх/Вых, при скорости связи адаптера 230.4 кбит/с 	
Модули Вх/Вых	Бюллетень 1771, включая 8-, 16-, 32-х точечные и интеллектуальные модули	

Аппаратная адресация	<p>2 слотовая</p> <ul style="list-style-type: none"> Любое расположение 8-ми точечных модулей 16-ти точечные модули должны быть парами Вх/Вых Нет 32-х точечных модулей <p>1 слотовая</p> <ul style="list-style-type: none"> любое сочетание 8-ми или 16-ти точечных модулей 32-х точечные модули должны быть парами Вх/Вых <p>1/2 слотовая – любое сочетание 8-, 16-, 32-х точечных модулей</p>																
Связь	<ul style="list-style-type: none"> DH+ DH при использовании 1785-КА Последовательная Ethernet (TCP/IP протокол, 15-ти контактный АUI порт приемопередатчика) Удаленные Вх/Вых (Remote I/O) Расширенные локальные Вх/Вых (только для процессоров PLC-5/40L и -5/60L) 																
Расположение	Шасси 1771-A1В, -A2В, -A3В, -A3В1, -A4В, крайний левый слот																
Ключи	<ul style="list-style-type: none"> Между 40 и 42 Между 54 и 56 																
Вес	<table> <tbody> <tr> <td>PLC-5/20, -5/26</td> <td>1.21 кг</td> </tr> <tr> <td>PLC-5/30</td> <td>1.20 кг</td> </tr> <tr> <td>PLC-5/40, -5/46, -5/40L</td> <td>1.42 кг</td> </tr> <tr> <td>PLC-5/60, -5/60L</td> <td>1.42 кг</td> </tr> <tr> <td>PLC-5/80, -5/86</td> <td>1.42 кг</td> </tr> <tr> <td>PLC-5/20E</td> <td>1.43 кг</td> </tr> <tr> <td>PLC-5/40E</td> <td>1.39 кг</td> </tr> <tr> <td>PLC-5/80E</td> <td>1.38 кг</td> </tr> </tbody> </table>	PLC-5/20, -5/26	1.21 кг	PLC-5/30	1.20 кг	PLC-5/40, -5/46, -5/40L	1.42 кг	PLC-5/60, -5/60L	1.42 кг	PLC-5/80, -5/86	1.42 кг	PLC-5/20E	1.43 кг	PLC-5/40E	1.39 кг	PLC-5/80E	1.38 кг
PLC-5/20, -5/26	1.21 кг																
PLC-5/30	1.20 кг																
PLC-5/40, -5/46, -5/40L	1.42 кг																
PLC-5/60, -5/60L	1.42 кг																
PLC-5/80, -5/86	1.42 кг																
PLC-5/20E	1.43 кг																
PLC-5/40E	1.39 кг																
PLC-5/80E	1.38 кг																
Сертификация агентств (если есть отметка на продукте)	<ul style="list-style-type: none"> CSA – класс I, раздел 2, группы А, В, С, D UL – указан CE – отмечен для всех применимых указаний 																

Характеристики процессора (продолжение)

Процессор (№ по кат.)	Максимальное количество слов памяти пользователя	Общее максимальное количество Вх/Вых (любое сочетание)	Типы портов связи	Максимальное количество рэков Вх/Вых (адреса рэков)	Максимальное количество шасси Вх/Вых		
					Всего	Расш.	Удал.
PLC-5/11 (1785-L11B)	8 К	<ul style="list-style-type: none"> 512 (л. соч.) или 384 вх + 384 вых (комплементарных) 	<ul style="list-style-type: none"> 1 DH+/Remote I/O (адаптер или сканер) 1 посл. порт, конфигурируемый как RS-232, -423 и -422A совместимый 	4 (0-3)	5	0	4 (должен быть рэк 3)
PLC-5/20 (1785-L20B) PLC-5/26 (1785-L26B)	16 К	<ul style="list-style-type: none"> 512 (л. соч.) или 512 вх + 512 вых (комплементарных) 	<ul style="list-style-type: none"> 1 DH+ (фиксированный) 1 DH+/Remote I/O (адаптер или сканер) 1 посл. порт, конфигурируемый как RS-232, -423 и -422A совместимый 	4 (0-3)	13	0	12
PLC-5/20E (1785-L20E)	16 К	<ul style="list-style-type: none"> 512 (л. соч.) или 512 вх + 512 вых (комплементарных) 	<ul style="list-style-type: none"> 1 DH+ (фиксированный) 1 DH+/Remote I/O (адаптер или сканер) 1 посл. порт, конфигурируемый как RS-232, -423 и -422A совместимый 1 канал Ethernet 	4 (0-3)	13	0	12
PLC-5/30 (1785-L30B)	32 К	<ul style="list-style-type: none"> 1024 (л. соч.) или 1024 вх + 1024 вых (комплементарных) 	<ul style="list-style-type: none"> 2 DH+/Remote I/O (адаптер или сканер) 1 посл. порт, конфигурируемый как RS-232, -423 и -422A совместимый 	8 (0-7)	29	0	28
PLC-5/40 (1785-40B) PLC-5/46 (1785-L46B)	48 К ¹	<ul style="list-style-type: none"> 2048 (л. соч.) или 2048 вх + 2048 вых (комплементарных) 	<ul style="list-style-type: none"> 4 DH+/Remote I/O (адаптер или сканер) 1 посл. порт, конфигурируемый как RS-232, -423 и -422A совместимый 	16 (0-17)	61	0	60
PLC-5/40E (1785-L40E)	48 К ¹	<ul style="list-style-type: none"> 2048 (л. соч.) или 2048 вх + 2048 вых (комплементарных) 	<ul style="list-style-type: none"> 2 DH+/Remote I/O (адаптер или сканер) 1 посл. порт, конфигурируемый как RS-232, -423 и -422A совместимый 1 канал Ethernet 	16 (0-17)	61	0	60
PLC-5/40L (1785-L40L)	48 К ¹	<ul style="list-style-type: none"> 2048 (л. соч.) или 2048 вх + 2048 вых (комплементарных) 	<ul style="list-style-type: none"> 2 DH+/Remote I/O (адаптер или сканер) 1 посл. порт, конфигурируемый как RS-232, -423 и -422A совместимый 1 расширенный Вх/Вых 	16 (0-17)	61	16	60
PLC-5/60 (1785-L40B)	64 К ²	<ul style="list-style-type: none"> 3072 (л. соч.) или 3072 вх + 3072 вых (комплементарных) 	<ul style="list-style-type: none"> 4 DH+/Remote I/O (адаптер или сканер) 1 посл. порт, конфигурируемый как RS-232, -423 и -422A совместимый 	24 (0-27)	93	0	92
PLC-5/60L (1785-L60L)	64 К ²	<ul style="list-style-type: none"> 3072 (л. соч.) или 3072 вх + 3072 вых (комплементарных) 	<ul style="list-style-type: none"> 2 DH+/Remote I/O (адаптер или сканер) 1 посл. порт, конфигурируемый как RS-232, -423 и -422A совместимый 1 расширенный Вх/Вых 	24 (0-27)	81	16	64

Процессор (№ по кат.)	Максимальное количество слов памяти пользователя	Общее максимальное количество Вх/Вых (любое сочетание)	Типы портов связи	Максимальное количество рэков Вх/Вых (адреса рэков)	Максимальное количество шасси Вх/Вых		
					Всего	Расш.	Удал.
PLC-5/80 (1785-L80B) PLC-5/86 (1785-L86B)	100 К ³	<ul style="list-style-type: none"> 3072 (л. соч.) или 3072 вх + 3072 вых (комплементарных) 	<ul style="list-style-type: none"> 1 DH+/Remote I/O (адаптер или сканер) 1 посл. порт, конфигурируемый как RS-232, -423 и -422A совместимый 	24 (0-27)	93	0	92
PLC-5/80E (1785-L80E)	100 К ³	<ul style="list-style-type: none"> 3072 (л. соч.) или 3072 вх + 3072 вых (комплементарных) 	<ul style="list-style-type: none"> 2 DH+/Remote I/O (адаптер или сканер) 1 посл. порт, конфигурируемый как RS-232, -423 и -422A совместимый 1 канал Ethernet 	24 (0-27)	65	0	64

¹ Процессоры PLC-5/40, -5/40E, -5/40L имеют ограничение в 32 К слов для файла таблицы данных.

² Процессоры PLC-5/60, -5/60L имеют ограничение в 56 К слов для программного файла и в 32 К слов для файла таблицы данных.

³ Процессоры PLC-5/80, -5/80E имеют всего 64 К слов пространства для файлов таблиц данных с ограничением в 56 К слов для программного файла и в 32 К слов для файла таблиц данных.

Характеристики батареи (1770-ХУС)

При использовании батареи в процессоре:	При температуре:	Оценка наименьшего срока службы батареи		Срок службы батареи после загорания светодиодного индикатора ¹
		Отключен 100%:	Отключен 50%:	
PLC-5/11, -5/20 и -5/20E	60° С	256 дней	1.4 года	11.5 дней
	25° С	2 года	4 года	47 дней
PLC-5/30, -5/40, -5/40L, -5/40E, -5/60, -5/60L, -5/80 и -5/80E	60° С	84 дня	150 дней	5 дней
	25° С	1 год	1.2 года	30 дней

¹ Индикатор батареи (BATT) предупреждает вас о разрядке батареи. Эти сроки основываются при работе только от батареи (источник питания шасси отключен) с момента первого загорания светодиода.

Файл состояния процессора



Данные по состоянию процессора хранятся в файле данных-2.

Важно: Для дополнительной информации по любой из этих тем обращайтесь к описанию в руководстве или к документации по вашему п/о программирования.

S:0 - S:2

Слово файла состояния	Описание
S:0	Арифметические флаги: <ul style="list-style-type: none"> • бит 0 = заем • бит 1 = переполнение • бит 2 = ноль • бит 3 = знак
S:1	Состояние и флаги процессора
S:1/00	Контрольная сумма ОЗУ неверна при включении
S:1/01	Процессор в режиме RUN
S:1/02	Процессор в режиме TEST
S:1/03	Процессор в режиме PROGRAM
S:1/04	Процессор загружает данные в модуль памяти
S:1/05	Процессор в режиме загрузки программы
S:1/06	Процессор имеет изменения в режиме тестирования
S:1/07	Переключатель выбора режимов в положении REMOTE
S:1/08	Форсировка разрешена
S:1/09	Форсировка присутствует
S:1/10	Процессор успешно загрузил данные в модуль памяти
S:1/11	Выполняется программирование в режиме ONLINE
S:1/12	Не определено
S:1/13	Контрольная сумма программы вычислена
S:1/14	Последнее сканирование реленой логики или шага ПФС
S:1/15	Процессор выполняет первое сканирование программы или выполняет первое сканирование очередного шага в SFC

Слово файла состояния	Описание
S:2	Информация по установке переключателей
S:2/00	Канал 1A DN+ номер станции
по S:2/05	
S:2/06	Канала DN+ скорость передачи данных 0 57.6 кбит/сек 1 230.4 кбит/сек
S:2/07	Не определены
S:2/08	
S:2/09	Последнее состояние 0 выходы выключены 1 выходы сохраняют последнее состояние
S:2/11	Адресация Вх/Вых шасси
S:2/12	<u>бит 12 бит 11</u> 0 0 запрещенное состояние 1 0 1/2 слота 0 1 1 слот 1 1 2 слота
S:2/13	Перемещение модуля памяти
S:2/14	<u>бит 14 бит 13</u> 0 0 модуль памяти пересылает данные в память процессора, если память процессора недействительна 0 1 модуль памяти не пересылает данные в память процессора 1 1 модуль памяти пересылает данные в память процессора при включении питания
S:2/15	Защита памяти процессора 0 разрешена 1 запрещена

S:3 - S:10

**Это слово
файла
состояния:**

Хранит:

S:3 по S:6	Таблица активных узлов для канала 1A
	<u>Слово Биты DN+ номер станции</u>
	3 0-15 00-17
	4 0-15 20-37
	5 0-15 40-57
	6 0-15 60-77
S:7	Основные биты состояния: (смотри также S:27, S:32, S:33, S:34, и S:35) • S:7/0-7 биты ошибки река для рэков 0-7 • S:7/8-15 не используются

Слово файла состояния	Описание
S:8	Последнее сканирование программы (в мс)
S:9	Максимальное сканирование программы (в мс)
S:10	Неосновная ошибка (слово 1) Смотри также S:17
S:10/00	Батарея разряжена (замените в течение 1-2 дней)
S:10/01	Таблица активных узлов DH+ изменена
S:10/02	STI задержка слишком мала, перекрытие программы прерываний
S:10/03	При включении питания модуль памяти переслал программу
S:10/04	Изменения не позволяют продолжение ПФС; размер таблицы данных изменился во время режима PROGRAM; сбрасывается автоматически во время режима RUN
S:10/05	Неверный файл состояния Vx/Вых
S:10/06	Зарезервировано
S:10/07	Больше не существует блоков управления чтобы осуществить блок-трансфер
S:10/08	Недостаточно памяти на модуле памяти для записи программы процессора
S:10/09	Не сконфигурировано ни одного MCP для выполнения
S:10/10	MCP не допускается
S:10/11	Номер слова PII не в локальном рэке
S:10/12	Перекрытие PII
S:10/13	Не существует блоков управления для получения PII
S:10/14	Арифмитическое переполнение
S:10/15	Перекрытие процесса «затянувшегося» ПФС – шаг уже был активен, когда он был ре-активирован

S11

Слово файла состояния:	Описание
S:11	Слово основной ошибки
S:11/00	Разрушен файл программы (коды 10-19). Смотри коды основных ошибок (S:12).
S:11/01	Неверный адрес в релейно-контактной программе (коды 20-29). Смотри коды основных ошибок (S:12).
S:11/02	Ошибка программирования (коды 30-49). Смотри коды основных ошибок (S:12).
S:11/03	Процессор обнаружил ошибку ПФС (коды 71-79). Смотри коды основных ошибок (S:12).
S:11/04	Процессор обнаружил ошибку пока ассемблировал файл релейно-контактной программы (код 70); обнаружено дублирование меток.
S:11/05	Ошибка защиты при запуске. Процессор устанавливает этот бит основной ошибки при включении в режиме RUN, если установлен пользовательский бит управления S:26/1.
S:11/06	Ошибка периферийного устройства
S:11/07	Ошибка созданная пользователем; процессор перешел на подпрограмму обработки ошибки (коды 0-9). Смотри коды основных ошибок (S:12).
S:11/08	Ошибка таймера программно сторожа
S:11/09	Система неправильно сконфигурирована (коды 80-89). Смотри коды основных ошибок (S:12).
S:11/10	Устранимая ошибка оборудования
S:11/11	МСП не существует, либо не релейно-контактная или ПФС
S:11/12	Файл PII не существует, либо он не релейно-контактный
S:11/13	STI файл не существует, либо он не релейно-контактный файл
S:11/14	Подпрограмма обработки ошибки не существует, либо она не релейно-контактный файл
S:11/15	Программа, в которой произошла ошибка, не содержит релейно-контактной логики.

S:12

Это слово хранит следующие коды ошибки:

Код ошибки	Описание	Ошибка	
00-09	<p>Зарезервирован для кодов ошибки, определенных пользователем.</p> <p>Вы можете использовать определяемые пользователем коды ошибок для распознавания различных типов ошибок или условий приведших к ошибке в вашей программе с помощью создания вашей собственной устранимой ошибки. Для использования этих кодов ошибки выберите условие входа, которое установит нужно ли перейти к файлу подпрограммы обработки ошибок, затем, примените инструкцию JSR как средство перехода к файлу подпрограммы обработки ошибок.</p> <p>Для использования инструкции JSR введите номер кода ошибки 0-9 (непосредственное значение) как первый входной параметр инструкции. Любые другие входные параметры игнорируются (даже если у вас есть инструкция SBR в начале подпрограммы обработки ошибок. Используя инструкции JSR/SBR, вы не можете передать параметры подпрограмме обработки ошибок).</p> <p>Вы не обязаны использовать коды ошибки, определенные пользователем для создания вашей собственной ошибки. Если вы программируете JSR без входных параметров, процессор запишет ноль в поле кода ошибки. Цель применения кодов ошибки, определенных пользователем – позволить вам различать различные типы ошибок или коды ошибок, основанные на номерах кода ошибки (0-9).</p> <p>Когда входное условие верно, процессор копирует номер кода ошибки, введенного как первый входной параметр JSR инструкции в 12 слово файла состояния процессора (S:12), являющегося полем кода ошибки. Процессор устанавливает основную ошибку S:11/7 «Созданная пользователем ошибка». Процессор затем остается в состоянии ошибки, пока вы не очистите слово основной ошибки (S:11) или специфический бит ошибки с помощью релейно-контактной логики в подпрограмме обработки ошибки.</p>	<p><i>Устраняемая:</i></p> <p>подпрограмма обработки ошибок может указать процессору очистить ошибку и, затем, возобновить сканирование программы.</p> <p>Подпрограмма обработки ошибки выполняется тогда, когда возникает любая из этих ошибок.</p>	
10	Проверка таблицы данных во время выполнения программы неудачна	<p><i>Устраняемая:</i></p> <p>подпрограмма обработки ошибок может указать процессору очистить ошибку и, затем, возобновить сканирование программы.</p> <p>Подпрограмма обработки ошибки выполняется тогда, когда возникает любая из этих ошибок.</p>	
11	Несовпадающая контрольная сумма программы пользователя		
12	Неправильный тип целого операнда, восстановить новый файл памяти процессора		
13	Неправильный тип операции смешанного режима, восстановить новый файл памяти процессора		
14	Недостаточно операндов для инструкции, восстановить новый файл памяти процессора		
15	Слишком много операндов для инструкции, восстановить новый файл памяти процессора		
16	Испорченная инструкция, вероятно из-за восстановления несовместимого файла памяти процессора (неправильный код операции)		
17	Конец выражения не найден, восстановить новый файл памяти процессора		
18	Нет завершения зоны редактирования, восстановить новый файл памяти процессора		
19	Загрузка программы прервана		
20	Вы ввели слишком большой номер элемента в не прямой адресации		
21	Вы ввели отрицательный номер элемента в не прямой адресации		
22	Вы пытались доступиться к несуществующему файлу программы		
23	Вы использовали отрицательный номер файла, вы использовали номер файла больше, чем количество существующих файлов, или вы пытались непрямо адресоваться к файлам 0, 1, или 2		
24	Вы пытались непрямо адресоваться к файлу неверного типа		
30	Вы пытались перейти к подпрограмме со слишком большой глубиной вложения вызовов		<p>Неустраняемая:</p> <p>подпрограмма обработки ошибки будет выполнена, но не может очистить бит 2 основной ошибки.</p>
31	Вы ввели недостаточно параметров подпрограммы		
32	Вы перешли к неверному (не релейно-контактному) файлу		
33	Вы ввели подпрограмму CAR, которая не представляет собой коды процессора 68000		
34	Вы ввели отрицательное предустановленное, либо накопленное значение в инструкции таймера		
35	Вы ввели отрицательное значение переменной времени в инструкции PID	<p><i>Устраняемая</i></p>	
36	Вы ввели уставку (setpoint) за пределами диапазона в инструкции PID		
37	Вы обратились к неверному модулю в блок-трансфере, инструкции немедленного ввода или инструкции немедленного вывода		

Код ошибки	Описание	Ошибка
38	Вы ввели инструкцию RET в файле, который не является подпрограммой	Неустраняемая
39	Инструкция FOR с отсутствующей инструкцией NXT	
40	Файл управления слишком мал для инструкций PID, BTR, BTW, или MSG	Устраняемая
41	Инструкция NXT с отсутствующей инструкцией FOR	Неустраняемая
42	Вы пытались перейти к несуществующей метке	
43	Не ПФС файл	
44	Ошибка в использовании ПФС, эта ошибка возникает, если: <ul style="list-style-type: none"> • вы пытались осуществить сброс в одновременный путь; • вы указали номер ссылки к найденному шагу или этот номер не привязан к шагу (это переход); • предыдущий ПФС на другой шаг не завершен. 	
45	Введен неверный номер канала	Устраняемая
46-69	Зарезервировано	
70	Процессор обнаружил дублирующиеся метки	
71	Процессор пытался запустить уже запущенную подсхему ПФС	
72	Процессор пытался остановить незапущенную подсхему ПФС	
73	Процессор пытался запустить больше, чем позволено, подсхем	
74	Обнаружена ошибка файла ПФС	
75	У ПФС слишком много активных функций	
76	Шаг ПФС зацикливается сам на себе	
77	ПФС ссылается на шаг, переход, подсхему или на SC файл, который отсутствует, пуст или слишком мал	
78	Процессор не может продолжить выполнение ПФС после потери питания	
79	Вы пытались загрузить ПФС в процессор, который не может выполнять ПФС	
80	У вас есть ошибка в конфигурации Вх/Вых	Неустраняемая
81	Вы неправильно установили переключатели на плате шасси Вх/Вых, установив оба переключателя 4 и 5 в ON	
82	Неверный тип картриджа для выбранной операции. Эта ошибка возникает также если у процессора нет модуля памяти, но переключатели на плате шасси установлены для использования этих модулей. Убедитесь, что переключатели в верном положении (установите переключатель 6 в ON и переключатель 7 в OFF, если у процессора нет модуля памяти).	Неустраняемая
83	Ошибка программного сторожа	Неустраняемая
84	Ошибка блок-трансфера в конфигурируемом пользователем режиме адаптера	
85	Плохой модуль памяти	
86	Модуль памяти несовместим с хостом	
87	Перекрытие в списке просмотра рэков	
88	Каналы просмотра перегружают удаленный буфер Вх/Вых; слишком много данных для выполнения процессором. Если вы встретите код ошибки 88, проверьте, следовали ли вы пунктам руководства на странице 4-9. В особенности, убедитесь, что вы: <ul style="list-style-type: none"> • сгруппировали вместе 1/4-рэки и 1/2-рэки каждого логического рэка, не разбивайте их по различным рэкам; • при использовании адресации к дополнительным Вх/Вых и группировке рэков обрабатываете индивидуально адреса дополнительных рэков; номера основных рэков отдельно от номеров дополнительных рэков. 	

Код ошибки	Описание	Ошибка
90	Расширенный тест памяти Sidescar модуля не прошел. Обратитесь к вашему представителю Allen-Bradley за обслуживанием	Устраняемая
91	Не определен тип сообщения Sidescar модуля	
92	Sidescar модуль запрашивает неопределенный пул	
93	Неверный максимальный размер пула Sidescar модуля	
94	Неверное ASCII сообщение Sidescar модуля	
95	Sidescar модуль выдал ошибку, которая может быть результатом неправильной программы Sidescar или аппаратной ошибки	
96	Sidescar модуль физически не подсоединен к процессору PLC-5	
97	Sidescar модуль запросил размер пула, слишком малый для команд PC (наступает при включении)	
98	Возникла ошибка при тестировании первых/последних 16 байт ОЗУ Sidescar модуля	
99	Ошибка пересылки данных из Sidescar модуля в процессор	
100	Ошибка пересылки данных из процессора в Sidescar модуль	
101	Ошибка пересылки данных по окончании сканирования в Sidescar модуль	Устраняемая
102	Значение указанного номера файла для пересылки сырых данных через Sidescar модуль неверно	Устраняемая
103	Значение номера элемента для пересылки сырых данных через Sidescar модуль неверно	
104	Запрошенный размер пересылки данных через Sidescar модуль неверен	
105	Значение смещения в сегменте пересылки сырых данных Sidescar модуля неверно	
106	Нарушение защиты пересылки Sidescar модуля; только для процессоров PLC-5/26, -5/46 и -5/86	
200	Отсутствует пересылка выхода ControlNet	
201	Отсутствуют входные данные ControlNet	
202	Отсутствуют данные по диагностике ControlNet	
203	Переполнение расписания передачи данных ControlNet	
204	Слишком сложная для процессора PLC-5 конфигурация ControlNet	
205	Конфигурация ControlNet превысила ширину диапазона PLC-5	

S:13 - S:24

Слово файла состояния	Описание
S:13	Программный файл, в котором наступила ошибка
S:14	Номер цепи, в которой наступила ошибка
S:15	Файл состояния VME
S:16	Файл состояния Вх/Вых

Слово файла состояния	Описание
S:17	Неосновная ошибка (слово 2). Смотри также S:10.
S:17/00	Очередь блок-трансферов к удаленным Вх/Вых полна
S:17/01	Очередь полна - канал 1A; использован максимум удаленных блок-трансферов
S:17/02	Очередь полна - канал 1B; использован максимум удаленных блок-трансферов
S:17/03	Очередь полна - канал 2A; использован максимум удаленных блок-трансферов
S:17/04	Очередь полна - канал 2B; использован максимум удаленных блок-трансферов
S:17/05	Нет модема на последовательном порте
S:17/06	<ul style="list-style-type: none"> Удаленный рэк Вх/Вых в таблице локальных рэков, или удаленный рэк Вх/Вых больше, чем размер образа. Эта ошибка может быть вызвана также локальным рэком, если он установлен на восьмиричную плотность сканирования, и каждая из таблиц образа Вх/Вых меньше, чем 64 слова (8 рэков).
S:17/07	Ревизия для пар каналов 1A/1B или 2A/2B не совпадает с ревизией процессора
S:17/08	Ошибка ASCII инструкции
S:17/09	Дублирующиеся адреса узлов
S:17/10	Ошибка в списке опроса мастера DF1
S:17/11	Нарушение в элементе таблицы защищенных данных процессора
S:17/12	Нарушение в защищенном файле процессора
S:17/13	Используются все 32 ControlNet MSG
S:17/14	Используются все 32 ControlNet 1771 READ и/или 1771 WRITE CIO
S:17/15	Используются все 8 ControlNet Flex I/O CIO
S:18	Часы процессора: год
S:19	Часы процессора: месяц
S:20	Часы процессора: день
S:21	Часы процессора: Час
S:22	Часы процессора: минута
S:23	Часы процессора: секунда
S:24	Смещение индексной адресации
S:25	Зарезервировано

S:26 - S:35

Слово файла состояния	Описание
S:26	Пользовательские биты управления
S:26/00	Перезапустить/продолжить ПФС: когда сброшен, процессор перезапускается на первом шаге ПФС. Когда установлен, процессор после потери питания или изменения на RUN продолжает с последнего активного шага.
S:26/01	Защита при запуске после потери питания: когда сброшен, защиты нет. Когда установлен, процессор устанавливает бит основной ошибки S:11/5 при включении в режиме RUN.
S:26/02	Определяет адрес локального рэка: когда сброшен, адрес локального рэка равен 0. Когда установлен, адрес локального рэка равен 1.
S:26/03	Устанавливает дополнительные Вх/Вых (только для серии А): когда сброшен дополнительные Вх/Вых не разрешены. Когда установлен дополнительные Вх/Вых разрешены.
S:26/04	Бит совместимости локальных блок-трансферов: когда сброшен, обычная работа. При установленном значении убираются частые ошибки контрольной суммы на определенных блок-трансфер модулях.
S:26/05	Бит совместимости со сканером PLC-3: когда установлен (1), отклик канала адаптера задержан на 1 мс; когда сброшен (0) работа с обычным временем отклика.
S:26/06	Бит запрета изменения таблиц данных. Когда установлен (1), пользователь не может редактировать таблицу данных или изменять форсировки, пока ключ процессора находится в положении RUN. Вы управляете этим битом с помощью вашего пакета программирования
S:26/07-15	Не определены
S:27	Биты управления рэком: (смотри также S:7, S:32, S:33, S:34, и S:35) <ul style="list-style-type: none"> • S:27/0-7 – биты запрета рэка Вх/Вых (для рэков 0-7) • S:27/8-15 – биты сброса рэка Вх/Вых (для рэков 0-7)
S:28	Уставка времени превышения цикла программы (программного сторожа)
S:29	Файл подпрограммы обработки ошибки
S:30	Уставка STI
S:31	Номер STI файла
S:32	Основные биты состояния: (смотри также S:7, S:27, S:33, S:34, и S:35) <ul style="list-style-type: none"> • S:32/0-7 биты ошибок рэка для рэков 10-17 (восьмеричное) • S:32/8-15 не используются
S:33	Биты управления рэком: (смотри также S:7, S:27, S:32, S:34, и S:35) <ul style="list-style-type: none"> • S:33/0-7 биты запрета рэка Вх/Вых (для рэков 10-17) • S:33/8-15 биты сброса рэка Вх/Вых (для рэков 10-17)
S:34	Основные биты состояния: (смотри также S:7, S:27, S:32, S:33, и S:35) <ul style="list-style-type: none"> • S:34/0-7 биты ошибок рэка для рэков 20-27 (восьмеричное) • S:34/8-15 не используются
S:35	Биты управления рэком: (смотри также S:7, S:27, S:32, S:33, и S:34) <ul style="list-style-type: none"> • S:35/0-7 биты запрета рэка Вх/Вых (для рэков 20-27) • S:35/8-15 биты сброса рэка Вх/Вых (для рэков 20-27)

S:36 - S:78

Слово файла состояния	Описание
S:36-S:45	Зарезервированы
S:46	Номер файла программы PII
S:47	Группа модуля PII
S:48	Битовая маска PII
S:49	Значение сравнения PII
S:50	Обратный отсчет PII
S:51	Измененный бит PII
S:52	События PII со времени последнего прерывания
S:53	Время сканирования STI (в мс)
S:54	Максимальное время сканирования STI (в мс)
S:55	Последнее время сканирования PII (в мс)
S:56	Максимальное время сканирования PII (в мс)
S:57	Контрольная сумма пользовательской программы
S:58	Зарезервировано
S:59	Сканирование пересылки дискретных данных расширенного локального канала Вх/Вых (в мс)
S:60	Максимальное сканирование дискретных данных расширенного локального канала Вх/Вых (в мс)
S:61	Сканирование блок-трансферов расширенного локального канала Вх/Вых (в мс)
S:62	Максимальное сканирование блок-трансферов расширенного локального канала Вх/Вых (в мс)
S:63	Номер файла защиты для защищенной таблицы данных процессора
S:64	Количество командных блоков удаленных блок-трансферов, используемых парой каналов 1А/1В.
S:65	Количество командных блоков удаленных блок-трансферов, используемых парой каналов 2А/2В.
S:66	Зарезервировано
S:77	Часть времени связи для системных функций связи (в мс)
S:78	Биты запрета обновления Вх/Вых между МСР Бит 0 для МСР А Бит 1 для МСР В и т.д.

S:79 - S:127

Слово файла состояния	Описание
S:79	Биты запрета МСР Бит 0 для МСР А Бит 1 для МСР В и т.д.
S:49-S:127	Номер файла МСР Время сканирования МСР (в мс) Максимальное время сканирования МСР (в мс) Вышеуказанная последовательность применяется к каждой МСР; следовательно, каждое МСР содержит 3 слова состояния. Например, слово 80: номер файла для МСР А слово 81: время сканирования для МСР А слово 82: максимальное время сканирования для МСР А слово 83: номер файла для МСР В слово 84: время сканирования для МСР В и т.д.

Для заметок _____

Краткий справочник по комплекту инструкций

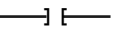
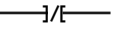



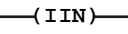
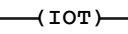
Назначение этой главы

Сведения	Страница
Релейные инструкции	22-2
Инструкции таймера	22-3
Инструкции счетчика	22-4
Инструкции сравнения	22-5
Инструкции вычисления	22-7
Логические инструкции	22-14
Инструкции преобразования	22-15
Инструкции перемещения и изменения битов	22-16
Файловые инструкции	22-17
Инструкции диагностики	22-18
Инструкции регистрового сдвига	22-19
Инструкции секвенсора	22-20
Инструкции управления программой	22-21
Инструкции передачи сообщений и управления процессами	22-23
Инструкции блок-трансферов	22-24
ASCII инструкции	22-25
Требования к памяти и времена выполнения инструкций бита и слова	22-28
Требования к памяти и времена выполнения файловых инструкций, инструкций управления программой и ASCII	22-31



Важно: Для более детального описания каждой из этих инструкций, обратитесь к Справочнику по комплекту инструкций пакета программирования PLC-5, публикация 1785-6.1.

Релейные инструкции

Инструкция	Описание	
I : 012  07	Проверка на включено XIC	Проверяет бит I:012/07 таблицы данных, который соответствует клемме 7 модуля входа в рэке 1, группа 2. Если этот бит таблицы данных установлен (1), то значение инструкции - истина.
I : 012  07	Проверка на выключено XIO	Проверяет бит I:012/07 таблицы данных, который соответствует клемме 7 модуля входа в рэке 1, группа 2. Если этот бит таблицы данных сброшен (0), то значение инструкции - истина.
O : 013  01	Включить выход OTE	Если входные условия, предшествующие этой выходной инструкции в той же цепи, становятся истиной, устанавливает (1) бит O:013/01, который соответствует клемме 1 модуля выхода в рэке 1, группа 3.
O : 013  01	Зафиксировать выход OTL	Если входные условия, предшествующие этой выходной инструкции в той же цепи, становятся истиной, устанавливает (1) бит O:013/01, который соответствует клемме 1 модуля выхода в рэке 1, группа 3. Этот бит таблицы данных остается установленным даже если условие цепи становится ложным.
O : 013  01	Расфиксировать выход OTU	Если входные условия, предшествующие этой выходной инструкции в той же цепи, становятся истиной, сбрасывает (0) бит O:013/01, который соответствует клемме 1 модуля выхода в рэке 1, группа 3. Это необходимо, чтобы сбросить бит который был зафиксирован.
01 	Немедленный ввод IIN	Эта инструкция обновляет слово битов образа входов до следующего нормального обновления образа входов. Адресуйте эту инструкции с помощью рэка и группы (RRG). Для локального шасси сканирование программы прервано, пока сканируются входы адресованной группы Vх/Вых; для удаленного шасси, просмотр программы прерван только для обновления образа входов последними состояниями, найденными в буфере удаленных Vх/Вых.
01 	Немедленный вывод IOT	Эта инструкция обновляет слово битов образа выходов до следующего нормального обновления образа выходов. Адресуйте эту инструкции с помощью рэка и группы (RRG). Для локального шасси сканирование программы прервано, пока выходы адресованной группы Vх/Вых не будут обновлены. Для удаленного шасси, просмотр программы прерван только для обновления буфера удаленных Vх/Вых последними состояниями, найденными в образе выходов.

Инструкции таймера

Инструкция			Описание																													
<div style="border: 1px solid black; padding: 5px;"> TON TIMER ON DELAY Timer T4:1 Time Base 1.0 Preset 15 Accum 0 </div>	Таймер с задержкой на включение TON Биты состояния: EN – разрешен TT – идет отсчет DN – завершен	Если условия входа становятся истинными, таймер T4:1 начинает увеличиваться с интервалом в 1 секунду. Когда накопленное значение больше, либо равно предустановленному (15), таймер останавливается и устанавливает бит завершения таймера.	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>Условия цепи</th> <th>EN</th> <th>TT</th> <th>DN</th> <th>ACC значение</th> <th>TON состояние</th> </tr> </thead> <tbody> <tr> <td>ложь</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>сброс</td> </tr> <tr> <td>истина</td> <td>1</td> <td>1</td> <td>0</td> <td>увеличивается</td> <td>отсчет</td> </tr> <tr> <td>истина</td> <td>1</td> <td>0</td> <td>1</td> <td>>= уставка</td> <td>завершен</td> </tr> </tbody> </table>	Условия цепи	EN	TT	DN	ACC значение	TON состояние	ложь	0	0	0	0	сброс	истина	1	1	0	увеличивается	отсчет	истина	1	0	1	>= уставка	завершен					
Условия цепи	EN	TT	DN	ACC значение	TON состояние																											
ложь	0	0	0	0	сброс																											
истина	1	1	0	увеличивается	отсчет																											
истина	1	0	1	>= уставка	завершен																											
<div style="border: 1px solid black; padding: 5px;"> TOF TIMER OFF DELAY Timer T4:1 Time Base .01 Preset 180 Accum 0 </div>	Таймер с задержкой на отключение TOF Биты состояния: EN – разрешен TT – идет отсчет DN – завершен	Если условия входа ложны, таймер T4:1 начинает увеличиваться с интервалами в 10 мс, пока цепь остается ложной. Когда накопленное значение больше, либо равно предустановленному (180), таймер останавливается и сбрасывает бит завершения таймера.	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>Условия цепи</th> <th>EN</th> <th>TT</th> <th>DN</th> <th>ACC значение</th> <th>TOF состояние</th> </tr> </thead> <tbody> <tr> <td>ложь</td> <td>1</td> <td>0</td> <td>1</td> <td>0</td> <td>сброс</td> </tr> <tr> <td>истина</td> <td>0</td> <td>1</td> <td>1</td> <td>увеличивается</td> <td>отсчет</td> </tr> <tr> <td>истина</td> <td>0</td> <td>0</td> <td>0</td> <td>>= уставка</td> <td>завершен</td> </tr> </tbody> </table>	Условия цепи	EN	TT	DN	ACC значение	TOF состояние	ложь	1	0	1	0	сброс	истина	0	1	1	увеличивается	отсчет	истина	0	0	0	>= уставка	завершен	Смотри страницу 24-8 для описания операции пресканирования этой инструкции.				
Условия цепи	EN	TT	DN	ACC значение	TOF состояние																											
ложь	1	0	1	0	сброс																											
истина	0	1	1	увеличивается	отсчет																											
истина	0	0	0	>= уставка	завершен																											

Инструкция	Описание																															
RTO RETENTIVE TIMER ON Timer T4:10 Time Base 1.0 Preset 10 Accum 0	Накапливающий таймер с задержкой на включение RTO Биты состояния: EN – разрешен TT – идет отсчет DN – завершен	Если условия входа становятся истинными, таймер T4:10 начинает увеличиваться с интервалом в 1 секунду, пока цепь остается истинной. Когда цепь становится ложной, таймер останавливается. Если цепь становится снова истинным, таймер продолжает работу. Когда накопленное значение больше, либо равно предустановленному (10), таймер останавливается и активизирует бит завершения таймера. <table border="1"> <thead> <tr> <th>Условия цепи</th> <th>EN</th> <th>TT</th> <th>DN</th> <th>ACC значение</th> <th>RTO состояние</th> </tr> </thead> <tbody> <tr> <td>ложь</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>сброс</td> </tr> <tr> <td>истина</td> <td>1</td> <td>1</td> <td>0</td> <td>увеличивается</td> <td>отсчет</td> </tr> <tr> <td>ложь</td> <td>0</td> <td>0</td> <td>0</td> <td>сохраняется</td> <td>запрещен</td> </tr> <tr> <td>истина</td> <td>1</td> <td>0</td> <td>1</td> <td>>= уставка</td> <td>завершен</td> </tr> </tbody> </table>	Условия цепи	EN	TT	DN	ACC значение	RTO состояние	ложь	0	0	0	0	сброс	истина	1	1	0	увеличивается	отсчет	ложь	0	0	0	сохраняется	запрещен	истина	1	0	1	>= уставка	завершен
Условия цепи	EN	TT	DN	ACC значение	RTO состояние																											
ложь	0	0	0	0	сброс																											
истина	1	1	0	увеличивается	отсчет																											
ложь	0	0	0	сохраняется	запрещен																											
истина	1	0	1	>= уставка	завершен																											
T4:1 —(RES)—	Сброс таймера RES	Если входные условия становятся верными, таймер T4:1 сбрасывается. Эта инструкция сбрасывает таймеры и счетчики, так же как и блоки управления. Необходимо сбрасывать накопленное значение RTO.																														

Инструкции счетчика

Инструкция	Описание																															
CTU COUNT UP Counter C5:1 Preset 10 Accum 0	Прямой счетчик CTU Биты состояния: CU – прямой счет CD – обратный счет DN – завершен OV – переполнен UN – обратно-переполнен	Если условия входа становятся истинными, счетчик C5:1 начинает отсчет, увеличиваясь на 1 каждый раз, когда цепь переходит от ложного к истинному значению. Когда накопленное значение больше, либо равно предустановленному (10), счетчик устанавливает бит завершения счета. <table border="1"> <thead> <tr> <th>Условия цепи</th> <th>CU</th> <th>DN</th> <th>OV</th> <th>ACC значение</th> <th>CTU состояние</th> </tr> </thead> <tbody> <tr> <td>ложь</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>сброс</td> </tr> <tr> <td>переход к истине</td> <td>1</td> <td>0</td> <td>0</td> <td>увеличивается на 1</td> <td>счет</td> </tr> <tr> <td>истина</td> <td>1</td> <td>1</td> <td>0</td> <td>>= уставка</td> <td>завершен</td> </tr> <tr> <td>истина</td> <td>1</td> <td>1</td> <td>1</td> <td>>32767</td> <td>переполнен</td> </tr> </tbody> </table> Смотри страницу 24-8 для описания операции пресканирования этой инструкции.	Условия цепи	CU	DN	OV	ACC значение	CTU состояние	ложь	0	0	0	0	сброс	переход к истине	1	0	0	увеличивается на 1	счет	истина	1	1	0	>= уставка	завершен	истина	1	1	1	>32767	переполнен
Условия цепи	CU	DN	OV	ACC значение	CTU состояние																											
ложь	0	0	0	0	сброс																											
переход к истине	1	0	0	увеличивается на 1	счет																											
истина	1	1	0	>= уставка	завершен																											
истина	1	1	1	>32767	переполнен																											

Инструкция	Описание																																					
CTD COUNT DOWN Counter C5:1 Preset 10 Accum 35	Обратный счетчик STU Биты состояния: CU - прямой счет CD - обратный счет DN - завершен OV - переполнен UN - обратно-переполнен	Если условия входа становятся истинными, счетчик C5:1 начинает отсчет, уменьшаясь на 1 каждый раз, когда цепь переходит от ложного к истинному значению. Когда накопленное значение меньше предустановленного (10), счетчик сбрасывает бит завершения счета. <table border="1"> <thead> <tr> <th>Условия цепи</th> <th>CU</th> <th>DN</th> <th>OV</th> <th>ACC значение</th> <th>CTD состояние</th> </tr> </thead> <tbody> <tr> <td>ложь</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>сброс</td> </tr> <tr> <td>ложь</td> <td>0</td> <td>1</td> <td>0</td> <td>>= уставка</td> <td>предварительно загружен</td> </tr> <tr> <td>переход к истине</td> <td>1</td> <td>1</td> <td>0</td> <td>уменьшается на 1</td> <td>счет</td> </tr> <tr> <td>истина</td> <td>1</td> <td>0</td> <td>0</td> <td>< уставка</td> <td>завершен</td> </tr> <tr> <td>истина</td> <td>1</td> <td>0</td> <td>1</td> <td>< -32768</td> <td>обратно-переполнен</td> </tr> </tbody> </table> <p>Смотри страницу 24-8 для описания операции пресканирования этой инструкции.</p>	Условия цепи	CU	DN	OV	ACC значение	CTD состояние	ложь	0	0	0	0	сброс	ложь	0	1	0	>= уставка	предварительно загружен	переход к истине	1	1	0	уменьшается на 1	счет	истина	1	0	0	< уставка	завершен	истина	1	0	1	< -32768	обратно-переполнен
Условия цепи	CU	DN	OV	ACC значение	CTD состояние																																	
ложь	0	0	0	0	сброс																																	
ложь	0	1	0	>= уставка	предварительно загружен																																	
переход к истине	1	1	0	уменьшается на 1	счет																																	
истина	1	0	0	< уставка	завершен																																	
истина	1	0	1	< -32768	обратно-переполнен																																	

Инструкции сравнения

Инструкция	Описание																													
LIM LIMIT TEST Low limit N7:10 3 Test N7:15 4 High limit N7:20 22	Тест на пределы LIM	Если испытуемое значение (N7:15) >= нижнего предела (N7:10) и <= верхнего предела (N7:20), то эта инструкция истинна. <table border="1"> <thead> <tr> <th>Нижний предел</th> <th>Испытуемое значение</th> <th>Верхний предел</th> <th>LIM</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>10</td> <td>истина</td> </tr> <tr> <td>-5</td> <td>5</td> <td>10</td> <td>истина</td> </tr> <tr> <td>5</td> <td>11</td> <td>10</td> <td>ложь</td> </tr> <tr> <td>10</td> <td>0</td> <td>0</td> <td>истина</td> </tr> <tr> <td>10</td> <td>5</td> <td>-5</td> <td>ложь</td> </tr> <tr> <td>10</td> <td>11</td> <td>5</td> <td>истина</td> </tr> </tbody> </table>	Нижний предел	Испытуемое значение	Верхний предел	LIM	0	0	10	истина	-5	5	10	истина	5	11	10	ложь	10	0	0	истина	10	5	-5	ложь	10	11	5	истина
Нижний предел	Испытуемое значение	Верхний предел	LIM																											
0	0	10	истина																											
-5	5	10	истина																											
5	11	10	ложь																											
10	0	0	истина																											
10	5	-5	ложь																											
10	11	5	истина																											
MEQ MASKED EQUAL Source D9:5 0000 Mask D9:6 0000 Compare D9:10 0000	Маскированное сравнение на равно MEQ	Процессор берет величину в источнике (D9:5), сравниваемую величину (D9:10) и пропускает эти величины через маску (D9:6). Затем процессор сравнивает результаты между собой. Если результаты равны, инструкция истинна. <table border="1"> <thead> <tr> <th>Источник</th> <th>Маска</th> <th>Значение для сравнения</th> <th>MEQ</th> </tr> </thead> <tbody> <tr> <td>0008</td> <td>0008</td> <td>0009</td> <td>истина</td> </tr> <tr> <td>0008</td> <td>0001</td> <td>0001</td> <td>ложь</td> </tr> <tr> <td>0087</td> <td>000F</td> <td>0007</td> <td>истина</td> </tr> <tr> <td>0087</td> <td>00F0</td> <td>0007</td> <td>ложь</td> </tr> </tbody> </table>	Источник	Маска	Значение для сравнения	MEQ	0008	0008	0009	истина	0008	0001	0001	ложь	0087	000F	0007	истина	0087	00F0	0007	ложь								
Источник	Маска	Значение для сравнения	MEQ																											
0008	0008	0009	истина																											
0008	0001	0001	ложь																											
0087	000F	0007	истина																											
0087	00F0	0007	ложь																											

Инструкция	Описание																																																													
CMP COMPARE Expression N7:5 = N7:10	Сравнение CMP	Если выражение истинное, эта входная инструкция истинна. Инструкция CMP может выполнить операции: равно (=), меньше чем (<), меньше чем или равно (<=), больше чем (>), больше чем или равно (>=), не равно (<>), и сложные выражения (вплоть до 80 символов).																																																												
xxx XXXXXXXXXXXXXX Source A N7:5 3 Source B N7:10 1	<table border="1"> <thead> <tr> <th>Источник A</th> <th>Источник B</th> <th>EQU</th> <th>GEQ</th> <th>GRT</th> <th>LEQ</th> <th>LES</th> <th>NEQ</th> </tr> </thead> <tbody> <tr> <td>10</td> <td>10</td> <td>истина</td> <td>истина</td> <td>ложь</td> <td>истина</td> <td>ложь</td> <td>ложь</td> </tr> <tr> <td>5</td> <td>6</td> <td>ложь</td> <td>ложь</td> <td>ложь</td> <td>истина</td> <td>истина</td> <td>истина</td> </tr> <tr> <td>21</td> <td>20</td> <td>ложь</td> <td>истина</td> <td>истина</td> <td>ложь</td> <td>ложь</td> <td>истина</td> </tr> <tr> <td>-30</td> <td>-31</td> <td>ложь</td> <td>истина</td> <td>истина</td> <td>ложь</td> <td>ложь</td> <td>истина</td> </tr> <tr> <td>-15</td> <td>-14</td> <td>ложь</td> <td>ложь</td> <td>ложь</td> <td>истина</td> <td>истина</td> <td>истина</td> </tr> </tbody> </table>	Источник A	Источник B	EQU	GEQ	GRT	LEQ	LES	NEQ	10	10	истина	истина	ложь	истина	ложь	ложь	5	6	ложь	ложь	ложь	истина	истина	истина	21	20	ложь	истина	истина	ложь	ложь	истина	-30	-31	ложь	истина	истина	ложь	ложь	истина	-15	-14	ложь	ложь	ложь	истина	истина	истина	<table border="1"> <tbody> <tr> <td>Равно EQU</td> <td>Если величина источника A (N7:5) = величине источника B (N7:10), эта инструкция истинна.</td> </tr> <tr> <td>Больше или равно GEQ</td> <td>Если величина источника A (N7:5) > или = величины источника B (N7:10), эта инструкция истинна.</td> </tr> <tr> <td>Больше GRT</td> <td>Если величина источника A (N7:5) > величины источника B (N7:10), эта инструкция истинна.</td> </tr> <tr> <td>Меньше или равно LEQ</td> <td>Если величина источника A (N7:5) < или = величины источника B (N7:10), эта инструкция истинна.</td> </tr> <tr> <td>Меньше LES</td> <td>Если величина источника A (N7:5) < величины источника B (N7:10), эта инструкция истинна.</td> </tr> <tr> <td>Не равно NEQ</td> <td>Если величина источника A (N7:5) не равна величине источника B (N7:10), эта инструкция истинна.</td> </tr> </tbody> </table>	Равно EQU	Если величина источника A (N7:5) = величине источника B (N7:10), эта инструкция истинна.	Больше или равно GEQ	Если величина источника A (N7:5) > или = величины источника B (N7:10), эта инструкция истинна.	Больше GRT	Если величина источника A (N7:5) > величины источника B (N7:10), эта инструкция истинна.	Меньше или равно LEQ	Если величина источника A (N7:5) < или = величины источника B (N7:10), эта инструкция истинна.	Меньше LES	Если величина источника A (N7:5) < величины источника B (N7:10), эта инструкция истинна.	Не равно NEQ	Если величина источника A (N7:5) не равна величине источника B (N7:10), эта инструкция истинна.
Источник A	Источник B	EQU	GEQ	GRT	LEQ	LES	NEQ																																																							
10	10	истина	истина	ложь	истина	ложь	ложь																																																							
5	6	ложь	ложь	ложь	истина	истина	истина																																																							
21	20	ложь	истина	истина	ложь	ложь	истина																																																							
-30	-31	ложь	истина	истина	ложь	ложь	истина																																																							
-15	-14	ложь	ложь	ложь	истина	истина	истина																																																							
Равно EQU	Если величина источника A (N7:5) = величине источника B (N7:10), эта инструкция истинна.																																																													
Больше или равно GEQ	Если величина источника A (N7:5) > или = величины источника B (N7:10), эта инструкция истинна.																																																													
Больше GRT	Если величина источника A (N7:5) > величины источника B (N7:10), эта инструкция истинна.																																																													
Меньше или равно LEQ	Если величина источника A (N7:5) < или = величины источника B (N7:10), эта инструкция истинна.																																																													
Меньше LES	Если величина источника A (N7:5) < величины источника B (N7:10), эта инструкция истинна.																																																													
Не равно NEQ	Если величина источника A (N7:5) не равна величине источника B (N7:10), эта инструкция истинна.																																																													

Инструкции вычисления

Инструкция	Описание										
<div style="border: 1px solid black; padding: 5px; margin-bottom: 10px;"> CPT COMPUTE Dest N7:3 3 Expression N7:4 - (N7:6 * N7:10) </div>	<div style="display: flex; justify-content: space-between;"> <div style="width: 20%;"> <p>Вычисление CPT</p> </div> <div style="width: 80%;"> <p>Если входные условия становятся верными, вычисляет выражение N7:4 - (N7:6 * N7:10) и сохраняет результат в назначении (N7:3). Инструкция CPT может выполнить операции: сложения (+), вычитания (-), умножения (*), деления (/), преобразования из BCD (FRD), преобразования в BCD (TOD), квадратного корня (SQR), логического И (AND), логического ИЛИ (OR), логического отрицания (NOT), исключающего ИЛИ (XOR), смены знака (-), очистки (0), возведения X в степень Y (**), преобразования в радианы (RAD), преобразования в градусы (DEG), логарифма (LOG), натурального логарифма (LN), синуса (SIN), косинуса (COS), тангенса (TAN), арксинуса (ASN), арккосинуса (ACS), арктангенса (ATN) и вычисления сложных выражений (вплоть до 80 символов). <i>Примечание:</i> Любая введенная величина (например, 2.3) расширяется до 8 символов (2.3000000).</p> </div> </div>										
<div style="border: 1px solid black; padding: 5px; margin-bottom: 10px;"> ACS ARCCOSINE Source F8:19 0.7853982 Dest F8:20 0.6674572 </div>	<div style="display: flex; justify-content: space-between;"> <div style="width: 20%;"> <p>Арккосинус ACS</p> </div> <div style="width: 80%;"> <p>Если входные условия становятся верными, вычисляет косинус величины в F8:19 и сохраняет результат в F8:20.</p> <table border="1" style="width: 100%; border-collapse: collapse; margin-top: 10px;"> <thead> <tr> <th style="width: 10%;">Бит состояния</th> <th>Описание</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">C</td> <td>всегда сбрасывается</td> </tr> <tr> <td style="text-align: center;">V</td> <td>устанавливается при переполнении, иначе сбрасывается</td> </tr> <tr> <td style="text-align: center;">Z</td> <td>устанавливается, если результат равен нулю, иначе сбрасывается</td> </tr> <tr> <td style="text-align: center;">S</td> <td>всегда сбрасывается</td> </tr> </tbody> </table> </div> </div>	Бит состояния	Описание	C	всегда сбрасывается	V	устанавливается при переполнении, иначе сбрасывается	Z	устанавливается, если результат равен нулю, иначе сбрасывается	S	всегда сбрасывается
Бит состояния	Описание										
C	всегда сбрасывается										
V	устанавливается при переполнении, иначе сбрасывается										
Z	устанавливается, если результат равен нулю, иначе сбрасывается										
S	всегда сбрасывается										
<div style="border: 1px solid black; padding: 5px; margin-bottom: 10px;"> ADD ADD Source A N7:3 3 Source B N7:4 1 Dest N7:12 4 </div>	<div style="display: flex; justify-content: space-between;"> <div style="width: 20%;"> <p>Сложение ADD</p> </div> <div style="width: 80%;"> <p>Когда входные условия истинные, добавляет величину в источнике A (N7:3) к величине в источнике B (N7:4) и сохраняет результат в назначении (N7:12).</p> <table border="1" style="width: 100%; border-collapse: collapse; margin-top: 10px;"> <thead> <tr> <th style="width: 10%;">Бит состояния</th> <th>Описание</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">C</td> <td>устанавливается при возникновении переноса, иначе сбрасывается</td> </tr> <tr> <td style="text-align: center;">V</td> <td>устанавливается при переполнении, иначе сбрасывается</td> </tr> <tr> <td style="text-align: center;">Z</td> <td>устанавливается, если результат равен нулю, иначе сбрасывается</td> </tr> <tr> <td style="text-align: center;">S</td> <td>устанавливается, если результат отрицателен, иначе сбрасывается</td> </tr> </tbody> </table> </div> </div>	Бит состояния	Описание	C	устанавливается при возникновении переноса, иначе сбрасывается	V	устанавливается при переполнении, иначе сбрасывается	Z	устанавливается, если результат равен нулю, иначе сбрасывается	S	устанавливается, если результат отрицателен, иначе сбрасывается
Бит состояния	Описание										
C	устанавливается при возникновении переноса, иначе сбрасывается										
V	устанавливается при переполнении, иначе сбрасывается										
Z	устанавливается, если результат равен нулю, иначе сбрасывается										
S	устанавливается, если результат отрицателен, иначе сбрасывается										

Инструкция		Описание										
ASN ARCSINE Source F8:17 0.7853982 Dest F8:18 0.9033391	Арксинус ASN	Когда входные условия становятся верными, вычисляет арксинус величины в F8:17 и сохраняет результат в F8:18. Бит состояния <table border="1"> <thead> <tr> <th>Состояние</th> <th>Описание</th> </tr> </thead> <tbody> <tr> <td>C</td> <td>всегда сбрасывается</td> </tr> <tr> <td>V</td> <td>устанавливается при переполнении, иначе сбрасывается</td> </tr> <tr> <td>Z</td> <td>устанавливается, если результат равен нулю, иначе сбрасывается</td> </tr> <tr> <td>S</td> <td>всегда сбрасывается</td> </tr> </tbody> </table>	Состояние	Описание	C	всегда сбрасывается	V	устанавливается при переполнении, иначе сбрасывается	Z	устанавливается, если результат равен нулю, иначе сбрасывается	S	всегда сбрасывается
Состояние	Описание											
C	всегда сбрасывается											
V	устанавливается при переполнении, иначе сбрасывается											
Z	устанавливается, если результат равен нулю, иначе сбрасывается											
S	всегда сбрасывается											
ATN ARCTANGENT Source F8:21 0.7853982 Dest F8:22 0.6657737	Арктангенс ATN	Когда входные условия становятся верными, вычисляет арктангенс величины в F8:21 и сохраняет результат в F8:22. Бит состояния <table border="1"> <thead> <tr> <th>Состояние</th> <th>Описание</th> </tr> </thead> <tbody> <tr> <td>C</td> <td>устанавливается при возникновении переноса, иначе сбрасывается</td> </tr> <tr> <td>V</td> <td>устанавливается при переполнении, иначе сбрасывается</td> </tr> <tr> <td>Z</td> <td>устанавливается, если результат равен нулю, иначе сбрасывается</td> </tr> <tr> <td>S</td> <td>устанавливается, если результат отрицателен, иначе сбрасывается</td> </tr> </tbody> </table>	Состояние	Описание	C	устанавливается при возникновении переноса, иначе сбрасывается	V	устанавливается при переполнении, иначе сбрасывается	Z	устанавливается, если результат равен нулю, иначе сбрасывается	S	устанавливается, если результат отрицателен, иначе сбрасывается
Состояние	Описание											
C	устанавливается при возникновении переноса, иначе сбрасывается											
V	устанавливается при переполнении, иначе сбрасывается											
Z	устанавливается, если результат равен нулю, иначе сбрасывается											
S	устанавливается, если результат отрицателен, иначе сбрасывается											
AVE AVERAGE FILE File #N7:1 Dest N7:0 Control R6:0 Length 4 Position 0	Усреднение файла AVE Биты состояния: EN – разрешен DN – завершен ER – ошибка	Когда входные условия изменяются из ложных в истинные, вычисляет среднее значение файла #N7:1 и сохраняет результат в N7:0. Бит состояния <table border="1"> <thead> <tr> <th>Состояние</th> <th>Описание</th> </tr> </thead> <tbody> <tr> <td>C</td> <td>всегда сброшен</td> </tr> <tr> <td>V</td> <td>устанавливается при переполнении, иначе сбрасывается</td> </tr> <tr> <td>Z</td> <td>устанавливается, если результат равен нулю, иначе сбрасывается</td> </tr> <tr> <td>S</td> <td>устанавливается, если результат отрицателен, иначе сбрасывается</td> </tr> </tbody> </table>	Состояние	Описание	C	всегда сброшен	V	устанавливается при переполнении, иначе сбрасывается	Z	устанавливается, если результат равен нулю, иначе сбрасывается	S	устанавливается, если результат отрицателен, иначе сбрасывается
Состояние	Описание											
C	всегда сброшен											
V	устанавливается при переполнении, иначе сбрасывается											
Z	устанавливается, если результат равен нулю, иначе сбрасывается											
S	устанавливается, если результат отрицателен, иначе сбрасывается											

Инструкция	Описание											
<pre>CLR CLR Dest D9:34 0000</pre>	Очистка CLR	<p>Когда входные условия становятся верными, очищает (устанавливает в ноль) слово 34 десятичного файла 9.</p> <table border="1" data-bbox="798 336 1444 627"> <thead> <tr> <th data-bbox="810 336 917 398">Бит состояния</th> <th data-bbox="949 369 1045 398">Описание</th> </tr> </thead> <tbody> <tr> <td data-bbox="837 398 853 425">C</td> <td data-bbox="949 398 1141 425">всегда сбрасывается</td> </tr> <tr> <td data-bbox="837 465 853 492">V</td> <td data-bbox="949 465 1141 492">всегда сбрасывается</td> </tr> <tr> <td data-bbox="837 533 853 560">Z</td> <td data-bbox="949 533 1141 560">всегда сбрасывается</td> </tr> <tr> <td data-bbox="837 600 853 627">S</td> <td data-bbox="949 600 1141 627">всегда сбрасывается</td> </tr> </tbody> </table>	Бит состояния	Описание	C	всегда сбрасывается	V	всегда сбрасывается	Z	всегда сбрасывается	S	всегда сбрасывается
Бит состояния	Описание											
C	всегда сбрасывается											
V	всегда сбрасывается											
Z	всегда сбрасывается											
S	всегда сбрасывается											
<pre>COS COSINE Source F8:13 0.7853982 Dest F8:14 0.7071068</pre>	Косинус COS	<p>Когда входные условия становятся верными, вычисляет косинус величины в F8:13 и сохраняет результат в F8:14.</p> <table border="1" data-bbox="798 716 1444 1041"> <thead> <tr> <th data-bbox="810 716 917 779">Бит состояния</th> <th data-bbox="949 750 1045 779">Описание</th> </tr> </thead> <tbody> <tr> <td data-bbox="837 779 853 806">C</td> <td data-bbox="949 779 1141 806">всегда сбрасывается</td> </tr> <tr> <td data-bbox="837 846 853 873">V</td> <td data-bbox="949 846 1268 902">устанавливается при переполнении, иначе сбрасывается</td> </tr> <tr> <td data-bbox="837 920 853 947">Z</td> <td data-bbox="949 920 1348 976">устанавливается, если результат равен нулю, иначе сбрасывается</td> </tr> <tr> <td data-bbox="837 976 853 1003">S</td> <td data-bbox="949 976 1364 1032">устанавливается, если результат отрицателен, иначе сбрасывается</td> </tr> </tbody> </table>	Бит состояния	Описание	C	всегда сбрасывается	V	устанавливается при переполнении, иначе сбрасывается	Z	устанавливается, если результат равен нулю, иначе сбрасывается	S	устанавливается, если результат отрицателен, иначе сбрасывается
Бит состояния	Описание											
C	всегда сбрасывается											
V	устанавливается при переполнении, иначе сбрасывается											
Z	устанавливается, если результат равен нулю, иначе сбрасывается											
S	устанавливается, если результат отрицателен, иначе сбрасывается											

Инструкция		Описание										
DIV DIVIDE Source A N7:3 3 Source B N7:4 1 Dest N7:12 3	Деление DIV	<p>Когда входные условия истинны, делит величину в источнике A (N7:3) на величину в источнике B (N7:4) и сохраняет результат в назначении (N7:12).</p> <table border="1"> <thead> <tr> <th>Бит состояния</th> <th>Описание</th> </tr> </thead> <tbody> <tr> <td>C</td> <td>всегда сбрасывается</td> </tr> <tr> <td>V</td> <td>устанавливается при переполнении или делении на ноль, иначе сбрасывается</td> </tr> <tr> <td>Z</td> <td>устанавливается, если результат равен нулю, иначе сбрасывается; неопределен при V=1</td> </tr> <tr> <td>S</td> <td>устанавливается, если результат отрицателен, иначе сбрасывается; неопределен при V=1</td> </tr> </tbody> </table>	Бит состояния	Описание	C	всегда сбрасывается	V	устанавливается при переполнении или делении на ноль, иначе сбрасывается	Z	устанавливается, если результат равен нулю, иначе сбрасывается; неопределен при V=1	S	устанавливается, если результат отрицателен, иначе сбрасывается; неопределен при V=1
Бит состояния	Описание											
C	всегда сбрасывается											
V	устанавливается при переполнении или делении на ноль, иначе сбрасывается											
Z	устанавливается, если результат равен нулю, иначе сбрасывается; неопределен при V=1											
S	устанавливается, если результат отрицателен, иначе сбрасывается; неопределен при V=1											
LN NATURAL LOG Source N7:0 5 Dest F8:20 1.609438	Натуральный логарифм LN	<p>Когда входные условия становятся верными, вычисляет натуральный логарифм величины в N7:0 и сохраняет результат в F8:20.</p> <table border="1"> <thead> <tr> <th>Бит состояния</th> <th>Описание</th> </tr> </thead> <tbody> <tr> <td>C</td> <td>всегда сбрасывается</td> </tr> <tr> <td>V</td> <td>устанавливается при переполнении, иначе сбрасывается</td> </tr> <tr> <td>Z</td> <td>устанавливается, если результат равен нулю, иначе сбрасывается</td> </tr> <tr> <td>S</td> <td>устанавливается, если результат отрицателен, иначе сбрасывается</td> </tr> </tbody> </table>	Бит состояния	Описание	C	всегда сбрасывается	V	устанавливается при переполнении, иначе сбрасывается	Z	устанавливается, если результат равен нулю, иначе сбрасывается	S	устанавливается, если результат отрицателен, иначе сбрасывается
Бит состояния	Описание											
C	всегда сбрасывается											
V	устанавливается при переполнении, иначе сбрасывается											
Z	устанавливается, если результат равен нулю, иначе сбрасывается											
S	устанавливается, если результат отрицателен, иначе сбрасывается											
LOG LOG BASE 10 Source N7:2 5 Dest F8:3 0.6989700	Десятичный логарифм LOG	<p>Когда входные условия становятся верными, вычисляет десятичный логарифм величины в N7:2 и сохраняет результат в F8:3.</p> <table border="1"> <thead> <tr> <th>Бит состояния</th> <th>Описание</th> </tr> </thead> <tbody> <tr> <td>C</td> <td>всегда сброшен</td> </tr> <tr> <td>V</td> <td>устанавливается при переполнении, иначе сбрасывается</td> </tr> <tr> <td>Z</td> <td>устанавливается, если результат равен нулю, иначе сбрасывается</td> </tr> <tr> <td>S</td> <td>устанавливается, если результат отрицателен, иначе сбрасывается</td> </tr> </tbody> </table>	Бит состояния	Описание	C	всегда сброшен	V	устанавливается при переполнении, иначе сбрасывается	Z	устанавливается, если результат равен нулю, иначе сбрасывается	S	устанавливается, если результат отрицателен, иначе сбрасывается
Бит состояния	Описание											
C	всегда сброшен											
V	устанавливается при переполнении, иначе сбрасывается											
Z	устанавливается, если результат равен нулю, иначе сбрасывается											
S	устанавливается, если результат отрицателен, иначе сбрасывается											

Инструкция		Описание										
MUL MULTIPLY Source A N7:3 3 Source B N7:4 1 Dest N7:12 3	Умножение MUL	<p>Когда входные условия истинны, умножает величину в источнике A (N7:3) на величину в источнике B (N7:4) и сохраняет результат в назначении (N7:12).</p> <table border="1"> <thead> <tr> <th>Бит состояния</th> <th>Описание</th> </tr> </thead> <tbody> <tr> <td>C</td> <td>всегда сбрасывается</td> </tr> <tr> <td>V</td> <td>устанавливается при переполнении, иначе сбрасывается</td> </tr> <tr> <td>Z</td> <td>устанавливается, если результат равен нулю, иначе сбрасывается</td> </tr> <tr> <td>S</td> <td>устанавливается, если результат отрицателен, иначе сбрасывается</td> </tr> </tbody> </table>	Бит состояния	Описание	C	всегда сбрасывается	V	устанавливается при переполнении, иначе сбрасывается	Z	устанавливается, если результат равен нулю, иначе сбрасывается	S	устанавливается, если результат отрицателен, иначе сбрасывается
Бит состояния	Описание											
C	всегда сбрасывается											
V	устанавливается при переполнении, иначе сбрасывается											
Z	устанавливается, если результат равен нулю, иначе сбрасывается											
S	устанавливается, если результат отрицателен, иначе сбрасывается											
NEG NEGATE Source N7:3 3 Dest N7:12 -3	Смена знака NEG	<p>Когда входные условия истинны, берет источник (N7:3) с противоположным знаком и сохраняет результат в назначении (N7:12). Эта инструкция преобразует положительные величины в отрицательные, а отрицательные – в положительные.</p> <table border="1"> <thead> <tr> <th>Бит состояния</th> <th>Описание</th> </tr> </thead> <tbody> <tr> <td>C</td> <td>устанавливается при переносе, иначе сбрасывается</td> </tr> <tr> <td>V</td> <td>устанавливается при переполнении, иначе сбрасывается</td> </tr> <tr> <td>Z</td> <td>устанавливается, если результат равен нулю, иначе сбрасывается</td> </tr> <tr> <td>S</td> <td>устанавливается, если результат отрицателен, иначе сбрасывается</td> </tr> </tbody> </table>	Бит состояния	Описание	C	устанавливается при переносе, иначе сбрасывается	V	устанавливается при переполнении, иначе сбрасывается	Z	устанавливается, если результат равен нулю, иначе сбрасывается	S	устанавливается, если результат отрицателен, иначе сбрасывается
Бит состояния	Описание											
C	устанавливается при переносе, иначе сбрасывается											
V	устанавливается при переполнении, иначе сбрасывается											
Z	устанавливается, если результат равен нулю, иначе сбрасывается											
S	устанавливается, если результат отрицателен, иначе сбрасывается											
SIN SINE Source F8:11 0.7853982 Dest F8:12 0.7071068	Синус SIN	<p>Когда входные условия становятся верными, вычисляет синус величины в F8:11 и сохраняет результат в F8:12.</p> <table border="1"> <thead> <tr> <th>Бит состояния</th> <th>Описание</th> </tr> </thead> <tbody> <tr> <td>C</td> <td>всегда сброшен</td> </tr> <tr> <td>V</td> <td>устанавливается при переполнении, иначе сбрасывается</td> </tr> <tr> <td>Z</td> <td>устанавливается, если результат равен нулю, иначе сбрасывается</td> </tr> <tr> <td>S</td> <td>устанавливается, если результат отрицателен, иначе сбрасывается</td> </tr> </tbody> </table>	Бит состояния	Описание	C	всегда сброшен	V	устанавливается при переполнении, иначе сбрасывается	Z	устанавливается, если результат равен нулю, иначе сбрасывается	S	устанавливается, если результат отрицателен, иначе сбрасывается
Бит состояния	Описание											
C	всегда сброшен											
V	устанавливается при переполнении, иначе сбрасывается											
Z	устанавливается, если результат равен нулю, иначе сбрасывается											
S	устанавливается, если результат отрицателен, иначе сбрасывается											

Инструкция	Описание											
SQR SQUARE ROOT Source N7:8 25 Dest N7:12 5	Квадратный корень SQR	Когда входные условия истинны, вычисляет квадратный корень источника (N7:8) и сохраняет результат в назначении (N7:12). Бит состояния <table border="1"> <thead> <tr> <th>Бит состояния</th> <th>Описание</th> </tr> </thead> <tbody> <tr> <td>C</td> <td>всегда сбрасывается</td> </tr> <tr> <td>V</td> <td>устанавливается в случае переполнения при преобразовании из плавающей запятой в целое, иначе сбрасывается</td> </tr> <tr> <td>Z</td> <td>устанавливается, если результат равен нулю, иначе сбрасывается</td> </tr> <tr> <td>S</td> <td>всегда сбрасывается</td> </tr> </tbody> </table>	Бит состояния	Описание	C	всегда сбрасывается	V	устанавливается в случае переполнения при преобразовании из плавающей запятой в целое, иначе сбрасывается	Z	устанавливается, если результат равен нулю, иначе сбрасывается	S	всегда сбрасывается
Бит состояния	Описание											
C	всегда сбрасывается											
V	устанавливается в случае переполнения при преобразовании из плавающей запятой в целое, иначе сбрасывается											
Z	устанавливается, если результат равен нулю, иначе сбрасывается											
S	всегда сбрасывается											
SRT SORT File #N7:1 Control R6:0 Length 4 Position 0	Сортировка SRT Биты состояния: EN – разрешен DN – завершен ER - ошибка	Когда входные условия изменяются из ложных в истинные, значения в N7:1, N7:2, N7:3 и N7:4 располагаются в порядке возрастания.										
STD STANDARD DEVIATION File #N7:1 Dest N7:0 Control R6:0 Length 4 Position 0	Стандартное отклонение STD Биты состояния: EN – разрешен DN – завершен ER - ошибка	Когда входные условия изменяются из ложных в истинные, вычисляет среднеквадратичное отклонение величин в файле #N7:1 и сохраняет результат в назначении (N7:0). Бит состояния <table border="1"> <thead> <tr> <th>Бит состояния</th> <th>Описание</th> </tr> </thead> <tbody> <tr> <td>C</td> <td>всегда сбрасывается</td> </tr> <tr> <td>V</td> <td>устанавливается при переполнении, иначе сбрасывается</td> </tr> <tr> <td>Z</td> <td>устанавливается, если результат равен нулю, иначе сбрасывается</td> </tr> <tr> <td>S</td> <td>всегда сбрасывается</td> </tr> </tbody> </table>	Бит состояния	Описание	C	всегда сбрасывается	V	устанавливается при переполнении, иначе сбрасывается	Z	устанавливается, если результат равен нулю, иначе сбрасывается	S	всегда сбрасывается
Бит состояния	Описание											
C	всегда сбрасывается											
V	устанавливается при переполнении, иначе сбрасывается											
Z	устанавливается, если результат равен нулю, иначе сбрасывается											
S	всегда сбрасывается											

Инструкция		Описание								
SUB SUBTRACT Source A N7:3 3 Source B N7:4 1 Dest N7:12 2	Вычитание SUB	Когда входные условия истинны, вычитает из величины в источнике A (N7:3) величину в источнике B (N7:4) и сохраняет результат в назначении (N7:12). Бит состояния Описание <table border="1"> <tr> <td>C</td> <td>устанавливается при заеме, иначе сбрасывается</td> </tr> <tr> <td>V</td> <td>устанавливается при переполнении, иначе сбрасывается</td> </tr> <tr> <td>Z</td> <td>устанавливается, если результат равен нулю, иначе сбрасывается</td> </tr> <tr> <td>S</td> <td>устанавливается, если результат отрицателен, иначе сбрасывается</td> </tr> </table>	C	устанавливается при заеме, иначе сбрасывается	V	устанавливается при переполнении, иначе сбрасывается	Z	устанавливается, если результат равен нулю, иначе сбрасывается	S	устанавливается, если результат отрицателен, иначе сбрасывается
C	устанавливается при заеме, иначе сбрасывается									
V	устанавливается при переполнении, иначе сбрасывается									
Z	устанавливается, если результат равен нулю, иначе сбрасывается									
S	устанавливается, если результат отрицателен, иначе сбрасывается									
TAN TANGENT Source F8:15 0.7853982 Dest F8:16 1.000000	Тангенс TAN	Когда входные условия становятся верными, вычисляет тангенс величины в F8:15 и сохраняет результат в F8:16. Бит состояния Описание <table border="1"> <tr> <td>C</td> <td>всегда сбрасывается</td> </tr> <tr> <td>V</td> <td>устанавливается при переполнении, иначе сбрасывается</td> </tr> <tr> <td>Z</td> <td>устанавливается, если результат равен нулю, иначе сбрасывается</td> </tr> <tr> <td>S</td> <td>устанавливается, если результат отрицателен, иначе сбрасывается</td> </tr> </table>	C	всегда сбрасывается	V	устанавливается при переполнении, иначе сбрасывается	Z	устанавливается, если результат равен нулю, иначе сбрасывается	S	устанавливается, если результат отрицателен, иначе сбрасывается
C	всегда сбрасывается									
V	устанавливается при переполнении, иначе сбрасывается									
Z	устанавливается, если результат равен нулю, иначе сбрасывается									
S	устанавливается, если результат отрицателен, иначе сбрасывается									
XPY X TO POWER OF Y Source A N7:4 5 Source B N7:5 2 Dest N7:6 25	X в степени Y XPY	Когда входные условия становятся верными, берет величину в N7:4, возводит в степень, хранящуюся в N7:5 и сохраняет результат в N7:6. Бит состояния Описание <table border="1"> <tr> <td>C</td> <td>всегда сброшен</td> </tr> <tr> <td>V</td> <td>устанавливается при переполнении, иначе сбрасывается</td> </tr> <tr> <td>Z</td> <td>устанавливается, если результат равен нулю, иначе сбрасывается</td> </tr> <tr> <td>S</td> <td>устанавливается, если результат отрицателен, иначе сбрасывается</td> </tr> </table>	C	всегда сброшен	V	устанавливается при переполнении, иначе сбрасывается	Z	устанавливается, если результат равен нулю, иначе сбрасывается	S	устанавливается, если результат отрицателен, иначе сбрасывается
C	всегда сброшен									
V	устанавливается при переполнении, иначе сбрасывается									
Z	устанавливается, если результат равен нулю, иначе сбрасывается									
S	устанавливается, если результат отрицателен, иначе сбрасывается									

Логические инструкции

Инструкция			Описание															
AND BITWISE AND Source A D9:3 3F37 Source B D9:4 00FF Dest D9:5 0037	Побитное И AND		Когда входные условия истинны, процессор выполняет побитное И между источником A (D9:3) и источником B (D9:4) и сохраняет результат в назначении (D9:5). Таблица истинности для функции И: <table style="margin-left: auto; margin-right: auto; border-collapse: collapse;"> <thead> <tr> <th style="padding: 2px 10px;">Источник A</th> <th style="padding: 2px 10px;">Источник B</th> <th style="padding: 2px 10px;">Результат</th> </tr> </thead> <tbody> <tr><td style="padding: 2px 10px;">0</td><td style="padding: 2px 10px;">0</td><td style="padding: 2px 10px;">0</td></tr> <tr><td style="padding: 2px 10px;">1</td><td style="padding: 2px 10px;">0</td><td style="padding: 2px 10px;">0</td></tr> <tr><td style="padding: 2px 10px;">0</td><td style="padding: 2px 10px;">1</td><td style="padding: 2px 10px;">0</td></tr> <tr><td style="padding: 2px 10px;">1</td><td style="padding: 2px 10px;">1</td><td style="padding: 2px 10px;">1</td></tr> </tbody> </table>	Источник A	Источник B	Результат	0	0	0	1	0	0	0	1	0	1	1	1
Источник A	Источник B	Результат																
0	0	0																
1	0	0																
0	1	0																
1	1	1																
NOT NOT Source D9:3 00FF Dest D9:5 FF00	Побитное отрицание NOT		Когда входные условия истинны, процессор выполняет побитное НЕ источника (D9:3) и сохраняет результат в назначении (D9:5). Таблица истинности для функции НЕ: <table style="margin-left: auto; margin-right: auto; border-collapse: collapse;"> <thead> <tr> <th style="padding: 2px 10px;">Источник</th> <th style="padding: 2px 10px;">Результат</th> </tr> </thead> <tbody> <tr><td style="padding: 2px 10px;">0</td><td style="padding: 2px 10px;">1</td></tr> <tr><td style="padding: 2px 10px;">1</td><td style="padding: 2px 10px;">0</td></tr> </tbody> </table>	Источник	Результат	0	1	1	0									
Источник	Результат																	
0	1																	
1	0																	
OR BITWISE INCLUSIVE OR Source A D9:3 3F37 Source B D9:4 00FF Dest D9:5 3FFF	Побитное ИЛИ OR		Когда входные условия истинны, процессор выполняет побитное ИЛИ между источником A (D9:3) и источником B (D9:4) и сохраняет результат в назначении (D9:5). Таблица истинности для функции ИЛИ: <table style="margin-left: auto; margin-right: auto; border-collapse: collapse;"> <thead> <tr> <th style="padding: 2px 10px;">Источник A</th> <th style="padding: 2px 10px;">Источник B</th> <th style="padding: 2px 10px;">Результат</th> </tr> </thead> <tbody> <tr><td style="padding: 2px 10px;">0</td><td style="padding: 2px 10px;">0</td><td style="padding: 2px 10px;">0</td></tr> <tr><td style="padding: 2px 10px;">1</td><td style="padding: 2px 10px;">0</td><td style="padding: 2px 10px;">1</td></tr> <tr><td style="padding: 2px 10px;">0</td><td style="padding: 2px 10px;">1</td><td style="padding: 2px 10px;">1</td></tr> <tr><td style="padding: 2px 10px;">1</td><td style="padding: 2px 10px;">1</td><td style="padding: 2px 10px;">1</td></tr> </tbody> </table>	Источник A	Источник B	Результат	0	0	0	1	0	1	0	1	1	1	1	1
Источник A	Источник B	Результат																
0	0	0																
1	0	1																
0	1	1																
1	1	1																
XOR BITWISE EXCLUSIVE OR Source A D9:3 3F37 Source B D9:4 3F37 Dest D9:5 0000	Побитное исключающее ИЛИ XOR		Когда входные условия истинны, процессор выполняет побитное исключающее ИЛИ между источником A (D9:3) и источником B (D9:4) и сохраняет результат в назначении (D9:5). Таблица истинности для функции исключающего ИЛИ: <table style="margin-left: auto; margin-right: auto; border-collapse: collapse;"> <thead> <tr> <th style="padding: 2px 10px;">Источник A</th> <th style="padding: 2px 10px;">Источник B</th> <th style="padding: 2px 10px;">Результат</th> </tr> </thead> <tbody> <tr><td style="padding: 2px 10px;">0</td><td style="padding: 2px 10px;">0</td><td style="padding: 2px 10px;">0</td></tr> <tr><td style="padding: 2px 10px;">1</td><td style="padding: 2px 10px;">0</td><td style="padding: 2px 10px;">1</td></tr> <tr><td style="padding: 2px 10px;">0</td><td style="padding: 2px 10px;">1</td><td style="padding: 2px 10px;">1</td></tr> <tr><td style="padding: 2px 10px;">1</td><td style="padding: 2px 10px;">1</td><td style="padding: 2px 10px;">0</td></tr> </tbody> </table>	Источник A	Источник B	Результат	0	0	0	1	0	1	0	1	1	1	1	0
Источник A	Источник B	Результат																
0	0	0																
1	0	1																
0	1	1																
1	1	0																
Бит состояния	Описание																	
C	всегда сброшен																	
V	всегда сброшен																	
Z	устанавливается, если результат = 0, иначе сбрасывается																	
S	устанавливается, если старший бит (бит 15 в десятичной системе, 17 – в восьмеричной) установлен (1), иначе сбрасывается																	

Инструкции преобразования

Инструкция	Преобразование	Описание																						
<table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td style="padding: 2px;">FRD</td><td style="padding: 2px;"> </td></tr> <tr><td style="padding: 2px;">FROM BCD</td><td style="padding: 2px;"> </td></tr> <tr><td style="padding: 2px;">Source</td><td style="padding: 2px;">D9:3</td></tr> <tr><td style="padding: 2px;"></td><td style="padding: 2px;">0037</td></tr> <tr><td style="padding: 2px;">Dest</td><td style="padding: 2px;">N7:12</td></tr> <tr><td style="padding: 2px;"></td><td style="padding: 2px;">37</td></tr> </table>	FRD		FROM BCD		Source	D9:3		0037	Dest	N7:12		37	Преобразование из BCD FRD	Когда входные условия истинны, преобразует двоично-десятичное (BCD) значение источника (D9:3) в целое и сохраняет результат в назначении (N7:12). Источник должен быть в диапазоне 0-9999 (BCD). <table border="1" style="width: 100%; border-collapse: collapse; margin-top: 5px;"> <thead> <tr> <th style="text-align: left; padding: 2px;">Бит состояния</th> <th style="text-align: left; padding: 2px;">Описание</th> </tr> </thead> <tbody> <tr> <td style="padding: 2px;">C</td> <td style="padding: 2px;">всегда сбрасывается</td> </tr> <tr> <td style="padding: 2px;">V</td> <td style="padding: 2px;">всегда сбрасывается</td> </tr> <tr> <td style="padding: 2px;">Z</td> <td style="padding: 2px;">устанавливается, если результат равен нулю, иначе сбрасывается</td> </tr> <tr> <td style="padding: 2px;">S</td> <td style="padding: 2px;">всегда сбрасывается</td> </tr> </tbody> </table>	Бит состояния	Описание	C	всегда сбрасывается	V	всегда сбрасывается	Z	устанавливается, если результат равен нулю, иначе сбрасывается	S	всегда сбрасывается
FRD																								
FROM BCD																								
Source	D9:3																							
	0037																							
Dest	N7:12																							
	37																							
Бит состояния	Описание																							
C	всегда сбрасывается																							
V	всегда сбрасывается																							
Z	устанавливается, если результат равен нулю, иначе сбрасывается																							
S	всегда сбрасывается																							
<table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td style="padding: 2px;">TOD</td><td style="padding: 2px;"> </td></tr> <tr><td style="padding: 2px;">TO BCD</td><td style="padding: 2px;"> </td></tr> <tr><td style="padding: 2px;">Source</td><td style="padding: 2px;">N7:3</td></tr> <tr><td style="padding: 2px;"></td><td style="padding: 2px;">44</td></tr> <tr><td style="padding: 2px;">Dest</td><td style="padding: 2px;">D9:5</td></tr> <tr><td style="padding: 2px;"></td><td style="padding: 2px;">0044</td></tr> </table>	TOD		TO BCD		Source	N7:3		44	Dest	D9:5		0044	Преобразование в BCD TOD	Когда входные условия истинны, преобразует целое значение источника (N7:3) в двоично-десятичный (BCD) формат и сохраняет результат в назначении (D9:5). <table border="1" style="width: 100%; border-collapse: collapse; margin-top: 5px;"> <thead> <tr> <th style="text-align: left; padding: 2px;">Бит состояния</th> <th style="text-align: left; padding: 2px;">Описание</th> </tr> </thead> <tbody> <tr> <td style="padding: 2px;">C</td> <td style="padding: 2px;">всегда сбрасывается</td> </tr> <tr> <td style="padding: 2px;">V</td> <td style="padding: 2px;">устанавливается, если значение источника отрицательно или больше, чем 9999 (т.е. за диапазоном 0-9999)</td> </tr> <tr> <td style="padding: 2px;">Z</td> <td style="padding: 2px;">устанавливается, если результат равен нулю, иначе сбрасывается</td> </tr> <tr> <td style="padding: 2px;">S</td> <td style="padding: 2px;">всегда сбрасывается</td> </tr> </tbody> </table>	Бит состояния	Описание	C	всегда сбрасывается	V	устанавливается, если значение источника отрицательно или больше, чем 9999 (т.е. за диапазоном 0-9999)	Z	устанавливается, если результат равен нулю, иначе сбрасывается	S	всегда сбрасывается
TOD																								
TO BCD																								
Source	N7:3																							
	44																							
Dest	D9:5																							
	0044																							
Бит состояния	Описание																							
C	всегда сбрасывается																							
V	устанавливается, если значение источника отрицательно или больше, чем 9999 (т.е. за диапазоном 0-9999)																							
Z	устанавливается, если результат равен нулю, иначе сбрасывается																							
S	всегда сбрасывается																							
<table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td style="padding: 2px;">DEG</td><td style="padding: 2px;"> </td></tr> <tr><td style="padding: 2px;">RADIANS TO DEGREE</td><td style="padding: 2px;"> </td></tr> <tr><td style="padding: 2px;">Source</td><td style="padding: 2px;">F8:7</td></tr> <tr><td style="padding: 2px;"></td><td style="padding: 2px;">0.7853982</td></tr> <tr><td style="padding: 2px;">Dest</td><td style="padding: 2px;">F8:8</td></tr> <tr><td style="padding: 2px;"></td><td style="padding: 2px;">45</td></tr> </table>	DEG		RADIANS TO DEGREE		Source	F8:7		0.7853982	Dest	F8:8		45	Преобразовать в градусы DEG	Когда входные условия истинны, преобразует радианы (значение источника) в градусы и сохраняет результат в назначении (источник умножается на 180/π). <table border="1" style="width: 100%; border-collapse: collapse; margin-top: 5px;"> <thead> <tr> <th style="text-align: left; padding: 2px;">Бит состояния</th> <th style="text-align: left; padding: 2px;">Описание</th> </tr> </thead> <tbody> <tr> <td style="padding: 2px;">C</td> <td style="padding: 2px;">всегда сбрасывается</td> </tr> <tr> <td style="padding: 2px;">V</td> <td style="padding: 2px;">устанавливается при переполнении, иначе сбрасывается</td> </tr> <tr> <td style="padding: 2px;">Z</td> <td style="padding: 2px;">устанавливается, если результат равен нулю, иначе сбрасывается</td> </tr> <tr> <td style="padding: 2px;">S</td> <td style="padding: 2px;">устанавливается, если результат отрицателен, иначе сбрасывается</td> </tr> </tbody> </table>	Бит состояния	Описание	C	всегда сбрасывается	V	устанавливается при переполнении, иначе сбрасывается	Z	устанавливается, если результат равен нулю, иначе сбрасывается	S	устанавливается, если результат отрицателен, иначе сбрасывается
DEG																								
RADIANS TO DEGREE																								
Source	F8:7																							
	0.7853982																							
Dest	F8:8																							
	45																							
Бит состояния	Описание																							
C	всегда сбрасывается																							
V	устанавливается при переполнении, иначе сбрасывается																							
Z	устанавливается, если результат равен нулю, иначе сбрасывается																							
S	устанавливается, если результат отрицателен, иначе сбрасывается																							

Инструкция		Описание										
RAD DEGREES TO RADIAN Source N7:9 45 Dest F8:10 0.7853982	Преобразовать в радианы RAD	<p>Когда входные условия истинны, преобразует градусы (значение источника) в радианы и сохраняет результат в назначении (источник умножается на $\pi/180$).</p> <table border="1"> <thead> <tr> <th>Бит состояния</th> <th>Описание</th> </tr> </thead> <tbody> <tr> <td>C</td> <td>всегда сбрасывается</td> </tr> <tr> <td>V</td> <td>устанавливается при переполнении, иначе сбрасывается</td> </tr> <tr> <td>Z</td> <td>устанавливается, если результат сбрасывается, иначе сбрасывается</td> </tr> <tr> <td>S</td> <td>устанавливается, если результат отрицателен, иначе сбрасывается</td> </tr> </tbody> </table>	Бит состояния	Описание	C	всегда сбрасывается	V	устанавливается при переполнении, иначе сбрасывается	Z	устанавливается, если результат сбрасывается, иначе сбрасывается	S	устанавливается, если результат отрицателен, иначе сбрасывается
Бит состояния	Описание											
C	всегда сбрасывается											
V	устанавливается при переполнении, иначе сбрасывается											
Z	устанавливается, если результат сбрасывается, иначе сбрасывается											
S	устанавливается, если результат отрицателен, иначе сбрасывается											

Инструкции перемещения и изменения битов

Инструкция		Описание										
MOV MOVE Source N7:3 20 Dest F8:12 20.000000	Перемещение MOV	<p>Когда входные условия истинны, перемещает копию значения источника (N7:3) в назначение (F8:12), преобразуя один тип данных в другой. Первоначальное значение назначения уничтожается.</p> <table border="1"> <thead> <tr> <th>Бит состояния</th> <th>Описание</th> </tr> </thead> <tbody> <tr> <td>C</td> <td>всегда сбрасывается</td> </tr> <tr> <td>V</td> <td>устанавливается в случае переполнения при преобразовании из плавающей запятой в целое, иначе сбрасывается</td> </tr> <tr> <td>Z</td> <td>устанавливается, если результат равен нулю, иначе сбрасывается</td> </tr> <tr> <td>S</td> <td>устанавливается, если старший бит результата установлен (1), иначе сбрасывается</td> </tr> </tbody> </table>	Бит состояния	Описание	C	всегда сбрасывается	V	устанавливается в случае переполнения при преобразовании из плавающей запятой в целое, иначе сбрасывается	Z	устанавливается, если результат равен нулю, иначе сбрасывается	S	устанавливается, если старший бит результата установлен (1), иначе сбрасывается
Бит состояния	Описание											
C	всегда сбрасывается											
V	устанавливается в случае переполнения при преобразовании из плавающей запятой в целое, иначе сбрасывается											
Z	устанавливается, если результат равен нулю, иначе сбрасывается											
S	устанавливается, если старший бит результата установлен (1), иначе сбрасывается											
MVM MASKED MOVE Source D9:3 478F Mask D9:5 00FF Dest D9:12 008F	Маскированное перемещение MVM	<p>Когда входные условия истинны, пропускает значение источника (D9:3) через маску (D9:5) и сохраняет результат в назначении (D9:12). Первоначальное значение назначения уничтожается.</p> <table border="1"> <thead> <tr> <th>Бит состояния</th> <th>Описание</th> </tr> </thead> <tbody> <tr> <td>C</td> <td>всегда сбрасывается</td> </tr> <tr> <td>V</td> <td>всегда сбрасывается</td> </tr> <tr> <td>Z</td> <td>устанавливается, если результат равен нулю, иначе сбрасывается</td> </tr> <tr> <td>S</td> <td>устанавливается, если старший бит результата установлен (1), иначе сбрасывается</td> </tr> </tbody> </table>	Бит состояния	Описание	C	всегда сбрасывается	V	всегда сбрасывается	Z	устанавливается, если результат равен нулю, иначе сбрасывается	S	устанавливается, если старший бит результата установлен (1), иначе сбрасывается
Бит состояния	Описание											
C	всегда сбрасывается											
V	всегда сбрасывается											
Z	устанавливается, если результат равен нулю, иначе сбрасывается											
S	устанавливается, если старший бит результата установлен (1), иначе сбрасывается											

Инструкция	Описание
BTD BITFIELD DISTRIBUTOR Source N7:3 0 Source bit 3 Dest N7:4 0 Dest bit 10 Length 6	Распределение битов BTD Когда входные условия истинны, процессор копирует количество битов, указанное в длине (length), начиная с бита 3 (source bit) источника (N7:3), и помещает их в назначение (N7:4), начиная с бита 10 (dest bit).

Файловые инструкции

Инструкция	Описание
FAL FILE ARITH/LOGICAL Control R6:1 Length 8 Position 0 Mode ALL Dest #N15:10 Expression #N14:0 - 256	Арифметика и логика файла FAL Биты состояния: EN – разрешен DN – завершен ER – ошибка Когда входные условия изменяются из ложных в истинные, процессор читает 8 элементов N14:0 и вычитает 256 (константу) из каждого элемента. Этот пример показывает, что результат будет хранится в восьми элементах, начиная с N15:10. Управляющий элемент R6:1 контролирует операцию. Режим определяет, выполняет ли процессор выражение во всех элементах в файлах (ALL) за одно сканирование программы, один элемент в файлах (INC) за каждый переход от ложного к истинному состоянию, или указанное количество элементов (NUM) за сканирование. Инструкция FAL может выполнять операции: сложения (+), вычитания (-), умножения (*), деления (/), преобразования из двоично-десятичного числа (FRD), преобразования в двоично-десятичное число (TOD), вычисления квадратного корня (SQR), логического И (AND), логического ИЛИ (OR), логического НЕ (NOT), исключающего ИЛИ (XOR), отрицания (-), очистки (0), перемещения, и новые математические инструкции (смотри список CPT).
FSC FILE SEARCH/COMPARE Control R9:0 Length 90 Position 0 Mode 10 Expression #B4:0 <> #B5:0	Файловые поиск и сравнение FSC Биты состояния: EN – разрешен DN – завершен ER – ошибка IN – запрет FD – обнаружен Когда входные условия изменяются из ложных в истинные, процессор выполняет сравнение на НЕ РАВНО в 10 элементах между файлами B4:0 и B5:0. Режим определяет, выполняет ли процессор выражение во всех элементах в файлах (ALL) за одно сканирование программы, один элемент в файлах (INC) за каждый переход от ложного к истинному состоянию, или указанное количество элементов (NUM) за сканирование. Управляющий элемент R6:1 контролирует операцию. Когда соответствующие исходные элементы не равны (элемент B4:4 и B5:4 в этом примере), процессор останавливает поиск и устанавливает биты обнаружения .FD и запрета .IN так, что ваша релейно-контактная программа может предпринять соответствующее действие. Чтобы продолжить сравнение поиска, Вы должны сбросить бит .IN. Для ознакомления со списком доступных сравнений, смотри варианты, указанные для инструкции CMP.

Инструкция	Описание
COP COPY FILE Source #N7:0 Dest #N12:0 Length 5	<p>Копирование файла COP</p> <p>Когда входные условия истинны, процессор копирует содержимое исходного файла (N7) в файл назначения (N12). Исходные данные не изменяются. Инструкция COP копирует количество элементов, указанное в длине (length). В противоположность инструкции MOV, эта инструкция не производит преобразования типов данных.</p>
FLL FILL FILE Source N10:6 Dest #N12:0 Length 5	<p>Заполнение файла FLL</p> <p>Когда входные условия истинны, процессор копирует величину в источнике (N10:6) в элементы назначения (N12). Инструкция FLL заполняет только то количество элементов, которое определено в длине (length). В противоположность инструкции MOV, эта инструкция не производит преобразования типов данных.</p>

Инструкции диагностики

Инструкция	Описание
FBC FILE BIT COMPARISON Source #I:31 Reference #B3:1 Result #N7:0 Cmp Control R6:4 Length 48 Position 0 Result Control R6:5 Length 10 Position 0	<p>Побитное сравнение файлов FBC</p> <p>Биты состояния: EN – разрешен DN – завершен ER – ошибка IN – запрет FD – обнаружен</p> <p>Когда входные условия изменяются из ложных в истинные, процессор сравнивает количество битов определенных в длине (Cmp Control, length = 48) файла-источника (#I:031) с битами в файле-этalone (#B3:1). Процессор загружает результаты (номера несовпадающих битов) в файл назначения (#N7:0). Файл R6:4 управляет сравнением, а файл R6:5 управляет файлом, который содержит результаты. Файл, содержащий результаты может содержать до 10 (число, заданное в области длины) несовпадений между сравниваемыми файлами. Примечание: чтобы избежать возможной ошибки выполнения этой инструкции, добавьте релейно-контактную цепь, которая очищает S:24 (индексное смещение адресации) непосредственно перед инструкцией FBC.</p>
DDT DIAGNOSTIC DETECT Source #I:30 Reference #B3:1 Result #N10:0 Cmp Control R6:0 Length 20 Position 0 Result Control R6:1 Length 5 Position 0	<p>Диагностика DDT</p> <p>Биты состояния: EN – разрешен DN – завершен ER – ошибка IN – запрет FD – обнаружен</p> <p>Когда входные условия изменяются из ложных в истинные, процессор сравнивает количество битов определенных в длине (Cmp Control, length = 20) файла-источника (#I:030) с битами в файле-этalone (#B3:1). Процессор загружает результаты (номера несовпадающих битов) в файл назначения (#N10:0). Файл R6:0 управляет сравнением, а файл R6:1 управляет файлом, который содержит результаты. Файл, содержащий результаты может содержать до 5 (число, заданное в области длины) несовпадений между сравниваемыми файлами. Процессор копирует биты источника в эталонный файл для следующего сравнения. Различие между DDT и инструкцией FBC – то, что всякий раз, когда инструкция DDT находит несовпадение, процессор изменяет бит эталона, в соответствии с исходным битом. Вы можете использовать инструкцию DDT, чтобы обновить ваш файл-эталон, для отображения изменения состояния машины или процесса. Примечание: чтобы избежать возможной ошибки выполнения этой инструкции, добавьте релейно-контактную цепь, которая очищает S:24 (индексное смещение адресации) непосредственно перед инструкцией DDT.</p>

Инструкция	Описание
DTR DATA TRANSITION Source I:002 Mask 0FFF Reference N63:11	Изменение данных DTR Инструкция DTR сравнивает биты источника (I:002) через маску (0FFF) с битами эталона (N63:11). Когда маскированный источник отличается от эталона, инструкция истинна в течение 1 сканирования. Исходные биты записываются в эталон для следующего сравнения. Когда замаскированный источник и ссылка одинаковы, инструкция остается ложной.

Инструкции регистрового сдвига

Инструкция	Описание
BSL BIT SHIFT LEFT File #B3:1 Control R6:53 Bit Address I:022/12 Length 5	Сдвиг битов влево BSL Биты состояния: EN – разрешен DN – завершен ER – ошибка UL – выгрузка Когда входные условия изменяются из ложных в истинные, инструкция BSL сдвигает количество битов, определенных длиной (5) в файле (B3), начиная с бита 16 (B3:1/0 = B3/16), влево на одну битовую позицию. Бит-источник (I:022/12) копируется в первую битовую позицию, B3:1/0 (B3/16). Пятый бит, B3:1/4 (B3/20), сдвигается в бит UL структуры управления (R6:53).
BSR BIT SHIFT RIGHT File #B3:2 Control R6:54 Bit Address I:023/06 Length 3	Сдвиг битов вправо BSR Биты состояния: EN – разрешен DN – завершен ER – ошибка UL – выгрузка Когда входные условия изменяются из ложных в истинные, инструкция BSR сдвигает количество битов, определенных длиной (3) в файле (B3), начиная с B3:2/0 (=B3/32), вправо на одну битовую позицию. Бит-источник (I:023/06) копируется в третью битовую позицию, B3/34. Первый бит, (B3/32), сдвигается в бит UL структуры управления (R6:54).
FFL FIFO LOAD Source N60:1 FIFO #N60:3 Control R6:51 Length 64 Position 0	Загрузить в очередь FFL Биты состояния: EN – загрузка разрешена DN – завершен EM – пуст Когда входные условия изменяются из ложных в истинные, процессор загружает N60:1 в следующий доступный элемент файла FIFO #N60:3, туда, куда указывает R6:51. Всякий раз, когда цепь изменяется от лжи к истине, процессор загружает следующий элемент. Когда файл FIFO (очередь) полон, (загружено 64 слова), устанавливается бит DN. Смотри страницу 24-8 для описания операции пресканирования этой инструкции.
FFU FIFO UNLOAD Source N60:3 FIFO #N60:2 Control R6:51 Length 64 Position 0	Выгрузить из очереди FFU Биты состояния: EU – выгрузка разрешена DN – завершен EM – пуст Когда входные условия изменяются из ложных в истинные, процессор выгружает элемент из #N60:3 в N60:2. Всякий раз, когда цепь изменяется от лжи к истине, процессор выгружает следующую величину. Все данные в файле #N60:3 сдвигаются на одну позицию в сторону N60:3. Когда файл пуст, устанавливается бит EM. Смотри страницу 24-8 для описания операции пресканирования этой инструкции.

Инструкция	Описание	
LFL LIFO LOAD Source N70:1 FIFO #N70:3 Control R6:61 Length 64 Position 0	Загрузить в стек LFL Биты состояния: EN – загрузка разрешена DN – завершен EM – пуст	Когда входные условия изменяются из ложных в истинные, процессор загружает N70:1 в следующий доступный элемент файла LIFO #N70:3, туда, куда указывает R6:61. Всякий раз, когда цепь изменяется от лжи к истине, процессор загружает следующий элемент. Когда файл LIFO (стек) полон, (загружено 64 слова), устанавливается бит DN. Смотри страницу 24-8 для описания операции пресканирования этой инструкции.
LFU LIFO UNLOAD Source N70:3 FIFO #N70:2 Control R6:61 Length 64 Position 0	Выгрузить из стека LFU Биты состояния: EU – выгрузка разрешена DN – завершен EM – пуст	Когда входные условия изменяются из ложных в истинные, процессор выгружает элемент из #N70:3 в N70:2. Всякий раз, когда цепь изменяется от лжи к истине, процессор выгружает следующую величину. Когда файл пуст, устанавливается бит EM. Смотри страницу 24-8 для описания операции пресканирования этой инструкции.

Инструкции секвенсора

Инструкция	Описание	
SQL SEQUENCER INPUT File #N7:11 Mask FFF0 Source I:031 Control R6:21 Length 4 Position 0	Вход секвенсора SQL Биты состояния: EN – разрешен DN – завершен EN – ошибка	Инструкция SQL фильтрует данные образа входов источника (I:031) через маску (FFF0) и сравнивает результат с эталонными данными (#N7:11) для проверки равенства этих величин. Операция контролируется информацией в файле управления R6:21. Когда состояние всех незамаскированных битов слова, на которое указывает элемент управления R6:21, соответствуют битам эталона, цепь остается истинной, если этой инструкции предшествуют истинные условия цепи.
SQL SEQUENCER LOAD File #N7:20 Source I:002 Control R6:22 Length 5 Position 0	Загрузка секвенсора SQL Биты состояния: EN – разрешен DN – завершен EN – ошибка	Инструкция SQL загружает данные в файл секвенсора (#N7:20) из слова-источника (I:002), проходя через элементы, количество которых определяется длиной (5) источника (I:002), начиная с позиции (0). Операция контролируется информацией в файле управления R6:22. Когда цепь изменяется от ложного состояния к истинному, инструкция SQL переходит к следующему шагу в файле секвенсора и загружает в него данные при каждом сканировании, пока цепь остается истинной. Смотри страницу 24-8 для описания операции пресканирования этой инструкции.
SQO SEQUENCER OUTPUT File #N7:1 Mask 0F0F Dest O:014 Control R6:20 Length 4 Position 0	Выход секвенсора SQO Биты состояния: EN – разрешен DN – завершен EN – ошибка	Когда цепь изменяется от ложного состояния к истинному, инструкция SQO переходит к следующему шагу в файле секвенсора (#N7:1). Данные из файла секвенсора передаются через маску (0F0F) в назначение (O:014) при каждом сканировании, пока цепь остается истинной. Смотри страницу 24-8 для описания операции пресканирования этой инструкции.

Инструкции управления программой

Инструкция		Описание
_____ (MCR) _____	Сброс основного управления MCR	Если входные условия истинны, программа сканирует цепи между цепями с инструкциями MCR и обрабатывает выходы нормально. Если входное условие является ложью, цепи между цепями с инструкциями MCR выполняются как ложные.
10 _____ (JMP) _____	Прыжок JMP	Если входные условия истинны, процессор пропускает цепи, переходя к цепи отмеченной меткой (10).
10 _____ [LBL] _____	Метка LBL	Когда процессор читает инструкцию JMP, которая соответствует метке 10, процессор перескакивает на цепь, содержащую метку и начинает ее выполнять. Важно: Должна быть первой инструкцией в цепи.
FOR FOR Label Number 0 Index N7:0 Initial Value 0 Terminal Value 10 Step Size 1	Цикл ДЛЯ FOR	Процессор выполняет цепи между FOR и инструкцией NXT многократно в одном сканировании программы, пока он не достигнет конечной величины (10), или пока инструкция BRK не завершит операцию аварийно. Размер шага (step size) - степень увеличения индекса (index) цикла . Смотри страницу 24-8 для описания операции пресканирования этой инструкции.
NXT NEXT Label Number 0	Продолжить цикл NXT	Инструкция NXT возвращает процессор к соответствующей инструкции FOR, идентифицируемой номером метки, определенной в инструкции FOR. NXT должна программироваться в безусловной цепи, которая является последней цепью, повторяющейся в цикле FOR-NEXT.
_____ [BRK] _____	Прервать цикл BRK	Когда входные условия становятся верными, инструкция BRK аварийно завершает цикл FOR-NEXT.
JSR JUMP TO SUBROUTINE Program File 90 Input par N16:23 Input par N16:24 Input par 231 Return par N19:11 Return par N19:12	Переход к подпрограмме JSR	Если входные условия истинны, процессор начинает выполнять файл подпрограммы (90). Процессор передает входные параметры (N16:23, N16:24 и 231) подпрограмме, а инструкция RET передает возвращаемые параметры (N19:11, N19:12) обратно основной программе туда, где процессор встретил инструкцию JSR.
SBR SUBROUTINE Input par N43:0 Input par N43:1 Input par N43:2	Подпрограмма SBR	Инструкция SBR является первой инструкцией в файле подпрограммы. Эта инструкция идентифицирует входные параметры (N43:0, N43:1 и N43:2), которые процессор получает от соответствующей инструкции JSR. Вам не нужна инструкция SBR, если вы не передаете входные параметры в подпрограмму.
RET RETURN Return par N43:3 Return par N43:4	Возврат RET	Если входные условия истинны, инструкция RET заканчивает подпрограмму и сохраняет параметры возврата (N43:3, N43:4), которые должны быть возвращены инструкции JSR в основной программе.
_____ [AFI] _____	Всегда ложна AFI	Инструкция AFI блокирует цепь (то есть, цепь всегда ложна).

Инструкция		Описание
——(TND)——	Временное окончание TND	Если входные условия истинны, инструкция TND останавливает сканирование процессором остальной части программы (то есть, эта инструкция временно оканчивает программу).
B3 ——[ONS]—— 110	Одно включение ONS	Если входные условия, предшествующие инструкции ONS на той же цепи изменяются от ложных к истинным, инструкция ONS обуславливает цепь так, что выход истинен в течение одного сканирования. Цепь ложна при последующем сканировании. Смотрите страницу 24-8 для описания операции пресканирования этой инструкции.
OSF ONE SHOT FALLING Storage Bit B3/0 Output Bit 15 Output Word N7:0	Одно включение по заднему фронту OSF Биты состояния: OB – выходной бит SB – бит хранения	Инструкция OSF переключает событие для однократного наступления. Используйте инструкцию OSF всякий раз, когда событие должно наступить, базирываясь на изменении состояния цепи от истинного к ложному, но не на результирующем состоянии цепи. Выходной бит (N7:0/15) установлен (1) для одного сканирования программы, когда цепь изменяет состояние от истинного к ложному. Смотрите страницу 24-8 для описания операции пресканирования этой инструкции.
OSR ONE SHOT RISING Storage Bit B3/0 Output Bit 15 Output Word N7:0	Одно включение по переднему фронту OSR Биты состояния: OB – выходной бит SB – бит хранения	Инструкция OSR переключает событие для однократного наступления. Используйте инструкцию OSR всякий раз, когда событие должно наступить, базирываясь на изменении состояния цепи от ложного к истинному, но не на результирующем состоянии цепи. Выходной бит (N7:0/15) установлен (1) для одного сканирования программы, когда цепь изменяет состояние от ложного к истинному. Смотрите страницу 24-8 для описания операции пресканирования этой инструкции.
SFR SFC Reset Prog File Number 3 Restart Step At	Сброс ПФС SFR	Инструкция SFR сбрасывает логику последовательной функциональной схемы. Когда инструкция SFR становится верной, процессор выполняет последнее сканирование/постсканирование во всех активных шагах и действиях в выбранном файле, и, затем, сбрасывает логику в ПФС в следующем сканировании программы. ПФС остается в состоянии сброса, пока инструкция SFR не станет ложной.
——(EOT)——	Окончание перехода EOT	Инструкция EOT должна быть последней инструкцией в файле перехода. Если вы не используете инструкцию EOT, процессор всегда оценивает переход как верный. Смотрите страницу 24-8 для описания операции пресканирования этой инструкции.
——(UID)——	Запретить прерывания пользователя UID	Инструкция UID временно запрещает выполняемым по прерыванию релейно-контактным программам (например, STI или PII) прерывать программу, выполняющуюся в текущий момент.
——(UIE)——	Разрешить прерывания пользователя UIE	Инструкция UIE вновь разрешает выполняемым по прерыванию релейно-контактным программам прерывать программу, выполняющуюся в текущий момент.

Инструкции передачи сообщений и управления процессами

Инструкция	Описание																		
<div style="border: 1px solid black; padding: 5px; margin-bottom: 10px;"> PID PID Control Block PD10:0 Process Variable N15:13 Tieback N15:14 Control Variable N20:21 </div>	<p>ПИД-регулятор PID</p> <p>Биты состояния: EN – разрешен DN – завершен (только для блоков управления типа N)</p>																		
<div style="border: 1px solid black; padding: 5px; margin-bottom: 10px;"> MSG READ/WRITE MESSAGE Control Block MG7:10 </div> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 15%;">№ бита</th> <th style="width: 85%;">Бит состояния</th> </tr> </thead> <tbody> <tr><td style="text-align: center;">15</td><td>EN – разрешен</td></tr> <tr><td style="text-align: center;">14</td><td>ST – старт</td></tr> <tr><td style="text-align: center;">13</td><td>DN – завершен</td></tr> <tr><td style="text-align: center;">12</td><td>ER – ошибка</td></tr> <tr><td style="text-align: center;">11</td><td>CO – непрерывный</td></tr> <tr><td style="text-align: center;">10</td><td>EW – разрешен-ожидание</td></tr> <tr><td style="text-align: center;">9</td><td>NR – нет ответа</td></tr> <tr><td style="text-align: center;">8</td><td>TO – тайм-аут</td></tr> </tbody> </table>	№ бита	Бит состояния	15	EN – разрешен	14	ST – старт	13	DN – завершен	12	ER – ошибка	11	CO – непрерывный	10	EW – разрешен-ожидание	9	NR – нет ответа	8	TO – тайм-аут	<p>Сообщение MSG</p>
№ бита	Бит состояния																		
15	EN – разрешен																		
14	ST – старт																		
13	DN – завершен																		
12	ER – ошибка																		
11	CO – непрерывный																		
10	EW – разрешен-ожидание																		
9	NR – нет ответа																		
8	TO – тайм-аут																		

Блок управления (PD10:0) содержит информацию для инструкции PID. PID получает переменную процесса из N15:13 и посылает выход PID в N20:21. Обратная связь в N15:14 поддерживает работу со станцией ручного управления.

Если вы используете блок управления N, цепь должна переходить от ложного состояния к истинному для выполнения инструкции. Если вы используете блок управления PD, то бит завершения отсутствует. Также, входные условия цепи должны быть истинными. Смотрите страницу 24-8 для описания операции пресканирования этой инструкции.

Если входные условия изменяют состояние от ложного к истинному, то данные пересылаются согласно параметрам инструкции, которые вы установили, когда вводили инструкцию сообщения. Блок управления (MG7:10) содержит состояние и параметры инструкции. Вы можете также использовать блоки управления N.

Для непрерывных MSGs, состояние цепи должно быть истинным только в течение одного сканирования. Смотрите страницу 24-8 для описания операции пресканирования этой инструкции.

Инструкции блок-трансферов

Целый (N) блок управления

Смещение слова	Описание
0	биты состояния (смотрите ниже)
1	число запрашиваемых слов
2	число переданных слов
3	номер файла
4	номер элемента

Блок управления блок-трансфером (BT)

Мнемоника слова	Описание
.EN до .RW	биты состояния
.RLEN	число запрашиваемых слов
.DLEN	число переданных слов/код ошибки
.FILE	номер файла
.ELEM	номер элемента
.RGS	рэк/группа/слот

Слово 0

15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
EN	ST	DN	ER	CO	EW	NR	TO	RW	рэк			группа		слот	

Инструкция	Описание
BTR BLOCK TRANSFER READ Rack 1 Group 0 Module 0 Control Block BT11:100 Data File N10:110 Length 40 Continuous Y	Блок-трансфер / чтение BTR Если входные условия изменяют состояние от ложного к истинному, то инициализируется блок-трансфер/чтение для модуля Вх/Вых, расположенного по адресу: рэк 1, группа 0, модуль 0. Блок управления (BT11:100, файл из 6 слов) содержит состояние пересылки. Файл данных (N10:110) – это место, где сохраняются данные, прочитанные из модуля. Длина BT (40) идентифицирует количество слов в пересылке. Не непрерывный блок-трансфер ставится в очередь и выполняется только один раз при переходе цепи от ложного состояния к истинному; непрерывный блок-трансфер ставится в очередь периодически. Вы можете также использовать для блока управления данные типа N. Смотрите страницу 24-8 для описания операции пресканирования этой инструкции.

Процессоры PLC-5/30, -5/40, -5/40E, -5/40L, -5/60, -5/60L, -5/80, -5/80E		Процессоры PLC-5/40, -5/40E, -5/40L, -5/60, -5/60L, -5/80, -5/80E		Процессоры PLC-5/60, -5/60L, -5/80, -5/80E	
№ бита	Очередь BT	№ бита	Очередь BT	№ бита	Очередь BT
S:7	полна для рэка	S:32	полна для рэка	S:34	полна для рэка
08 ¹	0	08	10	08	20
09 ¹	1	09	11	09	21
10 ¹	2	10	12	10	22
11 ¹	3	11	13	11	23
12	4	12	14	12	24
13	5	13	15	13	25
14	6	14	16	14	26
15	7	15	17	15	27

¹ Также для процессоров PLC-5/11, -5/20, -5/20E

Инструкция	Описание																		
<table border="1"> <tr><td>BTW</td><td></td></tr> <tr><td>BLOCK TRANSFER WRITE</td><td></td></tr> <tr><td>Rack</td><td>1</td></tr> <tr><td>Group</td><td>0</td></tr> <tr><td>Module</td><td>0</td></tr> <tr><td>Control Block</td><td>BT11:0</td></tr> <tr><td>Data File</td><td>N10:10</td></tr> <tr><td>Length</td><td>40</td></tr> <tr><td>Continuous</td><td>Y</td></tr> </table>	BTW		BLOCK TRANSFER WRITE		Rack	1	Group	0	Module	0	Control Block	BT11:0	Data File	N10:10	Length	40	Continuous	Y	<p>Блок-трансфер / запись BTW</p> <p>Если входные условия изменяют состояние от ложного к истинному, то инициализируется блок-трансфер/запись для модуля Вх/Вых, расположенного по адресу: рэк 1, группа 0, модуль 0. Блок управления (BT11:0, файл из 6 слов) содержит состояние пересылки. Файл данных (N10:10) – это место, где хранятся данные, предназначенные для записи в модуль. Длина BT (40) идентифицирует количество слов в пересылке. Не непрерывный блок-трансфер ставится в очередь и выполняется только один раз при переходе цепи от ложного состояния к истинному; непрерывный блок-трансфер ставится в очередь периодически.</p> <p>Вы можете также использовать для блока управления данные типа N. Смотрите страницу 24-8 для описания операции пресканирования этой инструкции.</p>
BTW																			
BLOCK TRANSFER WRITE																			
Rack	1																		
Group	0																		
Module	0																		
Control Block	BT11:0																		
Data File	N10:10																		
Length	40																		
Continuous	Y																		

Инструкции ASCII

Биты состояния:

EN – разрешен	EM – пуст
DN – завершен	EU – очередь
ER – ошибка	FD – найден

Инструкция	Описание																				
<table border="1"> <tr><td>ABL</td><td></td></tr> <tr><td>ASCII TEST FOR LINE</td><td></td></tr> <tr><td>Channel</td><td>0</td></tr> <tr><td>Control</td><td>R6:32</td></tr> <tr><td>Characters</td><td></td></tr> </table>	ABL		ASCII TEST FOR LINE		Channel	0	Control	R6:32	Characters		<p>ASCII тест на наличие строки ABL</p> <p>Если входные условия изменяют состояние от ложного к истинному, то процессор сообщает о количестве символов в буфере, вплоть до символов окончания строки, включая их, и помещает эту величину в слово позиции структуры управления (R6:32.POS). Процессор также отображает эту величину в текстовом поле дисплея.</p> <p>Смотрите страницу 24-8 для описания операции пресканирования этой инструкции.</p>										
ABL																					
ASCII TEST FOR LINE																					
Channel	0																				
Control	R6:32																				
Characters																					
<table border="1"> <tr><td>ACB</td><td></td></tr> <tr><td>ASCII CHARS IN BUFFER</td><td></td></tr> <tr><td>Channel</td><td>0</td></tr> <tr><td>Control</td><td>R6:32</td></tr> <tr><td>Characters</td><td></td></tr> </table>	ACB		ASCII CHARS IN BUFFER		Channel	0	Control	R6:32	Characters		<p>ASCII символов в строке ACB</p> <p>Если входные условия изменяют состояние от ложного к истинному, то процессор сообщает об общем числе символов в буфере и помещает эту величину в слово позиции (.POS) структуры управления. Процессор также отображает эту величину в текстовом поле дисплея.</p> <p>Смотрите страницу 24-8 для описания операции пресканирования этой инструкции.</p>										
ACB																					
ASCII CHARS IN BUFFER																					
Channel	0																				
Control	R6:32																				
Characters																					
<table border="1"> <tr><td>ACI</td><td></td></tr> <tr><td>STRING TO INTEGER</td><td></td></tr> <tr><td>Source</td><td>ST38:90</td></tr> <tr><td>Dest</td><td>N7:123</td></tr> <tr><td></td><td>75</td></tr> </table>	ACI		STRING TO INTEGER		Source	ST38:90	Dest	N7:123		75	<p>Преобразование ASCII строки в целое ACI</p> <p>Если входные условия истинны, процессор преобразовывает строку в ST38:90 в целое и сохраняет результат в N7:123.</p> <table border="1"> <thead> <tr> <th>Бит состояния</th> <th>Описание</th> </tr> </thead> <tbody> <tr> <td>C</td> <td>устанавливается, если возник перенос во время преобразования; иначе сбрасывается</td> </tr> <tr> <td>V</td> <td>устанавливается, если источник > 32767 или < -32768; иначе сбрасывается</td> </tr> <tr> <td>Z</td> <td>устанавливается, если источник равен нулю; иначе сбрасывается</td> </tr> <tr> <td>S</td> <td>устанавливается, если результат отрицателен; иначе сбрасывается</td> </tr> </tbody> </table>	Бит состояния	Описание	C	устанавливается, если возник перенос во время преобразования; иначе сбрасывается	V	устанавливается, если источник > 32767 или < -32768; иначе сбрасывается	Z	устанавливается, если источник равен нулю; иначе сбрасывается	S	устанавливается, если результат отрицателен; иначе сбрасывается
ACI																					
STRING TO INTEGER																					
Source	ST38:90																				
Dest	N7:123																				
	75																				
Бит состояния	Описание																				
C	устанавливается, если возник перенос во время преобразования; иначе сбрасывается																				
V	устанавливается, если источник > 32767 или < -32768; иначе сбрасывается																				
Z	устанавливается, если источник равен нулю; иначе сбрасывается																				
S	устанавливается, если результат отрицателен; иначе сбрасывается																				

Инструкция		Описание
ACN STRING CONCATENATE Source A ST38:90 Source B ST37:91 Dest ST52:76	Объединение ASCII строк ACN	Если входные условия истинны, процессор объединяет строку в ST38:90 со строкой в ST37:91 и сохраняет результат в ST52:76.
AEX STRING EXTRACT Source ST38:40 Index 42 Number 10 Dest ST52:75	Извлечение ASCII строки AEX	Если входные условия истинны, процессор извлекает 10 символов начиная с 42-го символа ST38:40 и сохраняет результат в ST52:75.
AIC INTEGER TO STRING Source 876 Dest ST38:42	Преобразование целого в ASCII строку AIC	Если входные условия истинны, процессор преобразует величину 876 в строку и сохраняет результат в ST38:42.
AHL ASCII HANDSHAKE LINES Channel 0 AND Mask 0001 OR Mask 0003 Control R6:23 Channel Status	ASCII квитирование линий AHL Биты состояния: EN – разрешен DN – завершен ER – ошибка	Если входные условия изменяют состояние от ложного к истинному, процессор использует маски И и ИЛИ, чтобы определить установить или сбросить линии DTR (бит 0) и RTS (бит 1), или оставить их неизменными. Бит 0 и 1 маски И приводят к сбросу линии(й), если бит равен 1, и оставляет линию(и) неизменной(ыми), если он равен 0. Бит 0 и 1 маски ИЛИ приводят к установке линии(й), если бит равен 1, и оставляет линию(и) неизменной(ыми), если он равен 0. Смотрите страницу 24-8 для описания операции пресканирования этой инструкции.
ARD ASCII READ Channel 0 Dest ST52:76 Control R6:32 String Length 50 Characters Read	ASCII чтение ARD Биты состояния: EN – разрешен DN – завершен ER – ошибка UL – выгрузка EM – пуст EU – очередь	Если входные условия изменяют состояние от ложного к истинному, читает 50 символов из буфера и помещает их в ST52:76. Количество прочитанных символов сохраняется в R6:32.POS и отображается в поле (characters read) инструкции. Смотрите страницу 24-8 для описания операции пресканирования этой инструкции.
ARL ASCII READ LINE Channel 0 Dest ST50:72 Control R6:30 String Length 18 Characters Read	ASCII чтение линии ARD Биты состояния: EN – разрешен DN – завершен ER – ошибка UL – выгрузка EM – пуст EU – очередь	Если входные условия изменяют состояние от ложного к истинному, читает 18 символов (или до конца строки) из буфера и помещает их в ST50:72. Количество прочитанных символов сохраняется в R6:30.POS и отображается в поле (characters read) инструкции. Смотрите страницу 24-8 для описания операции пресканирования этой инструкции.

Инструкция		Описание
ASC STRING SEARCH Source ST38:40 Index 35 Search ST52:80 Result 42	Поиск ASCII строк ASC	Если входные условия истинны, ищет в ST52:80, начиная с 35-го символа, подстроку, находящуюся в ST38:40. В этом примере, строка была найдена по индексу 42. Если строка не найдена, устанавливается бит неосновной ошибки инструкции ASCII (S:17/8), а результат устанавливается в ноль.
ASR ASCII STRING COMPARE Source A ST37:42 Source B ST38:90	Сравнение ASCII строк ASR	Если строка в ST37:42 идентична строке в ST38:90, инструкция истинна. Имейте в виду, что это - входная инструкция. Строка неправильной длины вызывает установку бита неосновной ошибки инструкции ASCII (S:17/8), а инструкция является ложной.
AWA ASCII WRITE APPEND Channel 0 Source ST52:76 Control R6:32 String Length 50 Characters Sent	ASCII запись с добавлением AWA Биты состояния: EN – разрешен DN – завершен ER – ошибка UL – выгрузка EM – пуст EU – очередь	Если входные условия изменяют состояние от ложного к истинному, то читает 50 символов из ST52:76 и записывает их в канал 0, а также добавляет два символа, выбранные в конфигурации канала (умолчание – CR/LF). Количество посланных символов сохраняется в R6:32.POS и отображается в поле (characters sent) инструкции. Смотрите страницу 24-8 для описания операции пресканирования этой инструкции.
AWT ASCII WRITE Channel 0 Source ST37:40 Control R6:23 String Length 40 Characters Sent	ASCII запись AWT Биты состояния: EN – разрешен DN – завершен ER – ошибка UL – выгрузка EM – пуст EU – очередь	Если входные условия изменяют состояние от ложного к истинному, записывает 40 символов из ST37:40 в канал 0. Количество посланных символов сохраняется в R6:23.POS и отображается в поле (characters sent) инструкции. Смотрите страницу 24-8 для описания операции пресканирования этой инструкции.

Требования к памяти и времена выполнения

Инструкции бита и слова

Категория	Код	Название	Время выполнения (мкс) – целые операнды		Время выполнения (мкс) – плавающая запятая		Слов ¹ памяти
			Истина	Ложь	Истина	Ложь	
Релейные	XIC	проверка на включено	0.32	0.16			1 ²
	XIO	проверка на отключено	0.32	0.16			1 ²
	OTL	зафиксировать выход	0.48	0.16			1 ²
	OTU	расфиксировать выход	0.48	0.16			1 ²
	OTE	включить выход	0.48	0.48			1 ²
Ветвление		конец ветвления	0.16	0.16			1
		следующая ветвь					1
		начало ветвления					1
Таймеры и счетчики	TON	таймер на включение					2-3
		(база 0.01 с)	3.8	2.6			
	(база 1 с)	4.1	2.5				
	TOF	таймер на отключение					2-3
		(база 0.01 с)	2.6	3.2			
	(база 1 с)	2.6	3.2				
RTO	накапливающий таймер					2-3	
	(база 0.01 с)	3.8	2.4				
(база 1 с)	4.1	2.3					
CTU	прямой счетчик	3.4	3.4			2-3	
CTD	обратный счетчик	3.3	3.4			2-3	
RES	сброс	2.2	1.0			2-3	

¹ Используйте большее число для адресов за первыми 2048 словами в таблице данных процессора.

² Для каждого битового адреса выше первых 256 слов памяти в таблице данных, добавьте 0.16 мкс и 1 слово памяти.

Категория	Код	Название	Время выполнения (мкс) – целые операнды		Время выполнения (мкс) – плавающая запятая		Слов ¹ памяти
			Истина	Ложь	Истина	Ложь	
Арифметические	ADD	сложение	6.1	1.4	14.9	1.4	4-7
	SUB	вычитание	6.2	1.4	15.6	1.4	4-7
	MUL	умножение	9.9	1.4	18.2	1.4	4-7
	DIV	деление	12.2	1.4	23.4	1.4	4-7
	SQR	квадратный корень	9.9	1.3	35.6	1.3	3-5
	NEG	смена знака	4.8	1.3	6.0	1.3	3-5
	CLR	очистка	3.4	1.1	3.9	1.1	2-3
	AVE	усреднение файла	152+E*25.8	30	162+E*22.9	36	4-7
	STD	среднеквадратическое отклонение	262+E*92.5	34	295+E*85.5	34	4-7
	TOD	преобразование в BCD	7.8	1.3			3-5
	FRD	преобразование из BCD	8.1	1.3			3-5
	RAD	радианы	57.4	1.4	50.1	1.4	3-5
	DEG	градусы	55.9	1.4	50.7	1.4	3-5
	SIN	синус			414	1.4	3-5
	COS	косинус			404	1.4	3-5
	TAN	тангенс			504	1.4	3-5
	ASN	арксинус			426	1.4	3-5
	ACS	арккосинус			436	1.4	3-5
	ATN	арктангенс			375	1.4	3-5
	LN	натуральный логарифм	409	1.4	403	1.4	3-5
	LOG	десятичный логарифм	411	1.4	403	1.4	3-5
	XPY	X в степени Y	897	1.5	897	1.5	4-7
	SRT	сортировка файла (-5/11, -5/20) (-5/30, -5/40, -5/60, -5/80)	276+12*E ^{1.34} 224+25*E ^{1.34}	227 189	278+16*E ^{1.35} 230+33*E ^{1.35}	227 189	3-5

¹ Используйте большее число для адресов за первыми 2048 словами в таблице данных процессора.

E = количество элементов, обрабатываемое за сканирование.

Времена выполнения для случая SRT+«истина» – только аппроксимация. Фактическое время зависит от степени разброса («случайности») чисел.

Категория	Код	Название	Время выполнения (мкс) – целые операнды		Время выполнения (мкс) – плавающая запятая		Слов ¹ памяти
			Истина	Ложь	Истина	Ложь	
Логические	AND	И	5.9	1.4			4-7
	OR	ИЛИ	5.9	1.4			4-7
	XOR	исключающее ИЛИ	5.9	1.4			4-7
	NOT	НЕ	4.6	1.3			3-5
Перемещения	MOV	перемещение	4.5	1.3	5.6	1.3	3-5
	MVM	маскированное перемещение	6.2	1.4			4-7
	BTD	распределение битов	10.0	1.7			6-9
Сравнения	EQU	равно	3.8	1.0	4.6	1.0	3-5
	NEQ	не равно	3.8	1.0	4.5	1.0	3-5
	LES	меньше	4.0	1.0	5.1	1.0	3-5
	LEQ	меньше или равно	4.0	1.0	5.1	1.0	3-5
	GRT	больше	4.0	1.0	5.1	1.0	3-5
	GEQ	больше или равно	4.0	1.0	5.1	1.0	3-5
	LIM	пределы	6.1	1.1	8.4	1.1	4-7
	MEQ	маскированное сравнение на равно	5.1	1.1			4-7
Сравнить	CMP	все	$2.48 + \sum(0.8+i)$	$2.16 + Wi * 0.56$	$2.48 + \sum(0.8+i)$	$2.16 + Wi * 0.56$	$2 + Wi$
Вычислить	CPT	все	$2.48 + \sum(0.8+i)$	$2.16 + Wi * 0.56$	$2.48 + \sum(0.8+i)$	$2.16 + Wi * 0.56$	$2 + Wi$

¹ Используйте большее число для адресов за первыми 2048 словами в таблице данных процессора.

i = время выполнения каждой инструкции (операции, например, ADD, SUB, и т.п.) использованных в пределах выражения CMP или CPT.

Wi = количества слов памяти использованных инструкцией (операцией, например, ADD, SUB, и т.п.) в пределах выражения CMP или CPT.

Расчеты приведены для инструкций CMP и CPT с короткой прямой адресацией.

Инструкции: файловые, управления программой и ASCII

Категория	Код	Название	Время выполнения (мкс) – целые операнды		Время выполнения (мкс) – плавающая запятая		Слов ¹ памяти
			Истина	Ложь	Истина	Ложь	
Арифметика и логика файла	FAL	все	$11+E*\Sigma(2.3+i)$	$6.16+Wi*0.16$	$11+E*\Sigma(2.3+i)$	$6.16+Wi*0.16$	3-5+Wi
Файловый поиск и сравнение	FSC	все	$11+E*\Sigma(2.3+i)$	$6.16+Wi*0.16$	$11+E*\Sigma(2.3+i)$	$6.16+Wi*0.16$	3-5+Wi
Файловые	COP	копирование	$16.2+E*0.72$	1.4	$17.8+E*1.44$	1.4	4-6
		счетчик, таймер и структура управления	$15.7+E*2.16$	1.4			
	FLL	заполнение	$15.7+E*0.64$	1.5	$18.1+E*0.80$	1.5	4-6
		счетчик, таймер и структура управления	$15.1+E*1.60$	1.5			
Регистровый сдвиг	BSL	сдвиг битов влево	$10.6+B*0.025$	5.2			4-7
	BSR	сдвиг битов вправо	$11.1+B*0.025$	5.2			4-7
	FFL	загрузка очереди	8.9	3.8			4-7
	FFU	выгрузка из очереди	$10.0+E*0.43$	3.8			4-7
	LFL	загрузка стека	9.1	3.7			4-7
	LFU	выгрузка из стека	10.6	3.8			4-7
Диагностические	FBC	0 несовпадений	$15.4+B*0.055$	2.9			6-11
		1 несовпадение	$22.4+B*0.055$	2.9			
		2 несовпадения	$29.9+B*0.055$	2.9			
	DDT	0 несовпадений	$15.4+B*0.055$	2.9			6-11
		1 несовпадение	$24.5+B*0.055$	2.9			
		2 несовпадения	$34.2+B*0.055$	2.9			
	DTR	изменение данных	5.3	5.3			4-7

¹ Используйте большее число для адресов за первыми 2048 словами в таблице данных процессора.

i = время выполнения каждой инструкции (операции, например, ADD, SUB, и т.п.) использованных в пределах выражения FAL или FSC.

E = количество элементов, обрабатываемых за сканирование

B = количество битов обрабатываемых за сканирование

Wi = количества слов памяти использованных инструкцией (операцией, например, ADD, SUB, и т.п.) в пределах выражения FAL или FSC.

Расчеты приведены для инструкций FAL и FSC с короткой прямой адресацией.

Категория	Код	Название	Время выполнения (мкс) – целые операнды		Время выполнения (мкс) – плавающая запятая		Слов ¹ памяти
			Истина	Ложь	Истина	Ложь	
Секвенсор	SQI	вход секвенсора	7.9	1.3			5-9
	SQL	загрузка секвенсора	7.9	3.5			4-7
	SQO	выход секвенсора	9.7	3.7			5-9
Немедленный Вх/Вых ²	IIN	немедленный ввод • PLC-5/11, -5/20 и -5/20E • PLC-5/30, -5/40, -5/40E, -5/40L, -5/60, -5/60L, -5/80, -5/80E	• 357 • 307	1.1			2
	IOT	немедленный вывод • PLC-5/11, -5/20 и -5/20E • PLC-5/30, -5/40, -5/40E, -5/40L, -5/60, -5/60L, -5/80, -5/80E	• 361 • 301	1.1			2
Управление участком	MCR	основное управление	0.16	0.16			1
Управление программой	JMP	прыжок	8.9+(№ файла- 2)*0.96	1.4			2
	JSR ³ /RET	переход к подпрограмме /возврат – 0 параметров – 1 параметр – увеличение/параметр	12.3 16.1 3.8	1.0 1.0 неприменимо	неприменимо 17.3 5.0	неприменимо 1.0 неприменимо	3+пара- метры /1+пара- метры
	SBR						1+пара- метры

¹ Используйте большее число для адресов за первыми 2048 словами в таблице данных процессора.

² Времена для инструкций немедленного Вх/Вых – это времена для постановки инструкций в очередь.

³ Вычисляйте время выполнения следующим образом: (время) + (количество дополнительных параметров)(время/параметр). Например: если вы передаете с помощью JSR три целых параметра в процессоре PLC-5/11, то время выполнения = 16.1 + (2)*(3.8) = 23.7 мкс

Категория	Код	Название	Время выполнения (мкс) – целые операнды		Время выполнения (мкс) – плавающая запятая		Слов ¹ памяти
			Истина	Ложь	Истина	Ложь	
Управление программой	LBL	метка	0.16	0.16			2
	END	окончание	пренебрежимо				1
	TND	временное окончание	мало				1
	EOT	окончание перехода					1
	AFI	всегда ложна	0.16	0.16			1
	ONS	одно включение	3.0	3.0			2-3
	OSR	одно включение по спаду	6.2	6.0			4-6
	OSF	одно включение по нарастанию	6.2	5.8			4-6
	FOR/ NXT	цикл FOR-NEXT	$8.1+L*15.9+$ (№ файла-2)* 0.96	$5.3+N*0.75$			FOR 5-9 NXT 2
	BRK	прервать цикл	$11.3+N*0.75$	0.9			1
	UID	запретить пользовательские прерывания • PLC-5/11, -5/20 • PLC-5/30, -5/40, -5/60, -5/80	• 175 • 119	1.0			1
	UIE	разрешить пользовательские прерывания • PLC-5/11, -5/20 • PLC-5/30, -5/40, -5/60, -5/80	• 170 • 100	1.0			1

¹ Используйте большее число для адресов за первыми 2048 словами в таблице данных процессора.

L = число циклов FOR/NXT

N = число слов памяти между FOR/NXT или BRK/NXT

Категория	Код	Название	Время выполнения (мкс) – целые операнды		Время выполнения (мкс) – плавающая запятая		Слов ¹ памяти
			Истина	Ложь	Истина	Ложь	
Управление процессом Коэффициенты	PID	регулятор ПИД					5-9
		Независимые • PLC-5/11, -5/20 и -5/20E • PLC-5/30, -5/40, -5/40E, -5/40L, -5/60, -5/60L, -5/80, -5/80E	• 462 • 655	3.0	1120	58	
		ISA • PLC-5/11, -5/20 и -5/20E • PLC-5/30, -5/40, -5/40E, -5/40L, -5/60, -5/60L, -5/80, -5/80E	• 560 • 895		1180		
		Ручной • PLC-5/11, -5/20 и -5/20E • PLC-5/30, -5/40, -5/40E, -5/40L, -5/60, -5/60L, -5/80, -5/80E	• 372 • 420		1150		
		Установить выход • PLC-5/11, -5/20 и -5/20E • PLC-5/30, -5/40, -5/40E, -5/40L, -5/60, -5/60L, -5/80, -5/80E	• 380 • 440		1130		
Каскадирование		Подчиненный			1530		
		Внешний			1080		

¹ Используйте большее число для адресов за первыми 2048 словами в таблице данных процессора.

Категория	Код	Название	Время выполнения (мкс) – целые операнды		Время выполнения (мкс) – плавающая запятая		Слов ¹ памяти
			Истина	Ложь	Истина	Ложь	
ASCII ²	ABL ²	тест на наличие строки • PLC-5/11, -5/20 и -5/20E • PLC-5/30, -5/40, -5/40E, -5/40L, -5/60, -5/60L, -5/80, -5/80E	• 316 • 388	• 214 • 150			3-5
	ACB ²	число символов в буфере • PLC-5/11, -5/20 и -5/20E • PLC-5/30, -5/40, -5/40E, -5/40L, -5/60, -5/60L, -5/80, -5/80E	• 316 • 389	• 316 • 388			3-5
	ACI	строка в целое • PLC-5/11, -5/20 и -5/20E • PLC-5/30, -5/40, -5/40E, -5/40L, -5/60, -5/60L, -5/80, -5/80E	• 220+C*11 • 140+C*21.4	1.4			3-5
	ACN	объединение строк • PLC-5/11, -5/20 и -5/20E • PLC-5/30, -5/40, -5/40E, -5/40L, -5/60, -5/60L, -5/80, -5/80E	• 237+C*2.6 • 179+C*5.5	1.9			4-7
	AEX	извлечение строки • PLC-5/11, -5/20 и -5/20E • PLC-5/30, -5/40, -5/40E, -5/40L, -5/60, -5/60L, -5/80, -5/80E	• 226+C*1.1 • 159+C*2.2	1.9			5-9
	AHL ²	установка/сброс линий • PLC-5/11, -5/20 и -5/20E • PLC-5/30, -5/40, -5/40E, -5/40L, -5/60, -5/60L, -5/80, -5/80E	• 318 • 526	• 213 • 157			5-9

¹ Используйте большее число для адресов за первыми 2048 словами в таблице данных процессора.

² Времена выполнения для ASCII инструкций – это времена для постановки инструкций в очередь обслуживания канала 0.

C = число ASCII символов

Категория	Код	Название	Время выполнения (мкс) – целые операнды		Время выполнения (мкс) – плавающая запятая		Слов ¹ памяти
			Истина	Ложь	Истина	Ложь	
ASCII	AIC	целое в строку • PLC-5/11, -5/20 и -5/20E • PLC-5/30, -5/40, -5/40E, -5/40L, -5/60, -5/60L, -5/80, -5/80E	• 260 • 270	1,4			3-5
	ARD ²	чтение символов • PLC-5/11, -5/20 и -5/20E • PLC-5/30, -5/40, -5/40E, -5/40L, -5/60, -5/60L, -5/80, -5/80E	• 315 • 380	• 214 • 149			4-7
	ARL ²	чтение линии • PLC-5/11, -5/20 и -5/20E • PLC-5/30, -5/40, -5/40E, -5/40L, -5/60, -5/60L, -5/80, -5/80E	• 316 • 388	• 214 • 151			4-7
	ASC	поиск строки • PLC-5/11, -5/20 и -5/20E • PLC-5/30, -5/40, -5/40E, -5/40L, -5/60, -5/60L, -5/80, -5/80E	• 222+C*1.7 • 151+C*3.0	1,9			5-9
	ASR	сравнение строк • PLC-5/11, -5/20 и -5/20E • PLC-5/30, -5/40, -5/40E, -5/40L, -5/60, -5/60L, -5/80, -5/80E	• 234+C*1.3 • 169+C*2.4	• 202 • 119			3-5

¹ Используйте большее число для адресов за первыми 2048 словами в таблице данных процессора.

² Времена выполнения для ASCII инструкций – это времена для постановки инструкций в очередь обслуживания канала 0.

C = число ASCII символов

Категория	Код	Название	Время выполнения (мкс) – целые операнды		Время выполнения (мкс) – плавающая запятая		Слов ¹ памяти
			Истина	Ложь	Истина	Ложь	
ASCII	AWA ²	запись с соединением • PLC-5/11, -5/20 и -5/20E • PLC-5/30, -5/40, -5/40E, -5/40L, -5/60, -5/60L, -5/80, -5/80E	• 260 • 270	• 215 * 154			4-7
	AWT ²	запись • PLC-5/11, -5/20 и -5/20E • PLC-5/30, -5/40, -5/40E, -5/40L, -5/60, -5/60L, -5/80, -5/80E	• 315 • 380	• 215 • 151			4-7

¹ Используйте большее число для адресов за первыми 2048 словами в таблице данных процессора.

² Времена выполнения для ASCII инструкций – это времена для постановки инструкций в очередь обслуживания канала 0.

C = число ASCII символов

Для заметок _____

Установка переключателей

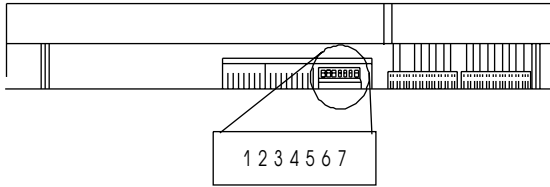
Назначение этой главы

Сведения	Страница
Установка адреса процессора в сети DN+ переключателем 1 для усовершенствованного и Ethernet процессоров PLC-5	23-2
Определение типа электрического интерфейса переключателем 2 для усовершенствованного и Ethernet процессоров PLC-5	23-3
Шасси Вх/Вых, содержащие процессор PLC-5	23-4
Шасси Вх/Вых, содержащие адаптер удаленных Вх/Вых 1771-ASB	23-5
Разъем шасси для определения внешнего источника питания или источника питания слотового исполнения	23-6
1771-ASB без использования комплементарных Вх/Вых	23-7
Модуль адаптера 1771-ALX	23-9

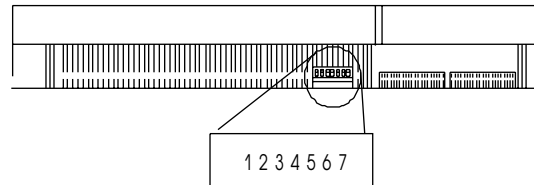
Переключатели процессора

Переключатель 1

Вид сбоку на сборку переключателей SW1 процессоров PLC-5/11, -5/20, -5/26, -5/20E



Вид сбоку на сборку переключателей SW1 процессоров PLC-5/30, -5/40, -5/46, -5/40L, -5/60, -5/60L, -5/80, -5/86, -5/40E и -5/80E



Чтобы установить

Установите переключатель:

В:

Адрес DH+ от 1 до 6 (см.таблицу снизу)



Нажат вниз
on (включено)

Скорость передачи данных по сети DH+ для канала 1A 7 on (включено) 57.6 кбит/сек
off(выключено) 230.4 кбит/сек



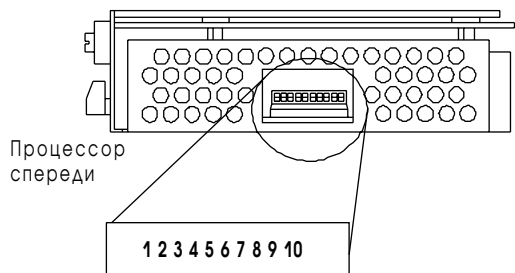
Поднят вверх
off (выключено)

Адрес станции DH+	Переключатель					
	1	2	3	4	5	6
0	on	on	on	on	on	on
1	off	on	on	on	on	on
2	on	off	on	on	on	on
3	off	off	on	on	on	on
4	on	on	off	on	on	on
5	off	on	off	on	on	on
6	on	off	off	on	on	on
7	off	off	off	on	on	on
10	on	on	on	off	on	on
11	off	on	on	off	on	on
12	on	off	on	off	on	on
13	off	off	on	off	on	on
14	on	on	off	off	on	on
15	off	on	off	off	on	on
16	on	off	off	off	on	on
17	off	off	off	off	on	on
20	on	on	on	on	off	on
21	off	on	on	on	off	on
22	on	off	on	on	off	on
23	off	off	on	on	off	on
24	on	on	off	on	off	on
25	off	on	off	on	off	on
26	on	off	off	on	off	on
27	off	off	off	on	off	on
30	on	on	on	off	off	on
31	off	on	on	off	off	on
32	on	off	on	off	off	on
33	off	off	on	off	off	on
34	on	on	off	off	off	on
35	off	on	off	off	off	on
36	on	off	off	off	off	on
37	off	off	off	off	off	on

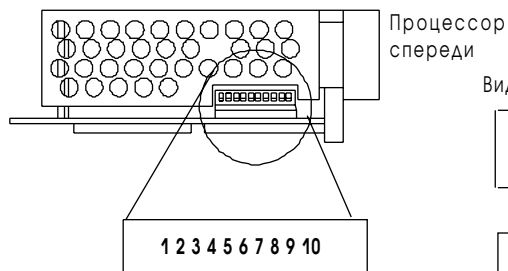
Адрес станции DH+	Переключатель					
	1	2	3	4	5	6
40	on	on	on	on	on	off
41	off	on	on	on	on	off
42	on	off	on	on	on	off
43	off	off	on	on	on	off
44	on	on	off	on	on	off
45	off	on	off	on	on	off
46	on	off	off	on	on	off
47	off	off	off	on	on	off
50	on	on	on	off	on	off
51	off	on	on	off	on	off
52	on	off	on	off	on	off
53	off	off	on	off	on	off
54	on	on	off	off	on	off
55	off	on	off	off	on	off
56	on	off	off	off	on	off
57	off	off	off	off	on	off
60	on	on	on	on	off	off
61	off	on	on	on	off	off
62	on	off	on	on	off	off
63	off	off	on	on	off	off
64	on	on	off	on	off	off
65	off	on	off	on	off	off
66	on	off	off	on	off	off
67	off	off	off	on	off	off
70	on	on	on	off	off	off
71	off	on	on	off	off	off
72	on	off	on	off	off	off
73	off	off	on	off	off	off
74	on	on	off	off	off	off
75	off	on	off	off	off	off
76	on	off	off	off	off	off
77	off	off	off	off	off	off

Переключатель 2

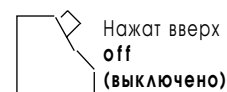
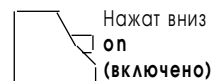
Сборка переключателей SW2. Вид снизу на процессоры PLC-5/11, -5/20, -5/26 и -5/20E



Сборка переключателей SW2. Вид снизу на процессоры PLC-5/30, -5/40, -5/46, -5/40L, -5/60, -5/60L, -5/80, -5/86, 5/40E и -5/80E



Вид сбоку



Переключите в :

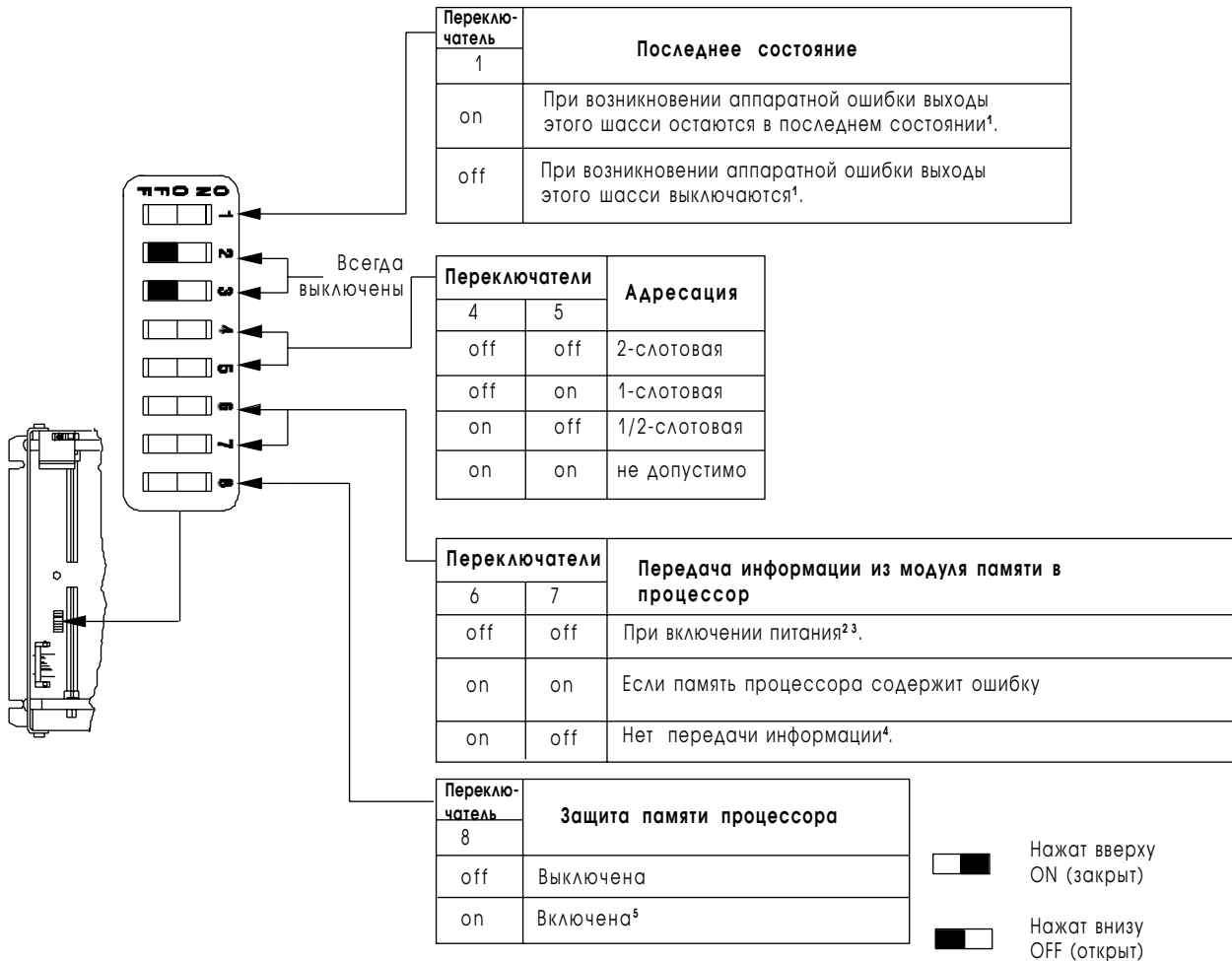
Чтобы

установить :

	1	2	3	4	5	6	7	8	9	10
RS-232C	on	on	on	off	off	on	on	off	on	off
RS-422A	off	off	on	off	off	off	off	off	on	off
RS-423	on	on	on	off	off	on	off	off	on	off

Монтажная плата шасси Вх/Вых

Процессор PLC-5 в шасси Вх/Вых



¹ Независимо от положения этих переключателей, выходы всех модулей отключаются в следующих ситуациях:

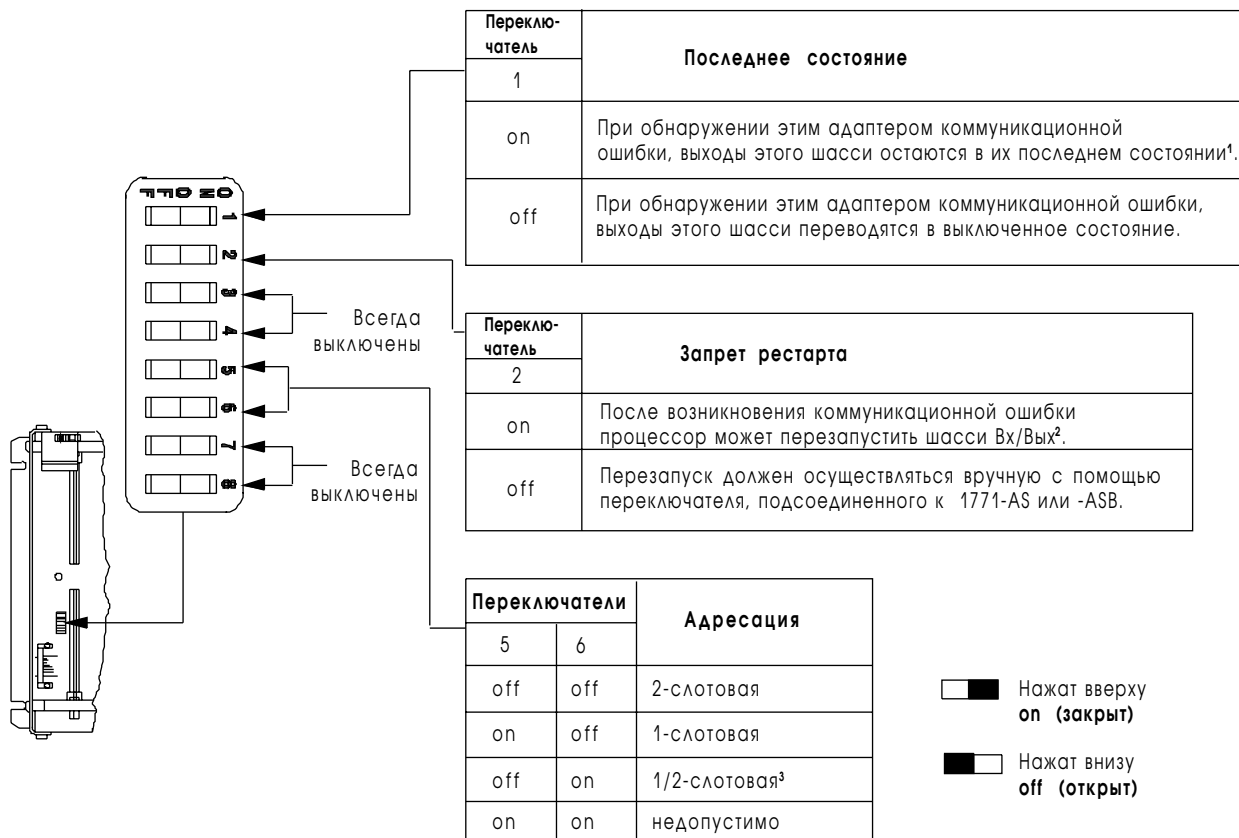
- процессор обнаруживает основную ошибку;
- происходит ошибка монтажной платы шасси;
- вы находитесь в режимах TEST или PROGRAM;
- вы установили бит в файле статуса для сброса локального рэка.

² Если модуль памяти не установлен, и память процессора не содержит ошибок, то индикатор процессора PROC LED будет мигать, и процессор установит бит S:11/9 в статусном слове основных ошибок. Выключите питание шасси процессора, а также, установите модуль памяти или переведите переключатель 6 в положение ON.

³ Если ключ процессора находится в положении REMOTE, после включения питания процессор войдет в режим RUN, и память процессора перепишется из модуля памяти.

⁴ Когда этот переключатель включен, то вы не сможете очистить память процессора.

Адаптер удаленных Вх/Вых 1771-ASB или адаптер локальных расширенных Вх/Вых 1771-ALX



19308

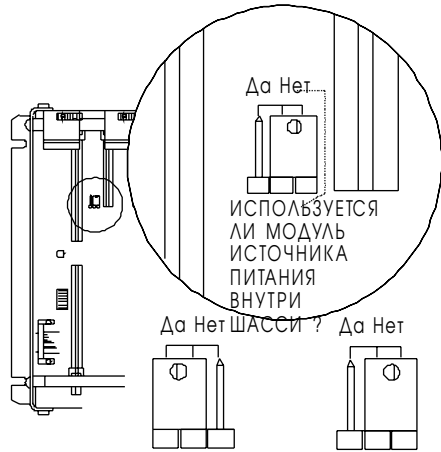
¹ВНИМАНИЕ: При установке этого переключателя в положение ON, выходы этого шасси при возникновении коммуникационной ошибки сохраняют свое прежнее состояние для предотвращения рывков в движении оборудования. Для отключения управляемых этим шасси выходных цепей, при обнаружении ошибки, мы рекомендуем устанавливать переключатель 1 в положение OFF.

Если выходы управляются входами из другого рэка, и возникает ошибка удаленного рэка Вх/Вых (во входном рэке), выходы также остаются в их последнем состоянии, предшествующем ошибке. В этом случае, выходы не могут корректно управляться и могут вызвать повреждение оборудования и травмирование персонала. Если вы хотите предотвратить подобную ситуацию и не оставлять выходы в их последнем состоянии, вы должны ввести в программу блок обработки ошибок.

² При планировании использовать автоконфигурацию рэка Вх/Вых установите переключатель в положение ON.

³ Адаптер 1771-ASB серии А не поддерживает 1/2-слотовую адресацию.

Разъем шасси для конфигурации источника питания



Ответьте "Да", если устанавливаете модуль источника питания в шасси.

Ответьте "Нет", если устанавливаете внешний источник питания.

1. Найдите разъем для конфигурации источника питания (между первыми двумя самыми левыми слотами).

2. Установите перемычку разъема.

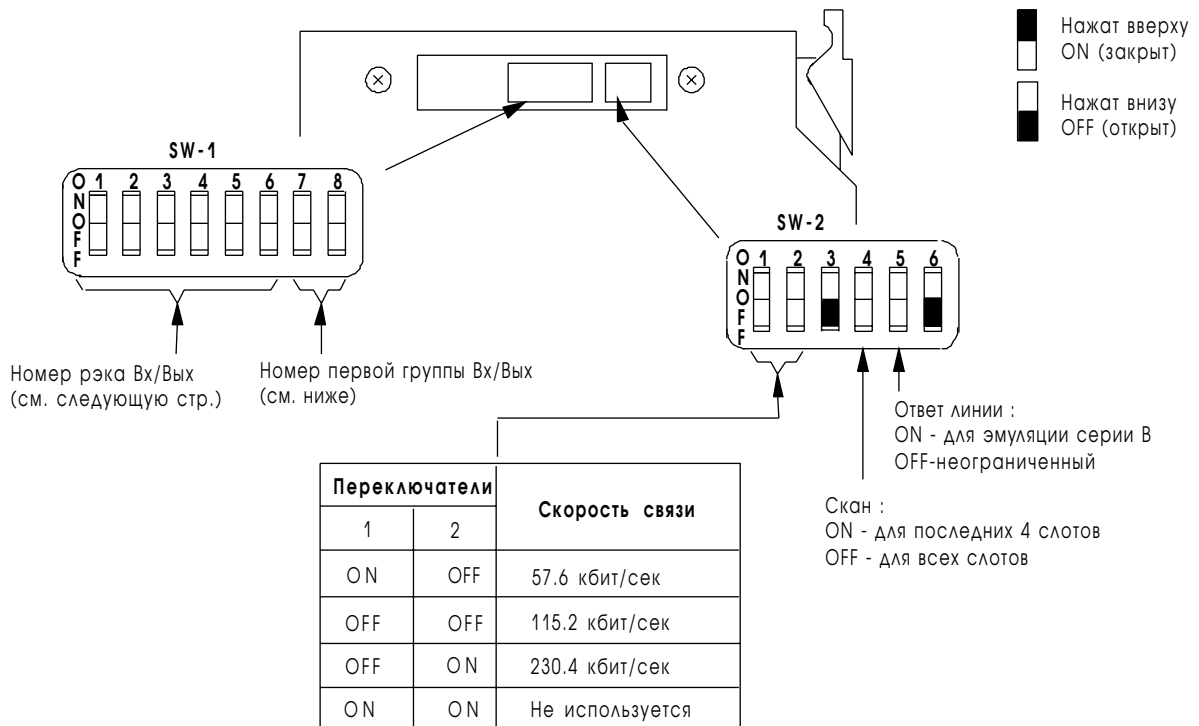
Положение по умолчанию - Нет (т.е. не используется модуль источника питания шасси внутренней установки).

Важно : Одновременная запитка шасси Вх/Вых внутренним модулем и внешним источником питания невозможна.

17075

Модуль адаптера удаленных Вх/Вых

Модули 1771-ASB серий С и D без комплементарных Вх/Вых



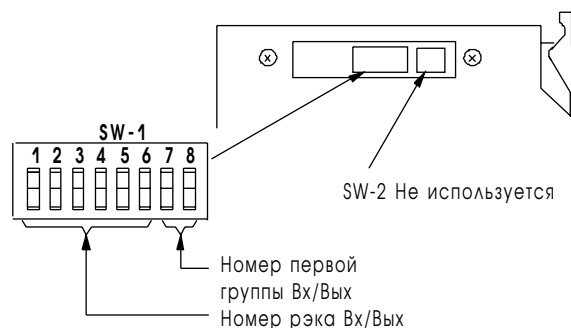
Начальный номер группы Вх/Вых :	7	8
	0	on
2	on	off
4	off	on
6	off	off

Номер рэков Вх/Вых (1771-ASB серий С и D) без комплементарных Вх/Вых

Рэк	1	2	3	4	5	6
01	on	on	on	on	on	off
02	on	on	on	on	off	on
03	on	on	on	on	off	off
04	on	on	on	off	on	on
05	on	on	on	off	on	off
06	on	on	on	off	off	on
07	on	on	on	off	off	off
10	on	on	off	on	on	on
11	on	on	off	on	on	off
12	on	on	off	on	off	on
13	on	on	off	on	off	off
14	on	on	off	off	on	on
15	on	on	off	off	on	off
16	on	on	off	off	off	on
17	on	on	off	off	off	off
20	on	off	on	on	on	on
21	on	off	on	on	on	off
22	on	off	on	on	off	on
23	on	off	on	on	off	off
24	on	off	on	off	on	on
25	on	off	on	off	on	off
26	on	off	on	off	off	on
27	on	off	on	off	off	off

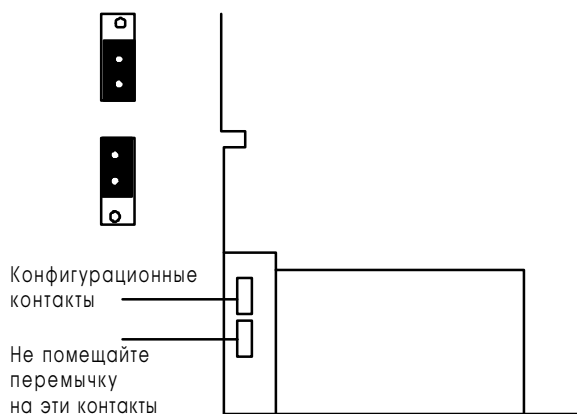
Модуль адаптера локальных расширенных Вх/Вых

Переключатель SW-1 модуля 1771-ALX



Рэк	1	2	3	4	5	6
01	on	on	on	on	on	off
02	on	on	on	on	off	on
03	on	on	on	on	off	off
04	on	on	on	off	on	on
05	on	on	on	off	on	off
06	on	on	on	off	off	on
07	on	on	on	off	off	off
10	on	on	off	on	on	on
11	on	on	off	on	on	off
12	on	on	off	on	off	on
13	on	on	off	on	off	off
14	on	on	off	off	on	on
15	on	on	off	off	on	off
16	on	on	off	off	off	on
17	on	on	off	off	off	off
20	on	off	on	on	on	on
21	on	off	on	on	on	off
22	on	off	on	on	off	on
23	on	off	on	on	off	off
24	on	off	on	off	on	on
25	on	off	on	off	on	off
26	on	off	on	off	off	on
27	on	off	on	off	off	off

Конфигурационные контакты модуля 1771-ALX



1. Положите модуль на его правую сторону.
Контакты конфигурации видны на нижней задней части модуля.
2. Установите перемычку, как указано ниже, в соответствии с вашими требованиями.

17341

Если вы используете :	Но не нижеперечисленные :	Установите перемычку:
Модуль Вх/Вых на 32 точки и любой из методов адресации	1771-IX или 1771-IY	на 2 нижних контакта
Модули 1771-IX и 1771-IY и любой метод адресации	32-точечные модули	на 2 верхних контакта

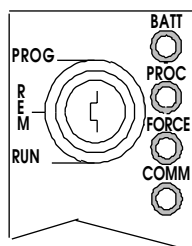
Поиск и устранение неисправностей

Назначение этой главы

Сведения	Страница
Общие проблемы процессора PLC-5 и канала 0	24-2
Проблемы сканера удаленных Вх/Вых, адаптера, или канала DH+ PLC-5, Ethernet PLC-5	24-3
Проблемы процессорных портов локальных расширенных Вх/Вых PLC-5/40L или PLC-5/60L	24-4
Проблемы Ethernet канала процессора PLC-5E	24-4
Модуль 1771-ASB	24-5
Модуль 1771-ALX	24-7
Неожиданное поведение PLC-5 при переходе в режим RUN	24-8

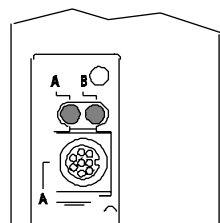
Процессор PLC-5

Общие проблемы



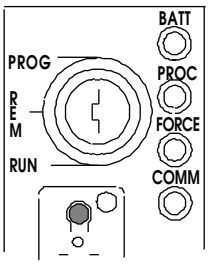
Индикатор	Цвет	Описание	Возможная причина	Действия по устранению
BATT	Красный	Батарея разряжена	Разряд батареи	Замените батарею в течение 10 дней
	Выключен	Батарея в норме	В норме	Нет
PROC	Зеленый	Нормальная работа процессора в режиме RUN	В норме	Нет
	Зеленый (мигающий)	Память процессора перемещается в модуль памяти		
	Красный (мигающий)	Основная ошибка	Основная ошибка	<p>Определите описание ошибки в файле статуса S:11</p> <p>Сбросьте бит ошибки, устраните причину и возвратитесь в режим "RUN"</p>
	Красный	Аппаратная ошибка	<ul style="list-style-type: none"> Ошибка контрольной суммы процессора Ошибка модуля памяти Сбой внутренней диагностики 	<ul style="list-style-type: none"> Сбросьте память и перезагрузите программу Проверьте установки переключателей соединительной платы или/и вставьте корректный модуль памяти Отключите процессор, выньте и вставьте процессор обратно. Включите питание, затем, сбросьте память и перезагрузите программу. Замените модуль памяти с новой программой, затем, при необходимости, замените процессор.
	Выключен	Процессор загружает программу, находится в режиме TEST или нет питания		Проверьте блок питания и соединения
FORCE	Желтый	ПФС, Вх/Вых и/или расширенное форсирование разрешено	В норме	Нет
	Желтый (мигающий)	ПФС, Вх/Вых и/или расширенное форсирование есть, но не разрешено		
	Выключен	ПФС, Вх/Вых и/или расширенного форсирования нет		
COMM	Выключен	Нет передачи информации через канал 0	В норме, если канал не используется	
	Зеленый (мигающий)	Есть передача информации через канал 0	В норме, если канал используется	

Поиск неисправностей в коммуникационном канале процессора

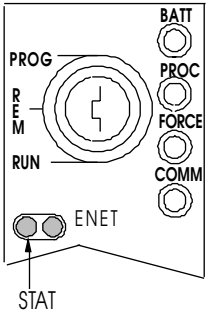


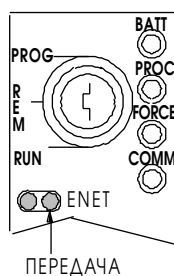
Индикатор	Цвет	Режим канала	Описание	Возможная причина	Действия по устранению
А или В	Зеленый	Сканер удаленных Вх/Вых	Связь с удаленными Вх/Вых активна, все модули есть, ошибок нет	Нормальное функционирование	Не требуются
		Адаптер удаленных Вх/Вых	Есть связь со сканером		
		DN+	Процессор передает или принимает данные по каналу DN+.		
Зеленый (мигает медленно или быстро)	Зеленый	Сканер удаленных Вх/Вых	По крайней мере один из адаптеров имеет ошибку или неисправен.	<ul style="list-style-type: none"> Нет питания удаленного рэка Повреждение кабеля 	<ul style="list-style-type: none"> Восстановите питание рэка Восстановите кабель
		DN+	Отсутствие в сети других узлов		
Красный	Красный	Сканер удаленных Вх/Вых	Неисправность аппаратуры	Неисправность аппаратуры	Выключите питание, и включите снова. Проверьте, соответствует ли программная конфигурация установкам аппаратуры. Замените процессор.
		Адаптер удаленных Вх/Вых			
Красный (мигает медленно или быстро)	Красный	Сканер удаленных Вх/Вых	Все адаптеры неисправны	<ul style="list-style-type: none"> Кабель не исправен или не подсоединен Отключено питание удаленных рэков 	<ul style="list-style-type: none"> Восстановите кабель Восстановите питание рэков
		DN+	Плохая связь по сети DN+	Обнаружены дубликаты узлов	Исправьте сетевой адрес станции
Выключен	Выключен	Сканер, адаптер удаленных Вх/Вых; DN+	Канал выключен (offline)	Канал не используется	Если необходимо подключите канал (online)

Поиск и устранение неисправностей в локальных расширенных Вх/Вых

Индикатор	Цвет	Режим канала	Описание	Возможная причина	Действия по устранению
<p>Только процессоры PLC-5/40L и PLC-5/60L</p> 	Зеленый	Сканер локальных расширенных Вх/Вых	Связь локальных расширенных Вх/Вых активна, все адаптеры присутствуют и работают нормально	Нормальная работа	Не требуются
	Зеленый (мигает быстро или медленно)	(Extended local I/O Scanner)	По крайней мере, один из адаптеров неисправен	<ul style="list-style-type: none"> Нет питания локальных расширенных рэков Вх/Вых Неисправность связи Поврежден кабель 	<ul style="list-style-type: none"> Восстановите питание рэка Перезапустите адаптеры, используя кнопку рестарта Восстановите кабель
	Красный		Неисправность аппаратуры	Аппаратный сбой	Выключите, а затем включите питание. Проверьте, что программная конфигурация соответствует установкам аппаратуры. Замените процессор.
	Красный (мигает быстро или медленно)	Сканер локальных расширенных Вх/Вых (Extended local I/O Scanner)	Все адаптеры неисправны	<ul style="list-style-type: none"> Кабель отключен или поврежден Отсутствие терминатора Нет питания расширенных Вх/Вых 	<ul style="list-style-type: none"> Восстановите кабель Восстановите или замените терминатор Обеспечьте питание рэкам
Выключен		Канал выключен (offline)	Канал не используется	Если необходимо включите канал (online)	

Индикатор состояния Ethernet

Индикатор	Цвет	Описание	Возможная причина	Действия по устранению	
	STAT	Ровный красный	Критичная неисправность аппаратуры	Процессор нуждается во внутреннем ремонте	Свяжитесь с региональным представительством фирмы Allen-Bradley
	Мигающий красный	Аппаратный или программный сбой (обнаруженный и отображенный кодом)	Зависит от кода ошибки	Свяжитесь со службой глобальной технической поддержки (GTS)	
	Выключен	Ethernet интерфейс работает нормально, но не соединен с активной Ethernet сетью	Нормальная работа	Соедините процессор с активной Ethernet сетью	
	Зеленый	Ethernet канал 2 работает нормально и обнаружил, что соединен с активной Ethernet сетью	Нормальная работа	Не требуются	



Индикаторы передачи информации канала Ethernet

Интерфейс канала Ethernet содержит светодиод зеленого цвета, быстро мигающий, когда Ethernet порт передает пакет. Прием пакета этот светодиод не индицирует.

Система удаленных Вх/Вых

Поиск и устранение неисправностей в модулях адаптеров 1771-ASB серий C и D

ACTIVE	●	Активен
ADAPTER FAULT	●	Ошибка адаптера
I/O RACK FAULT	●	Ошибка рэка Вх/Вых

Индикаторы			Описание	Возможная причина	Действия по устранению
Active	Adapter fault	I/O Rack fault			
Вкл	Выкл	Выкл	Нормальное и полное функционирование.		
Выкл	Вкл	Выкл		Ошибка модуля памяти, таймаут программного сторожа.	Замените модуль.
Вкл	Мигает	Выкл	Ошибка расположения.	Модуль Вх/Вых установлен не по месту.	Установите модуль в корректный слот.
Мигают в унисон		Выкл	Неправильный номер начальной группы Вх/Вых.	Ошибка в назначении начального номера группы или адреса рэка Вх/Вых.	Проверьте установки переключателей.
Вкл	Вкл	Вкл	Нет коммуникации.	Некорректная установка скорости связи.	
Выкл	Вкл	Вкл	Нет коммуникации.	Переключатель скана установлен в позиции "4 последних слота" в 1/4 рэка.	Сбросьте переключатель скана.
Мигает	Выкл	Выкл	Удаленный адаптер не контролирует Вх/Вых (связь сканер-адаптер в норме) ⁴ .	Процессор находится в режиме "PROGRAM" или "TEST". Сканер удерживает адаптер в состоянии ошибки.	Ошибка должна быть сброшена сканером Вх/Вых.
Последовательно мигают сверху вниз			Нет коммуникации.	В сети присутствует другой удаленный адаптер Вх/Вых, с таким же адресом.	Скорректируйте адрес.
Мигают в противофазе		Выкл	Модуль адаптера не контролирует Вх/Вых ² . Модуль адаптера находится в процессе блокировки рестарта (связь сканер-адаптер в норме).	Переключатель блокировки рестарта процессора на сборке переключателей соединительной платы находится во включенном положении ¹ .	Нажмите кнопку сброса, чтобы устранить блокировку или передерните питание ³ . Если после повторных попыток индикаторы по-прежнему мигают, проверьте: <ul style="list-style-type: none"> • соединения, подходящие к модулю адаптера; • модуль адаптера сбрасывается процессором или сканером и сразу входит в состояние ошибки.

¹ Режим работы модуля адаптера Вх/Вых должен определяться в соответствии с комплектацией удаленным сканером Вх/Вых/распределительной панелью, удаленным интерфейсным модулем программного сканера Вх/Вых или модулем сканера Вх/Вых, поддерживающего сообщения. Обратите пристальное внимание на деактивацию режима поиска в модулях 1771-SD, -SD2.

² Шасси Вх/Вых находится в режиме ошибки, как выбрано переключателем последнего состояния на соединительной плате шасси.

³ Выключение и последующее включение питания очищает очередь запросов блоков передачи (block-transfer request queue). Все текущие блоки информации будут утеряны. В программе должен быть предусмотрен повтор запроса на передачу блоков.

⁴ Если при наступлении сбоя процессор находится в режиме "RUN", но в действительности находится в зависимом режиме, реакция на наступление ошибки шасси выбирается переключателем последнего состояния соединительной платы шасси.

Поиск и устранение неисправностей в модулях адаптеров 1771-ASB серий C и D (продолжение)

Индикаторы			Описание	Возможная причина	Действия по устранению
Active	Adapter fault	I/O Rack fault			
Выкл	Выкл	Вкл	Неисправность шасси Vх/Вых ² . Нет связи.	Проблема существует между : <ul style="list-style-type: none"> •адаптером и модулем в шасси; модуль будет в режиме ошибки до тех пор, пока ошибка не будет исправлена; •короткое замыкание печатных проводников соединительной платы шасси или модуля Vх/Вых. 	Передерните питание шасси, чтобы устранить проблему, возникшую в результате действия электрических помех ³ . <ul style="list-style-type: none"> •Одновременно извлеките все модули, а затем снова вставьте в шасси. • Если проблема не устраняется, то неисправность находится внутри шасси или модуля Vх/Вых.
Мигает	Выкл	Вкл	Связь есть. Возможно короткое замыкание соединительной платы.	<ul style="list-style-type: none"> •Электрические помехи •Короткие замыкания соединительной платы •В шасси вставлен неисправный блок 	<ul style="list-style-type: none"> • Исклучите помехи • Экранируйте помехи • Примените помехоподавление • Замените шасси • Замените неисправный блок
Мигает	Вкл	Выкл	Повреждена цепь идентификации модуля.	Чрезмерные электрические помехи.	Проверьте блок питания и заземление шасси.
Выкл	Выкл	Выкл	Нет связи модуля.	<p>Неисправность блока питания.</p> <p>Разрушена связь между сканером и адаптером.</p> <p>Неправильная конфигурация сканера.</p> <p>Одно неисправное шасси внутри пространства адресов рэка вызывает ошибку распределяющей панели сканера всех шасси данного рэка (при нахождении в режиме запрета поиска).</p>	<p>Проверьте блок питания, все соединения, удостоверьтесь, что модуль адаптера полностью вставлен в шасси.</p> <p>Исправьте дефекты проводов и кабелей.</p> <p>См. документ 1772-2.18 для конфигурации сканера.</p> <p>Последовательно проверьте на мельчайшие ошибки каждый из модулей в каждом шасси.</p>

¹ Режим работы модуля адаптера Vх/Вых должен определяться в соответствии с комплектацией удаленным сканером Vх/Вых/распределительной панелью, удаленным интерфейсным модулем программного сканера Vх/Вых или модулем сканера Vх/Вых, поддерживающего сообщения. Обратите пристальное внимание на деактивацию режима поиска в модулях 1771-SD, -SD2.

² Шасси Vх/Вых находится в режиме ошибки, как выбрано переключателем последнего состояния на соединительной плате шасси.

³ Выключение и последующее включение питания очищает очередь запросов блоков передачи (block-transfer request queue). Все текущие блоки информации будут утеряны. В программе должен быть предусмотрен повтор запроса на передачу блоков.

⁴ Если при наступлении сбоя процессор находится в режиме "RUN", но в действительности находится в зависимом режиме, реакция на наступление ошибки шасси выбирается переключателем последнего состояния соединительной платы шасси.

Система локальных расширенных Вх/Вых

Поиск и устранение неисправностей для модуля адаптера 1771-ALX

ACTIVE		Активен
ADAPTER FAULT		Ошибка адаптера
I/O RACK FAULT		Ошибка рэка Вх/Вых

Индикаторы			Описание	Возможная причина	Действия по устранению
Active	Adapter fault	I/O Rack fault			
Вкл	Выкл	Выкл	Удаленный адаптер работает нормально		
Выкл	Вкл	Выкл	Ошибка локального адаптера ² .	Локальный адаптер не работает и будет находиться в состоянии ошибки до устранения причины неисправности.	Выключите и включите вновь питание адаптера ³ . Замените адаптер, если ошибка не исчезает.
Выкл	Выкл	Вкл	Неисправность шасси Вх/Вых ² .	Существует проблема между: • адаптером и модулем в шасси, модуль будет в режиме ошибки до тех пор, пока ошибка не будет исправлена; • короткое замыкание печатных проводников объединительной платы шасси или модуля Вх/Вых.	Передерните питание шасси, чтобы устранить проблему, возникшую в результате действия электрических помех ³ . • Одновременно извлеките и замените все Вх/Вых модули. • Замените адаптер. • Если проблема не устраняется, то неисправность находится внутри шасси или модуля Вх/Вых.
Мигает	Выкл	Выкл	Выходы сброшены.	Процессор находится в режиме PROGRAM или TEST. Локальный сканер Вх/Вых удерживает адаптер в состоянии ошибки.	Не требуются. Ошибка должна быть сброшена сканером локальных расширенных Вх/Вых.
Мигают в противофазе		Выкл	Модуль адаптера не контролирует Вх/Вых ² . Модуль адаптера находится в процес-се блокировки рестарта (связь сканер-адаптер в норме).	Переключатель блокировки рестарта процессора на сборке переключателей объединительной платы находится во включенном положении ¹ .	Чтобы устранить блокировку нажмите кнопку сброса шасси или передерните питание; если после повторных попыток индикаторы по-прежнему мигают, проверьте, что модуль адаптера сбрасывается процессором или сканером и после этого входит в состояние ошибки.
Выкл	Выкл	Выкл	Нет связи или питания.	Неисправность блока питания.	Проверьте блок питания, надежность соединения кабеля. Убедитесь, что модуль адаптера надежно установлен в разьеме шасси.
Вкл	Мигает	Off	Неправильное расположение модуля в локальном шасси расширения Вх/Вых	Неправильное размещение модулей с высокой емкостью Вх/Вых.	Проверьте режим адресации и установки переключателей.

¹ Шасси Вх/Вых находится в состоянии ошибки, как выбрано переключателем 1 (последнего состояния) на объединительной плате шасси.

² Выключение и последующее включение питания очищает очередь запросов блок-трансферов. Все текущие блоки информации будут утеряны. В программе должен быть предусмотрен повтор запроса на блок-трансфер.

³ Если при наступлении сбоя, процессор находится в режиме RUN, но в действительности находится в зависимом режиме, реакция шасси на наступление ошибки выбирается переключателем 1 (последнего состояния) объединительной платы шасси.

Неожиданное поведение программы при переходе в режим «RUN»

Если при входе в режим RUN возникают неожиданные операции, проверьте описанные ниже операции прескана (prescan operation). Эти инструкции выполняются по-разному в течение прескана и во время нормального скана.

Функция *прескана* - это промежуточный скан при переходе из режима программирования в режим RUN, во время которого все цепи сканируются, как «ложь». Прескан проверяет все файлы и инструкции релейно-контактной логики и, в зависимости от результатов программы, инициализирует таблицы данных.

Например, редко вызываемая подпрограмма может содержать неправильный косвенный адрес и в дальнейшем вызовет основную ошибку (major fault). Это может произойти после выполнения множества нормальных сканов. Прескан дает возможность процессору проверить программу на ошибки, такие как вышеописанные, *перед* входом в режим RUN.

Инструкции уникальных операций прескана

Используйте таблицу, представленную ниже, для отслеживания операций прескана, отличающихся от нормальных инструкций.

Таблица 24.А

Инструкции операций во время прескана

Эта инструкция	Выполняет следующие действия во время прескана
ARD	Если бит EN установлен, а биты DN и ER сброшены, то управляющее слово очищается. Если установлен один из бит DN или ER, то EN сбрасывается, а бит DN устанавливается.
ARL	
AWT	
AWA	
ACB	
ABL	
AHL	
BTR	Все непользовательские биты конфигурации 15,14,13,12, 10 и 9 сбрасываются (для обоих типов файлов - целочисленных(INT) и блочковой передачи (BT)).
BTW	
CTU	Устанавливается бит CU/CD для предотвращения ложного счета во время первого скана режима RUN.
CTD	
EOT	Не учитывается, для возможности прескана всех инструкций релейной логики.
FFL	Для предотвращения ложной загрузки при первом скане режима RUN устанавливается бит EL.
LFL	
FFU	Для предотвращения ложной выгрузки при первом скане режима RUN устанавливается бит EU.
LFU	
FND	Не учитывается, для возможности прескана всех инструкций релейной логики.
FOR	Инструкции внутри цикла FOR/NXT проверяются пресканом.

Эта инструкция	Выполняет следующие действия во время прескана
MSG	Если сброшены стартовый бит ПФС и бит СО, тогда все неиспользуемые биты конфигурации 15,14,13,12,10 и 9 будут сброшены в обоих типах файлов - целочисленном (INT) и сообщений (MG). Тип файлов MG, также, сбрасывает биты 11, 7, 6, 5, 4, 2, 1 и 0.
ONS	Для предотвращения ложного срабатывания во время первого скана режима RUN, запрограммированный битовый адрес инструкции устанавливается в 1.
OSF	Для предотвращения ложного срабатывания во время первого скана режима RUN, запрограммированный битовый адрес инструкции сбрасывается.
OSR	Сбрасывается, также, выходной бит.
PID	Для файлов типа PD, сбрасывается бит INI. Файл типа INT очищает биты 8, 9, и 10 (зону нечувствительности, выходы верхнего и нижнего пределов тревоги). Регистр ошибок предыдущего скана устанавливается в 32767, это индицирует о том, что биты задания и ER не были инициализированы на предыдущем скане. Биты Integral Accumulator и Derivative Error сбрасываются.
SQL	Для предотвращения ложного инкремента указателя таблицы во время первого скана режима RUN, бит EN устанавливается в 1.
SQO	
TOF	Биты TI, TC, TE и TO сбрасываются, накопленное значение равно заданному (ACC=preset).
DTR ¹	Ссылочное значение обновляется (независимо от условий цепи).

¹ Инструкция DTR выполняется аналогично во время нормального скана.

Предлагаемые действия

Во избежании непредсказуемых действий, могущих возникнуть в результате выполнения операций прескана, следуйте указаниям:

- Не используйте индексную или косвенную адресацию в инструкциях, приведенных в таблице 24.A.
Если вы *должны* использовать индексную или косвенную адресацию, используйте бит первого скана (S:1/15) для прединициализации всех используемых переменных.
- При использовании косвенной адресации в любой инструкции релейно-контактной логики, не применяйте переменную, находящуюся в косвенном адресе для различных функций.

Справочник по кабелям

Назначение этой главы

Сведения	Страница
Назначение контактов канала 0 процессора	25-1
Назначение контактов кабеля последовательной связи	25-2
Диаграммы кабельных соединений	25-3
Спецификация кабеля программирования	25-5
Соединения кабеля Ethernet	25-10

Назначение контактов канала 0 процессора

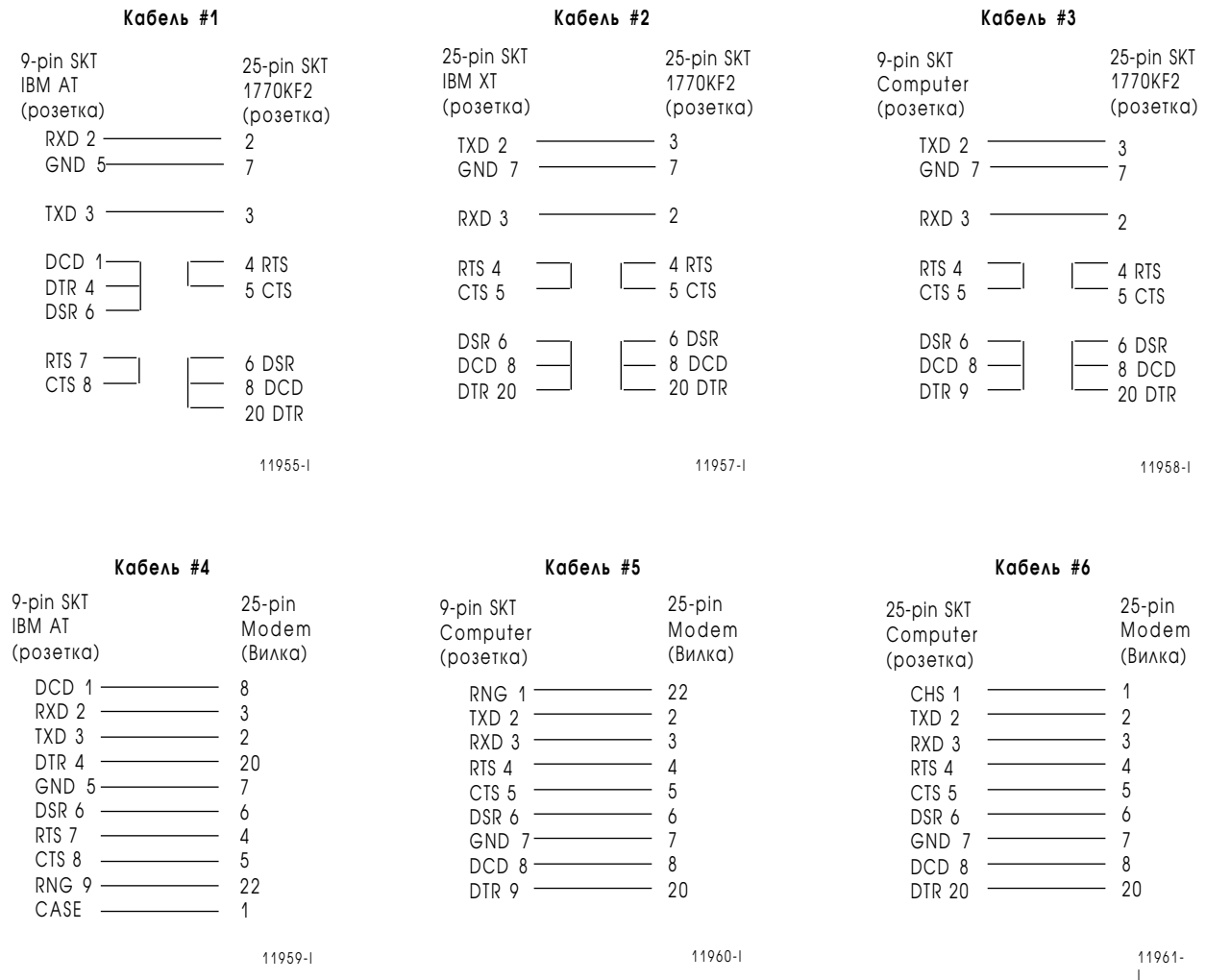
На боковой наклейке процессора дана таблица назначения контактов канала 0 (RS-порт). Представленная ниже таблица повторяет эту информацию.

Контакт	RS-232C	RS-422A	RS-423	Контакт	RS-232C	RS-422A	RS-423
1	C.GND	C.GND	C.GND	14	Не использован	TXD.OUT ⁻	SEND COM
2	TXD.OUT	TXD.OUT ⁺	TXD.OUT	15			
3	RXD.IN	RXD.IN ⁺	RXD.IN	16	Не использован	RXD.IN ⁻	REC COM
4	RTS.OUT	RTS.OUT ⁺	RTS.OUT	17			
5	CTS.IN	CTS.IN ⁺	CTS.IN	18			
6	DSR.IN	DSR.IN	DSR.IN	19	Не использован	RTS.OUT ⁻	Не использован
7	SIG.GND	SIG.GND	SIG.GND	20	DTR.OUT	DTR.OUT	DTR.OUT
8	DCD.IN	DCD.IN	DCD.IN	21			
9				22	Не использован	DSR.IN	Не использован
10	Не использован	DCD.IN	Не использован	23	Не использован	DTR.OUT	Не использован
11				24			
12				25			
13	Не использован	CTS.IN ⁻	Не использован				

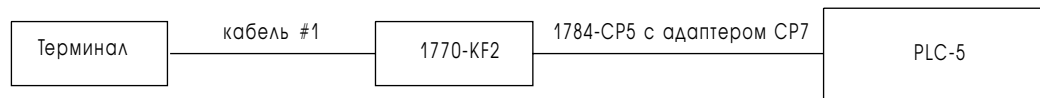
Серым фоном показаны зарезервированные контакты.

Назначение контактов кабеля последовательной связи

Следующие схемы дают соединения между контактами кабелей, необходимых для соединения с последовательным портом компьютера.



Диаграммы соединений



9-контактный последовательный порт

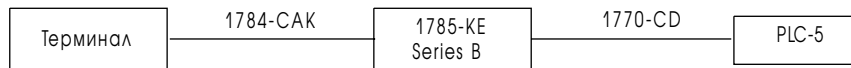
1784-T50

6160-T53

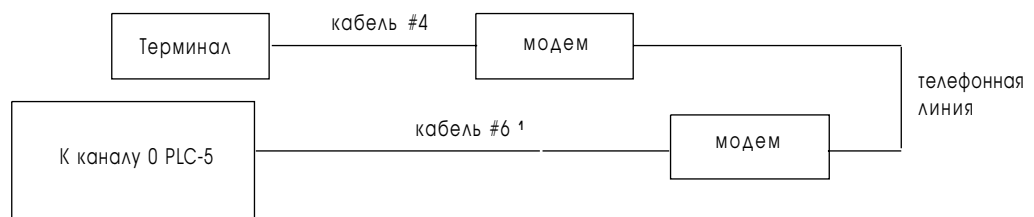
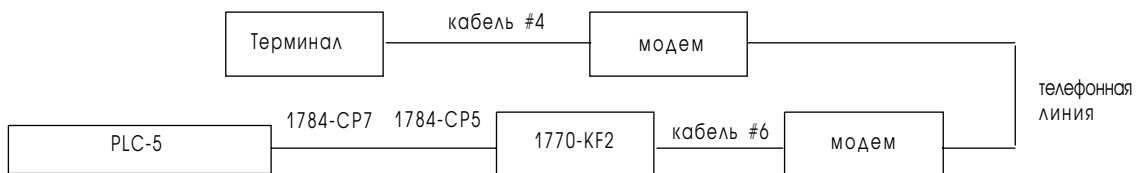
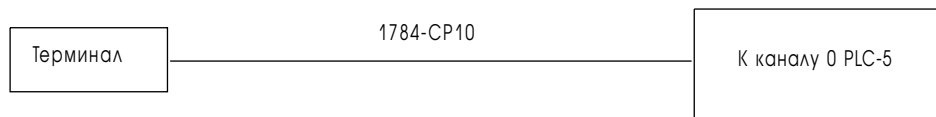
6160-T60

6160-T70

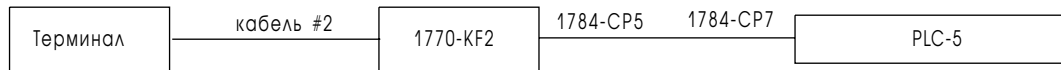
IBM PC/AT



Замечание: 1785-KE серии А использует кабель 1785-CP5 и адаптер 1785-CP7 с процессорами PLC-5 enhanced и PLC-5 Ethernet.



¹ Требуется либо родственная замена, либо кабель #2 с вилкой 25-контактного разъема.



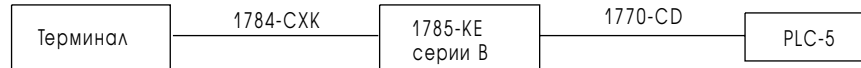
25-контактный последовательный порт

1784-T47

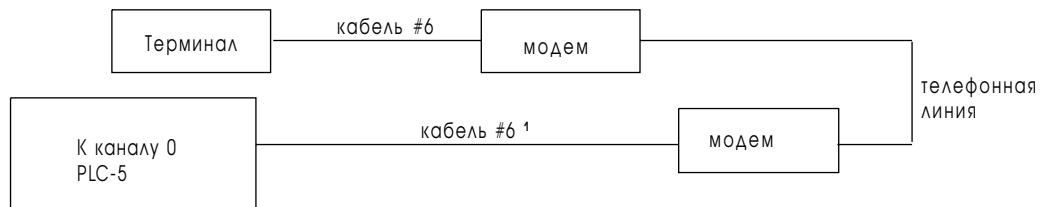
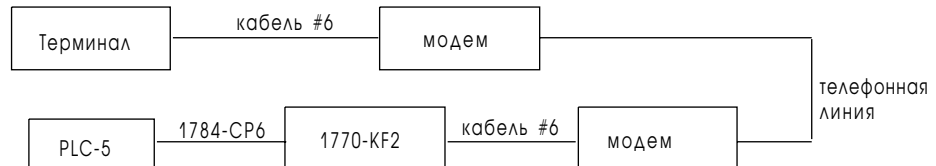
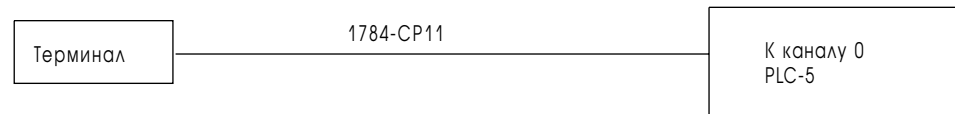
IBM XT

IBM PS/2 Model 30

IBM PS/2 Model 60



Замечание: 1785-KE серии А использует кабель 1784-CP5 и адаптер 1784-CP7 с процессорами PLC-5 enhanced и PLC-5 Ethernet.



¹ Требуется либо родственная замена, либо кабель #2 с вилкой 25-контактного разъема.

Характеристики кабеля программирования

Спецификации для каждого кабеля Allen-Bradley, используемого для DH+ коммуникаций, приведены на страницах, указанных в таблице 25.A

Таблица 25.A

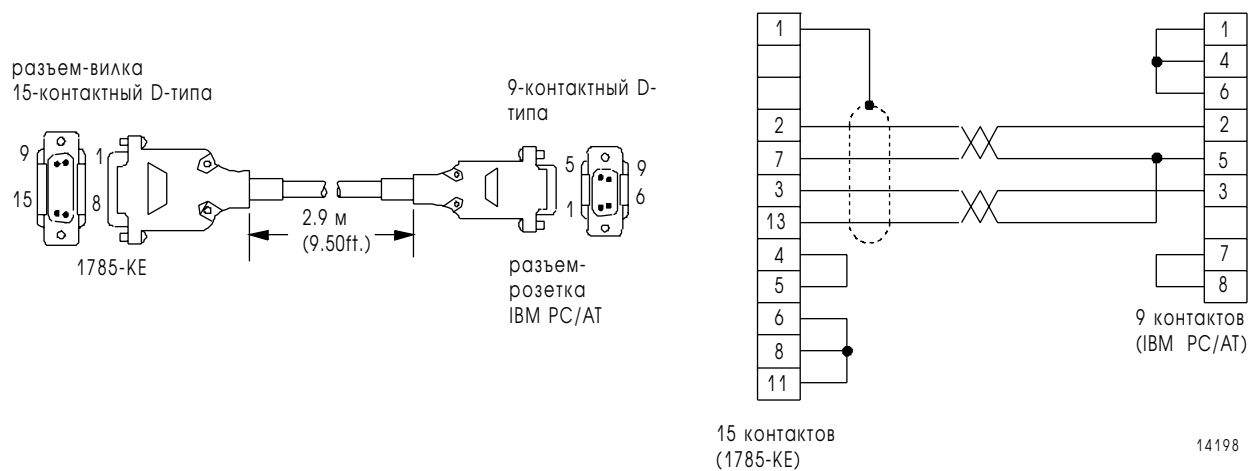
Характеристики кабеля программирования

От :	К :	Используйте кабель :	См. стр. :
6160-T53 6160-T60 6160-T70 IBM PC/AT	1785-KE	1784-САК	25-5
1784-T45 IBM PC IBM XT	1785-KE	1784-СХК	25-6
Процессор PLC-5, усовершенствованный или Ethernet	Терминал (с использованием 1784-КТ, -КТ2, -КЛ или -КЛ/В)	1784-СР6	25-6
		1784-СР с адаптером 1784-СР7 Адаптер 1784-СР8	25-7
	Терминал (с использованием 1784-КТК1)	1784-СР5 с адаптером 1784-СР7	25-7
	Терминал (с исполь- зованием последова- тельного 9-контакт- ного кабеля)	1784-СР10	25-8
	Терминал (с исполь- зованием последова- тельного 25-контакт- ного разъема)	1784-СР11	25-8
	Терминал (с исполь- зованием 1784-РСМК)	1784-РСМ5 с адаптером 1784-СР7	25-9 и 25-7

Рис. 25.1

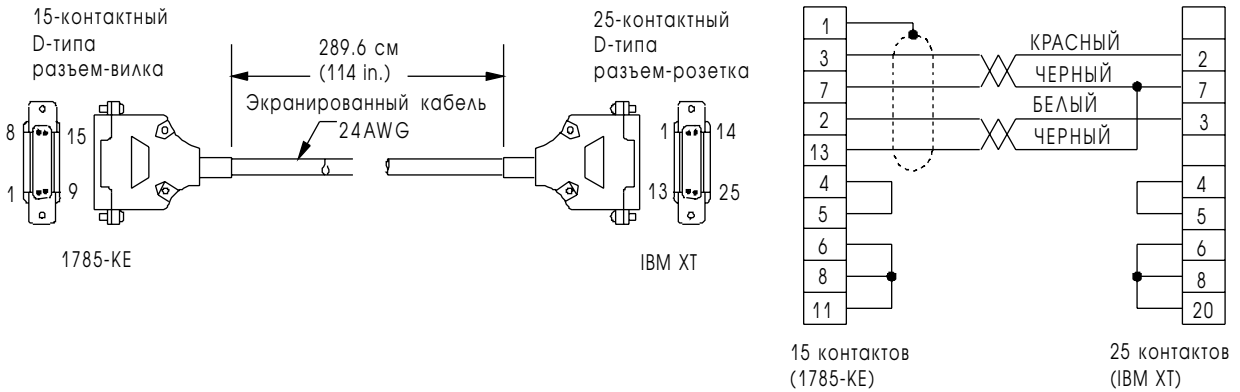
Кабель 1784-САК

Соединяет 1785-KE с 6160-T53, 6160-T60, 6160-T70 или IBM PC/AT



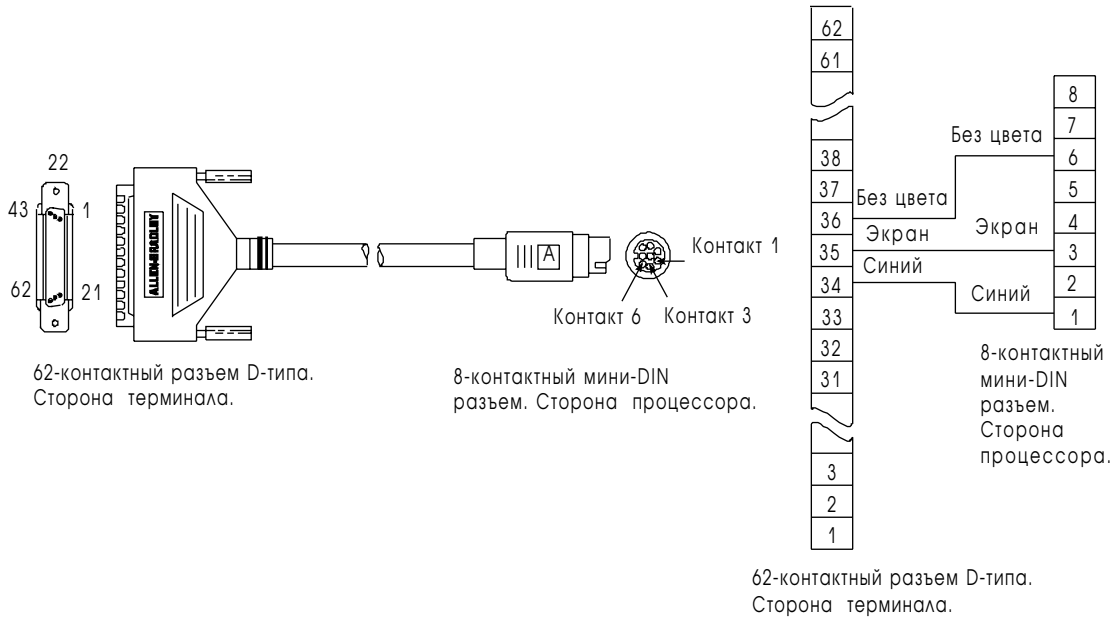
14198

Рис. 25.2
Кабель 1784-СХК
Соединяет 1785-КЕ с 1784-Т45, IBM PC или IBM XT



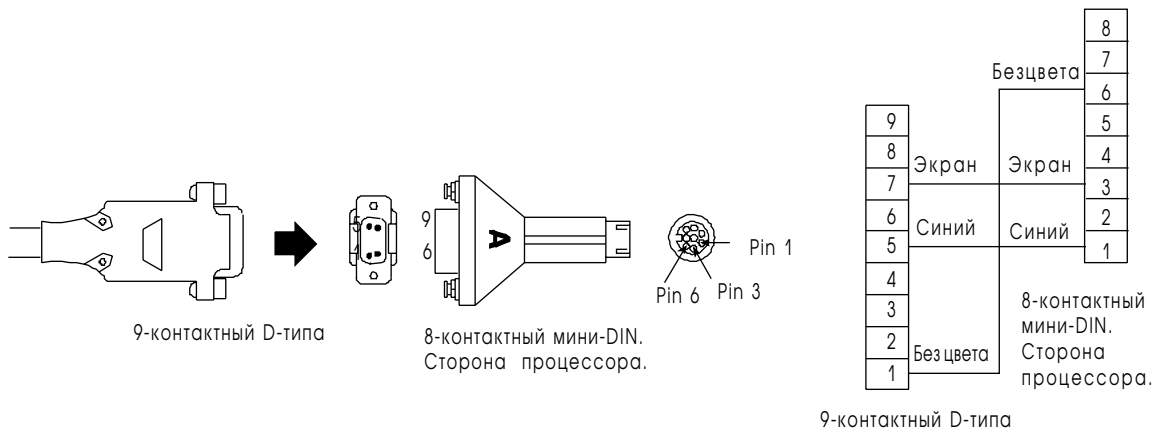
12727

Рис. 25.3
Кабель 1784-СР6
Соединяет терминал, использующий 1784-КТ, -КТ2 или -КЛ/В с процессором



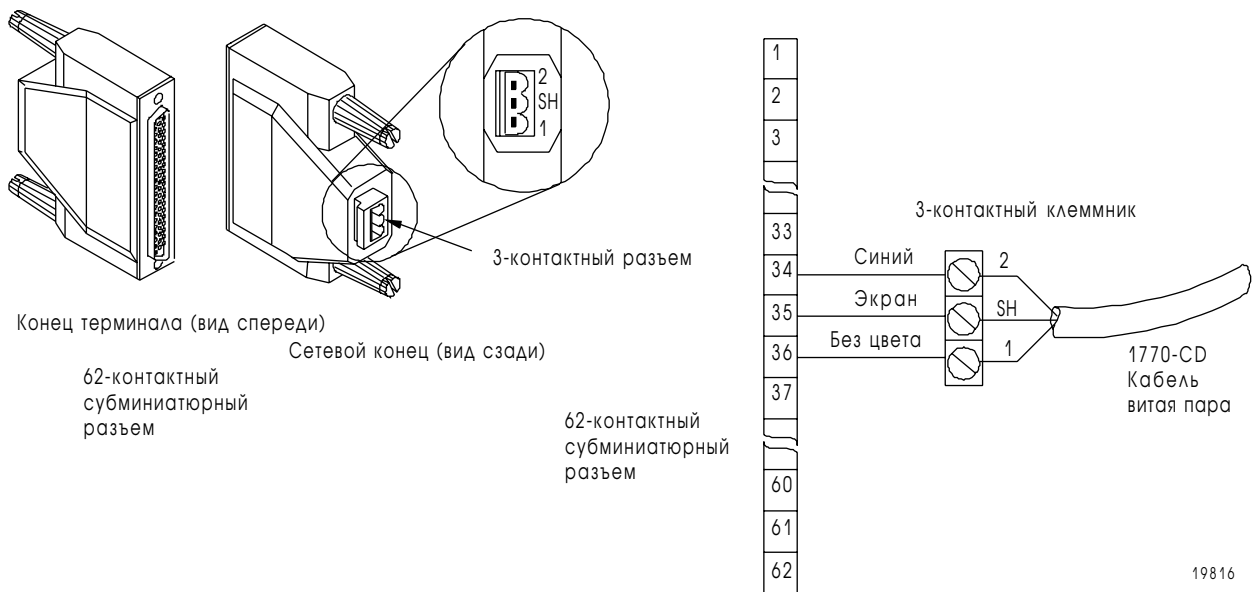
18378

Рис. 25.4
Кабель и адаптер - 1784-CP7, соединяемый с процессором через 9-контактный разъем D-типа для кабелей 1784-CP, -CP5 или -PCM5



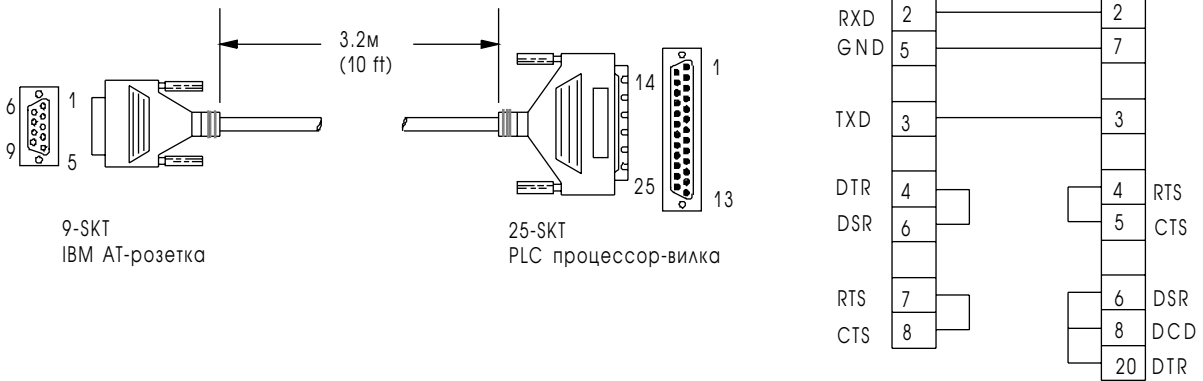
18377

Рис. 25.5
Кабель адаптер - 1784-CP8, соединяющий терминал, использующий платы 1784-КТ, -КТ2 или -КЛ со стационарной сетью ДН+



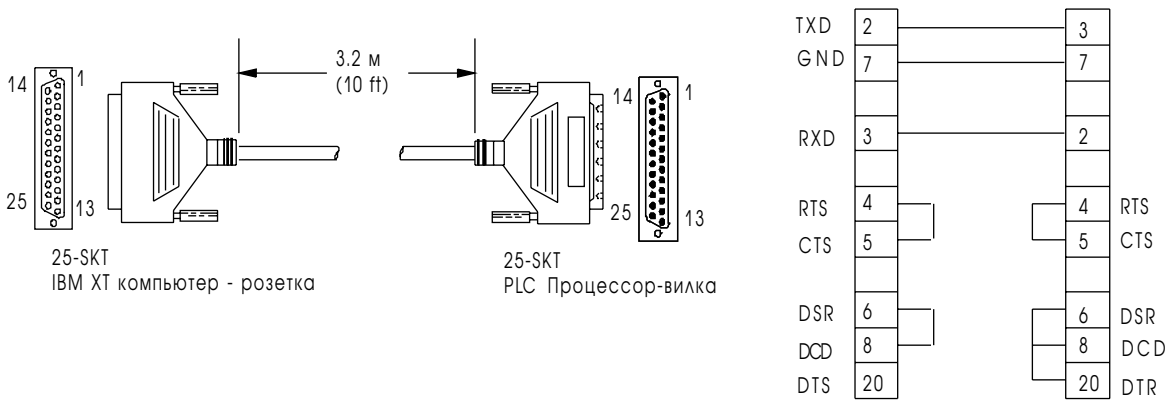
19816

Рис. 25.6
Кабель 1784-CP10
Соединяет терминал с процессором, используя последовательный порт



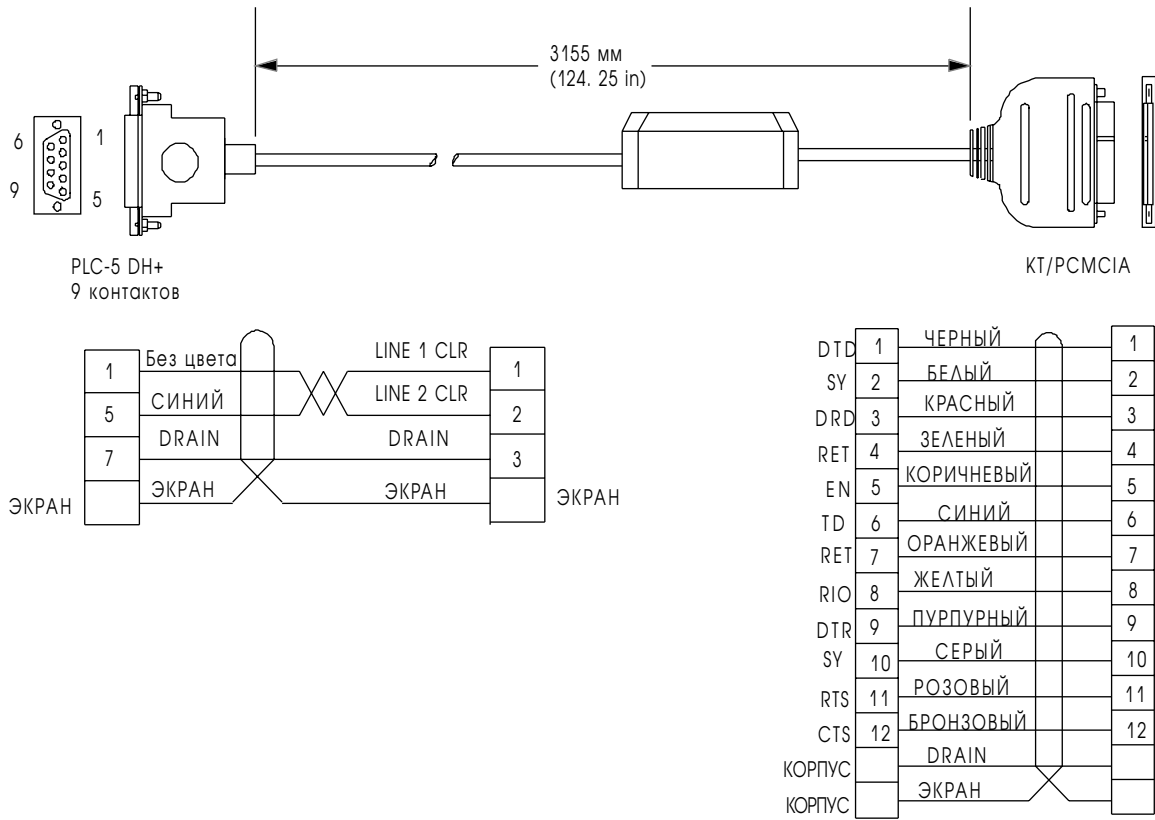
19870

Рис.25.7
Кабель 1784-CP11
Процессор к терминалу с использованием последовательного порта



19871

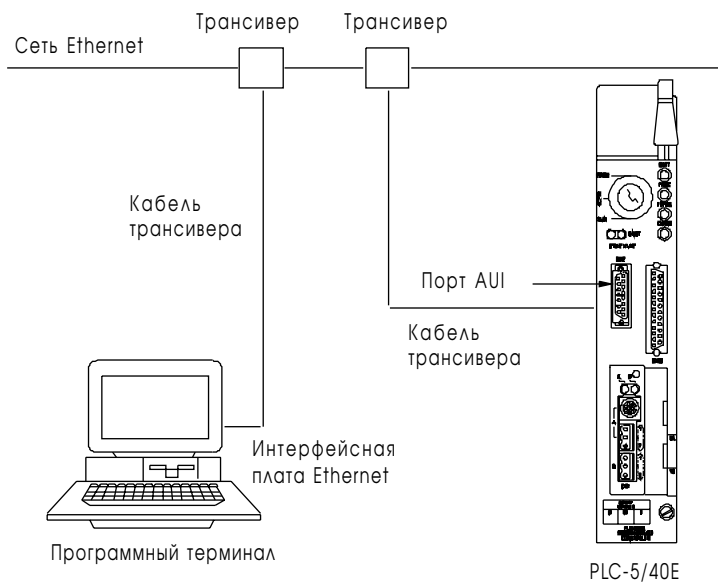
Рис. 25.8
Кабель 1784-PCM5
Процессор к терминалу (использующему 1784-PCMK)



19872

Кабельные соединения Ethernet

Ethernet порт соединяется либо с тонким, либо с толстым кабелем через 15-контактный трансивер или устройство доступа к носителю (Medium Access Unit - MAU).



Для подключения программного терминала к процессорам PLC-5/20E, -5/40E, -5/80E по сети Ethernet, необходимо следующее оборудование:

- Ethernet PC/AT-совместимая (6628-A5) или PCMCIA коммуникационная карта
- Ethernet кабель
- Трансиверы и трансиверные кабели

Существует два типа трансиверов Allen-Bradley.

Номер по каталогу	Описание
5810-AXMT	Трансивер для тонкого кабеля Ethernet/802.3
5810-AXMH	Трансивер для толстого кабеля Ethernet/802.3

Процессор соединяется с трансивером с помощью стандартного трансиверного кабеля, еще известный как AUI-кабель (Access Unit Interface). Allen-Bradley предлагает трансиверные кабели 2-х длин и четыре набора, состоящие из трансиверов и кабелей.

Номер по каталогу	Описание
5810-TC02/A	Толстый трансиверный кабель 2.0 м (6.5 ft)
5810-TC15/A	Толстый трансиверный кабель 15.0 м (49.2 ft)
5810-TAS/A (набор)	Трансивер тонкого кабеля и кабель 2.0 м (6.5 ft)
5810-TAM/A (набор)	Трансивер тонкого кабеля и кабель 15.0 м (49.2 ft)
5810-TER	Терминальные резисторы для тонкого Ethernet
5810-TBS/A (набор)	Трансивер толстого кабеля и кабель 2.0 м (6.5 ft)
5810-TBM/A (набор)	Трансивер толстого кабеля и кабель 15.0 м (49.2 ft)

Поддерживаются, также, соединения «10baseT» по оптоволоконным и широкополосным сетям, при покупке соответствующих трансиверов и кабелей третьих фирм.



Allen-Bradley
Publication Problem Report
(к русскому переводу)

Если у Вас возникли проблемы, связанные с нашей документацией, пожалуйста, заполните и вышлите нам эту форму.

Наименование публикации Усовершенствованные (Enhanced) и Ethernet Программируемые Контроллеры PLC-5No по каталогу _____ No публ. 1785-6.5.12 Дата публ. Октябрь 1995 Часть No. 95512279

Тип проблемы:	Описание проблемы	Только для внутреннего использования
<input type="checkbox"/> Техническая точность	<input type="checkbox"/> текст <input type="checkbox"/> иллюстрации	
<input type="checkbox"/> Законченность Какая информация пропущена?	<input type="checkbox"/> процедуры/шаги <input type="checkbox"/> иллюстрации <input type="checkbox"/> определение <input type="checkbox"/> примеры <input type="checkbox"/> требования <input type="checkbox"/> возможности <input type="checkbox"/> объяснение <input type="checkbox"/> другое	<input type="checkbox"/> информация в руководстве (accessibility) <input type="checkbox"/> информации нет в руководстве
<input type="checkbox"/> Понятность Что остается непонятным?		
<input type="checkbox"/> Размещение Что размещено не по порядку?		
<input type="checkbox"/> Прочие замечания Используйте следующую страницу для продолжения замечаний		

Ваше имя _____ Адрес/Телефон _____

Отошлите по адресу: Россия, г.Москва, Большой Строченовский пер. 22, Allen-Bradley
Phone: _____

Алфавитный указатель

Символы

1-слотовая адресация, 4-4, 23-4, 23-5
1/2-слотовая адресация, 4-4, 23-4, 23-5
1770-CD 10-1
1770-KF2 10-1
1770-XYC 20-1
1771-AF 6-5
1771-ALX 8-1
1771-ALX, 23-5
1771-AS 6-5
1771-ASB 6-5, 6-11
1771-ASB, 23-5
1771-DCM 6-5
1771-KRF 10-1
1771-SN 6-5
1772-SD, -SD2 6-5
1775-S4A, -S4B 6-5
1775-SR 6-5
1784-CAK, 25-5
1784-CP 10-1
1784-CP, 25-7
1784-CP10, 25-3, 25-5, 25-8
1784-CP11, 25-5, 25-8
1784-CP3 10-1
1784-CP5 10-1
1784-CP5, 25-5, 25-7
1784-CP6 10-1
1784-CP6, 25-5, 25-6
1784-CP7, 25-5, 25-7
1784-CP8, 25-5, 25-7
1784-CXK, 25-5, 25-6
1784-KL 10-1
1784-KL, 25-5, 25-6, 25-7
1784-KT 10-1
1784-KT, 25-5, 25-6, 25-7
1784-KT2 10-1
1784-KT2, 25-5, 25-6, 25-7
1784-KTK1, 25-5
1784-PCM5 10-1
1784-PCM5, 25-5, 25-7, 25-9
1784-PCMK 10-1
1784-PCMK, 25-5, 25-9
1785-KA 10-1
1785-KA5 10-1
1785-KE 10-1
1785-KE, 25-5, 25-7
2-слотовая адресация, 4-4, 23-4, 23-5
6008-SQH1, -SQH2 6-5

А

автоконфигурация 6-8, 6-9
автоматическая конфигурация, 8-11
адаптера режим
 конфигурация канала 7 - 3
 конфигурирование каналов 7 - 3
 Монитор статуса супервизорного процессора 7 -

16

определение 1-12
Пример программирования блок-трансфера 7 - 11
программирование блок-трансферов 7 - 8
связь через 7 - 1
статус канала адаптера 7 - 15, 7 - 16
Файл Конфигурации Дискретных Передатчиков 7 - 2
файл конфигурации дискретных передатчиков 7 - 7
Эффект программирования блок-трансферов дискретных 7 - 14
Адресация
 выбор режима,
 Ethernet 12 - 2
 Широковещательный адрес 12 - 11
адресация
 IP адрес 12 - 2
 выбор режима, 4-4, 23-4, 23-5
 индексная, 4-19
 концепция, 4-1
 косвенная, 4-18
 логическая, 4-16
 локальные расширенные Вх/Вых, 8-2
 мнемоника, 4-17
 назначение
 номеров рэков, 4-9
 определение
 адрес узла 10-3
 основы, 4-8
 отношение к биту устройства Вх/Вых, 4-3
 пример для модулей на 8 и 16 точек, 4-5
 пример для модуля на 32 точки 4-6
 рэка, 8-11
 рэков удаленных Вх/Вых, 4-10
 символическая, 4-20
 таблицы отображения Вх/Вых, 4-15
 термины, 4-2
 файл состояния Вх/Вых 6-23
 файлов данных, 4-15
 характеристики Вх/Вых 20-2
активные буферы 6-14
аппаратная
 адресация 20-2
 ошибка 16 - 2

Б

батарея 20-1, 20-4
бит, хранение данных, 4-11
бита и слова, инструкции 22-28
биты общего запрета, сброс 6-10
биты состояния множеств шасси 16 - 10
биты управления пользователя
 процедура запуска 15 - 2
 файл состояния процессора 21 - 9
блок-трансфер
 адаптер
 канал 7 - 8

- пример программирования / - 11
- адресация в модулях, 4-8
- активные буферы 6-14
- биты неосновных ошибок 6-15
- биты состояния 6-16
- запрос на передачу 6-14
- инструкции 22-24
- модуля, размещение 2-3
- передача блоков данных 6-13
- последовательность 6-16
- процессорные шасси 6-20
- размещение модулей, 5-3
- расчет времени
- локальные расширенные Вх/Вых 8-6
 - удаленный Вх/Вых 9-7
- редактирование онлайн 9-11
- статусные биты 7 - 15
- требования при программировании 6-20
- удаленные Вх/Вых, данные 6-15
- файлы управления 7 - 9
- Блок-трансфер данных Р - 3
- блоки данных, 4-11
- блоки питания, монтажные размеры, 3-6
- В**
- введение
- процессоры PLC-5 1-1
 - сканирование процессора, 5-1
 - удаленные Вх/Вых 6-3
- вес 20-2
- вибрационная стойкость, характеристики 20-1
- BOOTP 12 - 4
- IP адрес 12 - 8
 - аппаратный адрес 12 - 8
 - запрет 12 - 4
 - инсталляция 12 - 7
 - использование 12 - 5
 - Пример 12 - 8
 - Редактирование файла конфигурации 12 - 7
- BOOTP сервер 12 - 6
- Инсталляция 12 - 7
 - Редактирование файла конфигурации 12 - 7
- BOOTP.TAB.TXT 12 - 7
- время сканирования, вычисление 9-5
- Вх/Вых, адресация 20-2
- Вх/Вых модули
- выбор количества точек (плотности) 2-2
 - категории кабелей, 3-4
 - размещение 2-3
 - размещение блок-трансфер модуля 2-3
 - указания по выбору 2-1
- Вх/Вых, модули 20-1
- Вх/Вых скан
- определение, 5-2
- Вх/Вых, сканирование
- дискретных 20-1
- Вх/Вых файл
- адресация 6-23
 - конфигурация 6-7
 - распределение битов 6-24, 6-25
- выбор, модулей Вх/Вых 2-1
- выбор режима 1-9
- выбора режима 1-9
- выполнение программы 1-9
- выполнение, состояние программы 14 - 4
- выход, управление после ошибки 16 - 3
- вычисление
- время выполнения 9-5
 - время сканирования процессора 9-10, 9-11
 - время сканирования удаленного Вх/Вых 9-8
 - производительность 9-5
- вычисления, инструкции 22-7
- Г**
- главная программа управления 17 - 1
- глобальные биты состояния
- рэки 0-7 21 - 2
 - рэки 10-17 21 - 9
- глобальный бит блокировки, очистка, 8-11
- глобальных флагов состояния, файл 9-5
- Д**
- данных передача
- блок-трансфер, 8-6
 - время пересылки Вх/Вых 9-5
 - время пересылки через монтажную плату Вх/Вых 9-5
 - локальные расширенные Вх/Вых, 8-4
 - проектирование системы 9-9
 - типы 1-12
 - через блок-трансферы дискретных данных 7 - 14
- данных таблица
- файл, 4-13
 - файлы по умолчанию, 4-13
- данных файлы
- адресация, 4-13
 - диапазон значений, 4-14
 - неиспользованные, 4-13
 - привилегии чтение/запись 13 - 3
 - способы адресации, 4-15
- данных хранение
- бит, 4-11
 - блоки данных, 4-11
 - понятие, 4-10
 - структуры, 4-11
 - типы, 4-11
 - файлы, 4-11
 - члены, 4-11
- диагностики, инструкции 22-18
- дискретные передачи данных для режима адаптера 7 - 8
- дискретных данных передача
- локальные расширенные Вх/Вых 8-4
 - между сканером и каналом адаптера удаленных Вх/Вых 7 - 6
 - процессоро-резидентные Вх/Вых, 5-3
 - длина кабеля для RS-232, -422A и -423 11-5
 - дополнительные Вх/Вых 6-8
- Е**
- Ethernet

- IP адрес 12 - 3
 Адресация 12 - 2
 дополнительные функции 12 - 11
 коды ошибок 12 - 16
 конфигурация
 использование BOOTP 12 - 5
 использование программного обеспечения 12 - 3
 маска сегмента 12 - 12
 связь 12 - 1
 сообщения 12 - 18
 статусные данные 12 - 17
 требования к сети 12 - 1
 Характеристики процессоров 12 - 18
 широковежательный адрес 12 - 11
 шлюз 12 - 12
- З**
- Задание STI 18 - 1
 Задание Файла Конфигурации Дискретных Передач 7 - 4
 задачи супервизорного контроля и сбора данных 11-1
 заземление
 локальных расширенных Вх/Вых, 3-7
 процессорного шасси, 3-7
 удаленных Вх/Вых, 3-7
 защита от электрических помех, 3-4
 защита от ЭСП, 3-3
 защита программ 13 - 1
 защита процессоров 13 - 5
 защитный корпус, 3-3
- И**
- игнорирование пустых слотов 9-5
 изменения битов, инструкции 22-16
 индексная адресация, 4-19
 индикаторы
 1771-ALX, 24-7
 1771-ASB, 24-5
 Ethernet, 24-4, 24-5
 PLC-5/11, -5/20 1-3
 PLC-5/20E 1-6
 PLC-5/30 1-4
 PLC-5/40, -5/60 и -5/80 1-5
 PLC-5/40L и -5/60L 1-8
 PLC-5/40E и -5/80E 1-7
 коммуникации, 24-3
 локальных расширенных Вх/Вых, 24-4
 процессора, 24-3
 Инсталляция ,BOOTP, 12 - 7
 инструкции
 ASCII 22-25
 блок-трансферов 22-24
 время выполнения
 инструкции бита и слова 22-28
 файловые, управления программой и ASCII 22-31
 вычисления 22-7
 диагностики 22-18
 изменения битов 22-16
 краткий справочник 22-1
 логические 22-14
- память
 инструкции бита и слова 22-28
 файловые, управления программой и ASCII 22-31
 перемещения 22-16
 ПИД 22-23
 преобразования 22-15
 регистрового сдвига 22-19
 релейные 22-2
 секвенсора 22-20
 сообщений 22-23
 сравнения 22-5
 счетчика 22-4
 таймера 22-3
 управления программой 22-21
 файловые 22-17
 интервалы, 4-12
- К**
- кабели
 Belden 9463 6-3
 для программирования 10-1
 коммуникационные интерфейсы, 25-5
 назначение контактов, 25-2
 разметка прокладки, 3-3
 сети DN+ 10-1, 10-2
 трассировка соединений, 3-4
 удаленных Вх/Вых 6-5
 канал
 Ethernet
 использование BOOTP 12 - 5
 использование программного обеспечения 12 - 3
 конфигурация
 удаленный адаптер Вх/Вых 7 - 3
 Конфигурация адаптера 7 - 2
 конфигурация сканера локальных расширенных Вх/Вых, 8-9
 конфигурирование
 сеть DN+ 10-2, 10-3
 локальные расширенные Вх/Вых, 8-10
 сканер удаленных Вх/Вых 6-6
 канал связи по DN+ 10-2
 канала статус
 DN+ 10-5
 Ethernet 12 - 17
 локальные расширенные Вх/Вых, 8-13
 последовательный 11-18
 удаленных Вх/Вых адаптера 7 - 17
 классы, привилегии 13 - 3
 назначение на узел 13 - 4
 назначение на канал 13 - 4
 назначение на файл офлайн 13 - 4
 ключей, установка 20-2
 количество рэков, как оно влияет на время сканиров 9-7
 коммуникации
 1771-ASB 6-11
 изменение метода связи 11-17
 с оборудованием по последовательному порту 11-1

- скорость 9-6
 «точка-точка» 11-3, 11-6
 удаленные Вх/Вых 6-1
 характеристики 20-2
 часть времени связи для системных функций 21 - 10
 коммуникации мастера
 использование инструкции сообщений 11-4
 стандартный режим 11-4
 коммуникации
 локальные расширенные Вх/Вых, 8-1
 процессоро-резидентные Вх/Вых, 5-1
 сеть DH+ 10-1
 через канал адаптера 7 - 1
 комплементарные Вх/Вых, 4-22
 компоненты
 лицевая панель 1-2
 расстояния, 3-2
 конфигурирование
 Ethernet
 использование BOOTP 12 - 5
 использование программного обеспечения 12 - 3
 изменение метода связи 11-17
 канала сканера локальных расширенных Вх/Вых 8-9
 канала сканера удаленных Вх/Вых 6-6
 последовательного порта 11-2, 11-5
 протокола DF1 для станции Master 11-10
 протокола DF1 для станции Slave 11-8
 процедура подачи питания 15 - 2
 режим User 11-15
 «точка-точка» 11-6
 удаленные Вх/Вых канала адаптера 7 - 2
 файл состояния Вх/Вых 6-7
 шасси
 блок питания, 23-6
 локальных расширенных Вх/Вых, 23-10
 косвенная адресация, 4-18
- Л**
- логики скан
 определение, 5-2
 логики сканирование
 время 9-2
 логические инструкции 22-14
 логический адрес
 мнемоника, 4-17
 указание, 4-16
 локальные расширенные Вх/Вых
 адресация, 8-3
 блок-трансферы 6-13
 интерпретирование статуса, 8-13
 конфигурационные контакты, 23-10
 определение 1-14
 передача данных, 8-4
 подключение, 8-2
 подключение Вх/Вых модулей, 8-4
 поиск и устранение неисправностей, 24-4, 24-7
 расчет времени выполнения блок-трансферов, 8-6
 скан лист, 8-10
 требования по программированию, 8-9
 установка переключателей, 23-7, 23-9
- устройства, которые можно соединять, 8-1
- М**
- метод «ответвлений» 6-5
 метод связи, основанный на инструкциях сообщений 11-4
 метод «ответвлений» 10-2
 мнемоника, адресация, 4-17
 модули блок-трансферов, 4-8
 модуль памяти, передача информации, 23-4
 мониторинг
 PII 19-6
 каналов сканера удаленных Вх/Вых 6-21
 последовательный порт, состояние 11-18
 сеть DH+ 10-5
 сканер удаленных Вх/Вых 6-21
 состояние каналов связи DH+ 10-5
 состояния канала Ethernet 12 - 17
 статус локальных расширенных Вх/Вых 8-13
 статуса супервизорного процессора 7 - 16
 удаленных Вх/Вых канала адаптера 7 - 17
 монтаж
 размеры блока питания, 3-6
 размеры шасси Вх/Вых, 3-5
 монтажная плата
 локальные расширенные Вх/Вых, 23-9
 монтажная плата, 23-5
 потребление тока 20-1
 установка переключателей, 23-4, 23-5
 MCR P - 3
- Н**
- немедленные Вх/Вых
 использование с соседними модулями блок-трансферов 5-3
 локальные расширенные Вх/Вых 8-5
 передача данных 6-11
- О**
- Онлайн P - 3
 онлайн редактирование
 STIs 18 - 1
 служебные действия 9-4
 сообщения и блок-трансферы 9-11
 определение адаптера удаленных Вх/Вых 1-12
 определение сканера удаленных Вх/Вых 1-11
 опрос
 схемы 11-13
 требования 11-4
 оптимизация
 системы 9-5
 основная ошибка
 обнаружение 16 - 2
 очистка 16 - 6
 реакция 16 - 1
 основные программы управления 1-10
 относительная влажность, 3-1
 Офлайн P - 3
 офлайн файл, привилегии 13 - 4
 охлаждение, 3-1

- очереди ожидания 6-14
 ошибка, состояние программы 14 - 4
 ошибки
 монитор 16 - 11
 монитор статуса 16 - 11
 неосновные 16 - 11
 неосновные блок-трансферов 6-15
 основное повреждение 16 - 1
 основные и неосновные 16 - 10
 очистка 16 - 6
 расширенном локальном рэке Вх/Вых 16 - 3
 резидентном или расширенном локальном рэке Вх/Вых 16 - 3
 удаленного рэка Вх/Вых 16 - 3
 удаленного шасси Вх/Вых 16 - 3
- П**
- память
 защита, 23-4
 модули 20-1
 область, 4-10
 оптимизация, 4-21
 оставление интервалов, 4-12
 программных файлов, 4-14
 размеры файлов таблицы данных, 4-13
 Передача дискретных данных P - 3
 передача дискретных данных
 Монитор статуса канала адаптера 7 - 15
 Монитор статуса супервизорного процессора 7 - 16
 режим адаптера 7 - 8
 передняя панель
 PLC-5/11, -5/20 1-3
 PLC-5/20E 1-6
 PLC-5/30 1-4
 PLC-5/40, -5/46, -5/60, -5/80 и -5/86 1-5
 PLC-5/40L и -5/60L 1-8
 PLC-5/40E и -5/80E 1-7
 переключатели
 PLC-5/11, -5/20 1-3
 PLC-5/20E 1-6
 PLC-5/30 1-4
 PLC-5/40, -5/60, -5/80 1-5
 PLC-5/40L и -5/60L 1-8
 PLC-5/40E и -5/80E 1-7
 использование 1-9
 перемещения, инструкции 22-16
 ПИД 22-23
 плотность, модулей Вх/Вых 2-2
 подготовка к монтажу
 категории проводников, 3-4
 план трассировки, 3-3
 подача питания 15 - 2
 подключение Ethernet 12 - 1
 подпрограмма обработки ошибок
 Блок-трансфер данных 16 - 10
 в локальном рэке Вх/Вых 16 - 3
 изменение из релейно-контактной логики 16 - 8
 использование 16 - 1
 конфигурация 16 - 4
 обнаружение основных повреждений 16 - 2
 очистка 16 - 6
 проверка 16 - 10
 программирование 16 - 6
 программный сторож 16 - 5
 рекомендации по использованию 14 - 3
 специальные подпрограммы 14 - 2
 устранение с использованием релейно-контактной ло 16 - 9
 подпрограмма подачи питания
 обзор использования 14 - 2
 приоритет выполнения 14 - 4
 подпрограммы 1-10
 поиск и устранение неисправностей
 Ethernet, 24-5
 коммуникации, 24-3
 локальных расширенных Вх/Вых, 24-4, 24-7
 процессор, 24-3
 удаленных Вх/Вых, 24-5
 понятие
 PLC-5, процессор 1-1
 памяти процессора, 4-10
 последнее состояние, 23-4, 23-5
 последовательное подключение 10-2
 последовательные соединения 6-5
 последовательные функциональные схемы 1-10
 последовательный (serial)
 DF1 режим Master 11-3, 11-10
 состояние 11-18
 DF1 режим Slave 11-3, 11-8
 DF1 режим Slave, состояние 11-18
 DF1 режим «точка-точка» 11-3, 11-6
 выбор протокола связи 11-1
 изменение метода связи 11-17
 изменение режима 11-4
 использование канала 0 11-2
 кабель, 25-2
 конфигурирование 11-5
 назначение контактов, 25-1
 оборудование 11-2
 переключатель SW2 11-2
 подключение 11-5
 протоколы 11-5
 режим System («Системный») 11-2, 11-3
 режим User ("Пользователь") 11-2
 режим User (протокол ASCII) 11-15
 состояние 11-19
 состояние режима User (ASCII) 11-20
 состояние связи режима «точка-точка» 11-18
 «точка-точка» 11-6
 последовательный порт. 12 - 3
 посылка сообщений станцией Master 11-4
 преобразования, инструкции 22-15
 прерывания 9-3
 прескан, 24-8
 привелегии
 назначены класса на канал 13 - 4
 назначены класса на файл офлайн 13 - 4
 назначение файлу данных 13 - 5
 определение классов 13 - 3
 типы 13 - 1

- пример
 32-точечной адресации, 4-6
 8- и 16-точечной адресации, 4-5
 BOOTP 12 - 9
 время выполнения блок-трансфера в локальных
 расшир 8-8
 расчет времени процессора 9-11
 приоритеты выполнения 14 - 4
 проводники, 3-4
 программа, состояние 14 - 4
 программирование
 блок-трансферов для канала адаптера 7 - 8
 возможности 1-10
 кабель, соединение контактов, 25-5
 локальные расширенные Вх/Вых, 8-9
 множество блок-трансферов для канала адаптера
 7 - 10
 подпрограммы ошибок 16 - 6
 советы по разработке для улучшения
 производительности 9-12
 соглашения 14 - 1
 программные константы, допустимый диапазон, 4-14
 программный сторож 16 - 5
 программный файл
 максимальный размер 9-12
 память, 4-14
 хранение, 4-14
 программой, инструкции управления 22-21
 программы, сканирование
 MCR, 17 - 4
 введение, 5-2
 действия, которые могут влиять на время 9-1
 инструкции 9-2
 использование прерываний 9-3
 ложная и истинная логика 9-2
 определение, 5-2
 состояния входов 9-2
 эффекты служебного времени 9-4
 производительность
 Ethernet процессоров 12 - 18
 PII 19-4
 влияние онлайн редактирования 9-11
 время пересылки Вх/Вых 9-5
 время пересылки через монтажную плату Вх/Вых 9-
 5
 время сканирования процессора 9-10
 время сканирования удаленных Вх/Вых 9-6
 вставка цепей около предела в 56 к 9-12
 вычисление 9-5
 оптимизация времени выполнения инструкций, 4-21
 оптимизация памяти процессора, 4-21
 оптимизация производительности системы 9-1
 сеть DN+ 10-9
 служебные действия 9-4
 прохождение передачи 10-9
 процессом, инструкции управления 22-23
 процессора время
 обмен данными 9-10
 пример 9-11
 процессора, файл состояния 21 - 1
- процессорно-резидентные Вх/Вых
 блок-трансферы 6-20
 форсирование 14 - 1
 ПФС переходы, форсирование 14 - 2
- Р**
- PII
 когда использовать 14 - 3
 размеры
 блока питания, 3-6
 шасси, 3-5
 размеры шасси, 3-5
 размещение
 аппаратуры, 3-1
 Вх/Вых модулей 2-1, 2-3
 локальных расширенных Вх/Вых, 8-2
 модулей локальных расширенных Вх/Вых, 8-4
 шасси, 3-5
 размещение систем
 расположение, 3-5
 условия окружающей среды, 3-1
 разработка систем
 распределенное управление 1-2
 централизованное управление 1-1
 расключение
 сети удаленных Вх/Вых 6-5
 расположение 20-2
 распределенное управление 1-2
 расстояния от шасси, 3-2
 расчет времени
 блок-трансфер 6-13
 выполнения блок-трансферов, 8-6
 вычисление времени выполнения блок-трансфера
 9-7
 инструкции 9-2
 инструкции бита и слова 22-28
 использование прерываний 9-3
 ложная и истинная логика 9-2
 локальные расширенные Вх/Вых, 8-5
 немедленного обновления Вх/Вых 6-11
 оптимизация для удаленных Вх/Вых 9-8
 ответа сети DN+ 10-13
 пересылка дискретных данных, размещение для
 оптимизации 2-3
 прескана, 24-8
 пример 9-11
 скорость связи 9-6
 служебные действия 9-4
 состояния входов 9-2
 трансиверы, 25-10
 узлы 10-13
 файл глобальных флагов состояния 9-5
 файловые, управления программой и ASCII
 инструкции 22-31
 цикл сканирования Вх/Вых 6-11
 Расширенная локальная сеть Вх/Вых P - 3
 Расширенные локальные Вх/Вых P - 3
 Редактирование файла
 BOOTPTAB.TXT 12 - 7
 Редактирование файла конфигурации BOOTP 12 - 7

- режим PROGRAM (PROG) 1-9
 - режим Remote (REM) 1-9
 - режим RUN ("Работа") 1-9
 - режим User («Пользователь») 11-2, 11-15
 - режим связи «точка-точка» 11-3
 - режимы
 - адаптера 1-12
 - адресации, 4-4
 - ключ 1-9
 - сканера 1-11
 - Резидентные локальные Вх/Вых Р - 3
 - Релейно-контактная Р - 3
 - релейно-контактная логика 1-10
 - релейные, инструкции 22-2
 - рэка, биты управления
 - рэки 0-7 21 - 9
 - рэки 10-17 21 - 9
 - рэка размер
 - удаленных Вх/Вых 6-9
 - рэки, количество адресуемых 20-3
 - рэков
 - адресация, 8-10
 - назначение, локальных расширенных Вх/Вых, 8-3
 - определение, 4-2
 - процессоро-резидентный локальный Вх/Вых, 4-9
 - размер, относительно размера шасси и режима адреса 4-9
 - удаленные Вх/Вых, 4-10
- С**
- сборки переключателей
 - SW1, 23-2
 - SW2, 23-3
 - локальных расширенных Вх/Вых, 23-9
 - разъем шасси, 23-6
 - удаленных Вх/Вых, без комплементарных Вх/Вых, 23-7
 - шасси, 23-4, 23-5
 - связь в режиме Slave 11-8
 - сдвига регистрового, инструкции 22-19
 - секвенсора, инструкции 22-20
 - сертификация 20-2
 - Сеть удаленных Вх/Вых Р - 3
 - символ, адрес, 4-20
 - системный режим 11-2
 - DF1 режим Master 11-3
 - DF1 режим Slave 11-3
 - состояние 11-19
 - «точка-точка» 11-3
 - системы
 - проектирование 9-9
 - производительность 9-1
 - характеристики 20-1
 - сканирования список
 - изменение 6-10
 - ограничения 6-10
 - ограничения, 8-11
 - содержание 6-9
 - создание 6-9
 - локальных расширенных Вх/Вых, 8-10
 - сканер
 - изменение списка сканирования, 8-11
 - конфигурация каналов 6-8
 - создание скан листа, 8-12
 - сканер удаленных Вх/Вых, конфигурация 6-8
 - сканера режим
 - блок-трансфер 6-15
 - конфигурация канала 6-6
 - конфигурирование каналов локальных расширенных Вх/ 8-9
 - определение 1-11
 - просмотр состояния 6-21
 - связь с 1771-ASB 6-11
 - создание списка сканирования, 8-11
 - создание списка сканирования 6-9
 - сканирование
 - данных блок-трансфера 6-13
 - сканирование, список
 - как количество вхождений влияет на время сканирова 9-7
 - слова, хранение данных, 4-11
 - служебные действия
 - определение, 5-2
 - совет
 - адресация и размещение локальных расширенных Вх/Вы 8-2
 - адресация локальных расширенных Вх/Вых рэков, 8-3
 - группирование данных, 4-12
 - к программированию блок-трансферов 6-20
 - назначение номеров удаленным Вх/Вых рэкам, 4-10
 - назначение привилегии 13 - 3
 - написание PII программ 19-2
 - оптимизация времени выполнения инструкций, 4-21
 - организация файлов, 4-12
 - проектирование кабелей сети удаленных Вх/Вых 6-5
 - проектирование сети DN+ 10-14
 - проектирование сети удаленных Вх/Вых 6-4
 - разделение файлов глобальных флагов состояния 9-5
 - размещение модулей блок-трансферов, 5-3
 - редактирование списка сканирования для обновления 6-10
 - советы по разработке программы 9-12
 - требования для локальных расширенных Вх/Вых, 8-9
 - указания для PII 19-4
 - сообщения
 - коды ошибок Ethernet 12 - 16
 - редактирование онлайн 9-11
 - сеть DN+ 10-10
 - сообщения, инструкции 22-23
 - Состояние выполняемой программы 14 - 4
 - сравнения, инструкции 22-5
 - стандартный способ связи 11-4
 - станция Master 11-10
 - DF1 режим Master 11-3
 - статус
 - сканер удаленных Вх/Вых 6-21
 - статусных битов, монитор 16 - 11

- удаленных Вх/Вых 6-21
статуса информация
PII 19-6
STIs 18 - 4
статуса файл
процессора 21 - 1
размер, 4-13
структурированный текст 1-10
структуры, хранения данных, 4-11
счетчика, инструкции 22-4
счетчики диагностики
сеть DN+ 10-5
удаленных Вх/Вых 6-21
- T**
- таблица активных узлов 10-5
таблица отображения
адрес, 4-15
входов и выходов 4-1
таймера, инструкции 22-3
тепло, рассеиваемое 20-1
терминатор сети DN+ 10-2
терминаторы 6-5
типы данных, допустимые типы, 4-14
типы, хранение данных, 4-11
точка, определение, 4-2
трассировка, 3-3
- У**
- удаленное изменение режима 11-4
Удаленное шасси Вх/Вых P - 3
удаленные Вх/Вых
блок-трансферы 6-13, 6-15
введение 6-3
время сканирования 9-6
вычисление времени сканирования 9-8
длины кабеля 6-5
как блок-трансферы влияют на время сканирования 9-7
количество рэков в списке сканирования 9-7
максимальное количество оборудования 6-5
настройка канала сканера 6-6
настройка системы 6-4
обзор конфигурации 6-1
оптимизация времени сканирования 9-8
передача данных 6-11
проектирование 9-9
программирование блок-трансферов 7 - 8
связь 6-1
скорость связи 9-6
состояние 6-21
список сканирования 6-3, 6-9
терминаторы сети 6-5
файл состояния Вх/Вых 6-7
шаги конфигурации 6-12
удаленные Вх/Вых, конфигурация заземления, 3-6
ударостойкость, характеристики 20-1
указания
требования к программированию PII 19-4
- Вх/Вых, выбор 2-1
Вх/Вых, количество точек (плотность), выбор 2-2
выбор подпрограммы прерываний 14 - 3
использования типов адресации, 4-8
когда использовать подпрограмму прерываний 14 - 3
надлежащих условий эксплуатации, 3-1
по размещению локальных расширенных Вх/Вых, 8-4
применение сети DN+ 10-14
программирование STI 18 - 1
прокладка кабелей 3-4
размещение модулей Вх/Вых 2-3
уменьшение времени сканирования 17 - 2
условия эксплуатации
надлежащие, 3-1
характеристики 20-1
установка переключателей SW1 10-2
установка переключателей, справочник, 23-1
установочные переключатели
SW1, 23-2
SW2, 23-3
монтажная плата, 23-4
удаленные Вх/Вых, без комплементарных Вх/Вых, 23-7
устройства
в сети удаленных Вх/Вых 6-2
максимум 6-4
связь по последовательному порту 11-2
сеть DN + 10-1
- Ф**
- файл
DTLBOOTD.EXE 12 - 9
DTLBOOTW.EXE 12 - 9
Файл Конфигурации Дискретных Передач 7 - 2
файл основных флагов состояния 10-4, 10-5
файловые инструкции 22-17
файловые, управления программой и ASCII, инструкции 22-31
файлы, 4-14
хранение данных, 4-11
часто используемые, 4-21
форсирование
входов и выходов 14 - 1
переходов ПФС 14 - 2
- Х**
- характеристики 20-1
хранение
маски сегмента 12 - 12
программных файлов, 4-14
температура, 3-1
- Ц**
- Централизованное управление 1-1
- Ч**
- часы, процессора 20-1
члены, хранение данных, 4-11

Ш

- шасси
 - конфигурационные контакты, 23-6
 - размеры, 3-5
 - размещение 3-2
 - размещение в 20-2
 - установка переключателей, 23-4, 23-5
- широковещательный адрес 12 - 11
- шлюз 12 - 12
- Иностранные термины**
- ASCII
 - инструкции 22-25
 - настройка последовательного порта 11-15
- Belden 9463 6-3, 6-5
- BOOTP сервер 12 - 6
- CE, соответствие требованиям 20-2
- DF1 мастер
 - конфигурирование 11-10
 - последовательность передачи 11-13
 - связь 11-3
- DF1 режим Slave 11-3, 11-8
- DH+ связь
 - внутреннее время обработки 10-12
 - диагностические счетчики 10-5
 - длина кабеля 10-2
 - кому предназначено сообщение 10-11
 - конфигурация каналов 10-2
 - лучшее время ответа 10-13
 - определение адреса процессора 10-3
 - оценка работы 10-9
 - поиск и устранение неисправностей, 24-2
 - проектирование кабелей 10-2
 - просмотр состояния 10-5
 - прохождение передачи 10-9
 - размер и число сообщений 10-10
 - связь с устройствами 10-1
 - скорость передачи 10-2
 - советы по проектированию 10-14
 - счетчики диагностики 10-5
 - таблица активных узлов 10-5
 - терминатор 10-2
 - требования к приложениям 10-14
 - узлы/время 10-9
 - файл основных флагов состояния 10-5
- DTLBOOTD.EXE 12 - 9
- DTLBOOTW.EXE 12 - 9
- Ethernet P - 3
 - поиск и устранение неисправностей, 24-5
- ladder logic 1-10
- MCPs
 - использование 17 - 1
 - конфигурация 17 - 3
 - монитор 17 - 4
 - обработка 17 - 1
- PII 9-3, 9-4, 19-5
 - использование 19-1
 - конфигурирование 19-5
 - мониторинг 19-6
- написание релейно-контактной логики 19-2
- пример программы 19-2
- производительность 19-4
- с инструкциями блок-трансферов 19-3
- состояние 19-6
- требования к программированию 19-4
- Remote I/O 6-1
- RUN ("Работа") режим 1-9
- SCADA 11-1
- SFCs 1-10
- STI 9-3, 9-4, 16 - 10
 - время сканирования 18 - 3
 - задание 18 - 1
 - использование 18 - 1
 - конфигурация 18 - 3
 - пример выполнения релейно-контактной логики 18 - 2
 - с инструкцией блок-трансфера 18 - 2
 - с релейно-контактной логикой 18 - 1
 - статус 18 - 4



Allen-Bradley, подразделение Rockwell Automation помогает заказчикам улучшать производительность и качество больше 90 лет. Мы разрабатываем, производим и поддерживаем широкий диапазон средств автоматизации во всем мире. Сюда входят логические процессоры, силовые устройства и устройства управления движением, интерфейсы оператора, датчики и различное программное обеспечение. Rockwell - одна из мировых ведущих технологических компаний.

Представительства во всем мире



Австралия • Австрия • Англия • Аргентина • Бахрейн • Бельгия • Бразилия • Болгария • Венгрия • Венесуэла • Гватемала • Германия • Греция • Гондурас • Гонг Конг • Денмарк • Египет • Индия • Индонезия • Израиль • Италия • Иордания • Испания • Китай • Колумбия • Коста Рика • Кипр • Канада • Корея • Кувейт • Катар • Малайзия • Мексика • Новая Зеландия • Объединенные Арабские Эмираты • Оман • Пакистан • Перу • Польша • Португалия • Пуэрто Рико • Румыния • Россия • Сальвадор • Саудовская Аравия • Сингапур • Словакия • Словения • Тайвань • Таиланд • Турция • Уругвай • Финляндия • Франция • Филиппины • Хорватия • Чили • Швеция • Эквадор • Южная Африка • Югославия • Ямайка • Япония

Штаб-квартира Allen-Bradley, 1201 South Second Street, Milwaukee, WI 53204 USA, Тел: (1)414 382-2000, Факс: (1)414 382-4444

Представительство Allen-Bradley в СНГ, Большой Строченовский переулок, 22/25, 113054, Москва, Россия, Тел: (095) 956-0464.

Перевод выполнен ЗАО «ЭЛСИС», авторизованным дистрибутором Rockwell Automation, улица Орджоникидзе, 35, 654007, Новокузнецк, Россия, Тел: (3843) 45-53-66, Факс: (3843) 49-13-43, E-mail: root@elsys.kemerovo.su