

SIEMENS

SIMATIC

Загружаемый драйвер для PtP-соединения с протоколом MODBUS RTU-формата для модулей CP (в случае, когда S7 - ведомое устройство)

Руководство

Данное руководство является
частью пакета документации
с заказным номером:

6ES7870-1AB00-0YA0

Выпуск 05/2003
A5E00218418-04

Предисловие, содержание

Описание	1
Инсталляция	2
Режим работы	3
Настройка драйвера	4
Отладка FB	5
Интерфейс CPU - CP	6
Протокол передачи данных	7
Функциональные коды	8
Диагностика драйвера	9
Диагностика FB	10
Приложения	
Технические данные	A
Схемы многоточечного соединения	B
Список литературы	C
Глоссарий, предметный указатель	

Указания по технике безопасности

Данное руководство содержит указания, которые вы должны соблюдать для обеспечения собственной безопасности, а также защиты от повреждений оборудования. Эти замечания выделены в руководстве символами, как показано ниже, в соответствии с уровнем опасности:



Опасность

указывает, что несоблюдение надлежащих мер предосторожности может привести к гибели людей, тяжким телесным повреждениям или существенному имущественному ущербу.



Предупреждение

указывает, что несоблюдение надлежащих мер предосторожности может привести к гибели людей, тяжким телесным повреждениям или к существенному имущественному ущербу.



Предостережение

указывает, что несоблюдение надлежащих мер предосторожности может привести к легким телесным повреждениям и нанесению незначительного имущественного ущерба.

Предостережение

указывает, что несоблюдение надлежащих мер предосторожности может привести к нанесению незначительного имущественного ущерба.

Замечание

привлекает ваше внимание к особо важной информации о продукте, обращении с ним, или к соответствующей части документации.

Квалифицированный персонал

К монтажу и работе на этом оборудовании должен допускаться только квалифицированный персонал. Квалифицированный персонал – это люди которые, имеют право вводить в эксплуатацию, заземлять и маркировать электрические цепи, оборудование и системы в соответствии с установленной практикой и стандартами техники безопасности.

Надлежащее использование

Примите во внимание следующее:



Предупреждение

Это устройство и его компоненты могут использоваться только для применений, описанных в каталоге или технической документации, и совместно только с теми устройствами или компонентами других производителей, которые были одобрены или рекомендованы фирмой Siemens.

Этот продукт может правильно и надежно функционировать только в том случае, если он правильно транспортируется, хранится, устанавливается и монтируется, а также если эксплуатируется и обслуживается в соответствии с рекомендациями.

Товарные знаки

SIMATIC®, SIMATIC HMI® and SIMATIC NET® - это зарегистрированные товарные знаки SIEMENS AG.

Некоторые другие обозначения, использованные в этих документах, также являются зарегистрированными товарными знаками; если они используются третьей стороной для своих собственных целей, то соответствующие права собственности могут быть нарушены.

Copyright © Siemens AG 2003 Все права защищены

Воспроизведение, передача или использование этого документа или его содержания не допускаются без письменного разрешения правообладателя. Нарушители будут нести ответственность за нанесенный ущерб. Все права, включая права на патенты, практические модели и конструкции, защищены.

Отказ от ответственности

Мы проверили содержание этого руководства на соответствие с описанным аппаратным и программным обеспечением. Так как отклонения не могут быть полностью исключены, то мы не можем гарантировать полного соответствия. Однако данные, приведенные в этом руководстве, регулярно пересматриваются и все необходимые исправления вносятся в последующие издания. Мы будем благодарны за предложения по улучшению содержания.

Содержание

1	Описание	1-1
1.1	Возможности приложения	1-1
1.2	Требования к программному и аппаратному обеспечению	1-2
1.3	Краткие сведения о протоколе GOULD-MODBUS	1-4
1.4	Примечания	1-5
2	Инсталляция	2-1
2.1	Использование защитного ключа-заглушки	2-1
2.2	Интерфейсы	2-2
3	Режим работы Data Link	3-1
3.1	Компоненты ПО Data Link для совместной работы SIMATIC с ведомым MODBUS-устройством	3-1
3.2	Распределение задач	3-3
3.3	Используемые MODBUS функциональные коды	3-3
3.4	Области данных в SIMATIC CPU	3-4
3.5	Доступ с функциональными кодами для битов	3-5
3.6	Доступ с функциональными кодами для регистров	3-6
3.7	Блокировка / ограничение функций записи	3-8
4	Настройка драйвера	4-1
4.1	Инсталляция драйвера в STEP 7 в программаторе / в ПК	4-1
4.2	Деинсталляция драйвера	4-2
4.3	Конфигурирование данных коммуникаций	4-2
4.3.1	Конфигурирование данных коммуникаций для CP 341	4-2
4.3.2	Конфигурирование данных коммуникаций для CP 441-2	4-3
4.4	Назначение параметров для CP	4-4

4.4.1	Назначение параметров для CP 341	4-4
4.4.2	Назначение параметров для CP 441-2	4-5
4.5	Конфигурирование данных коммуникаций проекта	4-8
4.6	Назначение параметров для загружаемого драйвера	4-9
4.6.1	Протокол для ведомого устройства Modbus	4-10
4.6.2	Преобразование MODBUS-адреса для битовых функций	4-13
4.6.3	Преобразование Modbus-адресов для функций для регистров	4-17
4.6.4	Ограничения для функций записи	4-19
4.6.5	Интерфейс X27 (RS 422/485)	4-21
4.7	Загрузка данных конфигурации и параметризации для CP 341	4-23
4.8	Загрузка данных конфигурации и параметризации для CP 441-2	4-24
4.9	Характеристики запуска CP	4-25
4.10	Назначение параметров запуска ("Startup of the CPU")	4-26
5	Отладка коммуникационного FB	5-1
5.1	Инсталляция FB	5-1
5.2	Проект STEP 7	5-2
5.3	Параметры FB 80 (CP 341)	5-3
5.4	Параметры FB 180 (CP 441-2)	5-4
5.5	Вызов программы	5-5
5.6	Циклическая обработка программы	5-8
6	Интерфейс CPU - CP	6-1
6.1	Интерфейс CPU - CP для модуля CP 341	6-1
6.1.1	Общая информация	6-1
6.1.2	Параметр длины LENGTH при передаче данных между CP и CPU	6-1
6.1.3	Консистентность данных	6-2
6.2	Интерфейс CPU - CP для модуля CP 441-2	6-3
6.2.1	Общая информация	6-3
6.2.2	Параметр длины LENGTH при передаче данных между CP и CPU	6-3
6.2.3	Консистентность данных	6-4
7	Протокол обмена данными	7-1

8	Функциональные коды	8-1
8.1	Функциональный код 01 - Read Output Status (считывание состояния выхода)	8-2
8.2	Функциональный код 02 - Read Input Status (считывание состояния входа)	8-6
8.3	Функциональный код 03 - Read Output Registers (считывание выходных регистров)	8-9
8.4	Функциональный код 04 - Read Input Registers (считывание входных регистров)	8-13
8.5	Функциональный код 05 - Force Single Coil (назначение для одной катушки)	8-16
8.6	Функциональный код 06 - Preset Single Register (предустановки для одного регистра)	8-19
8.7	Функциональный код 08 - Loop Back Diagnostic Test (тестирование коммуникационного соединения)	8-22
8.8	Функциональный код 15 - Force Multiple Coils (назначения для нескольких катушек)	8-23
8.9	Функциональный код 16 - Preset Multiple Registers (предустановки для нескольких регистров)	8-26
9	Диагностика драйвера	9-1
9.1	Средства диагностики модуля CP 341	9-2
9.1.1	Диагностика с использованием элементов индикации CP 341	9-2
9.1.2	Диагностические сообщения функциональных блоков CP 341	9-3
9.2	Функции диагностики CP 441-2	9-4
9.2.1	Диагностика с использованием элементов индикации CP 441-2	9-4
9.2.2	Диагностические сообщения системных функциональных блоков CP 441-2	9-5
9.2.3	Считывание сообщений об ошибках из области SYSTAT для CP 441-2	9-6
9.2.4	Диагностика посредством области для сообщений об ошибках SYSTAT	9-7
9.3	Таблицы диагностических событий (Events) / ошибок (Errors)	9-8
9.3.1	Номера ошибок в области SYSTAT для "CPU Job Errors" ("Ошибки в работе CPU")	9-9
9.3.2	Номера ошибок в области SYSTAT для "Receive Errors" ("Ошибки при приеме")	9-9
9.3.3	Номера ошибок в области SYSTAT для "General Processing Errors" ("Общие ошибки обработки")	9-12
10	Диагностика коммуникационного FB	10-1
10.1	Диагностика посредством ERROR_NR, ERROR_INFO	10-1
10.1.1	Ошибки при инициализации	10-2
10.1.2	Ошибки при обработке функционального кода	10-2
10.1.3	Другие ошибки	10-4

A	Технические данные	A-1
B	Схемы многоточечных соединений	B-1
C	Список литературы	C-1
	Словарь терминов	G-1
	Предметный указатель	I-1

Предисловие

Назначение

Информация, которую Вы найдете в данном руководстве, поможет Вам устанавливать и настраивать канал обмена данными между CP и системой управления с протоколом обмена "Modbus".

Рекомендуемый базовый уровень подготовки пользователей

Для того, чтобы материал данного руководства был Вам понятен, Вы должны иметь общие знания по технике автоматизации.

Кроме того, Вы должны уметь обращаться с компьютерами или аналогичными устройствами (например, с программаторами) под управлением операционной системы Windows 95/98/2000/NT или XP. Так как загружаемый драйвер базируется на программном обеспечении STEP 7, то Вы также должны уметь использовать ПО STEP 7. Информацию по данному вопросу Вы можете найти в руководстве "Programming with STEP 7 V5.2" ("Программирование в STEP 7 версии 5.2").

Применимость данного руководства

Данное руководство может быть применимо для следующих пакетов ПО:

ПО	Заказной номер	Начиная с версии
Loadable Driver for Point-to-Point CPs (Загружаемый драйвер для PtP-соединений с использованием CP)	6ES7 870-1AA00-0YA0	3.0

Примечание

Описание драйвера в данном руководстве корректно на момент данной публикации. В описании дается информация об изменениях по сравнению с предыдущими версиями.

Содержание руководства

В этом руководстве описываются функции загружаемого драйвера и то, как его связать с программным обеспечением и аппаратурой коммуникационных процессоров CP 341 и CP 441-2.

Руководство состоит из глав, посвященных инструкциям, и из справочных разделов (приложений).

В руководстве рассматриваются следующие темы:

- Описание ПО / Инсталляция
- Режим работы коммуникационных данных (Data Link)
- Настройка драйвера / Инсталляция / Назначение параметров
- Отладка FB / Назначение параметров / Пример для STEP 7
- Интерфейс CPU - CP
- Протокол передачи / Функциональные коды
- Диагностика драйвера

Соглашения

В документации используется сокращение CP для обозначения коммуникационных процессоров CP 341 и/или CP 441-2.

Структура данного руководства

Для быстрого поиска нужной информации руководство имеет следующие особенности:

- В основных разделах в левом поле на каждой странице представлено название темы, в котором обобщается содержание соответствующего раздела.
- В последующем за приложениями глоссарии объяснены важные термины, используемые в руководстве.
- Подробный предметный указатель поможет Вам быстро найти нужную информацию, касающуюся отдельных объектов.

Примечание

Драйвер, описанный в данном руководстве, может применяться как загружаемый протокол обмена для CP.
Он может использоваться вместо стандартных протоколов 3964R, RK512, ASCII и протокола обмена для принтера.

Предупреждение

Для описанного здесь драйвера допускаются модификации или расширения последовательностей данных обмена между CP и CPU.

Указанные модификации или расширения могут касаться в частности классов событий или номеров событий в системе диагностики.

Более того, необходимо иметь в виду, что данное руководство описывает только модификации или расширения в сравнении со стандартными функциями. Общая информация по конкретным модулям CP может быть найдена в руководствах по использованию соответствующих модулей CP.

Для корректного использования рассматриваемого драйвера Вы должны обладать подробными сведениями о том, как функционирует выбранный Вами коммуникационный процессор.

Дополнительная помощь

Если у Вас возникли любые технические вопросы, то обратитесь к местному представителю фирмы Siemens. Вы можете также получить нашу поддержку с помощью Интернета, используя следующую ссылку:

<http://www.siemens.com/automation/partner>

Учебные курсы

Мы предоставляем множество курсов, чтобы помочь Вам начать использовать программируемые контроллеры SIMATIC S7. Вы можете поддерживать контакт с местным учебным центром или с центральным учебным центром в Нюрнберге:

Nuremberg, D-90327 Germany, тел. +49 (911) 895-3200.

Интернет: <http://www.sitrain.com>

Обслуживание и поддержка с помощью Интернета

В дополнение к документации, Вы можете также получить нашу поддержку с помощью Интернета используя следующую ссылку:

<http://www.siemens.com/automation/service&support>

Здесь Вы найдете:

- Новую информацию, которая постоянно обновляется, с последними данными по продуктам, которые Вы используете.
- Правовую документацию, необходимую Вам для поиска в службе обслуживания и поддержки с помощью Интернета (Service & Support).
- Форум для обмена информацией и опытом между специалистами и пользователями.
- Контактные реквизиты для ответов на вопросы по автоматизации и приводам (Automation & Drives) в нашей базе данных.
- Информацию по местным отделениям обслуживания, по ремонту и запчастям, а также многое другое под заголовком "Services" ("Обслуживание").

Техническая поддержка A&D (A&D Technical Support)

Круглосуточно, по всему миру:



<p>Всемирная (Нюрнберг) техническая поддержка</p> <p>Мест. вр. : 0:00 - 24:00 / 365 дней Тлф: +49 (0) 180 5050-222 Факс: +49 (0) 180 5050-223 E-mail: adsupport@siemens.com GMT: +1:00</p>		
<p>Европа / Африка (Нюрнберг) Авторизация</p> <p>Мест. вр. : Пнд.-Птн. 8:00 - 17:00 Тлф: +49 (0) 180 5050-222 Факс: +49 (0) 180 5050-223 E-mail: adautorisierung@siemens.com GMT: +1:00</p>	<p>США (Джонсон-Сити) Техническая поддержка и авторизация</p> <p>Мест. вр. : Пнд.-Птн. 8:00 - 17:00 Тлф: +1 (0) 423 262 2522 Факс: +1 (0) 423 262 2289 E-mail: simatic.hotline@sea.siemens.com GMT: -5:00</p>	<p>Азия / Австралия (Пекин) Техническая поддержка и авторизация</p> <p>Мест. вр. : Пнд.-Птн. 8:00 - 17:00 Тлф: +86 10 64 75 75 75 Факс: +86 10 64 74 74 74 E-mail: adsupport.asia@siemens.com GMT: +8:00</p>
<p>Язык общения для технической поддержки и авторизации: английский или немецкий.</p>		

1 Описание

1.1 Функциональные возможности приложения

Положение в системе автоматизированного управления

Драйвер, описанный в данном руководстве, является программным продуктом, предназначенным для работы с коммуникационными процессорами CP 341 (S7-300) и CP 441-2 (S7-400).

Коммуникационные процессоры CP 341 и CP 441-2 могут использоваться в автоматизированных системах для обеспечения обмена данными с партнерскими системами.

Функции драйвера

Указанный драйвер позволяет установить коммуникационное соединение между коммуникационным процессором CP 341 или CP 441-2 и системой управления, поддерживающей Modbus (например, Modicon или Honeywell TDC 3000).

Используемый протокол обмена **GOULD - MODBUS** имеет **RTU**-формат. Обмен данными происходит в соответствии с принципом Master - Slave (Ведущее устройство - Ведомое устройство).

Ведущее устройство (Master в SIMATIC S7) имеет активный статус во время передачи.

Функциональные коды (Function Codes) 01, 02, 03, 04, 05, 06, 08, 15 и 16 могут использоваться для коммуникаций между CP и основной системой.

Интерпретация MODBUS-адресов

MODBUS-адрес в сообщении-запросе от ведущего (master) устройства интерпретируется драйвером как S7-адрес.

Это означает, что имеется возможность:

- считывать и записывать меркеры, выходы и блоки данных
- считывать меркеры, входы, значения таймеров и счетчиков в S7 CPU.

Интерпретация MODBUS-адреса поясняется в следующих главах руководства.

Пригодные для использования интерфейсы и протоколы

Два последовательных интерфейса коммуникационного процессора CP 441-2 могут работать независимо, используя разные стандартные протоколы или протоколы загружаемых драйверов.

Вы можете использовать интерфейсные submodule RS 232, TTY или RS 422/485 (X27) в качестве интерфейса CP.

При использовании интерфейса RS 422/485 (X27) в двухпроводном режиме Вы можете подключить до 32 ведомых (slave) устройств к одному ведущему (master), тем самым обеспечивая многоточечное соединение (сеть).

Создание многоточечного соединения с другими типами физических интерфейсов требует дополнительного внешнего аппаратного обеспечения (например, Honeywell DHP).

Возможная конфигурация системы

Возможная конфигурация системы: PS, CPU 416, CP 441-2, интерфейсный submodule RS232C / TTY / X27

1.2 Требования к программному и аппаратному обеспечению

Используемые модули

Драйвер работает с модулями CP 341 и CP 441-2 с заказными номерами 6ES7 441-2AA02-0AE0 или с более новыми.

Модули CP 441-1 с номером MLFB 6ES7 441-1AA0x-0AE0 и CP 441-2 с номером MLFB 6ES7 441-2AA00-0AE0 или 6ES7 441-2AA01-0AE0 не могут быть использованы с загружаемыми драйверами.

Защитный ключ-заглушка (dongle)

Для использования в CP загружаемых драйверов Вам необходим защитный ключ-заглушка. Ключ-заглушка поставляется с драйвером.

Загрузочная память (Loading Memory) CPU (карта памяти)

Если Вы используете CP 441-2, то загружаемые драйвера загружаются в загрузочную память (loading memory) CPU после назначения параметров и пересылаются в память CP при запуске. Поэтому CPU должен иметь загрузочную память (loading memory) достаточного размера. По этой причине необходимо использовать карту памяти типа RAM или FLASH (заказной номер MLFB 6ES7 952-...).

Загружаемый драйвер для PtP-соединений CP
(S7 - ведомое устройство)
A5E00218418-04

Для каждого интерфейса CP, для которого назначается данный загружаемый драйвер, требуется загрузочная память в CPU в размере около 25 кбайтов.

Если Вы используете CP 341, то загружаемые драйвера загружаются непосредственно в CP 341. Для этого Вам не требуется загрузочная память (loading memory) в S7-300 CPU. Однако Вы должны иметь в виду, что это значит, что Вы не сможете менять модуль без программатора.

Требования к программному обеспечению

В системе должно быть установлено ПО STEP 7 Standard версии 4.02.1 или выше.

В системе должно быть установлено ПО *Point-to-Point Communication, Parameter Assignment (CP 341: PtP-соединение, назначение параметров)* (CP PtP Param) версии 4.1 или выше

Структуры данных

Перед конфигурированием в проекте структур S7-данных Вы должны убедиться, что они совместимы с пользовательскими программами ведомых MODBUS-систем (корректность функциональных кодов и Modbus-адресов).

1.3 Краткие сведения о протоколе GOULD-MODBUS

Функциональные коды

Способ обмена данными между MODBUS-системами определяется функциональными кодами (FC).

Обмен данными

Для обеспечения побитного обмена данными могут использоваться следующие функциональные коды (FC):

FC 01 Read coil (output) status (считывание состояния выхода)

FC 02 Read input status (считывание состояния входа)

FC 05 Force single coil (назначение для одной катушки)

FC 15 Force multiple coils (назначения для нескольких катушек)

Для обеспечения порегистрового обмена данными могут использоваться следующие функциональные коды (FC):

FC 03 Read holding registers (считывание выходных регистров)

FC 04 Read input registers (считывание входных регистров)

FC 06 Preset single register (предустановки для одного регистра)

FC 16 Preset multiple registers (предустановки для нескольких регистров)

Области данных

Как правило, отдельные функциональные коды FC работают в соответствии с представленной ниже таблицей:

Функциональный код	Данные	Тип данных		Вид доступа
01, 05, 15	Состояние катушки (coil) (выход)	Бит	Выход	Чтение/Запись
02	Состояние входа	Бит	Вход	Только чтение
03, 06, 16	Выходной регистр	Регистр (16 битов)	Выходной регистр	Чтение/Запись
04	Входной регистр	Регистр (16 битов)	Входной регистр	Только чтение

Представление адресов

Аналогично разделению областей с доступом для чтения/записи и с доступом только для чтения, данные на уровне пользователя могут быть представлены как показано в следующей таблице:

Функция	Тип данных	Представление адреса на пользовательском уровне
01, 05, 15	Выходной бит	0xxx
02	Входной бит	1xxx
04	Входной регистр	3xxx
03, 06, 16	Выходной регистр	4xxx

В сообщениях при передаче данных (**transmission messages**) в последовательной линии передаче адреса в пользовательской системе MODBUS указываются с 0.

Собственно в пользовательской системе MODBUS эти адреса считаются, начиная с 1.

Пример: Первый выходной удерживающий (holding) регистр в пользовательской системе представляется как регистр 40001. В сообщениях при передаче данных (transmission messages) 0000 Hex передается как адрес регистра при использовании FC 03, 06, или 16.

127-я катушка (coil) представляется как катушка 00127 в пользовательской системе и в сообщениях при передаче данных имеет адрес 007E Hex.

1.4 Примечания

Консистентность данных

Обмен данными между S7 CPU и CP выполняется поблочно (блок за блоком) посредством встроенных системных функций.

Вы можете также обратиться к разделу “Консистентность данных” в главе “Интерфейс CPU-CP” в данном руководстве.

2 Инсталляция

2.1 Использование защитного ключа-заглушки

Введение

Для того чтобы запускать CP с загружаемыми драйверами, необходим защитный ключ-заглушка. Если ключ-заглушка установлен в порт CP, то драйвер может быть загружен. Оба интерфейса CP 441-2 могут работать с загружаемыми драйверами.

Установка защитного ключа-заглушки

Для того чтобы вставить ключ-заглушку, необходимо снять модуль CP с монтажной шины.

На задней панели модуля над разъемами для подключения к системной шине есть слот, в который может быть установлен ключ-заглушка.

2.2 Интерфейсы

Интерфейс RS232C / TTY

Интерфейс RS232C / TTY позволяет создать PtP-соединение с ведущей (master) системой.

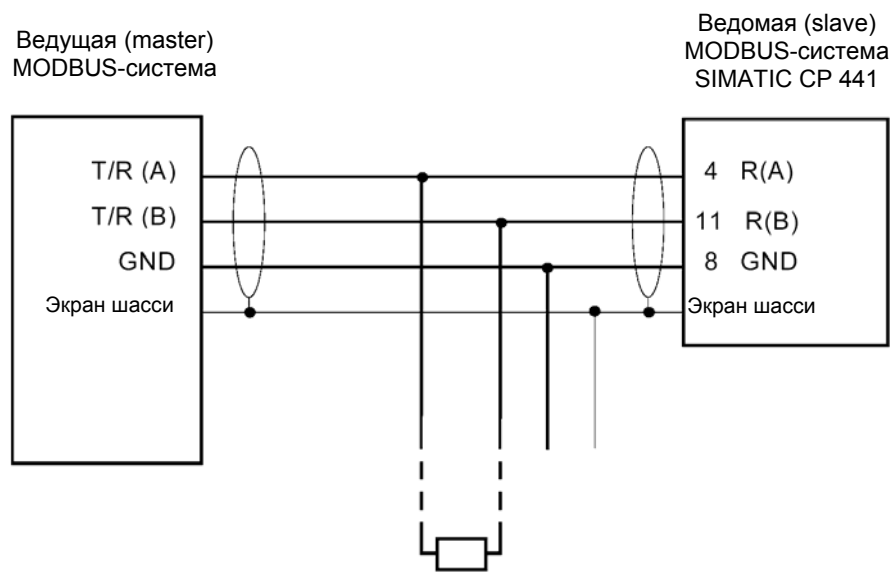
Дополнительную информацию по подключению к интерфейсам Вы можете получить в руководстве "CP 341 or CP 441-2 Point-to-Point Communication" ("PtP-соединение с CP 341 или CP 441-2").

Интерфейс X27 / RS485 (двухпроводный режим)

Интерфейс X27 / RS485 (2-хпроводный режим) позволяет установить многоточечное коммуникационное соединение (сеть) одной ведущей (master) системы с несколькими (до 32) ведомых (slave) устройств.

Драйвер CP обеспечивает переключение двухпроводной линии в режим приема или передачи.

Ниже представлена схема соединения 1 ведущая - 1 ведомая система:



Интерфейс X27 / RS422 (четырепроводный режим)

Интерфейс X27 / RS422 обеспечивает PtP-соединение с ведущей (master) системой.

Прямое многоточечное соединение (сеть) невозможно, так как оно не поддерживается аппаратурой интерфейса X27 (RS422 / RS485).

Дополнительную информацию Вы можете найти в главе В данного руководства и в руководстве "CP 341 или CP 441-2: Point-to-Point Communication" ("PtP-соединение с CP 341 или CP 441-2").

3 Режим работы Data Link

Общая информация

Поставляемое ПО Data Link обеспечивает преобразование адресов для обеспечения доступа к данным в памяти SIMATIC S7 CPU при использовании протокола MODBUS.

3.1 Компоненты ПО Data Link для совместной работы SIMATIC с ведомым MODBUS-устройством

ПО MODBUS Slave Data Link

Программные средства "Modbus slave data link" для совместной работы системы SIMATIC / ведомое MODBUS-устройство имеют в своем составе две части:

1. Загружаемый драйвер (Loadable Driver) для CP.
2. MODBUS-коммуникационный функциональный блок (MODBUS Communications Function Block) для SIMATIC S7 CPU.

MODBUS-коммуникационный функциональный блок (MODBUS Communications Function Block)

Кроме загружаемого драйвера ведомого (slave) MODBUS-устройства для работы ПО MODBUS Slave Data Link также необходим в S7 CPU специальный MODBUS-коммуникационный FB (**Communications FB**).

Этот функциональный блок может быть найден на прилагаемой дискете для Modbus в **STEP 7 - библиотеке Modbus**. Эта библиотека содержит MODBUS-коммуникационный функциональный блок FB180.

MODBUS-коммуникационный функциональный блок FB80, требуемый для CP 341, может быть также найден в **STEP 7 - библиотеке Modbus**.

Вызов FB показан в примере OB в **файле STEP7-проекта Examples\Modsl**. Вызов для модуля CP 441-2 в станции Station Modbus Slave 400, вызов для модуля CP 341 в станции Station Modbus Slave 300.

MODBUS-коммуникационные FB обрабатывают все функции, необходимые для связи данных (data link).

Поставляемые MODBUS-коммуникационные блоки FB180/80 для работы с ведомыми (slave) устройствами должны вызываться в циклически обрабатываемой пользовательской программе. MODBUS-коммуникационные блоки FB используют экземплярный блок данных как рабочую (work) область.

Примечание

В случае любых изменений предоставленного функционального блока делает гарантию его работоспособности недействительной. За возможные последствия фирма-поставщик не несет ответственности.

MODBUS-драйвер для ведомого устройства (MODBUS Slave Driver)

Загружаемый драйвер использует MODBUS-протокол и преобразует MODBUS-адреса в адреса областей памяти SIMATIC.

Загружаемый драйвер загружается в систему SIMATIC с использованием утилиты параметризации CP "CP: Point-to-Point Communication, Parameter Assignment" (CP: PtP-коммуникации, Назначение параметров), после чего драйвер автоматически передается в модуль CP.

Параметры

Параметры и рабочие режимы, перечисленные ниже, должны быть заданы для загружаемого драйвера с использованием утилиты назначения параметров.

- Скорость передачи, четность
- Адрес ведомого устройства CP
- Рабочий режим: "нормальный" или "режим с подавлением помех"
- Множитель для времени ожидания символа (character delay time)
- Адресные области для функциональных кодов FC01, 05, 15
- Адресные области для функциональных кодов FC02
- Адресные области для функциональных кодов FC03, 06, 16
- Адресные области для функциональных кодов FC04
- Границы области для доступа в режиме записи

3.2 Распределение задач

Функциональные коды 01, 02, 03, 04, 06 и 16 обрабатываются непосредственно CP.

Для функциональных кодов 05 и 15 коммуникационные FB побитный (бит за битом) ввод данных в области памяти SIMATIC.

3.3 Используемые MODBUS функциональные коды

Используемые функциональные коды

Следующие MODBUS функциональные коды поддерживаются драйвером:

Функциональный код	Функция в соответствии со спецификацией MODBUS	Общее описание
01	Считывание состояния выхода (Read Output Status)	Считывание битов
02	Считывание состояния входа (Read Input Status)	Считывание битов
03	Считывание выходных регистров (Read Output Registers)	Считывание регистров (слова)
04	Считывание входных регистров (Read Input Registers)	Считывание регистров (слова)
05	Назначение для одного выхода (катушки) (Force Single Coil)	Запись в один бит
06	Предустановки для одного регистра (Preset Single Register)	Запись в один регистр (слово)
08	Тестирование обратной связи (Loop Back Test)	Проверка линии
15	Назначения для нескольких катушек (Force Multiple Coils)	Запись в несколько битов
16	Предустановки для нескольких регистров (Preset Multiple Registers)	Запись в несколько регистров

3.4 Области данных в SIMATIC CPU

Области данных

Доступ для отдельных функциональных кодов в областях памяти SIMATIC

Функциональный код	Тип данных MODBUS	Тип данных SIMATIC	Тип доступа
01	Coil status (Состояние выходов (катушек))	Меркеры	Побитное чтение
		Выходы	Побитное чтение
		Таймеры	Побитное чтение*
		Счетчики	Побитное чтение*
02	Input status (Состояние входов)	Меркеры	Побитное чтение
		Входы	Побитное чтение
03	Holding registers (Выходные регистры)	Блок данных DB	Считывание слово за словом
04	Input registers (Входные регистры)	Блок данных DB	Считывание слово за словом
05	Single coil (Одиночный выход)	Меркеры	Побитная запись
		Выходы	Побитная запись
06	Single (holdin) register (Один вых. регистр)	Блок данных DB	Запись слово за словом
08	-	-	Прием "эхо"- сигнала
15	Multiple coils (Несколько выходов)	Меркеры	Побитная запись
		Выходы	Побитная запись
16	Multiple (hold) registers (Неск. вых. регистров)	Блок данных DB	Запись слово за словом

* с 16-битным интервалом

Преобразование адресов

MODBUS-адрес в сообщениях интерпретируется драйвером "S7-способом" и преобразуется в адрес в соответствующей области памяти SIMATIC.

Доступ к отдельным областям памяти SIMATIC может быть задан пользователем с помощью утилиты параметризации CP - "CP: Point-to-Point Communication, Parameter Assignment" (CP: PtP-коммуникации, Назначение параметров).

Загружаемый драйвер для PtP-соединений CP
(S7 - ведомое устройство)
A5E00218418-04

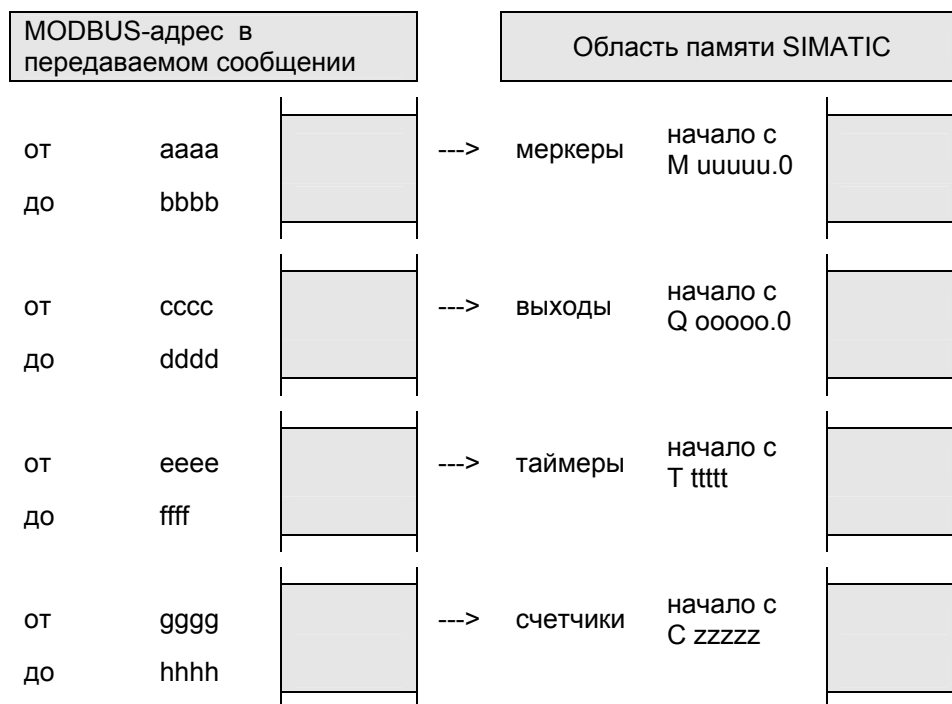
3.5 Доступ с функциональными кодами для битов

Функциональные коды 01, 05, 15

Функциональные коды 01, 05, 15 для битовых операций обеспечивают доступ в режиме чтения (read) и в режиме записи (write) в областях памяти SIMATIC, например, в областях **меркеров, выходов, таймеров, счетчиков**; таймеры и счетчики доступны только в режиме чтения (read-only) для FC01.

Вы можете использовать утилиту назначения параметров для определения границ ("от"/"до") доступа Modbus-адресов в областях меркеров, выходов, таймеров, счетчиков.

Более того, возможно назначить параметры для определения элемента данных, с которого будет начинаться доступная область памяти SIMATIC.



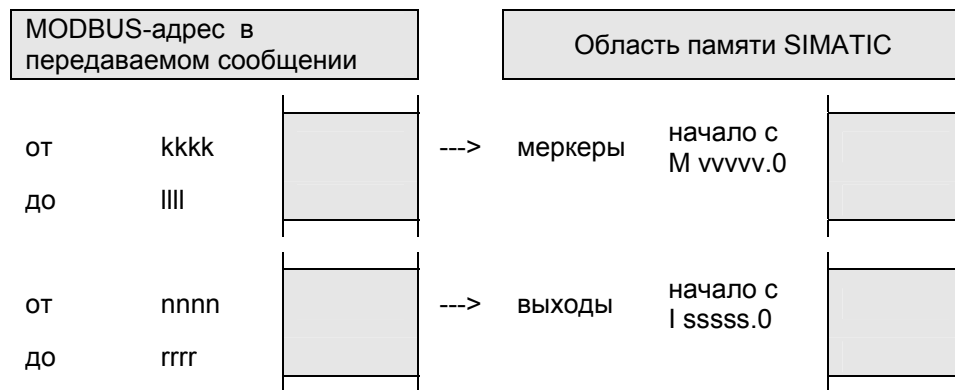
Функциональный код 02

Функциональный код 02 для битовых операций обеспечивает доступ только в режиме чтения (read-only) в областях памяти SIMATIC - в областях **меркеров и входов**.

Вы можете использовать утилиту назначения параметров для определения границ ("от"/"до") доступа Modbus-адресов в областях меркеров и входов.

Более того, возможно назначить параметры для определения элемента данных, с которого будет начинаться доступная область памяти SIMATIC.

Адресные области MODBUS и адресные области SIMATIC для FC 02 могут выбираться независимо от областей для кодов FC 01, 05 и 15.



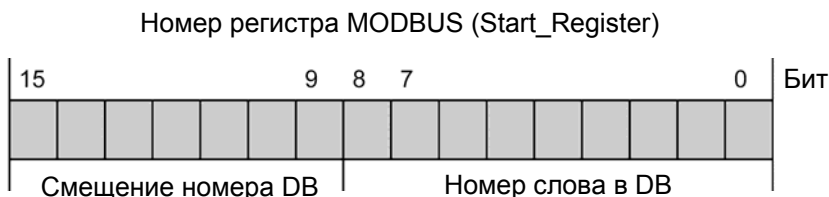
3.6 Доступ с функциональными кодами для регистров

Функциональные коды FC 03, 06, 16

Функциональные коды 03, 06 и 16 для регистровых операций обеспечивают доступ в режиме чтения (read) и в режиме записи (write) в областях **блоков данных** SIMATIC.

Вычисление нужного номера блока данных производится в два этапа:

- 1) Вы должны использовать утилиту для назначения параметров для определения базового номера (base DB number). Этот базовый блок DB - **первый блок DB**, к которому имеется доступ.
- 2) Номер начального регистра (MODBUS **Start_Register** address = Register Number) передаваемый в сообщении интерпретируется следующим образом:



"Итоговый" номер блока DB

"Итоговый" номер блока данных DB, к которому организуется доступ, рассчитывается следующим образом:

Итоговый номер блока = Базовый номер DB + Смещение номера DB.

Это значит, что возможно адресовать область DB, состоящую из 128 последовательных блоков данных внутри всей адресуемой области DB (65535 DB).

Номер слова в DB

С помощью указания слова возможно адресовать область от DBW 0 до DBW 1022 внутри каждого блока данных.

Блоки DB, которые обычно имеют байтовую организацию, в этом случае интерпретируются драйвером слово за словом, как показано ниже:

Номер слова	0	--- >	DBW 0	(= DBB 0/1)
	1		2	(2/3)
	2		4	(4/5)
	3		6	(6/7)
	:		:	(: / :)
	511		DBW 1022	(1022/1023)

Функциональный код 04

Функциональный код 04 для регистровых операций обеспечивает доступ только в режиме чтения (read-only) в областях памяти SIMATIC - в областях **блоков данных**.

Режим и использование этой функции соответствует описанию для функциональных кодов 03, 06 и 16.

Функциональный код 04 имеет свой собственный базовый номер DB, который можно свободно назначать с помощью утилиты назначения параметров. Это позволяет выбрать вторую **независимую** область из 128 блоков DB.

К блокам из этой области обеспечивается доступ только в режиме чтения (read-only).

3.7 Блокировка / ограничение функций записи

Функциональные коды FC 05, 06, 15, 16

При использовании функциональных кодов для записи 05, 06, 15 и 16 возможно заблокировать (disable) или ограничить доступ к соответствующим областям памяти SIMATIC.

Вы можете использовать утилиту для назначения параметров для определения границ ("от"/"до") области доступных для записи адресов из системы ведущего MODBUS-устройства (master system).

Если ведущее устройство обращается к области SIMATIC за пределами допусков для записи, то такое обращение блокируется и выдается сообщение об ошибке.

Допустимые области для записи

Область памяти SIMATIC				
Блоки данных ("итоговый" номер блока)	MIN	-	DB xxx	
	MAX	-	DB xxx	
Меркеры M	MIN	-	M (байт)	
	MAX	-	M (байт)	
Выходы Q	MIN	-	Q (байт)	
	MAX	-	Q (байт)	

4 Настройка драйвера

Общая информация

Весь материал в последующих разделах данного руководства, посвященный STEP 7, относится к ПО STEP 7 версии 4.02.

Последовательности операций, имена и имена каталогов (директорий) могут быть изменены в последующих версиях ПО.

4.1 Инсталляция драйвера в STEP 7 в программаторе/ПК

Для инсталляции драйвера, состоящего из программного модуля и определяемых им файлов, выполните следующие действия:

1. Вставьте компакт-диск Modbus Master CD в привод CD-ROM.
2. В панели управления Windows откройте диалоговое окно для установки программного обеспечения двойным щелчком на значке "Add / Remove Software" ("Установка / удаление программ").
3. В диалоговом окне выберите привод CD-ROM и запустите на выполнение программу инсталляции Setup.EXE.
4. Последовательно шаг за шагом выполняйте инструкции по установке, которые Вы увидите на экране.

Результат:

Драйвер и интерфейс назначения параметров (parameter assignment) устанавливаются в следующий раздел: Step7\S7ftp\S7Driver.

Раздел содержит следующие файлы:

- S7wfpb1a.dll
- S7wfpb1x.cod
- S7wfpb2x.cod

4.2 Деинсталляция драйвера

Драйвер может быть удален из STEP 7 под управлением Windows посредством утилиты из панели управления для установки и удаления программ ("Add / Remove Software").

Пользователь может проконтролировать, все ли файлы S7wfpb1?.*, S7wfpb2?.*, S7wfpb3?.* были успешно удалены из раздела Step7\S7ftp\S7Driver.

Примечание

До деинсталляции программного пакета *Point-to-Point Communication, Parameter Assignment (CP 341: PtP-соединение, назначение параметров)* Вы сначала должны удалить (деинсталлировать) все загружаемые драйверы.

4.3 Конфигурирование данных коммуникаций Data Link

Введение

Конфигурирование данных коммуникаций (Data Link) заключается в размещении оборудования в таблице конфигурации (configuration table) с использованием утилиты конфигурирования HW config. Конфигурация может выполняться с помощью средств STEP 7.

4.3.1 Конфигурирование данных коммуникаций для CP 341

Проект в S7

Перед выполнением конфигурирования Вы должны создать S7-проект, используя средства ПО STEP 7.

Компоненты проекта

Вставьте необходимые компоненты в открытый проект, используя SIMATIC Manager: станция S7-300.

Перед каждой вставкой Вы должны выбрать требуемый проект:

Insert (Вставка) -> Station (Станция) -> SIMATIC 300 Station

Загружаемый драйвер для PtP-соединений CP
(S7 - ведомое устройство)
A5E00218418-04

Конфигурация оборудования

Конфигурация оборудования заключается собственно в определении компонентов оборудования, а также в определении свойств этих компонентов.

Для запуска конфигурирования оборудования выберите станцию "SIMATIC 300 Station" и дважды щелкните на пункте "Hardware" ("Оборудование") или выберите опции меню:
Edit (Правка) -> Open Object (Открыть объект).

Используйте опции меню:

Insert (Вставка) -> Hardware Components (Компоненты оборудования) для вставки объекта стойки станции "RACK 300", модуля источника питания "PS-300", процессора "CPU-300" из "SIMATIC 300" и модуля "CP PtP" из "CP-300" с подходящим номером компонента.

Подробное описание конфигурирования модулей S7-300 Вы можете найти в руководстве пользователя по STEP 7.

4.3.2 Конфигурирование данных коммуникаций для CP 441-2

Для обеспечения данных коммуникаций для PtP-соединения Вы должны сконфигурировать станцию S7-400, связанную с ней партнерскую станцию, PtP-узлы и PtP-сеть.

Проект в S7

Перед выполнением конфигурирования Вы должны создать S7-проект средствами ПО STEP 7.

Компоненты проекта

Вставьте необходимые компоненты в открытый проект, используя SIMATIC Manager: станцию S7-400, партнерскую станцию и PtP-сеть.

Перед каждой вставкой Вы должны щелчком выбрать требуемый проект:

Insert (Вставка) -> Station (Станция) -> SIMATIC 400 Station

(для Вашей собственной S7-программы (Rack, PS, CPU, CP 441-2, ...),

Insert (Вставка) -> Station (Станция) -> Other Station (Другая станция)

(Для партнерской станции)

Insert (Вставка) -> Subnet (Подсеть) -> PtP

(Для PtP-подсети между станцией S7-400 и партнерской станцией)

Конфигурация оборудования

Конфигурация оборудования заключается собственно в определении компонентов оборудования, а также в определении свойств этих компонентов.

Для запуска конфигурирования оборудования выберите станцию "SIMATIC 400 Station" и дважды щелкните на пункте "Hardware" ("Оборудование") или выберите опции меню:

Edit (Правка) -> Open Object (Открыть объект).

Используйте опции меню:

Insert (Вставка) -> Hardware Components (Компоненты оборудования) для вставки объекта стойки станции "RACK 400", модуля источника питания "PS-400", процессора "CPU-400" из "SIMATIC 400" и модуля "CP PtP" из "CP-400" с подходящим номером компонента.

Подробное описание конфигурирования модулей S7-400 Вы можете найти в руководстве пользователя по STEP 7.

4.4 Назначение параметров для CP

Назначение параметров для модуля CP

После того, как Вы расположили модули в Вашей стойке, используя утилиту конфигурирования HW Config, Вы должны назначить для них параметры.

Для запуска утилиты назначения параметров дважды щелкните на модуле CP в "Hardware Configuration" ("Конфигурация оборудования") или щелкните на модуле CP и выберите опции меню:

Edit (Правка) -> Object Properties (Свойства объекта).

4.4.1 Назначение параметров для CP 341

- 1) Окно "Properties - CP" ("Свойства CP") -> вкладка "Basic Parameters" ("Основные параметры")

Щелкнув один раз на кнопке "Parameters" ("Параметры"), откройте интерфейс для выбора протокола "Parameter Assignment of Point-to-Point Connection" ("Назначение параметров для PtP-соединения"). Здесь Вы можете выбрать требуемый протокол обмена данными.

После выбора протокола Вы можете выполнить назначение параметров драйвера ("Parameter Assignment of the Driver") (запуск осуществляется двойным щелчком на символе "почтовый ящик").

Подробное описание процесса выбора протокола и назначения параметров для загружаемых драйверов Вы можете найти в разделе "Назначение параметров для загружаемого драйвера".

После назначения параметров Вы возвращаетесь в диалоговое окно "Properties - CP" ("Свойства CP").

2) Окно "Properties - CP" ("Свойства CP") -> вкладка "Addresses" ("Адреса")

Никаких установок не требуется выполнять на вкладке "Addresses" ("Адреса") диалогового окна "Properties - CP" ("Свойства CP").

3) Окно "Properties - CP" ("Свойства CP") -> вкладка "General" ("Общие свойства")

Никаких установок не требуется выполнять на вкладке "General" ("Общие свойства") диалогового окна "Properties - CP" ("Свойства CP").

Вы можете завершить параметризацию модуля CP щелчком на кнопке "OK" в диалоговом окне "Properties - CP" ("Свойства CP"). После этого Вы возвращаетесь в диалоговое окно "Hardware Configuration" ("Конфигурация оборудования").

Сохраните назначенные параметры и закройте диалоговое окно "Hardware Configuration" ("Конфигурация оборудования"). После этого Вы возвращаетесь в основное меню проекта STEP 7.

4.4.2 Назначение параметров для CP 441-2

1) Окно "Properties - CP 441-2" ("Свойства CP 441-2") -> вкладка "Basic Parameters" ("Основные параметры")

Определите требуемый интерфейс ("Interface") для модуля CP 441-2 (1= верхний (upper), 2= нижний (lower)) на вкладке "Basic Parameters" ("Основные параметры"). Выберите установленный интерфейсный подмодуль как "Module" ("Модуль").

Щелкнув один раз на кнопке "Parameters" ("Параметры"), откройте интерфейс для выбора протокола "Parameter Assignment of Point-to-Point Connection" ("Назначение параметров для PtP-соединения"). Здесь Вы можете выбрать требуемый протокол обмена данными.

После выбора протокола Вы можете выполнить назначение параметров драйвера ("Parameter Assignment of the Driver") (запуск осуществляется двойным щелчком на символе "почтовый ящик").

Подробное описание процесса выбора протокола и назначения параметров для загружаемых драйверов Вы можете найти в разделе "Назначение параметров для загружаемого драйвера".

После назначения параметров Вы возвращаетесь в диалоговое окно "Properties - CP 441-2" ("Свойства CP 441-2").

2) Окно "Properties - CP 441-2" ("Свойства CP 441-2") -> вкладка "Addresses" ("Адреса")

Никаких установок не требуется выполнять на вкладке "Addresses" ("Адреса") диалогового окна "Properties - CP 441-2" ("Свойства CP").

3) Окно "Properties - CP 441-2" ("Свойства CP 441-2") -> вкладка "General" ("Общие свойства")

На вкладке "General" ("Общие свойства") диалогового окна "Properties - CP 441-2" ("Свойства CP 441-2") Вы определяете PtP-сеть, к которой подключаются интерфейсы модуля CP.

PtP(1) соответствует верхнему интерфейсу, а PtP(2) - соответствует нижнему интерфейсу модуля CP.

Щелчок на PtP(1) или PtP(2) открывает диалоговое окно для конфигурирования соответствующей подсети проекта.

Выберите требуемую подсеть (subnet) и активируйте элемент управления checkbox "Partner is connected to the selected network" ("Партнер подсоединен к выбранной подсети").

Щелкните на кнопке "OK" для возврата в диалоговое окно "Properties - CP 441-2" ("Свойства CP 441-2"). Щелчком на кнопке "OK" в этом диалоговом окне Вы можете завершить параметризацию модуля CP 441-2, после чего Вы возвращаетесь в диалоговое окно "Hardware Configuration" ("Конфигурация оборудования").

Сохраните назначенные параметры и закройте диалоговое окно "Hardware Configuration" ("Конфигурация оборудования"). После этого Вы возвращаетесь в основное меню проекта STEP 7.

Назначение параметров для коммуникационного партнера

После вставки в S7-проект партнерской станции как описано выше, Вы должны определить свойства для этой станции.

В открытом проекте Вы можете щелчком выбрать станцию-коммуникационного партнера (other station).

Выберите опции меню:

Edit (Правка) -> Object Properties (Свойства объекта).

При этом откроется диалог "Properties - Other Station" ("Свойства - Другая станция")

1) Окно "Properties - Other Station" ("Свойства - Другая станция") -> вкладка "Node List" ("Список узлов")

Щелкните на кнопке "New" ("Новый") на вкладке "Node List" ("Список узлов"). В окне "Select Type" ("Выбор типа") выберите "PtP Nodes" ("PtP-узлы") и щелкните затем на кнопке "OK".

После этого появится диалоговое окно "Network Connection" ("Подключение к сети").

Выберите требуемую подсеть, которая соединяет интерфейс модуля CP с интерфейсом коммуникационного партнера, и активируйте элемент управления checkbox "Node is connected to the selected network" ("Узел подключен к выбранной подсети").

После этого щелкните затем на кнопке "OK" для возвращения на вкладку "Node List" ("Список узлов").

2) Окно "Properties - Other Station" ("Свойства - Другая станция") -> вкладка "General" ("Общие свойства")

Никаких установок не требуется выполнять на вкладке "General" ("Общие свойства").

Вы можете завершить параметризацию станции партнера щелчком на кнопке "OK". После этого Вы возвращаетесь в основное меню проекта STEP 7.

Партнерская станция может иметь несколько интерфейсов (другими словами, несколько PtP-узлов) и может быть посредством этих интерфейсов подключена к различным PtP-подсетям.

4.5 Конфигурирование данных коммуникаций проекта

Данный раздел касается только модуля CP 441-2. Если Вы используете модуль CP 341, то Вы можете пропустить данный раздел.

Коммуникационные связи

В коммуникационном соединении S7-CPU с коммуникационным партнером / в подключении шины посредством PtP-соединения коммуникационный процессор CP играет роль элемента, обеспечивающего связь. Для каждого последовательного интерфейса, подключаемого к коммуникационному партнеру / к шине, Вы должны выполнить конфигурирование данных коммуникаций.

Конфигурирование данных коммуникаций проекта

Выберите CPU в проекте STEP 7 в открытой станции S7-400 и двойным щелчком на "Connection" ("Соединение") откройте диалог "Project configuration of data link" ("Конфигурирование данных коммуникаций проекта"). При этом появится диалоговое окно "Carry out Project Configuration of Connections" ("Выполнение конфигурирования соединений проекта").

Выберите опцию меню:

Insert (Вставка) -> Connection (Соединение) ,

чтобы открыть окно диалога "New Connection" ("Новое соединение"). Здесь Вы можете выбрать коммуникационного партнера ("other station" - другую станцию) для нового соединения и выбрать в качестве типа соединения ("connection type") - тип "S7 Point-to-Point Connection" (S7 PtP-соединение).

Подтвердите свой выбор щелчком на кнопке "OK".

После этого откроется диалоговое окно "Connection Properties" ("Свойства соединения").

Диалоговое окно "Connection Properties" ("Свойства соединения")

- Вы получаете идентификатор ID соединения, который при необходимости Вы можете изменить.
- Выберите "Communication Direction 3: Local <-> Partner" (Направление соединения 3: локальный <-> партнер)
- Отображается rooting для параметризованного соединения.
- Оба обозначения номера CPU некорректны для обработки данным драйвером.

Подтвердите сделанные установки щелчком на кнопке "OK".

Сохраните "Project configuration of data link" ("Конфигурирование данных коммуникаций проекта") и закройте диалоговое окно.

Примечание

Вы должны помнить, что идентификатор соединения (Local ID) должен также использоваться при вызове функций SFB в пользовательской программе.

4.6 Назначение параметров для загружаемого драйвера

Открытие утилиты для назначения параметров CP-PtP (Parameter Assignment Tool CP-PtP)

Для запуска конфигурирования оборудования выберите станцию "SIMATIC Station" и дважды щелкните на пункте "Hardware" ("Оборудование") или выберите опции меню:
Edit (Правка) -> Open Object (Открыть объект).

Щелкните на CP и выберите опции меню:
Edit (Правка) -> Object Properties (Свойства объекта).

После того, как Вы выбрали интерфейс (только для CP 441-2) и интерфейсный submodule (только для CP 441-2), щелкните на кнопке "Parameters" ("Параметры"); при этом откроется интерфейс для выбора протокола "Parameter Assignment of Point-to-Point Connection" ("Назначение параметров для PtP-соединения").

Выбор протокола

Кроме стандартных протоколов в окне со списком также отображаются все установленные загружаемые драйверы. Выберите для загружаемого драйвера опцию "MODBUS Master" ("Ведущее MODBUS-устройство"). После этого дважды щелкните кнопкой манипулятора "мышь" на символе протокола передачи (значок "почтовый ящик"); при этом откроется диалоговое окно, в котором устанавливаются параметры, которые зависят от драйвера.

Параметры, зависящие от драйвера

Параметры, описываемые далее, могут быть заданы для данного драйвера в отдельных диалоговых окнах.

4.6.1 Протокол для ведомого устройства Modbus

Обзор параметров для обмена данными

Скорость, Фрейм символа			
Параметр	Описание	Диапазон значений	Значение по умолчанию
Скорость передачи (Transmission Rate)	Скорость передачи данных, измеряемая количеством битов в секунду	300	9600
		600	
		1200	
		2400	
		4800	
		9600	
		19200	
		38400	
		76800	
Дополнительная скорость передачи для CP 341 (№ 6ES7 341-1xH01-0AE0)		57600	
Дополнительная скорость передачи для CP 441-2 (№ 6ES7 441-2AA03-0AE0)		57600 115200	
Биты данных (Data Bits)	Битов на символ	8	8
Стоповые биты (Stop Bits)	Количество стоповых битов	1 2	1
Четность (Parity)	Бит четности не передается	Нет	Четный
	Количество битов данных заканчивается нечетным	Нечетный	
	Количество битов данных заканчивается четным	Четный	

Скорость передачи (Transmission Rate)

Скорость передачи данных, измеряемая количеством битов в секунду

Обратите внимание на максимальную общую скорость передачи модуля CP 441-2. Общая скорость передачи модуля равна сумме заданных при параметризации скоростей передачи для двух интерфейсов.

Максимальная скорость передачи для TTY-интерфейса равна 19200 бит/с.

Биты данных (Data Bits)

Количество битов данных в передаваемом символе

Стоповые биты (Stop Bits)

Количество стоповых битов определяет наименьший возможный временной интервал между двумя передаваемыми символами.

Четность (Parity)

Бит четности служит для контроля целостности данных; в зависимости от заданных параметров, он завершает количество передаваемых битов четным или нечетным номером.

Если выбрано значение “no” (“нет”), то это означает, что бит четности не передается, при этом уменьшается надежность сохранения целостности передаваемых данных.

Обзор параметров протокола обмена данными

Параметры протокола			
Параметр	Описание	Диапазон значений	Значение по умолчанию
Адрес ведомого устройства (Slave Address)	Собственный адрес модуля CP	1 ... 255	222
Рабочий режим (Operating Mode)	Нормальный режим (“Normal Operation”) Подавление помех (“Interference Suppression”)	Нормальный режим (“Normal Operation”) Подавление помех (“Interference Suppression”)	Нормальный режим (Normal)
Множитель для времени ожидания символа (Multiplier Character Delay Time)	Множитель для заданного времени ожидания символа (character delay time), зависящего от скорости передачи данных	1 ... 10	1

Адрес ведомого устройства (Slave Address)

В этом параметре задается MODBUS-адрес ведомого (slave) устройства, на который должен отвечать модуль CP. Модуль CP отвечает только на те сообщения, в которых принятый адрес ведомого (slave) устройства совпадает с заданным при параметризации адресом ведомого устройства. Сообщения с другими адресами ведомых устройств не проверяются и остаются без ответа.

Нормальный режим ("Normal Operation")

В данном рабочем режиме все распознанные ошибки при передаче данных и/или отказы, вызванные обрывом связи (BREAK) до и после принятых сообщений от ведомого устройства, приводят к соответствующим сообщениям об ошибках (error message).

Режим с подавлением помех ("Interference Suppression")

В данном рабочем режиме в случае распознавания отказа, вызванного обрывом линии связи ("BREAK"), при входящем сообщении или, если интерфейсный блок CP обнаружил ошибки при передаче, то драйвер рассматривает принятое сообщение ошибочным и игнорирует его.

Начало входящего сообщения от ведомого (slave) устройства распознается посредством корректно принятого адреса ведомого устройства (slave address).

Распознанные ошибки при передаче данных и/или отказы, вызванные обрывом связи (BREAK), также игнорируются, если они происходят после окончания приема сообщения (CRC-код).

Множитель для времени ожидания символа (Multiplier Character Delay Time)

Если коммуникационный партнер не может обеспечить временных параметров, соответствующих требованиям спецификаций MODBUS, то возможно использование множителя f_{MUL} для контролируемых периодов времени задержки символа (character delay time) ZVZ.

Параметр "время ожидания (задержки) символа" (character delay time) должен корректироваться, только если коммуникационный партнер не может обеспечить требуемых временных параметров.

В результате "время ожидания (задержки) символа" (character delay time) t_{ZVZ} рассчитывается следующим образом:

$$t_{ZVZ} = t_{ZVZ_TAB} \times f_{MUL};$$

t_{ZVZ_TAB} : табличное значение для ZVZ (см. главу "Протокол обмена данными")

f_{MUL} : множитель (коэффициент).

4.6.2 Преобразование MODBUS-адреса для битовых функций

Общий обзор функциональных кодов 01, 05, 15

Преобразование MODBUS-адреса для кодов 01, 05, 15			
Параметр		Вход	Значение
Область меркеров (SIMATIC)			
<i>MODBUS-адрес</i> в передаваемом сообщении (номер бита)	от	0 .. 65535 (десятичн.)	Начало для MODBUS-адреса
	до	0 .. 65535 (десятичн.)	Включительно для MODBUS-адреса
Память SIMATIC <i>область меркеров</i> (номер байта меркеров)	начало	0 .. 65535 (десятичн.)	Начало с данного байта в памяти (меркеров)
Область выходов (SIMATIC)			
<i>MODBUS-адрес</i> в передаваемом сообщении (номер бита)	от	0 .. 65535 (десятичн.)	Начало для MODBUS-адреса
	до	0 .. 65535 (десятичн.)	Включительно для MODBUS-адреса
Память SIMATIC <i>область выходов</i> (номер байта выходов)	начало	0 .. 65535 (десятичн.)	Начало с данного байта выходов
Область таймеров (SIMATIC)			
<i>MODBUS-адрес</i> в передаваемом сообщении (номер бита)	от	0 .. 65535 (десятичн.)	Начало для MODBUS-адреса
	до	0 .. 65535 (десятичн.)	Включительно для MODBUS-адреса
Память SIMATIC <i>область таймеров</i> (номер таймера)	начало	0 .. 65535 (десятичн.)	Начало с данного таймера (равен 16-битовому слову)
Область счетчиков (SIMATIC)			
<i>MODBUS-адрес</i> в передаваемом сообщении (номер бита)	от	0 .. 65535 (десятичн.)	Начало для MODBUS-адреса
	до	0 .. 65535 (десятичн.)	Включительно для MODBUS-адреса
Память SIMATIC <i>область счетчиков</i> (номер счетчика)	начало	0 .. 65535 (десятичн.)	Начало с данного счетчика (равен 16-битовому слову)

Modbus-адрес - "от" / "до"

Для задания Modbus-адреса Вы можете использовать значение, начиная "от" адреса, совпадающего с адресом начала соответствующей области; например, области меркеров, выходов и т.д. (= номер первого бита области).

Для задания Modbus-адреса Вы можете использовать значение, вплоть "до" адреса, совпадающего с адресом конца соответствующей области; например, области меркеров, выходов и т.д. (= номер последнего бита области).

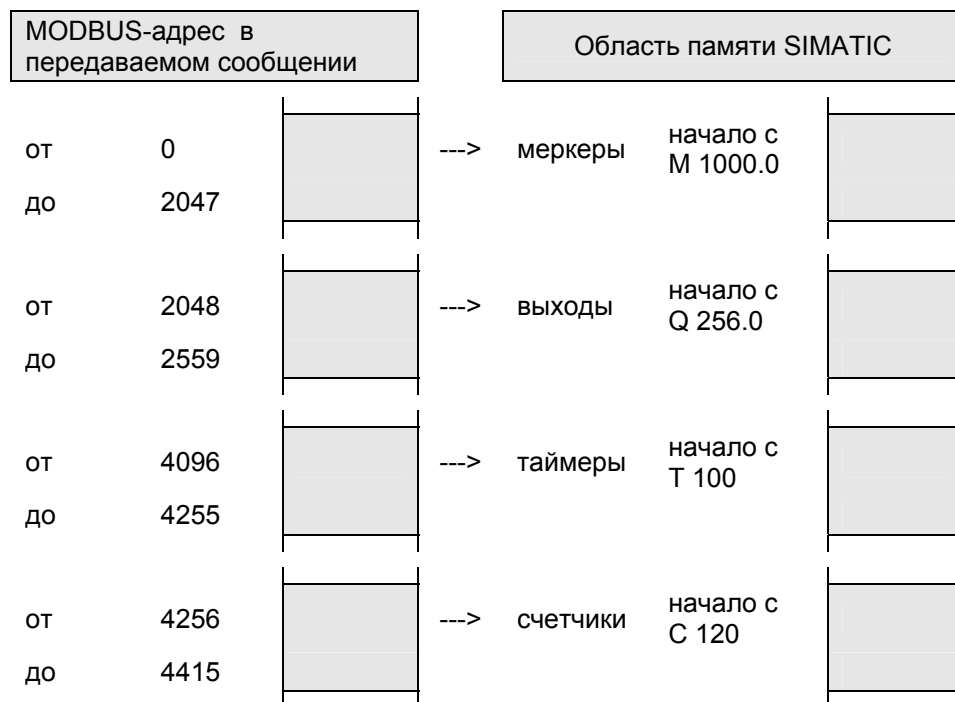
Границы "от" / "до" указывают область Modbus-адресов в передаваемом сообщении (номера битов начинаются с 0) для функциональных кодов FC 01, 05 и 15.

Отдельные области с границами "от" / "до" не должны перекрываться. Между отдельными областями с границами "от" / "до" допускается наличие зазоров.

"Начало" SIMATIC-области памяти

Вы можете использовать "начало" входа для определения начала SIMATIC-области, в которой размещается Modbus-область с границами "от" / "до" (= номеру первого байта меркеров, номеру выходного байта / номеру таймера / номеру счетчика SIMATIC-области).

Пример



Modbus-адреса от 0 до 2047 располагаются в SIMATIC-области меркеров (битов памяти), начиная с меркера M 1000.0; т.е. длина области составляет 2048 битов или 256 байтов, что означает, что последний меркер имеет адрес M 1255.7.

Modbus-адреса от 2048 до 2559 располагаются в SIMATIC-области выходов, начиная с выхода Q 256.0; т.е. длина области составляет 512 битов или 64 байта, что означает, что последний выход имеет адрес Q 319.7.

Modbus-адреса от 4096 до 4255 располагаются в SIMATIC-области таймеров, начиная с таймера T 100; т.е. длина области составляет 160 битов или 10 слов, что означает, что последний таймер имеет адрес T 109.

Modbus-адреса от 4256 до 4415 располагаются в SIMATIC-области счетчиков, начиная с счетчика C 120; т.е. длина области составляет 160 битов или 10 слов, что означает, что последний счетчик имеет адрес T 129.

Общий обзор функционального кода 02

Преобразование MODBUS-адреса для кода 02			
Параметр		Вход	Значение
Область меркеров (SIMATIC)			
MODBUS-адрес в передаваемом сообщении (номер бита)	от	0 .. 65535 (десятичн.)	Начало для MODBUS-адреса
	до	0 .. 65535 (десятичн.)	Включительно для MODBUS-адреса
Память SIMATIC область меркеров (номер байта меркеров)	начало	0 .. 65535 (десятичн.)	Начало с данного байта в памяти (меркеров)
Область входов (SIMATIC)			
MODBUS-адрес в передаваемом сообщении (номер бита)	от	0 .. 65535 (десятичн.)	Начало для MODBUS-адреса
	до	0 .. 65535 (десятичн.)	Включительно для MODBUS-адреса
Память SIMATIC область входов (номер байта входов)	начало	0 .. 65535 (десятичн.)	Начало с данного байта входов

Modbus-адрес - "от" / "до"

Для задания Modbus-адреса Вы можете использовать значение, начиная "от" адреса, совпадающего с адресом начала соответствующей области; например, области меркеров или входов (= с номером первого бита области).

Вы можете использовать значение, вплоть "до" адреса, совпадающего с адресом конца соответствующей области (= с номером последнего бита области).

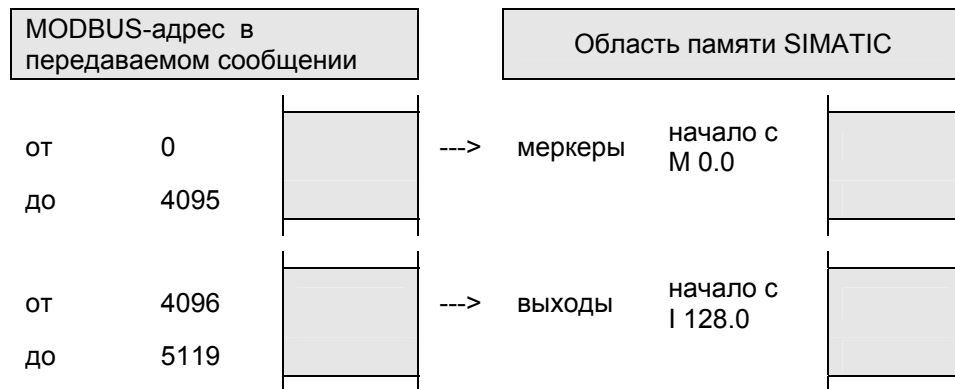
Границы "от" / "до" указывают область Modbus-адресов в передаваемом сообщении (номера битов начинаются с 0) для функционального кода 02.

Отдельные с не должны перекрываться. Между отдельными областями с границами "от" / "до" допускается наличие зазоров.

"Начало" SIMATIC-области памяти

Вы можете использовать "начало" входа для определения начала SIMATIC-области, в которой размещается Modbus-область с границами "от" / "до" (= номеру первого байта меркеров или номеру входного байта SIMATIC-области).

Пример



Modbus-адреса от 0 до 4095 располагаются в SIMATIC-области меркеров, начиная с меркера M 0.0 т.е. длина области составляет 4096 битов или 512 байтов, что означает, что последний меркер имеет адрес M 511.7.

Modbus-адреса от 4096 до 5119 располагаются в SIMATIC-области входов, начиная со входа I 128.0; т.е. длина области составляет 1024 бита или 128 байтов, что означает, что последний вход имеет адрес I 255.7.

Примечание

Вход (input) со значением "с начала области меркеров" ("commence at memory bit") для функционального кода 02 никак не зависит от входа (input) со значением "с начала области меркеров" ("commence at memory bit") для функциональных кодов 01, 05 и 15.

Это значит, что для FC 02 можно использовать вторую SIMATIC-область меркеров ("read-only" - "только для чтения"), которая полностью не зависит от первой.

Таким образом в определенных байтах меркеров (memory bytes) нет точек одновременно доступных для функциональных кодов FC01 и FC02.

4.6.3 Преобразование Modbus-адресов для функций для регистров

Общий обзор функциональных кодов FC 03, 06, 16

Преобразование MODBUS-адреса для кодов FC 03, 06, 16			
Параметр		Вход	Значение
Область блоков данных (SIMATIC)			
<i>MODBUS-адрес</i> = 0 в передаваемом сообщении (номер регистра) определяет доступ к			
Область памяти SIMATIC блоков данных (<i>Data Blocks</i>)	начало с DB	1 .. 65535 (десятичн.)	Начало с данного блока данных Начало с DBW 0 (= базовый номер DB)

Y

Начало с DB

Вы можете использовать "Начало с DB" ("commence at DB") для задания первого блока данных в SIMATIC-области, к которому нужен доступ - "базовый номер DB" ("base DB Number").

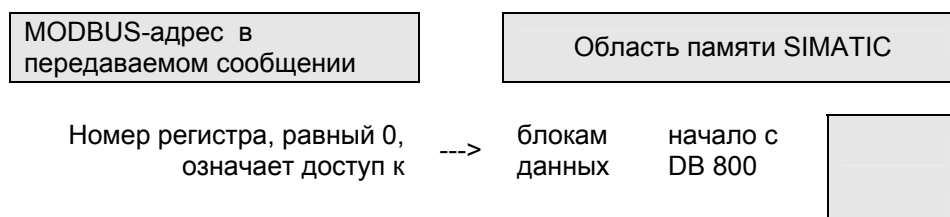
Этот блок DB доступен, когда номер регистра в Modbus-сообщении имеет значение 0, начиная со слова данных DBW 0.

Высшие номера Modbus-регистров обеспечивают доступ к последующим словам данных / блокам данных.

Таким образом могут быть адресованы до 127 последующих DB.

Драйвер интерпретирует биты 9 - 15 номера Modbus-регистра для доступа к отдельным последующим DBs (см. примечание на стр. 8-2).

Пример



Вы можете использовать Modbus-регистра с адресом 0 для доступа к блоку данных 800, начиная с DBW 0 в системе SIMATIC.

Более высокие адреса Modbus-регистров (начиная с 512) обеспечивают доступ к последующим блокам данных, например, DB 801.

Общий обзор функционального кода FC 04

Преобразование MODBUS-адреса для кода FC 04			
Параметр		Вход	Значение
Область блоков данных (SIMATIC)			
MODBUS-адрес = 0 в передаваемом сообщении (номер регистра) определяет доступ к			
Область памяти SIMATIC блоков данных (Data Blocks)	начало с DB	1 .. 65535 (десятичн.)	Начало с данного блока данных Начало с DBW 0 (= базовый номер DB)

Y

Начало с DB

Вы можете использовать "Начало с DB" ("commence at DB") для задания первого блока данных в SIMATIC-области, к которому нужен доступ - "базовый номер DB" ("base DB Number").

Этот блок DB доступен, когда номер регистра в Modbus-сообщении имеет значение 0, начиная со слова данных DBW 0.

Высшие номера Modbus-регистров обеспечивают доступ к последующим словам данных / блокам данных.

Таким образом могут быть адресованы до 127 последующих DB.

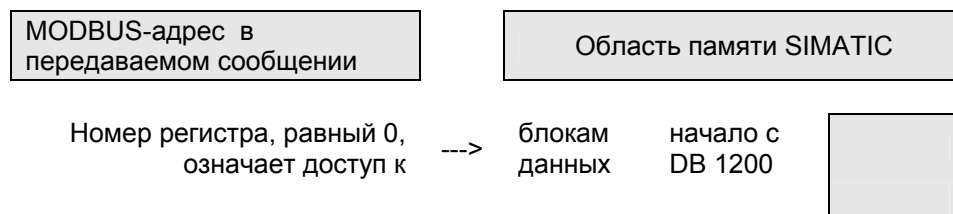
Драйвер интерпретирует биты 9 - 15 номера Modbus-регистра для доступа к отдельным последующим DBs (см. примечание на стр. 8-2).

Примечание

Поле со значением "Начало с DB" ("commence at DB") для кода 04 никак не зависит от "Начало с DB" ("commence at DB") для кодов 03, 06 и 16.

Это значит, что для FC 04 можно использовать вторую SIMATIC-область DB ("read-only" - "только для чтения"), которая не зависит от первой.

Пример



Вы можете использовать Modbus-регистра с адресом 0 для адресации блока данных 1200, начиная с DBW 0 в системе SIMATIC.

Более высокие адреса Modbus-регистров (как 512, 1024...) обеспечивают доступ к последующим блокам данных, например, DB 1201, DB 1202...

4.6.4 Ограничения для функций записи

Общий обзор функциональных кодов FC 05, 06, 15, 16

SIMATIC-границы для доступа в режиме записи FC 05, 06, 15, 16			
Параметр		Вход	Значение
Блоки данных (DB): (номер блока данных DB)	MIN	1 .. 65535	Первый доступный блок
	MAX	1 .. 65535	Последний доступный блок (если MAX=0, то недоступны все DB)
Меркеры M (номер байта меркеров)	MIN	0 .. 65535	Первый доступный байт меркеров
	MAX	1 .. 65535	Последний доступный байт меркеров (если MAX=0, то недоступны все байты меркеров)
Выходы Q (номер байта выходов)	MIN	0 .. 65535	Первый доступный байт выходов
	MAX	1 .. 65535	Последний доступный байт выходов (если MAX=0, то недоступны все байты выходов)

"MIN" / "MAX" в области памяти SIMATIC




Для функциональных кодов при адресации для записи возможно определять нижнюю и верхнюю границу (MIN / MAX). Доступ в режиме записи допускается только в диапазоне адресов в пределах граничных значений.

Если верхнее граничное значение равно 0, то это означает, что недоступной является вся область.

При выборе размера доступной области помните, какой CPU должен к ней обращаться.

Если ведущее (master) устройство пытается получить доступ для записи к области, которая лежит за пределами верхней / нижней границы, то такие попытки будут отменены CP с сообщением об ошибке.

Пример

Область памяти SIMATIC			
Блоки данных (итоговый номер DB)	MIN -	DB 600	
	MAX -	DB 699	
Меркеры M	MIN -	M 1000	
	MAX -	M 1127	
Выходы Q	MIN -	Q 256	
	MAX -	Q 319	

Блоки данных SIMATIC DB 600 ... DB 699 могут быть доступны в режиме записи с использованием функциональных кодов (FC 06, 16).

Байты меркеров SIMATIC MB 1000 ... MB 1127 могут быть доступны в режиме записи с использованием функциональных кодов (FC 05, 15).

Байты выходов SIMATIC QB 256 ... QB 319 могут быть доступны в режиме записи с использованием функциональных кодов (FC 05, 15).

4.6.5 Интерфейс X27 (RS 422/485)

Общий обзор

Интерфейс X27 (RS 422/485)			
Параметр	Описание	Диапазон значений	Значение по умолчанию
Presets of the Receiving Line (Предустановки для приемной линии)	No presets (нет предустановок) Preset "Break" (Предустановка "Обрыв") Preset "High" (Предустановка "Высокий уровень")	None R(A)5V,R(B)0V R(A)0V,R(B)5V	R(A)5V,R(B)0V
X27 Operating Mode (Рабочий режим X27)	Использование линии передачи T(A), T(B) для передачи данных, использование линии R(A), R(B) для приема данных. Переключение входной (приемной) линии R(A), R(B) при переключении между режимами передачи (send) и приема (receive).	Полный дуплексный режим (RS422), 4-хпроводный режим работы Полу-дуплексный режим (RS485), 2-хпроводный режим работы	Полный дуплексный режим (RS422), 4-хпроводный режим работы

Полный дуплексный режим (RS422), 4-хпроводный режим работы ("Full-Duplex (RS422) Four-Wire Operation")

В этом рабочем режиме для передачи данных используются линии передачи T(A), T(B), а для приема данных используются входные линии (для приема) R(A), R(B).

Обработка ошибок выполняется в соответствии с установленной функцией в параметре "Driver Operating Mode" ("Режим работы драйвера"): "нормальный режим" ("Normal") или "режим с подавлением помех" ("Interference Suppression").

Полудуплексный режим (RS485), 2-проводный режим работы ("Half-Duplex (RS485) Two-Wire Operation")

В этом рабочем режиме драйвер обеспечивает переключение 2-проводной приемной линии интерфейса R(A), R(B) при переключении между режимами передачи (send) и приема (receive).

В данном рабочем режиме все распознанные ошибки передачи и/или BREAK до и после входных сообщений игнорируются. Уровень обрыва (BREAK) в линиях в паузах между сообщениями также игнорируется.

Начало входных сообщений от ведомого (slave) устройства распознается при приеме корректного адреса ведомого устройства (slave address).

Рекомендуется использование предустановки R(A) 0V, R(B) 5V (High) (R(A) 0B, R(B) 5B (высокое)) для входных (приемных) линий интерфейса.

Предустановки для входных линий

"None" (плавающее значение)

При данном значении параметра нет предустановок для 2-проводной приемной линии интерфейса R(A),R(B). При этом текущие (мгновенные) значения сигналов в линиях определяются и обеспечиваются коммуникационным партнером.

Предустановка "R(A) 5V, R(B) 0V" (BREAK) ("Обрыв")

Такая предустановка для 2-проводной приемной линии интерфейса обеспечивается в CP следующим образом:

$R(A) \rightarrow +5B, R(B) \rightarrow 0B (V_A - V_B \geq +0.3V)$.

Это значит, что в линии возникает "уровень обрыва" (BREAK level) в CP при реальном событии - при обрыве линии.

Предустановка "R(A) 0V, R(B) 5V" (High) ("Высокое")

Такая предустановка для 2-проводной приемной линии интерфейса обеспечивается в CP следующим образом:

$R(A) \rightarrow 0B, R(B) \rightarrow +5B (V_A - V_B \leq -0.3V)$.

Это значит, что в линии возникает "высокий уровень" (HIGH level) в CP при реальном событии - при обрыве линии (и/или когда CP находится в состоянии ожидания, то есть, когда ведомое (slave) устройство не передает данных).

Состояние линии "BREAK" ("Обрыв") не может быть распознано.

Выбор параметров

Выберите требуемые значения параметров коммуникационных данных и выходите из соответствующего диалога, нажав кнопку "OK."

4.7 Загрузка данных конфигурации и параметризации для CP 341

Управление данными (Data Management)

При закрытии "Hardware Configuration" ("Конфигурация оборудования") данные автоматически сохраняются в Вашем проекте STEP 7.

Загрузка данных конфигурации и параметров

Заданные значения данных конфигурации и параметров теперь могут быть загружены в интерактивном (online) режиме из программатора в CPU. Для передачи данных в CPU используйте опции меню:
PLC -> Download (Загрузить).

Во время запуска CPU и всякий раз при переключении из режима STOP (СТОП) в RUN (ВЫПОЛНЕНИЕ) параметры модуля CP автоматически пересылаются в CP, как только модуль CP становится доступен посредством системной шины S7-300.

Программный модуль драйвера (коды) не сохраняются в CPU, однако непосредственно с помощью утилиты назначения параметров (parameter assignment tool) сохраняется в реманентной памяти модуля CP 341. Однако Вы должны иметь в виду, что это значит, что Вы не можете заменить модуль без программатора.

Дополнительная информация

Вы можете использовать руководство пользователя по программному обеспечению STEP 7 ("User Manual for STEP 7") для получения более подробной информации по следующим темам:

- Сохранение данных конфигурации и параметров.
- Загрузка (load) данных конфигурации и параметров в CPU.
- Считывание, изменение, копирование и распечатка данных конфигурации и параметров.

4.8 Загрузка данных конфигурации и параметризации для CP 441-2

Управление данными (Data Management)

При закрытии "Hardware Configuration" ("Конфигурация оборудования") и/или "Project Configuration of Connection" ("Конфигурация соединений в проекте") данные (включая данные модуля CP и программный (кодовый) модуль драйвера) автоматически сохраняются в Вашем проекте STEP 7.

Загрузка данных конфигурации и параметров

Заданные значения данных конфигурации и параметров теперь могут быть загружены в интерактивном (online) режиме из программатора в CPU. Для передачи данных в CPU используйте опции меню:
PLC -> Download (Загрузить).

Во время запуска CPU и всякий раз при переключении из режима STOP (СТОП) в RUN (ВЫПОЛНЕНИЕ) параметры модуля CP автоматически пересылаются в CP, как только модуль CP становится доступен посредством системной шины S7-400.

Дополнительная информация

Вы можете использовать руководство пользователя по программному обеспечению STEP 7 ("User Manual for STEP 7") для получения более подробной информации по следующим темам:

- Сохранение данных конфигурации и параметров.
- Загрузка (load) данных конфигурации и параметров в CPU.
- Считывание, изменение, копирование и распечатка данных конфигурации и параметров.

4.9 Характеристики запуска CP

Введение

Запуск модуля CP происходит в два этапа:

- Инициализация (включение питания CP)
- Назначение параметров

Инициализация (Initialization)

Как только подается напряжение на модуль CP и после отработки программы тестирования оборудования, микропрограмма в CP подготавливается к выполнению.

Назначение параметров (Parameter Assignment)

Во время назначения параметров модуль CP получает параметры, соответствующие слоту, в котором он установлен.

После этого коммуникационный процессор CP готов к выполнению своих функций.

4.10 Назначение параметров запуска ("Startup of the CPU" - "Запуск CPU")

Этот раздел относится только к коммуникационным процессорам CP 441-2. Если в качестве коммуникационного процессора Вы используете модуль CP 341, то Вы можете не принимать данный раздел во внимание.

Конфигурация оборудования

Чтобы не возникало проблем при запуске CPU-CP, при конфигурировании оборудования в диалоге "Hardware Configuration" ("Конфигурация оборудования") при назначении параметров CPU ("Parameter Assignment of the CPU") необходимо выполнить следующие установки.

После запуска утилиты для назначения параметров кнопкой манипулятора "мышь" дважды щелкните на CPU или один раз щелкните на CPU и затем выберите опции меню:

Edit (Правка) -> Object Properties (Свойства объекта);

после этого появится диалоговое окно "Properties - CPU" ("Свойства - CPU").

На вкладке "Startup" ("Запуск") установите минимальное значение контролируемого времени в окне ("Monitoring Time for"), равное 1000 (что означает 100 с) в пункте "Transfer of Parameters to Module (100ms):" ("Передача параметров в модуль (100 мс)").

Когда для интерфейса модуля CP 441-2 назначаются параметры путем загрузки драйвера, программный модуль (коды) драйвера передается в коммуникационный процессор CP, как и назначенные значения параметров. Процедура загрузки параметров в целом контролируется на предмет превышения заданного временного интервала, упомянутого выше; при этом значение этого параметра должно быть задано достаточно большим.

5 Отладка коммуникационного FB

5.1 Инсталляция FB

Дискета с дистрибутивом

MODBUS-коммуникационный FB (MODBUS Communications FB) - это компонент проекта STEP 7, который после инсталляции хранится в директории для примеров EXAMPLES в проекте STEP 7 под именем "Modsl" для CP 441-2 или CP 341. Он также сохраняется в библиотеке *Modbus*.

Перед инсталляцией FB необходимо, чтобы проект с указанным выше именем уже существовал.

1. Файл проекта *Modsl* полностью содержит проект STEP 7 в форме загружаемого примера.
2. Если Вы хотите продолжать работу с Вашим пользовательским проектом, то Вы должны загрузить один из MODBUS-коммуникационных блоков - **FB180** для CP 441-2 или **FB80** для CP 341 соответственно в Ваш проект.
3. Если требуется, перешлите организационные блоки OB запуска **OB100** и **OB101**, циклический блок **OB1**, а также **DB180** или **DB80** в Ваш проект.

Это позволит вызывать пример для коммуникационного FB, а также соответствующий экземплярный блок DB для FB.

Примечание

OB1 и OB100/OB101 могут быть сгенерированы самостоятельно. Если экземплярный блок DB не передан в проект, то он должен быть сгенерирован при вызове FB180 или FB80 в блоках OB1/OB100/OB101.

5.2 Проект STEP 7

Проект STEP 7

Файл проекта *Modsl* полностью содержит проект STEP 7 в форме загружаемого примера, заключающий в себе следующие компоненты:

- Проект конфигурации оборудования, включая UR1, PS, CPU и CP
- Назначенные параметры CP
- STEP 7 -программу с блоками OB и с MODBUS-коммуникационным FB.

Блоки в программном файле должны рассматриваться только как примеры и могут быть изменены пользователем в соответствии с его требованиями. При необходимости MODBUS-коммуникационный FB может быть переименован.

Содержание файла проекта Modsl для CP 341

FB 80	Блок FB	MODBUS FB для "slave"-модуля CP 341
DB 80	Экземплярный DB	Экз. DB и рабочий (work) блок для FB80
OB 1	Пример OB	Циклическая программа
OB 100	Example OB	"Холодный" (Cold) рестарт (полный)
FB 7	P_RCV_RK	Прием данных
FB 8	P_SND_RK	Передача данных
SFC 36	MSK_FLT	Маскирование синхронных ошибок
SFC 37	DMSK_FLT	Демаскирование синхронных ошибок
SFC 38	READ_ERR	Считывание регистра состояний
SFC 41	DIS_AIRT	Задержка сигналов (Delay alarms)
SFC 42	EN_AIRT	Разрешение сигналов (Enable alarms)
SFC 51	RDSYSST	Считывание системной области CPU

SFC встроены в CPU, таблицы переменных даны только для диагностики.

Содержание файла проекта Modsl для CP 441-2

FB 180	Блок FB	MODBUS FB для "slave"-модуля CP 441-2
DB 180	Экземплярный DB	Экз. DB и рабочий (work) блок для FB
OB 1	Пример OB	Циклическая программа
OB 100	Example OB	"Холодный" (Cold) рестарт (полный)
OB 101	Example OB	Перезапуск (Restart)
SFB 12	BSEND	Передача данных для DB
SFB 22	STATUS	Чтение информации об ошибках из области SYSTAT
SFC 36	MSK_FLT	Маскирование синхронных ошибок
SFC 37	DMSK_FLT	Демаскирование синхронных ошибок
SFC 38	READ_ERR	Считывание регистра состояний
SFC 41	DIS_AIRT	Задержка сигналов (Delay alarms)
SFC 42	EN_AIRT	Разрешение сигналов (Enable alarms)
SFC 51	RDSYSST	Считывание системной области CPU

SFC встроены в CPU, таблицы переменных даны только для диагностики.

5.3 Параметры FB 80 (CP 341)

Имя	Тип	Тип данных	Значение	Допустимые значения
LADDR	I	INT	Базовый (Base) адрес CP	Использование утилиты HW Config
START_TIMER	I	TIMER	Таймер для инициализации окончания времени	
START_TIME	I	S5TIMER	Задание времени для таймера	
OB_MASK	I	BOOL	Маскирование ошибок доступа к I/O, задержка сигналов	FALSE: Ошибки доступа к периферии (I/O) не маскируются TRUE: Маскируются ошибки доступа к несуществующим I/O и сигналы (alarms) задерживаются
CP_START	I	BOOL	Инициализация	
CP_START_FM	I	BOOL	Активация инициализации передним фронтом сигнала в CP_START	
CP_START_NDR	O	BOOL	Информация: задание на запись от CP	
CP_START_OK	O	BOOL	Инициализация выполнена без ошибок	TRUE: Инициализация может завершиться без ошибок даже после истечения времени мониторинга
CP_START_ERROR	O	BOOL	Инициализация выполнена с ошибкой	TRUE: Инициализация не может завершиться без ошибок даже после истечения времени мониторинга
ERROR_NR	O	WORD	Номер ошибки	Назначение, см. диагностику
ERROR_INFO	O	WORD	Дополнительная информация по ошибке	Назначение, см. диагностику

5.4 Параметры FB 180 (CP 441-2)

Имя	Тип	Тип данных	Значение	Допустимые значения
ID	I	INT	ID соединения	ID соединения из конфигурации связей данных проекта
START_TIMER	I	TIMER	Таймер для инициализации окончания времени	
START_TIME	I	S5TIMER	Задание времени для таймера	
STATUS_TIMER	I	TIMER	Чтение значения таймера для SYSTAT	
STATUS_TIME	I	S5TIMER	Чтение значения времени для SYSTAT	
OB_MASK	I	BOOL	Маскирование ошибок доступа к I/O, задержка сигналов	FALSE: Ошибки доступа к периферии (I/O) не маскируются TRUE: Маскируются ошибки доступа к несуществующим I/O и сигналы (alarms) задерживаются
CP_START	I	BOOL	Инициализация	
CP_START_FM	I	BOOL	Инициализация по переднему фронту сигнала CP_START	
CP_START_NDR	O	BOOL	Информация: задание на запись от CP	
CP_START_OK	O	BOOL	Инициализация выполнена без ошибок	TRUE: Инициализация может завершиться без ошибок и после контрольного времени
CP_START_ERROR	O	BOOL	Инициализация выполнена с ошибкой	TRUE: Инициализация не может завершиться без ошибок и после контрольного времени
ERROR_NR	O	WORD	Номер ошибки	Назначение, см. диагностику
ERROR_INFO	O	WORD	Дополнительная информация по ошибке	Назначение, см. диагностику

5.5 Вызов программы

Общая информация

Modbus коммуникационный блок FB для загружаемого драйвера ведомого (slave) Modbus-устройства должен вызываться в SIMATIC S7 CPU в циклически обрабатываемой части программы.

Коммуникационный блок FB инициализирует модуль CP и выполняет соответствующие Modbus-функции, на которые не способен сам драйвер. Коммуникационный блок FB для соответствующего ведомого (slave) Modbus-устройства должен вызываться в пользовательской программе, даже если указанные функциональные коды не используются системой ведущего (master) Modbus-устройства.

Коммуникации между CP и FB обеспечиваются функциями операционной системы CPU и функциональным блоком SFB BSEND (для CP 441-2) или блоками P_SND_RK и P_RCV_RK (для CP 341), вызываемыми из FB.

Более того, Modbus коммуникационный блок FB (только для CP 441-2 (FB180)) считывает информацию об ошибках из области SYSTAT модуля CP посредством SFB STATUS.

Запуск, инициализация

После каждого полного рестарта (complete restart) или перезапуска (restart) CPU Вы должны выполнить инициализацию Modbus коммуникационного блока FB.

Инициализация иницируется передним фронтом на входе CP_START. Сначала FB удаляет экземплярный DB, считывает области операндов I, Q, M, T, C в CPU с помощью SFC51 SZL_READ, и помещает их в экземплярный блок DB. Это позволяет Вам выяснить требования для режима записи в системе ведущего (master) Modbus-устройства к перекрытию отдельных областей.

Номер экземплярного блока DB и сигнал о прохождении инициализации посылаются в CP посредством выполнения задания SEND.

Как только задание SEND выполняется без ошибок, устанавливается выход CP_START_OK и инициализация FB завершается. Если задание SEND выполняется с ошибкой, то вход CP_START сбрасывается и устанавливается выход CP_START_ERROR.

Если инициализация выполняется с ошибкой, то связь между устройствами Modbus и SIMATIC невозможна.

При этом на все запросы от системы ведущего (master) Modbus-устройства выдается ответное сообщение с кодом ошибки (Exception Code).

Экземплярный DB

Все данные, относящиеся к Modbus FB располагаются в экземплярном блоке данных. Этот DB также является экземплярным блоком DB ("мультиэкземплярным") для используемых FB / SFB и рабочим блоком (work area) для Modbus коммуникационного блока FB. При этом других дополнительных областей данных не требуется.

Modbus FB использует только экземплярный блок DB и локальные данные.

К экземплярному блоку DB разрешен только доступ в режиме чтения.

Инициализация окончания контрольного времени (Timeout Initialization) (START_TIME)

После включения питания модулю CP необходимо несколько секунд для проверки оборудования и памяти, после чего он готов к работе. Попытки инициализации в течение этого времени Modbus FB завершаются с индикацией ошибки. Поэтому Modbus FB за этот период повторяет задание на инициализацию несколько раз.

Выход CP_START_OK устанавливается, если инициализация может быть выполнена без ошибок в течение времени параметризации START-TIME, контролируемого таймером START-TIMER. Если инициализация не может быть выполнена без ошибок после окончания контрольного времени (monitoring time), то устанавливается выход CP_START_ERROR.

Значение времени для считывания области SYSTAT (только в CP 441-2)

Циклическое считывание области SYSTAT (только для CP 441-2 (FB180)) в каждом цикле PLC или в каждом цикле другого PLC вызывает перегрузку CP и К-шины и снижает производительность при обработке данных. Поэтому для считывания области SYSTAT может быть установлен специальный временной интервал.

После истечения времени STATUS_TIME в счетчике STATUS_TIMER Modbus FB активирует SFB STATUS для считывания области SYSTAT.

Ошибки доступа к I/O, задержка сигналов

Входной параметр OB_MASK может использоваться для задания Modbus FB маскировать ошибки доступа к I/O. Тогда в случае попытки записи в несуществующие I/O CPU не переходит в режим STOP и не вызывает OB обработки ошибок.

Однако ошибки доступа распознаются FB и действие доступа заканчивается с сообщением в CP об ошибке. Ошибки доступа к I/O в режиме записи маскируются только, если параметр OB_MASK = TRUE (ИСТИНА).

Перед маскированием ошибок доступа все сообщения с высшим приоритетом задерживаются (SFC14); они вновь разблокируются после попытки FB доступа в режиме записи и после размаскирования ошибок доступа (SFC42).

Это обеспечивает распознавание ошибок доступа программами с высоким приоритетом (сообщения о времени и сообщения от процесса) в случае, когда обработка FB прерывается между командами маскирования и демаскирования.

Пример OB100/101

Сегмент 1

```

UN    M 180.0      // установка CP_START
S     M 180.0      // !
U     M 180.1      // сброс CP_START_FM
R     M 180.1      // !

```

Пример OB1 для CP 341

Сегмент 1

```

CALL FB 180 , DB80 // MODBUS SLAVE CP341 FB
LADDR :=256        // базовый адрес CP
START_TIMER :=T120 // таймер "Timeout initi."
START_TIME :=S5T#5S // значение времени "Timeout ."
OB_MASK :=TRUE     // маскирование ошибок доступа
CP_START :=M180.0  // запуск инициализации
CP_START_FM :=M180.1 // меркер запуска по фронту сигнала
CP_NDR :=M180.2    // новое задание на запись от CP
CP_START_OK :=M180.3 // инициализация без ошибок
CP_START_ERROR :=M180.4 // инициализация с ошибкой
CP_ERROR_NR :=MW182 // номер ошибки
CP_ERROR_INFO :=MW184 // дополнительная инфо по ошибке
:

```

Пример OB1 для CP 441-2

Сегмент 1

```

CALL FB 180 , DB180 // MODBUS SLAVE CP441 FB
ID :=W#16#1000      // ID соединения
START_TIMER :=T180  // таймер "Timeout initi."
START_TIME :=S5T#5S // значение времени "Timeout ."
STATUS_TIMER :=T181 // таймер "чтение SYSTAT"
STATUS_TIME :=S5T#2S // значение времени "чтение SYST"
OB_MASK :=TRUE     // маскирование ошибок доступа
CP_START :=M180.0  // запуск инициализации
CP_START_FM :=M180.1 // меркер запуска по фронту сигнала
CP_NDR :=M180.2    // новое задание на запись от CP
CP_START_OK :=M180.3 // инициализация без ошибок
CP_START_ERROR :=M180.4 // инициализация с ошибкой
CP_ERROR_NR :=MW182 // номер ошибки
CP_ERROR_INFO :=MW184 // дополнительная инфо по ошибке
:

```

5.6 Циклическая обработка программы

Коммуникационный FB

MODBUS коммуникационный FB выполняет все необходимые вызовы SFB и обрабатывает соответствующие функциональные коды, которые собственно модуль CP не может выполнить (побитная запись с FC05 или FC15 в области памяти SIMATIC для меркеров и выходов).

Область SYSTAT (только для CP 441-2)

Modbus-блок FB180 для CP 441-2 циклически считывает данные из области SYSTAT в CP и сохраняет эти данные в экземплярном блоке DB, начиная с адреса DBW 40.

Кроме того Вы должны запрограммировать обработку ошибок в пользовательской программе.

Байты с 4 по 15 (события 1 .. 6) области SYSTAT всегда сохраняются в экземплярном DB, начиная с адреса DBW 44, обеспечивая установление "бита ошибки" (Bit 2.0) в области SYSTAT.

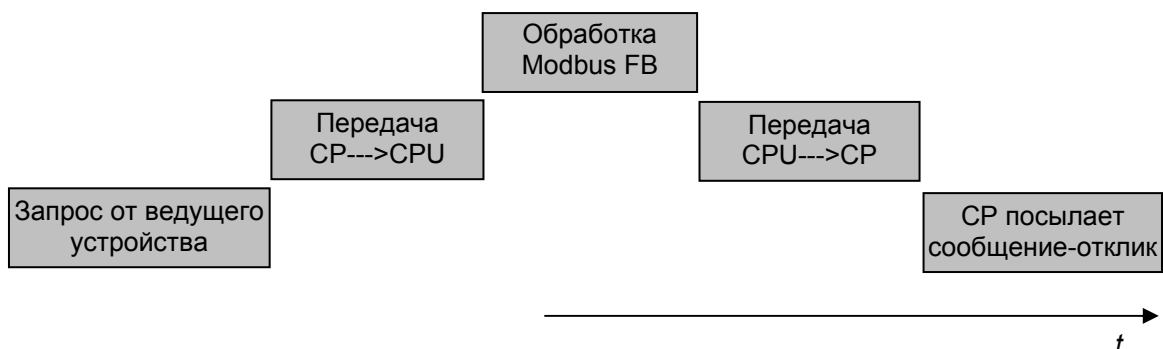
Обратитесь к главе "Диагностика" данного руководства и к руководству по CP 441-2 для получения подробного описания структуры области SYSTAT.

Время реакции

Для обработки кодов FC05, FC15 требуется период времени, складывающийся из времени обработки FB в одном цикле PLC и времени на передачу данных CP--->CPU и CPU--->CP.

Другим функциям, которые обрабатываются непосредственно в CP, требуется только время на передачу данных CP--->CPU или CPU--->CP.

Модуль CP не посылает сообщение отклика в систему ведущего устройства до передачи данных CPU--->CP. В этом примере стандартное время ожидания ответного сообщения (reply monitoring time) достигает 2 с.



Время реакции зависит от длительности цикла (cycle time) программы CPU (Modbus FB) и типа CPU (передача данных CPU<-->CP).

6 Интерфейс CPU - CP

6.1 Интерфейс CPU - CP для модуля CP 341

6.1.1 Общая информация

MODBUS - коммуникационный FB

Передача данных между CP и CPU выполняется с помощью функциональных блоков **P_SND_RK** и **P_RCV_RK**. Указанные блоки вызываются MODBUS-коммуникационным блоком FB (MODBUS Communications FB) из комплекта поставки.

Нет необходимости программировать другие вызовы FB в пользовательской программе SIMATIC.

Адрес модуля

Единственное, что остается сделать пользователю - это определить адрес модуля (LADDR) в MODBUS-коммуникационном FB.

6.1.2 Параметр длины LENGTH при передаче данных между CP и CPU

Параметр длины LENGTH при передаче данных

Передача данных между CP и CPU выполняется с помощью функциональных блоков **P_SND_RK** и **P_RCV_RK**.

Максимально возможная длина передаваемых данных: 1024 байта.

Это означает, что Modbus-драйвер ведомого (slave) устройства выдаст ответное сообщение с кодом ошибки (Exception_Code), в тех случаях, когда делается попытка считывания (read) или записи (write), и при этом превышаете указанная выше максимальная длина данных.

6.1.3 Консистентность данных

Размер блока

Для устойчивой работы системы сообщений в автоматической системе S7 передача данных между CPU и CP осуществляется поблочно (блок за блоком).

При этом размер блока (block size) при организации доступа к области памяти SIMATIC (блоки данных, область меркеров...) составляет 32 байта.

Консистентность данных

Консистентность данных при передаче данных между CPU и CP обеспечивается только для блоков указанного выше размера - 32 байта. При передаче большого количества данных данные передаются поблочно с размером блока, равным 32 байта, и с определенными временными интервалами между отдельными блоками.

При передаче большого количества данных отдельными блоками консистентность не гарантируется, так как в процессе передачи эти данные могут изменяться при обработке в программе пользователя.

Доступ к областям памяти CPU осуществляется при выполнении программы пользователя при обработке FB P_RCV_PK.

Ведомое (slave) Modbus-устройство

Для драйвера ведомого (slave) Modbus-устройства справедливо следующее: для обеспечения консистентности данных при считывании/записи регистров или битов количество передаваемых данных в отдельном сообщении должно быть ограничено указанным выше размером блока: например, максимальным количеством регистров - 16 для FC 03, 04, 16, или максимальным количеством битов - 256 для FC 01, 02, 15.

Если требуется, консистентность обработки связанных областей данных может быть обеспечена с помощью соответствующих механизмов координации в пользовательской программе.

6.2 Интерфейс CPU - CP для модуля CP 441-2

6.2.1 Общая информация

MODBUS - коммуникационный FB

Передача данных между CP и CPU выполняется с помощью встроенных системных функций **PUT** и **GET** и SFB **BSEND** и **STATUS**. Эти SFB-блоки вызываются MODBUS-коммуникационным блоком FB (MODBUS Communications FB) из комплекта поставки.

Системные функции PUT и GET - функции операционной системы, которые выполняются в фоновой программе CPU, независимо от пользовательской программы.

Нет необходимости программировать другие вызовы FB в пользовательской программе SIMATIC.

Коммуникационная связь

Единственное, что остается сделать пользователю - это определить идентификатор соединения (Connection ID) из конфигурации проекта в MODBUS-коммуникационном FB.

Параметр ID описывает уникальный адрес соединения с коммуникационным партнером.

6.2.2 Параметр длины LENGTH при передаче данных между CP и CPU

Параметр длины LENGTH при передаче данных

Передача данных между CP и CPU выполняется с помощью системных функций PUT и GET.

Длина передаваемых данных зависит от используемого типа CPU: (см. также справочное руководство "*System and Standard Functions*" (*Системные и стандартные функции*)).

Системные функции	S7-400
PUT / GET	450 байтов

Это означает, что Modbus-драйвер ведомого (slave) устройства выдаст ответное сообщение с кодом ошибки, когда делается попытка считывания / записи с превышением указанной выше максимальной длины данных.

Ниже в таблице показано максимальное количество данных для передачи

S7-400		
Modbus- функциональный код	Максимальное количество регистров	Максимальное количество битов
01, 02		2040
03, 04, 16	127	
15		2040

6.2.3 Консистентность данных

Размер блока

Передача данных между CPU и CP выполняется с помощью встроенных системных функций PUT и GET.

Для устойчивой работы системы сообщений в автоматической системе S7 передача данных между CPU и CP осуществляется поблочно (блок за блоком).

При этом размер блока (block size) при организации доступа к области памяти SIMATIC (блоки данных, область меркеров...) составляет 32 байта.

Консистентность данных

Доступ к областям памяти CPU с помощью встроенных системных функций PUT и GET выполняется асинхронно по отношению к программам-приложениям STEP 7. Поэтому консистентность данных при передаче данных между CPU и CP обеспечивается только для блоков размером 32 байта.

При передаче большего количества данных данные передаются поблочно с размером блока, равным 32 байта, и с временными интервалами между отдельными блоками.

При передаче большого количества данных отдельными блоками консистентность не гарантируется, так как в процессе передачи эти данные могут изменяться при обработке в программе пользователя.

Необходимо учитывать, что доступ к областям памяти CPU может осуществляться в любой момент времени во время выполнения пользовательской программы, так как встроенные системные функции выполняются в фоновой программе CPU.

Данные вопросы связаны с выбранным S7 CPU и описаны в справочном руководстве "*Системные и стандартные функции*" в главах 17 и 18 ("консистентность данных"). (см. также другие источники информации, указанные во введении данного руководства).

Ведомое (slave) Modbus-устройство

Для драйвера ведомого (slave) Modbus-устройства справедливо следующее: для обеспечения консистентности данных при считывании/записи регистров или битов количество передаваемых данных в отдельном сообщении должно быть ограничено указанным выше размером блока: например, для S7-400: максимальное количество регистров - 16 для FC 03, 04, 16, или максимальным количеством битов - 256 для FC 01, 02, 15.

Необходимо учитывать, что доступ к областям памяти CPU Modbus-функциями может осуществляться в любой момент времени во время выполнения пользовательской S7-программы.

Если требуется, консистентность обработки связанных областей данных может быть обеспечена с помощью соответствующих механизмов координации в пользовательской программе.

7 Протокол обмена данными

Общая информация

Используемая процедура "прозрачна" с точки зрения кодов, она является асинхронной и предполагает полудуплексный режим.

Передача данных выполняется без квитирования установления связи.

Связь ведущего и ведомого устройств (Master-Slave-связь)

Модуль CP инициирует передачу (CP - ведущее устройство - master), и после выдачи сообщения запроса он ожидает ответное сообщение от ведомого (slave) устройства в течение установленного времени ожидания ответа (reply monitoring time).

Структура сообщения

Передаваемые данные "Master-Slave" и/или "Slave-Master" начинаются с адреса ведомого устройства (Slave Address), за которым следует функциональный код (Function Code). Затем следуют пересылаемые данные. Структура полей данных зависит от используемого кода функции. Данные для CRC-проверки передаются в конце сообщения.

Адрес	Функция	Данные	CRC
1 байт	1 байт	n байтов	2 байта

Адрес	Адрес ведомого MODBUS-устройства (MODBUS Slave Address)
Функция	Функциональный код MODBUS (MODBUS Function Code)
Данные сообщения:	Byte_Count (счетчик байтов), Coil_Number (число катушек), Data (данные)
CRC	Контрольная сумма сообщения

Адрес ведомого устройства (Slave Address)

Значение для адреса ведомого устройства лежит в диапазоне от 1 до 255. Используемый адрес - это шинный адрес определенного ведомого устройства.

Рассылка сообщений

Ведущее (master) устройство использует нулевой адрес (zero) в качестве адреса ведомого устройства при адресации одновременно всех ведомых устройств шины. Рассылка сообщений (**Broadcast Messages**) разрешается только для функциональных кодов **05, 06, 15** и **16**.

Рассылка сообщений не сопровождается ответными сообщениями ведомых устройств.

Функциональный код (Function Code)

Функциональный код определяет значение, а также структуру сообщения. Модулем CP поддерживаются следующие функциональные коды:

Функциональный код	Функция в соответствии со спецификацией MODBUS
01	Считывание состояния выхода (Read Output Status)
02	Считывание состояния входа (Read Input Status)
03	Считывание выходных регистров (Read Output Registers)
04	Считывание входных регистров (Read Input Registers)
05	Назначение для одной катушки (Force Single Coil)
06	Предустановки для одного регистра (Preset Single Register)
08	Тестирование обратной связи (Loop Back Diagnostic Test)
15	Назначения для нескольких катушек (Force Multiple Coils)
16	Предустановки для нескольких регистров (Preset Multiple Registers)

Поле данных DATA

Поле данных DATA определяется функциональным кодом, как например: Bytecount (счетчик байтов), Coil_Start Address (начальный адрес катушки), Register_Start Address (начальный адрес регистра); Number_of_Coils (число катушек), Number_of_Registers (число регистров) и т.д.. См. также главу "Функциональные коды".

Проверка четности (CRC Check)

Конец сообщения идентифицируется с помощью контрольной суммы CRC 16, состоящей из 2 байтов. Она рассчитывается как полином:
 $x^{16} + x^{15} + x^2 + 1$.

Первым передается младший байт, затем - старший байт.

Конец сообщения

Загружаемый драйвер распознает конец сообщения, затем передачи нет в течение периода времени, требуемого для передачи трех с половиной символов ($3.5 \times \{\text{время ожидания символа (character delay time)}\}$) (см. справочное руководство по протоколу MODBUS (*MODBUS Protocol Reference Guide*)).

Следовательно, задержка TIME_OUT зависит от скорости передачи.

Скорость передачи, бит/с	TIME_OUT, мс
76800	0,5
38400	1
19200	2
9600	4
4800	8
2400	16
1200	32
600	64
300	128

После истечения задержки TIME_OUT по окончании передачи проверяется формат ответного сообщения, принятого ведомым (slave) устройством.

Сообщения об отказах (Exception Responses)

При распознавании ошибки в сообщении запроса от ведущего (master) устройства (например, некорректный адрес регистра), ведомое (slave) устройство устанавливает старший значащий бит в функциональном коде ответного сообщения.

За этим следует передача одного байта кода ошибки (Exception Code), который описывает причину ошибки.

Коды ошибок в сообщениях

Сообщение с кодом ошибки (error code) от ведомого устройства (slave) имеет следующую структуру: например, адрес ведомого устройства 5, функциональный код 5, код ошибки 02

Коды ошибок в сообщениях ведомого устройства EXCEPTION_CODE_хх:

05H	адрес ведомого устройства
85H	функциональный код
02H	код ошибки (1 ... 4)
ххH CRC	младший байт контрольного кода ("Low")
ххH CRC	старший байт контрольного кода ("High")

Код ошибки	Значение в соответствии со спецификациями MODBUS	Причина (краткое описание)
1	Некорректная функция (Illegal Function)	Некорректный функциональный код (function code)
2	Некорректный адрес данных (Illegal Data Address)	Указан некорректный адрес данных ведомого устройства Попытка доступа к области SIMATIC, которая недоступна (enabled) (см. данные назначения параметров - области, ограничения)
3	Некорректное значение данных (Illegal Data Value)	Некорректное значение данных ведомого устройства Размер (длина) больше 2040 битов или 127 регистров, в поле данных не FF00 или 0000 при FC 05, диагностический код не равен 0000 при FC 08.
4	Отказ соответствующего устройства (Failure in Associated Device)	В ведомом устройстве обнаружена внутренняя ошибка Инициализация Modbus-коммуникаций пока не выполнена (FB) или FB сообщает об ошибке, ошибка при передаче данных CP<->CPU (например, не существует DB, превышена максимальная длина данных (размер блока при передаче данных CPU<->CP)).

Вспомогательные сигналы RS 232C (Secondary Signals)

Следующие вспомогательные сигналы RS 232C могут использоваться в CP при использовании интерфейсного модуля RS 232C:

- DCD (входной) **Data Carrier Detect** - Обнаружение несущего сигнала
- DTR (выходной) **Data Terminal Ready** - CP готов к работе
- DSR (входной) **Data Set Ready** - партнер готов к работе
- RTS (выходной) **Request To Send** - CP готов к передаче (send)
- CTS (входной) **Clear To Send** - партнер может принять данные от CP (отклик на сигнал RTS = ON от CP)
- RI (входной) **Ring Indicator** - индикация входящего звонка

Если CP 441 включен, то выходные сигналы в состоянии OFF (выкл) (неактивен).

Вы должны задать параметры для способа использования управляющих сигналов DTR/DSR и RTS/CTS с помощью интерфейса параметризации: *CP 441: Point-to-Point Communication, Parameter Assignment (CP 341: PtP-соединение, назначение параметров)* или управляя ими с помощью вызова функций (FB) в пользовательской программе.

Использование вспомогательных сигналов RS 232C

Вспомогательные сигналы RS 232C могут быть использованы:

- при параметризации автоматического использования вспомогательных сигналов RS 232C
- посредством FB-функций V24_STAT и V24_SET

Примечание

При параметризации автоматического использования вспомогательных сигналов RS 232C невозможны ни RTS/CTS-управление потоками данных, ни RTS- и DTR-управление посредством FB-функции V24_SET. Однако с другой стороны, всегда возможно считывать все вспомогательные сигналы RS 232C с помощью FB V24_STAT.

В следующих разделах описывается как управлять вспомогательными сигналами RS 232C и как их проверять.

Автоматическое использование вспомогательных сигналов

Автоматическое использование вспомогательных сигналов RS 232C в CP производится следующим образом:

- Как только CP переключается с помощью параметризации в рабочий режим с автоматическим использованием вспомогательных сигналов RS 232C, он переключает линию RTS в выключенное состояние (OFF), а линию DTR - во включенное состояние (ON) (это означает, что CP готов к работе (режим - ready)).
Фреймы сообщений не могут посылаться и приниматься до тех пор, пока линия DTR находится в ON-состоянии. Пока DTR находится в OFF-состоянии, никакие данные не принимаются с помощью интерфейса RS 232C. Если выполняется запрос на передачу (send), то он отбрасывается с соответствующим сообщением об ошибке.
- Если выполняется запрос на передачу (send), то RTS устанавливается в ON-состояние и начинается отсчет заданного параметром времени ожидания вывода данных (data output waiting time). Если время ожидания вывода данных заканчивается и CTS = ON, то данные посылаются через интерфейс RS 232C.
- Если линия CTS не устанавливается в состояние ON в течение времени ожидания вывода данных (data output waiting time) для разрешения посылки данных или, если CTS переключается в состояние OFF во время передачи данных, то запрос на передачу (send) отбрасывается и генерируется сообщение об ошибке.
- После выполнения передачи данных линия RTS устанавливается в OFF после истечения заданного параметром времени для перехода RTS в OFF-состояние ("time to RTS OFF"). CP не ждет, пока CTS переключится в OFF.
- Данные могут приниматься с помощью RS 232C-интерфейса, как только линия DSR устанавливается в ON. Если возникает угроза переполнения приемного буфера CP, то CP не выдает ответного сообщения.
- Запрос на передачу (send) или прием данных отбрасывается с соответствующим сообщением об ошибке, если DSR переключается из состояния ON в OFF. При этом в диагностический буфер CP вводится сообщение "DSR = OFF (automatic use of V24 signals)" ("DSR = OFF (автоматическое использование сигналов V24)").

Примечание

Автоматическое использование вспомогательных сигналов RS 232C возможно только в полудуплексном режиме.

При выборе режима автоматического использования вспомогательных сигналов RS 232C невозможны ни RTS/CTS-управление потоками данных, ни RTS- и DTR-управление посредством FC-функции V24_SET.

Примечание

Параметр "Время для перехода RTS в OFF" ("time to RTS OFF") должен быть задан с помощью интерфейса параметризации так, чтобы партнер мог принять последние символы фрейма сообщения во всей их полноте до того, пока RTS и запрос на передачу будут выключены. Параметр "Время ожидания вывода данных" ("data output waiting time") должно выбираться так, чтобы партнер мог перейти в состояние готовности для приема данных до истечения этого времени.

Временная диаграмма

На рисунке 7-1 представлена обработка во времени запроса на передачу (send request):

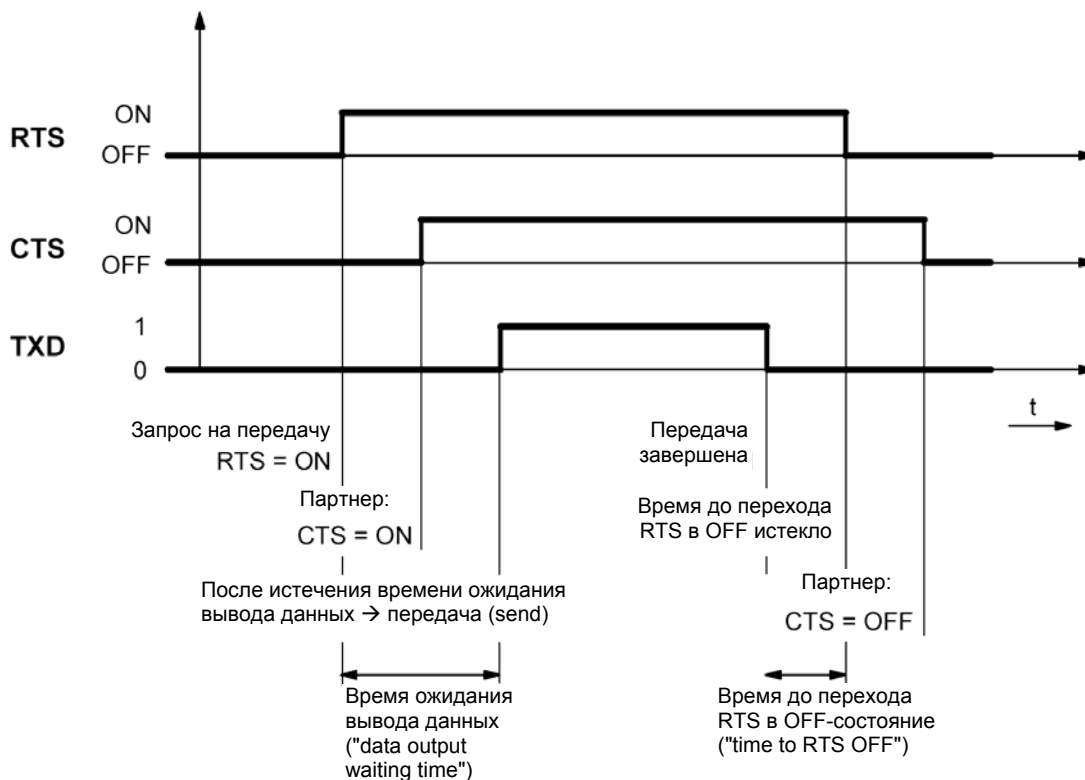


Рис. 7-1 Схема автоматического использования вспомогательных сигналов RS 232C

8 Функциональные коды

Используемые функциональные коды

Драйвером поддерживаются следующие MODBUS функциональные коды:

Функциональный код	Функция в соответствии со спецификацией MODBUS	Функция в SIMATIC S7	
01	Считывание состояния выхода (Read Output Status)	Побитное чтение	Меркеры M
		Побитное чтение	Выходы Q
		Побитное чтение (с 16-битным интервалом)	Таймеры T
		Побитное чтение (с 16-битным интервалом)	Счетчики C
02	Считывание состояния входа (Read Input Status)	Побитное чтение	Меркеры M
		Побитное чтение	Входы Q
03	Считывание выходных регистров (Read Output Registers)	Считывание слово за словом	Блок данных DB
04	Считывание входных регистров (Read Output Registers)	Считывание слово за словом	Блок данных DB
05	Назначение для одного выхода (катушки) (Force Single Coil)	Побитная запись	Меркеры M
		Побитная запись	Выходы Q
06	Предустановки для одного регистра (Preset Single Register)	Запись слово за словом	Блок данных DB
08	Тестирование линии связи (Loop Back Test)	-	-
15	Назначения для нескольких катушек (Force Multiple Coils)	Побитная запись (1..2040 битов)	Меркеры M
		Побитная запись (1..2040 битов)	Выходы Q
16	Предустановки для нескольких вых. регистров (Preset Multiple (holding) Registers)	Запись слово за словом (1..127 регистров)	Блок данных DB

Примечание

Все MODBUS-адреса, перечисленные ниже, относятся к передаваемым сообщениям; они не касаются уровня пользователя в системе ведущего MODBUS-устройства.

Это означает, что MODBUS-адреса в передаваемых сообщениях начинаются с номера 0000H.

Примечание

MODBUS-драйвер ведомого (slave) устройства поддерживает обработку блоков с максимальной длиной 512 слов данных при использовании всех функциональных кодов, которые используются для доступа к областям DB SIMATIC (FC 03, 04, 06, 16).

При преобразовании MODBUS-адреса в SIMATIC-адрес "прямой переход" от одного номера DB к последующему номеру DB внутри сообщения-запроса ведущего MODBUS-устройства невозможен. Ведомое (slave) MODBUS-устройство выдает в ответ на это сообщение-запрос сообщение отклика с кодом ошибки (Exception Code) и вводит в диагностический буфер "0E 39" (ошибка доступа в области SIMATIC блоков данных DB).

Например, адрес слова DBX 510 с длиной > 2 не может быть выведен, так как адрес для номера регистра > 2 (например, регистр 3) будет автоматически означать адрес слова 0 в DBX+1.

8.1 Функциональный код 01 (Read Coil (Output) Status) (Считывание состояния выхода (катушки))

Функция

Данная функция обеспечивает для системы ведущего Modbus-устройства доступ в режиме чтения (read) отдельных битов в областях памяти SIMATIC, как показано ниже.

Сообщение-запрос

ADDR (адрес)	FUNC (код)	start_address (начальный адрес)	bit_number (число битов)	CRC (четность)
-----------------	---------------	------------------------------------	-----------------------------	-------------------

Ответное сообщение

ADDR (адрес)	FUNC (код)	Byte_count n (счетчик байтов)	n Byte Data (число байтов данных)	CRC (четность)
-----------------	---------------	----------------------------------	--------------------------------------	-------------------

Start_address (начальный адрес)

Начальный Modbus-адрес бита "start_address" интерпретируется драйвером следующим образом:

Драйвер проверяет, размещен ли начальный адрес "start_address" в одной из областей, которые были определены при назначении параметров в диалоговом окне "Conversion of MODBUS Addressing for FC 01, 05, 15" ("Преобразование MODBUS-адресов для FC 01, 05, 15") ("from/to" ("от/до"): "memory bits" ("меркеры"), "outputs" ("выходы"), "timers" ("таймеры"), "counters" ("счетчики")).

Если начальный MODBUS-адрес start_address располагается в области	обращение производится к следующей области памяти SIMATIC
от <i>aaaaa</i> до <i>bbbbbb</i>	Область меркеров начало с <i>M uuuuu.0</i>
от <i>cccc</i> до <i>dddd</i>	Область выходов начало с <i>Q ooooo.0</i>
от <i>eeee</i> до <i>ffff</i>	Область таймеров начало с <i>T tttt</i>
от <i>ggggg</i> до <i>hhhhh</i>	Область счетчиков начало с <i>C zzzzz</i>

Адрес, к которому выполняется обращение, вычисляется следующим образом:

Начальный SIMATIC-адрес	Формула преобразования
Байт меркеров	$= ((start_address) - aaaaa) / 8) + uuuuu$
Байт выходов	$= ((start_address) - ccccc) / 8) + ooooo$
Таймер	$= ((start_address) - eeeee) / 16) + tttt$
Счетчик	$= ((start_address) - ggggg) / 16) + zzzzz$

Доступ к областям меркеров и выходов

При доступе к SIMATIC-областям меркеров и выходов оставшееся число битов вычисляется и используется для адресации соответствующих битов внутри байта в областях меркеров и выходов.

Число битов "bit_number"

Значения между 1 и 2040 допускаются в качестве количества битов "bit_number" ("число катушек" - "number of coils"). Данное количество битов считывается.

Доступ к областям таймеров и счетчиков

При расчете адресов необходимо, чтобы результат делился на 16: (start_address - eeeee) или (start_address - ggggg) без остатка (доступ в режиме слово за словом начинается только с границ между словами).

При доступе к областям таймеров и счетчиков SIMATIC необходимо, чтобы число битов ("bit_number") без остатка делилось на 16 (доступ только в режиме слово за словом).

Необходимо помнить, что при использовании CP 341 максимальное количество таймеров и счетчиков, которое может быть считано, равно 16.

Примечание

Ограничения, которые зависят от CPU, представлены в главе "Интерфейс CPU-CP".

Пример применения

Пример назначения параметров:

Преобразование MODBUS-адресов для функциональных кодов FC 01, 05, 15	
MODBUS-адрес в переданном сообщении	Область памяти SIMATIC
от 0 до 2047	Область меркеров начало с M 1000.0
от 2048 до 2559	Область выходов начало с Q 256.0
от 4096 до 4607	Область таймеров начало с T 100
от 4608 до 5119	Область счетчиков начало с C 200

Сообщение-запрос для функции 01:

05H	Адрес ведомого устройства ADDR
01H	Функциональный код FUNC
00H	Начальный битовый адрес (старший байт)
40H	Начальный битовый адрес (младший байт)
00H	Число битов (старший байт)
20H	Число битов (младший байт)
xxH	Код CRC-проверки (младший байт)
xxH	Код CRC-проверки (старший байт)

Ответное сообщение для функции 01:

05H	Адрес ведомого устройства ADDR
01H	Функциональный код FUNC
04H	Байтовый счетчик
01H	<Данные 1> M 1008.0 - M 1008.7
17H	<Данные 2> M 1009.0 - M 1009.7
02H	<Данные 3> M 1010.0 - M 1010.7
18H	<Данные 4> M 1011.0 - M 1011.7
xxH	Код CRC-проверки (младший байт)
xxH	Код CRC-проверки (старший байт)

Вычисление адреса

Начальный Modbus-адрес "start_address" 0040H (64 десятичное) размещается в области меркеров ("memory bit"):

Байт меркеров	= ((start_address) - aaaaa) / 8 + iiiii
	= ((64 - 0) / 8) + 1000
	= 1008

Оставшееся число битов (Rest Bit_Number):

Оставшееся	= ((start_address) - aaaaa) % 8
число битов	= ((64 - 0) % 8)
	= 0

Таким образом доступ обеспечивается с битового адреса M 1008.0 до M 1011.7 включительно.

Количество битов

Может быть считано следующее количество Modbus-битов ("bit_number"): 0020H (32 десятичное) = 32 бита или 4 байта.

Дополнительные примеры

Несколько примеров операций доступа показаны в таблице:

start_address		Доступ в SIMATIC-области			Начало с
(hex)	(десят.)	(десят.)			(десят.)
0000	0	Меркер	((0 - 0) / 8) + 1000		М 1000.0
0021	33	Меркер	((33 - 0) / 8) + 1000		М 1004.1
0400	1024	Меркер	((1024 - 0) / 8) + 1000		М 1128.0
0606	1542	Меркер	((1542 - 0) / 8) + 1000		М 1192.6
0840	2112	Выход	((2112 - 2048) / 8) + 256		Q 264.0
09E4	2532	Выход	((2532 - 2048) / 8) + 256		Q 316.4
1010	4112	Таймеры	((4112 - 4096) / 16) + 100		Т 101
10C0	4288	Таймеры	((4288 - 4096) / 16) + 100		Т 112
1200	4608	Счетчики	((4608 - 4608) / 16) + 200		С 200
13E0	5088	Счетчики	((5088 - 4608) / 16) + 200		С 230

8.2 Функциональный код 02 (Read Input Status) (Считывание состояния входа)

Функция

Данная функция обеспечивает для системы ведущего Modbus-устройства считывание (read) отдельных битов в областях памяти SIMATIC, как показано ниже.

Сообщение-запрос

ADDR (адрес)	FUNC (код)	start_address (начальный адрес)	bit_number (число битов)	CRC (четность)
-----------------	---------------	------------------------------------	-----------------------------	-------------------

Ответное сообщение

ADDR (адрес)	FUNC (код)	Byte_count n (счетчик байтов)	n Byte Data (число байтов данных)	CRC (четность)
-----------------	---------------	----------------------------------	--------------------------------------	-------------------

Start_address (начальный адрес)

Начальный Modbus-адрес бита "start_address" интерпретируется драйвером следующим образом.

Драйвер проверяет, размещен ли начальный адрес "start_address" в одной из областей, которые были определены при назначении параметров в диалоговом окне "Conversion of MODBUS Addressing for FC 02" ("Преобразование MODBUS-адресов для FC 02") ("from/to" ("от/до"): "memory bits" ("меркеры"), "inputs" ("входы")).

Если начальный MODBUS-адрес start_address располагается в области	обращение производится к следующей области памяти SIMATIC
от <i>kkkkk</i> до <i>lllll</i>	Область меркеров начало с <i>M vvvvv.0</i>
от <i>nnnnn</i> до <i>rrrrr</i>	Область входов начало с <i>I sssss.0</i>

Адрес, к которому выполняется обращение, вычисляется следующим образом:

Начальный SIMATIC-адрес	Формула преобразования
Байт меркеров	$= ((start_address) - kkkkk) / 8 + vvvvv$
Байт входов	$= ((start_address) - nnnnn) / 8 + sssss$

Доступ к областям меркеров и входов

При доступе к SIMATIC-областям меркеров и входов оставшееся число битов вычисляется и используется для адресации соответствующих битов внутри байта в областях меркеров и входов.

Число битов "bit_number"

Значения между 1 и 2040 допускаются в качестве количества битов "bit_number" ("число катушек" - "number of coils"). Данное количество битов считывается.

Примечание

Ограничения, которые зависят от CPU, представлены в главе "Интерфейс CPU-CP".

Пример применения

Преобразование MODBUS-адресов для функциональных кодов FC 02	
MODBUS-адрес в переданном сообщении	Область памяти SIMATIC
от 0 до 4095	Область меркеров начало с <i>M 2000.0</i>
от 4096 до 5119	Область входов начало с <i>I 128.0</i>

Сообщение-запрос для функции 02:

05H	Адрес ведомого устройства ADDR
02H	Функциональный код FUNC
10H	Начальный битовый адрес (старший байт)
30H	Начальный битовый адрес (младший байт)
00H	Число битов (старший байт)
18H	Число битов (младший байт)
xxH	Код CRC-проверки (младший байт)
xxH	Код CRC-проверки (старший байт)

Ответное сообщение для функции 02:

05H	Адрес ведомого устройства ADDR
02H	Функциональный код FUNC
03H	Байтовый счетчик
12H	<Данные 1> 134.0 - 134.7
34H	<Данные 2> 135.0 - 135.7
56H	<Данные 3> 136.0 - 136.7
xxH	Код CRC-проверки (младший байт)
xxH	Код CRC-проверки (старший байт)

Вычисление адреса

Начальный Modbus-адрес "start_address" 1030H (4144 десятичное) размещается в области входов:

Байт входов	= ((start_address) - nnnnn) / 8 + sssss
	= ((4144 - 4096) / 8) + 128
	= 134

Оставшееся число битов (Rest Bit_Number):

Оставшееся	= ((start_address) - aaaaa) % 8
число битов	= ((4144 - 4096) % 8)
	= 0

Таким образом доступ обеспечивается с битового адреса | 134.0 до | 136.7 включительно.

Количество битов

Может быть считано следующее количество Modbus-битов ("bit_number"): 0018H (24 десятичное) = 24 бита или 3 байта.

Дополнительные примеры

Несколько примеров операций доступа показаны в таблице:

start_address (hex)	(десят.)	Доступ в SIMATIC-области (десят.)	Начало с (десят.)
0000	0	Меркер ((0 - 0) / 8) + 2000	М 2000.0
0071	113	Меркер ((113 - 0) / 8) + 2000	М 2014.1
0800	2048	Меркер ((2048 - 0) / 8) + 2000	М 2256.0
0D05	3333	Меркер ((3333 - 0) / 8) + 2000	М 2416.5
1000	4096	Вход ((4096 - 4096) / 8) + 128	I 128.0
10A4	4260	Вход ((4260 - 4096) / 8) + 128	I 148.4

8.3 Функциональный код 03 (Read Output Registers) (СЧИТЫВАНИЕ ВЫХОДНЫХ РЕГИСТРОВ)

Функция

Данная функция позволяет системе ведущего Modbus-устройства считывать (read) слова данных из блока данных.

Сообщение-запрос

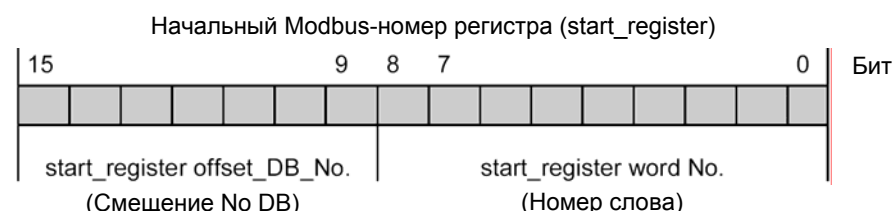
ADDR (адрес)	FUNC (код)	start_register (начальный регистр)	register_number (число регистров)	CRC (четность)
-----------------	---------------	---------------------------------------	--------------------------------------	-------------------

Ответное сообщение

ADDR (адрес)	FUNC (код)	Byte_count n (счетчик байтов)	n Byte Data (число байтов данных)	CRC (четность)
-----------------	---------------	----------------------------------	--------------------------------------	-------------------

Начальный регистр Start_register

Начальный Modbus-номер регистра "start_register" интерпретируется драйвером следующим образом:



Для последующей генерации адреса драйвер использует "базовый номер блока DB" ("base DB number") (начиная с DB xxxxx), который вводится в диалоговом окне "Conversion of MODBUS Addressing for FC 03" ("Преобразование MODBUS-адресов для FC 03").

Вычисление адреса (преобразование адреса) выполняется в два этапа, как показано ниже:

Доступ к SIMATIC-области	Формула преобразования
Блок данных DB (итоговый №)	= (Базовый номер DB xxxxx + start_register offset_DB_No.)
Слово данных DBW	= (start_register word_No. × 2)

Формула вычисления начального регистра Start_register (начальный регистр)

После того, как становится известным итоговый № блока данных DB, из которого следует считывать данные, начальный Modbus-номер регистра "start_register" интерпретируется драйвером следующим образом:

$$\text{start_register} = ((\text{итоговый \# DB} - \text{базовый \# DB}) \times 512) + (\text{data word_DBW} / 2)$$

Этот адрес базируется только на четных номерах слов данных.

Число регистров "register_number"

Значения между 1 и 127 допускаются в качестве количества регистров "register_number". Данное количество регистров может быть считано. Максимальное количество регистров, которое может быть считано, вычисляется следующим образом:

$$(\text{register_number})_{\text{max}} = 512 - \text{start_register}$$

Примечание

Ограничения, которые зависят от CPU, представлены в главе "Интерфейс CPU-CP".

Пример применения

Пример назначения параметров:

Преобразование MODBUS-адресов для функциональных кодов FC 03, 06, 16	
MODBUS-адрес в переданном сообщении	Область памяти SIMATIC
0	Область блоков данных (базовый номер DB) начало с DB 800

Сообщение-запрос для функции 03:

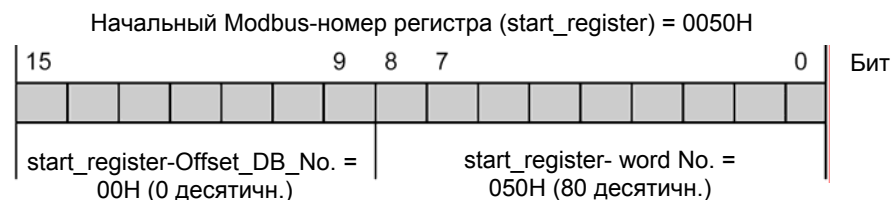
05H Адрес ведомого устройства ADDR
 03H Функциональный код FUNC
 00H Начальный адрес регистра (старший байт)
 50H Начальный адрес регистра (младший байт)
 00H Число регистров (старший байт)
 02H Число регистров (младший байт)
 ххH Код CRC-проверки (младший байт)
 ххH Код CRC-проверки (старший байт)

Ответное сообщение для функции 03:

05H Адрес ведомого устройства ADDR
 03H Функциональный код FUNC
 04H Байтовый счетчик
 87H <Данные 1> DBW 160 (старший байт)
 65H <Данные 2> DBW 160 (младший байт)
 43H <Данные 3> DBW 161 (старший байт)
 21H <Данные 4> DBW 161 (младший байт)
 ххH Код CRC-проверки (младший байт)
 ххH Код CRC-проверки (старший байт)

Вычисление начального адреса регистра Start_register

Начальный Modbus-номер регистра "start_register" 0050H (80 десятичн.) интерпретируется драйвером следующим образом:



Блок данных DB (итоговый №)	= (Базовый номер DB xxxxx + + start_register offset_DB_No.) = (800 + 0) = 800
Слово данных DBW	= (start_register word_No. × 2) = (80 × 2) = 160

Доступ производится к DB 800, слово данных DBW 160.

Количество регистров

Может быть считано следующее количество Modbus-регистров ("register_number"): 0002H (2 десятичное) = 2 регистра или 2 слова данных.

Дополнительные примеры

Несколько примеров операций доступа показаны в таблице:

start_register		Базовый № DB (десят.)	start_register		Итоговый № DB (десят.)	DBW (десят.)	
(hex)	(десят.)		Смещение DB_No (десят.)	Номер слова			
			(десят.)	(hex)	(десят.)		
0000	0	800	0	000	0	800	0
01F4	500	800	0	1F4	500	800	1000
0200	512	800	1	000	0	801	0
02FF	767	800	1	0FF	255	801	510
0300	768	800	1	100	256	801	512
03FF	1023	800	1	1FF	511	801	1022
0400	1024	800	2	000	0	802	0

8.4 Функциональный код 04 (Read Input Registers) (Считывание входных регистров)

Функция

Данная функция позволяет системе ведущего Modbus-устройства считывать (read) слова данных из блока данных.

Сообщение-запрос

ADDR (адрес)	FUNC (код)	start_register (начальный регистр)	register_number (число регистров)	CRC (четность)
-----------------	---------------	---------------------------------------	--------------------------------------	-------------------

Ответное сообщение

ADDR (адрес)	FUNC (код)	Byte_count n (счетчик байтов)	n Byte Data (число байтов данных)	CRC (четность)
-----------------	---------------	----------------------------------	--------------------------------------	-------------------

Начальный регистр Start_register

Начальный Modbus-номер регистра "start_register" интерпретируется драйвером следующим образом:



Для последующей генерации адреса драйвер использует "базовый номер блока DB" ("base DB number") (начиная с DB xxxxx), который вводится в диалоговом окне "Conversion of MODBUS Addressing for FC 04" ("Преобразование MODBUS-адресов для FC 04").

Вычисление адреса (преобразование адреса) выполняется в два этапа, как показано ниже:

Доступ к SIMATIC-области	Формула преобразования
Блок данных DB (итоговый №)	= (Базовый номер DB xxxxx + + start_register offset_DB_No.)
Слово данных DBW	= (start_register word_No. × 2)

Формула вычисления начального регистра Start_register (начальный регистр)

После того, как становится известным итоговый № блока данных DB, из которого следует считывать данные, начальный Modbus-номер регистра "start_register" интерпретируется драйвером следующим образом:

$$\text{start_register} = ((\text{итоговый \# DB} - \text{базовый \# DB}) \times 512) + (\text{data word_DBW} / 2)$$

Этот адрес фиксируется только на четных номерах слов данных.

Число регистров "register_number"

Значения между 1 и 127 допускаются в качестве количества регистров "register_number". Данное количество регистров может быть считано. Максимальное количество регистров, которое может быть считано, вычисляется следующим образом:

$$(\text{register_number})_{\text{max}} = 512 - \text{start_register}$$

Примечание

Ограничения, которые зависят от CPU, представлены в главе "Интерфейс CPU-CP".

Пример применения

Пример назначения параметров:

Преобразование MODBUS-адресов для функциональных кодов FC 04	
MODBUS-адрес в переданном сообщении	Область памяти SIMATIC
0	Область блоков данных (базовый номер DB) начало с DB 900

Сообщение-запрос для функции 04:

05H	Адрес ведомого устройства ADDR
04H	Функциональный код FUNC
02H	Начальный адрес регистра (старший байт)
C0H	Начальный адрес регистра (младший байт)
00H	Число регистров (старший байт)
03H	Число регистров (младший байт)
xxH	Код CRC-проверки (младший байт)
xxH	Код CRC-проверки (старший байт)

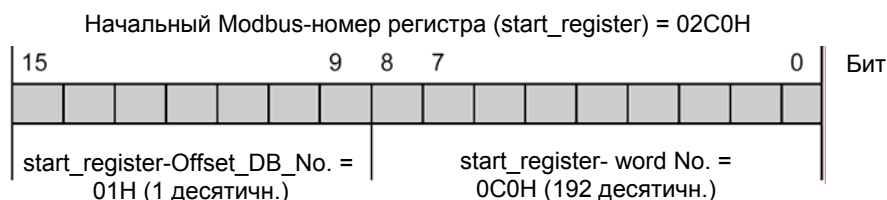
Загружаемый драйвер для PtP-соединений CP
(S7 - ведомое устройство)
A5E00218418-

Ответное сообщение для функции 04:

05H	Адрес ведомого устройства ADDR
04H	Функциональный код FUNC
06H	Байтовый счетчик
A1H	<Данные 1> DBW 384 (старший байт)
A2H	<Данные 2> DBW 384 (младший байт)
A3H	<Данные 3> DBW 385 (старший байт)
A4H	<Данные 4> DBW 385 (младший байт)
A5H	<Данные 5> DBW 386 (старший байт)
A6H	<Данные 6> DBW 386 (младший байт)
xxH	Код CRC-проверки (младший байт)
xxH	Код CRC-проверки (старший байт)

Вычисление начального адреса регистра Start_register

Начальный Modbus-номер регистра "start_register" 02C0H (704 десятичн.) интерпретируется драйвером следующим образом:



Блок данных DB (итоговый №)	= (Базовый номер DB xxxxx + start_register offset_DB_No.)
	= (900 + 0)
	= 900
Слово данных DBW	= (start_register word_No. × 2)
	= (192 × 2)
	= 384

Доступ производится к DB 901, слово данных DBW 384.

Количество регистров

Может быть считано следующее количество Modbus-регистров ("register_number"): 0003H (3 десятичное) = 3 регистра или 3 слова данных.

Дополнительные примеры

Несколько примеров операций доступа показаны в таблице:

start_register		Базовый № DB (десят.)	start_register			Итоговый № DB (десят.)	DBW (десят.)
(hex)	(десят.)		Смещение DB_No (десят.)	Номер слова			
				(hex)	(десят.)		
0000	0	900	0	000	0	900	0
0064	100	900	0	064	100	900	200
00C8	200	900	0	0C8	200	900	400
0190	400	900	0	190	400	900	800
1400	5120	900	10	000	0	910	0
1464	5220	900	10	064	100	910	200
14C8	5320	900	10	0C8	200	910	400

8.5 Функциональный код 05 (Force Single Coil) (Назначение сигнала для отдельного выхода)

Функция

Данная функция обеспечивает для системы ведущего Modbus-устройства запись (write) в отдельные биты в областях памяти SIMATIC, как показано ниже.

Сообщение-запрос

ADDR (адрес)	FUNC (код)	coil_address (адрес бита)	DATA-on/off (данные вкл/выкл)	CRC (четность)
-----------------	---------------	------------------------------	----------------------------------	-------------------

Ответное сообщение

ADDR (адрес)	FUNC (код)	coil_address (адрес бита)	DATA-on/off (данные вкл/выкл)	CRC (четность)
-----------------	---------------	------------------------------	----------------------------------	-------------------

coil_address (начальный адрес)

Начальный Modbus-адрес бита "coil_address" интерпретируется драйвером следующим образом.

Драйвер проверяет, размещен ли начальный адрес "coil_address" в одной из областей, которые были определены при назначении параметров в диалоговом окне "Conversion of MODBUS Addressing for FC 01, 05, 15" ("Преобразование MODBUS-адресов для FC 01, 05, 15") ("from/to" ("от/до"): "memory bits" ("меркеры"), "outputs" ("выходы")).

Если начальный MODBUS-адрес start_address располагается в области	обращение производится к следующей области памяти SIMATIC
от <i>aaaaa</i> до <i>bbbbbb</i>	Область меркеров начало с <i>M <i>uuuu</i>.0</i>
от <i>cccccc</i> до <i>dddddd</i>	Область выходов начало с <i>Q <i>oooo</i>.0</i>

Адрес, к которому выполняется обращение, вычисляется следующим образом:

Начальный SIMATIC-адрес	Формула преобразования
Байт меркеров	$= ((start_address) - aaaaa) / 8) + uuuuu$
Байт выходов	$= ((start_address) - ccccc) / 8) + ooooo$

Доступ к областям меркеров и выходов

При доступе к SIMATIC-областям меркеров и выходов оставшееся число битов вычисляется и используется для адресации соответствующих битов внутри байта в областях меркеров и выходов.

Доступ к областям таймеров и счетчиков

При доступе к SIMATIC-областям таймеров и счетчиков операция доступа прерывается с сообщением об ошибке для функционального кода 05.

DATA-on/off (Данные вкл/выкл)

Для этого параметра возможны следующие два значения:

FF00H --> set bit - установка бита

0000H --> delete bit - сброс бита.

Пример применения

Преобразование MODBUS-адресов для функциональных кодов FC 01, 05, 15	
MODBUS-адрес в переданном сообщении	Область памяти SIMATIC
от 0 до 2047	Область меркеров начало с <i>M 1000.0</i>
от 2048 до 2559	Область выходов начало с <i>Q 256.0</i>

Сообщение-запрос для функции 05:

05H	Адрес ведомого устройства ADDR
05H	Функциональный код FUNC
08H	Начальный битовый адрес (coil) (старший байт)
09H	Начальный битовый адрес (coil) (младший байт) A257.1
FFH	DATA-on/off (Данные вкл/выкл) (старший байт)
00H	DATA-on/off (Данные вкл/выкл) (младший байт)
xxH	Код CRC-проверки (младший байт)
xxH	Код CRC-проверки (старший байт)

Ответное сообщение для функции 05:

05H	Адрес ведомого устройства ADDR
05H	Функциональный код FUNC
08H	Начальный битовый адрес (coil) (старший байт)
09H	Начальный битовый адрес (coil) (младший байт) A257.1
FFH	DATA-on/off (Данные вкл/выкл) (старший байт)
00H	DATA-on/off (Данные вкл/выкл) (младший байт)
xxH	Код CRC-проверки (младший байт)
xxH	Код CRC-проверки (старший байт)

Вычисление адреса

Начальный Modbus-адрес "coil_address" 0809H (2057 десятичное) размещается в области выходов:

Байт входов	= ((coil_address) - ccccc) / 8 + 00000
	= ((2057 - 2048) / 8) + 256
	= 257

Оставшееся число битов (Rest Bit_Number):

Оставшееся	= ((coil_address) - ccccc) % 8
число битов	= ((2057 - 2048) % 8)
	= 1

Таким образом доступ обеспечивается к выходу Q 257.1 включительно.

Дополнительные примеры

Несколько примеров операций доступа показаны в соответствующей таблице для FC 01

8.6 Функциональный код 06 (Preset Single Register) (Назначение для отдельного регистра)

Функция

Данная функция позволяет системе ведущего Modbus-устройства записывать (write) слово данных в блок данных CPU.

Сообщение-запрос

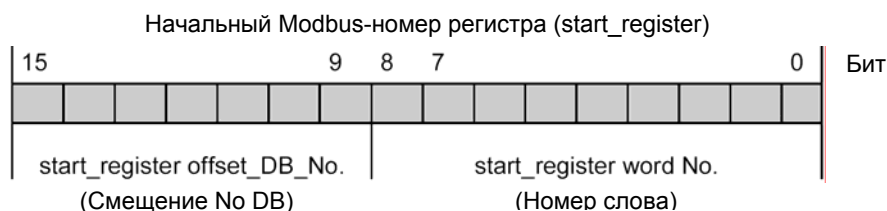
ADDR (адрес)	FUNC (код)	start_register (начальный регистр)	DATA value (High, Low) (младший/старший байт)	CRC (четность)
-----------------	---------------	---------------------------------------	--	-------------------

Ответное сообщение

ADDR (адрес)	FUNC (код)	start_register (начальный регистр)	DATA value (High, Low) (младший/старший байт)	CRC (четность)
-----------------	---------------	---------------------------------------	--	-------------------

Начальный регистр Start_register

Начальный Modbus-номер регистра "start_register" интерпретируется драйвером следующим образом:



Для последующей генерации адреса драйвер использует "базовый номер блока DB" ("base DB number") (начиная с DB xxxxx), который вводится в диалоговом окне "Conversion of MODBUS Addressing for FC 03, 06, 16" ("Преобразование MODBUS-адресов для FC FC 03, 06, 16").

Вычисление адреса (преобразование адреса) выполняется в два этапа, как показано ниже:

Доступ к SIMATIC-области	Формула преобразования
Блок данных DB (итоговый №)	= (Базовый номер DB xxxxx + + start_register offset_DB_No.)
Слово данных DBW	= (start_register word_No. × 2)

Формула вычисления начального регистра Start_register (начальный регистр)

После того, как становится известным итоговый № блока данных DB, из которого следует считывать данные, начальный Modbus-номер регистра "start_register" интерпретируется драйвером следующим образом:

$$\text{start_register} = ((\text{итоговый \# DB} - \text{базовый \# DB}) \times 512) + (\text{data word_DBW} / 2)$$

Этот адрес базируется только на четных номерах слов данных.

DATA value (значение для регистра)

Значения для записи в регистр могут быть любыми.

Пример применения

Пример назначения параметров:

Преобразование MODBUS-адресов для функциональных кодов FC 03, 06, 16	
MODBUS-адрес в переданном сообщении	Область памяти SIMATIC
0	Область блоков данных (базовый номер DB) начало с DB 800

Сообщение-запрос для функции 06:

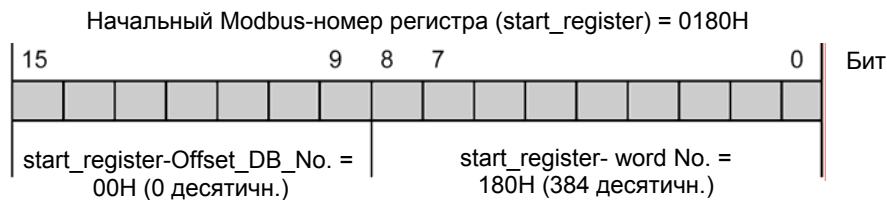
05H	Адрес ведомого устройства ADDR
06H	Функциональный код FUNC
01H	Начальный адрес регистра (старший байт)
80H	Начальный адрес регистра (младший байт) DBW 768
2BH	Значение регистра (старший байт)
1AH	Значение регистра (младший байт)
xxH	Код CRC-проверки (младший байт)
xxH	Код CRC-проверки (старший байт)

Ответное сообщение для функции 06:

05H	Адрес ведомого устройства ADDR
06H	Функциональный код FUNC
01H	Начальный адрес регистра (старший байт)
80H	Начальный адрес регистра (младший байт) DBW 768
2BH	Значение регистра (старший байт)
1AH	Значение регистра (младший байт)
xxH	Код CRC-проверки (младший байт)
xxH	Код CRC-проверки (старший байт)

Вычисление начального адреса регистра Start_register

Начальный Modbus-номер регистра "start_register" 0180H
(384 десятичн.) интерпретируется драйвером следующим образом:



Блок данных DB (итоговый №)	= (Базовый номер DB xxxxx + + start_register offset_DB_No.) = (800 + 0) = 800
Слово данных DBW	= (start_register word_No. × 2) = (384 × 2) = 768

Доступ производится к DB 800, слово данных DBW 768.

Дополнительные примеры

Несколько примеров операций доступа показаны в соответствующей таблице для FC 03 (см. выше).

8.7 Функциональный код 08 (Loop Back Diagnostic Test) (Тестирование соединительных линий)

Функция

Функция позволяет системе ведущего Modbus-устройства выполнять тестирование соединительных линий. Он не влияет на S7 CPU, на пользовательскую программу, на пользовательские данные. Сообщение ведущей системы драйвер возвращает обратно без изменений.

Сообщение-запрос

ADDR (адрес)	FUNC (код)	Diagnostic Code (High, Low) (диагностический код (старший/младший байт))	Test Data (данные теста)	CRC (четность)
-----------------	---------------	--	-----------------------------	-------------------

Ответное сообщение

ADDR (адрес)	FUNC (код)	Diagnostic Code (High, Low) (диагностический код (старший/младший байт))	Test Data (данные теста)	CRC (четность)
-----------------	---------------	--	-----------------------------	-------------------

Diagnostic Code (диагностический код)

Функция поддерживает только диагностический код 0000:

Test Data (данные теста)

Возможны любые значения (16 битов).

Сообщение-запрос для функции 08:

05H	Адрес ведомого устройства ADDR
08H	Функциональный код FUNC
00H	Diagnostic Code (диагностический код) (старший байт)
00H	Diagnostic Code (диагностический код) (младший байт)
A5H	Test Data (данные теста) (старший байт)
C3H	Test Data (данные теста) (младший байт)
xxH	Код CRC-проверки (младший байт)
xxH	Код CRC-проверки (старший байт)

Ответное сообщение для функции 08:

05H	Адрес ведомого устройства ADDR
08H	Функциональный код FUNC
00H	Diagnostic Code (диагностический код) (старший байт)
00H	Diagnostic Code (диагностический код) (младший байт)
A5H	Test Data (данные теста) (старший байт)
C3H	Test Data (данные теста) (младший байт)
xxH	Код CRC-проверки (младший байт)
xxH	Код CRC-проверки (старший байт)

8.8 Функциональный код 15 (Force Multiple Coil) (Назначение для нескольких выходов)

Функция

Данная функция обеспечивает для системы ведущего Modbus-устройства запись (write) в несколько битов в областях памяти SIMATIC, как показано ниже.

Сообщение-запрос

ADDR (адрес)	FUNC (код)	start_address (начальный адрес)	количество	байтовый счетчик	n-DATA	CRC (четность)
-----------------	---------------	------------------------------------	------------	---------------------	--------	-------------------

Ответное сообщение

ADDR (адрес)	FUNC (код)	start_address (начальный адрес)	количество	CRC (четность)
-----------------	---------------	------------------------------------	------------	-------------------

start_address (начальный адрес)

Начальный Modbus-адрес бита "start_address" интерпретируется драйвером следующим образом.

Драйвер проверяет, размещен ли начальный адрес "start_address" в одной из областей, которые были определены при назначении параметров в диалоговом окне "Conversion of MODBUS Addressing for FC 01, 05, 15" ("Преобразование MODBUS-адресов для FC 01, 05, 15") ("from/to" ("от/до"): "memory bits" ("меркеры"), "outputs" ("выходы")).

Если начальный MODBUS-адрес start_address располагается в области	обращение производится к следующей области памяти SIMATIC
от aaaaa до bbbbb	Область меркеров начало с M uuuu.0
от ccccc до ddddd	Область выходов начало с Q oooo.0

Адрес, к которому выполняется обращение, вычисляется следующим образом:

Начальный SIMATIC-адрес	Формула преобразования
Байт меркеров	$= ((start_address) - aaaaa) / 8 + uuuuu$
Байт выходов	$= ((start_address) - ccccc) / 8 + ooooo$

Доступ к областям меркеров и выходов

При доступе к SIMATIC-областям меркеров и выходов оставшееся число битов вычисляется и используется для адресации соответствующих битов внутри байта в областях меркеров и выходов.

Доступ к областям таймеров и счетчиков

При доступе к SIMATIC-областям таймеров и счетчиков операция доступа прерывается с сообщением об ошибке для функционального кода 15.

Количество (число битов)

Для этого параметра возможны любые значения от 1 до 1024.

Примечание

Ограничения, которые зависят от CPU, представлены в главе "Интерфейс CPU-CP".

DATA

Состояние битов (bit status) - для этого параметра возможны любые значения.

Пример применения

Преобразование MODBUS-адресов для функциональных кодов FC 01, 05, 15	
MODBUS-адрес в переданном сообщении	Область памяти SIMATIC
от 0 до 2047	Область меркеров начало с M 1000.0
от 2048 до 2559	Область выходов начало с Q 256.0

Пример

Если системе ведущего устройства требуется записать состояние "включено" ("ON") в меркеры M 1144.0 ... M 1144.7 и M 1145.0 ... M 1145.3 (при этом состояние "выключено" записываем как "OFF")

Меркер	7	6	5	4	3	2	1	0	Бит
M 1144	ON	ON	OFF	OFF	ON	ON	OFF	ON	

Меркер	7	6	5	4	3	2	1	0	Бит
M 1145	-	-	-	-	ON	OFF	OFF	ON	

Сообщение-запрос для функции 15:

05H	Адрес ведомого устройства ADDR
0FH	Функциональный код FUNC
04H	Начальный битовый адрес (старший байт)
80H	Начальный битовый адрес (младший байт) M 1144.0
00H	Количество (старший байт)
0CH	Количество (младший байт) (12 битов)
02H	Счетчик байтов
CDH	Состояние битов (M 1144.0... M 1144.7)
09H	Состояние битов (M 1145.0... M 1145.3)
xxH	Код CRC-проверки (младший байт)
xxH	Код CRC-проверки (старший байт)

Ответное сообщение для функции 15:

05H	Адрес ведомого устройства ADDR
0FH	Функциональный код FUNC
04H	Начальный битовый адрес (старший байт)
80H	Начальный битовый адрес (младший байт)
00H	Количество (старший байт)
0CH	Количество (младший байт)
xxH	Код CRC-проверки (младший байт)
xxH	Код CRC-проверки (старший байт)

Вычисление адреса

Начальный Modbus-адрес "coil_address" 0480H (1152 десятичное) размещается в области меркеров:

Байт меркеров	= ((coil_address) - aaaaa) / 8 + uuuu
	= ((1152 - 0) / 8) + 1000
	= 1144

Оставшееся число битов (Rest Bit_Number):

Оставшееся	= ((coil_address) - aaaaa) % 8
число битов	= ((1152 - 0) % 8)
	= 0

Таким образом доступ обеспечивается к меркерам, начиная с M 1144.0.

Дополнительные примеры

Несколько примеров операций доступа показаны в соответствующей таблице для FC 01

Формула вычисления начального регистра Start_register (начальный регистр)

После того, как становится известным итоговый № блока данных DB, из которого следует считывать данные, начальный Modbus-номер регистра "start_register" интерпретируется драйвером следующим образом:

$$\text{start_register} = ((\text{итоговый \# DB} - \text{базовый \# DB}) \times 512) + (\text{data word_DBW} / 2)$$

Этот адрес фиксируется только на четных номерах слов данных.

Количество

Значения между 1 и 127 допускаются в качестве количества регистров. Максимальное количество регистров, значения для которых могут быть предустановлены, вычисляется следующим образом:

$$(\text{Количество})_{\text{max}} = 512 - \text{start_register}$$

Примечание

Ограничения, которые зависят от CPU, представлены в главе "Интерфейс CPU-CP".

DATA

Для этого параметра возможны любые значения.

Пример применения

Пример назначения параметров:

Преобразование MODBUS-адресов для функциональных кодов FC 03, 06, 16	
MODBUS-адрес в переданном сообщении	Область памяти SIMATIC
0	Область блоков данных (базовый номер DB) начало с DB 800

Пример

Если системе ведущего MODBUS-устройства требуется записать значения CD09H, DE1AH, EF2BH в слова данных DBW 100, DBW 102, DBW 104 блока DB 800.

Сообщение-запрос для функции 16:

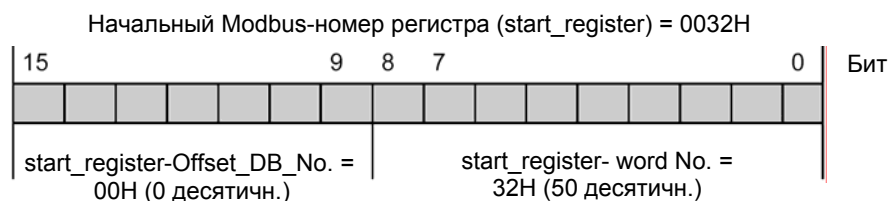
05H	Адрес ведомого устройства ADDR
10H	Функциональный код FUNC
00H	Начальный адрес регистра (старший байт)
32H	Начальный адрес регистра (младший байт) DBW 100
00H	Количество (старший байт)
03H	Количество (младший байт) (3 регистра)
06H	Байтовый счетчик
CDH	Значение регистра (старший байт) DBW 100
09H	Значение регистра (младший байт)
DEH	Значение регистра (старший байт) DBW 102
1AH	Значение регистра (младший байт)
EFH	Значение регистра (старший байт) DBW 104
2BH	Значение регистра (младший байт)
xxH	Код CRC-проверки (младший байт)
xxH	Код CRC-проверки (старший байт)

Ответное сообщение для функции 16:

05H	Адрес ведомого устройства ADDR
10H	Функциональный код FUNC
00H	Начальный адрес регистра (старший байт)
32H	Начальный адрес регистра (младший байт)
00H	Количество (старший байт)
03H	Количество (младший байт) (3 регистра)
xxH	Код CRC-проверки (младший байт)
xxH	Код CRC-проверки (старший байт)

Вычисление начального адреса регистра Start_register

Начальный Modbus-номер регистра "start_register" 0032H
(50 десятичн.) интерпретируется драйвером следующим образом:



Блок данных DB (итоговый №)	= (Базовый номер DB xxxxx + + start_register offset_DB_No.) = (800 + 0) = 800
Слово данных DBW	= (start_register word_No. × 2) = (50 × 2) = 100

Доступ производится к DB 800, слово данных DBW 100.

Дополнительные примеры

Несколько примеров операций доступа показаны в соответствующей таблице для FC 03 (см. выше).

9 Диагностика драйвера

Функции диагностики

Функции диагностики модуля CP позволяют оперативно определить место возникновения ошибки. Пользователю доступны следующие диагностические функции:

- диагностика посредством элементов индикации модуля CP
- диагностика посредством выходного параметра "состояний" STATUS функциональных блоков
- диагностика посредством сообщений об ошибках в области SYSTAT (только для CP 441-2)
- диагностика посредством диагностического буфера CP

Диагностика посредством элементов индикации модуля (светодиоды)

Элементы индикации показывают рабочее состояние и/или возможные ошибочные состояния модуля CP. Элементы индикации дают предварительную обзорную информацию внутренних или внешних ошибок, а также ошибок интерфейса.

Диагностика посредством выходного параметра "состояний" STATUS функциональных блоков

Каждый функциональный блок / системный функциональный блок имеет выходной параметр STATUS для диагностики ошибок в модуле CP. Считывая значения этого параметра, пользователь может получить информацию по ошибкам, которые возникают в соединениях. Выходной параметр STATUS может быть проверен в пользовательской программе.

Диагностика посредством сообщений об ошибках в области SYSTAT (только для CP 441-2)

Область SYSTAT в модуле CP 441-2 предназначена для хранения подробной информации о диагностических событиях и об обнаруженных ошибках. Вы можете считывать эту информацию из пользовательской программы, запрограммировав системный функциональный блок SYSTAT.

Диагностика посредством диагностического буфера CP

Все ошибки и диагностические события, описанные в разделе 7.3, вводятся в диагностический буфер CP. В руководстве по CP Вы можете найти информацию о том, как считывать записи из диагностического буфера.

9.1 Средства диагностики модуля CP 341

9.1.1 Диагностика с использованием элементов индикации CP 341

Введение

Элементы индикации на передней панели CP 341 выдают информацию о его состоянии. Имеются следующие функции индикации:

- **Индикатор групповых сообщений (Group alarm)**
 - SF (светится красным при отказе или при повторной параметризации)
- **Специальные индикаторы (Special)**
 - TXD (светится зеленым при передаче; загорается, когда CP 341 передает пользовательские данные через интерфейс)
 - RXD (светится зеленым при приеме; загорается, когда CP 341 принимает пользовательские данные через интерфейс)

Индикатор групповых сообщений (Group alarm) SF

Индикатор групповых сообщений (Group alarm) SF всегда загорается после включения питания (POWER ON) и выключается после инициализации. Если данные параметризации были сгенерированы для CP 341, светодиод SF вновь загорается на короткое время во время повторной параметризации.

Индикатор групповых сообщений (Group alarm) SF загорается при следующих событиях:

- Отказ оборудования
- Сбой микропрограммы
- Ошибка при параметризации или отсутствие параметризации
- Обрыв линии (размыкание сигнального кабеля между CP 341 и коммуникационным партнером)

9.1.2 Диагностические сообщения функциональных блоков CP 341

Введение

Каждый функциональный блок имеет параметр STATUS для диагностики ошибок. Номера сообщений об одних и тех же ошибках STATUS всегда имеют одинаковые значения, независимо от конкретного системного функционального блока.

Схема нумерации классов и номеров событий:

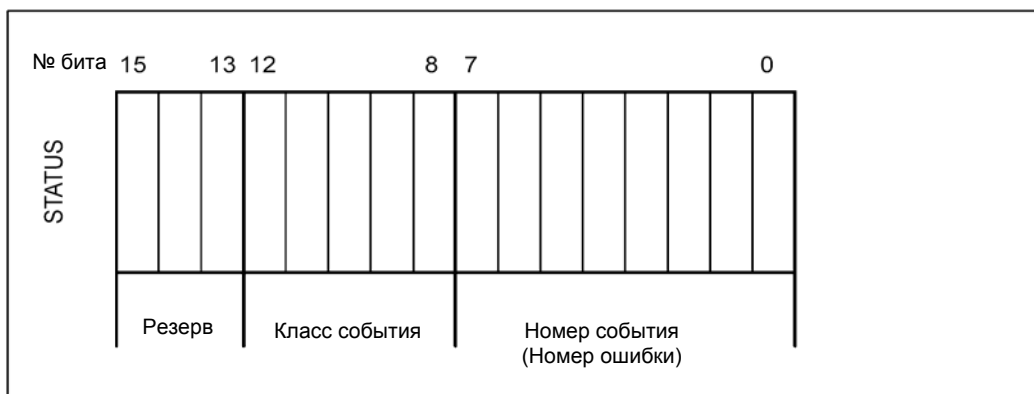


Рис. 9.1 Структура параметра STATUS

В разделе 9.3 представлен список отдельных диагностических событий / ошибок.

9.2 Средства диагностики модуля CP 441-2

9.2.1 Диагностика с использованием элементов индикации CP 441-2

Функции индикации

Элементы индикации на передней панели модуля CP 441-2 выдают информацию о его состоянии. Имеются следующие функции индикации:

- **Индикатор групповых сообщений (Group Error Displays)**
 - INTF (индикатор внутренней ошибки)
 - EXTf (индикатор внешней ошибки)
- **Специальные индикаторы (Special Displays)**
 - TXD (индикатор выполнения задания BSEND в CP)
 - RXD (индикатор передачи принятых CP данных в CPU)
- **Индикатор групповых сообщений (Group Error Displays)**
 - FAULT (индикатор ошибки интерфейса)

Индикация сообщений об ошибках

Сообщение об ошибке	Описание ошибки / диагностическое событие	Действия
Светодиод INTF светится	CP сообщает о внутренней ошибке; ошибка аппаратного или программного обеспечения	Запрограммируйте SFB STATUS для считывания подробной информации
Светодиод EXTf светится	CP 441-2 сообщает о внешней ошибке; например, об обрыве входных линий связи (BREAK)	Запрограммируйте SFB STATUS для считывания подробной информации
Светодиод FAULT не светится	Интерфейс готов к работе или интерфейсный submodule не вставлен в модуль	-
Светодиод FAULT редко мигает	Интерфейс инициализирован и готов к работе, но невозможно соединение посредством системной шины S7-400	Проверьте общую конфигурацию и коммуникацию данных в конфигурации проекта
Светодиод FAULT часто мигает	Некорректен параметр или сбой интерфейсного submodule (не совпадают параметры submodule и интерфейса)	Проверьте установки параметров в утилите для назначения параметров или в интерфейсном submodule
Светодиод FAULT светится	Нет доступа к параметрам интерфейса или существенная ошибка submodule (ошибка оборудования)	Задайте параметры с утилитой для назначения параметров или проверьте интерфейсный submodule

9.2.2 Диагностические сообщения системных функциональных блоков CP 441-2

Введение

В каждом системном функциональном блоке имеется параметр STATUS, предназначенный для диагностики ошибочных состояний. Номера сообщений об одних и тех же ошибках STATUS всегда имеют одинаковые значения, независимо от используемого системного блока. Наиболее важные для модуля CP STATUS-сообщения описываются в следующей ниже таблице. Вы можете найти полное описание STATUS-сообщений в справочном руководстве "Системное программное обеспечение для S7-300/400, Системные и стандартные функции" ("System Software for S7-300/400, System and Standard Functions").

Сообщения в выходном параметре STATUS блоков SFB

STATUS	Описание
0	Нет ошибок
1	Проблемы в соединении CP - CPU
2	Негативное квитирование, функция не может быть выполнена (например, партнер не отвечает или выдает негативное квитирование)
3	Неизвестный идентификатор R-ID; устройство недоступно
4	Некорректны число данных или отдельные типы данных
5	Принят запрос на выполнение сброса (RESET)
6	Удаленный блок недоступен (disabled)
7	Удаленный партнер находится в некорректном состоянии
8	Нет доступа к удаленному объекту; ошибка доступа в сервере (GET/PUT)
9	Предупреждение о переполнении (ERROR = 0): принятые данные были перезаписаны новыми значениями
10	Нет доступа к локальной памяти пользователя (например, удален DB)
11	Предупреждение (ERROR = 0): новое задание неактивно, так как пока не завершено предыдущее.
12	Экземпляр не совместим с системным вызовом; вызван не экземплярный, а обычный блок DB
13	Ошибка в описании формата
14	Указанная связь данных (относящаяся к приложению) не существует. Неизвестный ID (Вы должны задать локальный ID из данных коммуникаций в конфигурации проекта)
15	Связь данных сгенерирована в соответствии с указанным ID.
16	Связь данных не генерируется из-за недостатка ресурсов.

Отображение и проверка информации в выходном параметре STATUS

Информация в параметре STATUS системного функционального блока может быть отображена и считана с использованием таблицы переменных STEP 7.

Примечание

Считывание информации из области SYSTAT с помощью задания STATUS предоставит Вам подробную информацию по ошибкам / событиям, которые имели место во время коммуникационных связей CP, соответствующего CPU и подключенного коммуникационного партнера.

9.2.3 Считывание сообщений об ошибках из области SYSTAT для CP 441-2

Считывание информации с помощью коммуникационных FB

Сообщения об ошибках из области SYSTAT может быть считана с помощью **MODBUS коммуникационных FB**.

FB циклически считывают информацию из области SYSTAT и сохраняют данные в экземплярном DB, начиная со слова данных DBW 40.

(см. также главу "Отладка коммуникационных FB").

Примечание

Благодаря тому, что задание STATUS выполняется асинхронно по отношению к другим заданиям в сети, не предоставляется возможным разместить SFB для заданного R_ID. Это означает, что даже если можно отобразить информацию о произошедших ошибках в коммуникационных данных (data link), но невозможно вычислить, какой из вызовов SFB стал их причиной.

9.2.4 Диагностика посредством области для сообщений об ошибках SYSTAT для CP 441-2

Введение

Область для сообщений об ошибках SYSTAT - это область данных в CP 441-2, в которую вводятся все распознанные CP ошибки / диагностические события в деталях. Область SYSTAT содержит поля для информации о 6 событиях для каждого интерфейса, а также информацию о рабочих состояниях CP и состоянии собственно области SYSTAT.

Структура области SYSTAT

Информация о первых шести распознанных CP ошибках / событиях вводится в область SYSTAT. Последующие ошибки / события могут быть введены в область SYSTAT, только после того как область SYSTAT будет очищена.

Ошибки / события вводятся в параметр LOCAL следующим образом:

- Байт 0 Рабочее состояние CP (02H для RUN, 05H для отказа)
- Байт 1 Зарезервирован
- Байт 2 Бит 0 - F ошибка введена в SYSTAT
 Бит 1 - U переполнение (Error overflow)
 Бит 2 - B обрыв (BREAK)
- Байт 3 Зарезервирован
- Байты 4/5 Событие 1
- Байты 6/7 Событие 2
- Байты 8/9 Событие 3
- Байты 10/11 Событие 4
- Байты 12/13 Событие 5
- Байты 14/15 Событие 6

Очистка (Deleting) области SYSTAT

После того, как SYSTAT-информация была считана с помощью SFB STATUS, все SYSTAT-сообщения автоматически удаляются.

Схема нумерации



Рис. 9.2 Схема нумерации событий в области SYSTAT

Классификация классов и номеров событий представлена в следующих разделах, а также в руководстве "PtP-коммуникации CP 441-2" ("CP 441-2 Point-to-Point Communication").

Примечание

В отличие от стандартных драйверов классы и номера событий для SYSTAT-области частично изменены для использования с загружаемыми драйверами. В следующих разделах представлены все **измененные классы и номера событий** для их использования с загружаемыми драйверами.

Если событие не представлено в данном руководстве, Вы можете рассматривать его соответствующим стандартному применению и найти информацию по нему в руководстве по CP 441-2.

9.3 Таблицы событий (Events) / ошибок (Errors)

Классы событий

Определены следующие классы событий:

Классы событий	Описание	Справочник
1	Ошибки аппаратуры CP	Руководство по CP
2	Ошибки при инициализации	Руководство по CP
3	Ошибки при параметризации РВК	Руководство по CP
4	Ошибки при обмене данных CP - CPU	Руководство по CP
5	Ошибки в работе CPU	Руководство по CP / Описание драйвера
6	Ошибки при обработке задания партнера	Руководство по CP
7	Ошибки при передаче	Руководство по CP
8	Ошибки при приеме	Описание драйвера
9	Сообщение с кодом ошибки принято от коммуникационного партнера	Не используется
10	Ошибки распознаны CP в ответном сообщении партнера	Не используется
14	Общие ошибки обработки загружаемого драйвера	Описание драйвера

9.3.1 Коды ошибок в области SYSTAT для "CPU Job Errors" ("Ошибки в работе CPU")

Класс события 5 (05H): "Ошибки в работе CPU" ("CPU Job Errors ")			
Класс / номер события (HEX)	Класс / номер события (десятич.)	Текст события	Действия
(05)18H	24	Transmission length during transmission is too large (> 4 KB), or transmission length for SEND is too small. (Для передачи задан размер > 4 kb - слишком большой или слишком малый для параметров интерфейса)	Проверьте параметр LEN для функции SEND.

9.3.2 Номера ошибок в области SYSTAT для "Receive Errors" ("Ошибки при приеме")

Класс события 8 (08H): "Ошибки при приеме CPU" ("CPU Receive Errors ")			
Класс / номер события (HEX)	Класс / номер события (десятич.)	Текст события	Действия
(08)06H	6	Character delay time (ZVZ) exceeded. (Превышено время ожидания (задержки) символа)	Устраните ошибку в партнерском устройстве или помехи в линии передачи.
(08)0CH	12	Transmission error (parity error, overflow error, stop bit error (frame)) recognized in a character. (Обнаружена ошибка при передаче - в фрейме символа (ошибка четности, ошибка переполнения, ошибка стоповых битов)	Проверьте наличие помех в линии передачи. Если требуется, измените структуру системы и/или прокладку кабелей. Проверьте, одинаковы ли в CP и в партнере параметры протокола: скорость передачи, количество битов данных, четность, количество стоповых битов.

Класс события 8 (08H): "Ошибки при приеме CPU" ("CPU Receive Errors ")			
Класс / номер события (HEX)	Класс / номер события (десятич.)	Текст события	Действия
(08)0DH	13	BREAK Receiving line to partner device is interrupted. (Обрыв - нарушена целостность приемной линии связи с партнерским устройством)	Восстановите соединение между устройствами или включите устройство-партнер. При использовании ТТУ-соединения, проверьте ток в линии в состоянии покоя. При использовании RS422 / 485 (X27) - соединения, проверьте (измените, если требуется) назначение соединительных контактов 2-х проводной приемной линии R(A), R(B).
(08)30H	48	Broadcast not allowed with this function code. (Рассылка сообщения с этим функциональным кодом не разрешается)	Система ведущего Modbus -устройства допускает рассылку сообщений только с разрешенными для этого функциональными кодами.
(08)31H	49	Received function code not allowed. (Принятый функциональный код не допускается).	Принятый функциональный код не допускается для данного драйвера.
(08)32H	50	Maximum amount of bits or registers exceeded, or amount of bits cannot be divided by 16 when accessing SIMATIC memory areas timers or counters. (Максимальное количество битов или регистров превышено или количество битов не делится на 16 при попытке обращения к областям счетчиков / таймеров)	Максимальное количество битов - до 2040. Максимальное количество регистров - до 127. Доступ к областям счетчиков / таймеров - только с интервалом 16 битов.

Класс события 8 (08H): "Ошибки при приеме CPU" ("CPU Receive Errors ")			
Класс / номер события (HEX)	Класс / номер события (десятич.)	Текст события	Действия
(08)33H	51	Amount of bits or registers for function codes FC 15/16 and message element byte_count do not match. (Количество битов или регистров для функциональных кодов FC 15/16 и значение в параметре сообщения byte_count не равны).	Исправьте или количество битов или регистров, или значение в параметре сообщения byte_count.
(08)34H	52	Illegal bit coding recognized for "set bit / reset bit." (Неверное значение бита, распознанное при операции "Установка бита"/"Сброс бита")	Допускается только 0000H или FF00H для FC05.
(08)35H	53	Illegal diagnostic subcode (≠ 0000Hex) recognized for function code FC 08 "Loop Back Test." (Неверное значение диагностического кода (≠ 0000H), распознанное для кода FC 08 (тестирование линии передачи))	Используйте только код 0000H для FC 08.
(08)36H	54	The internally-generated value of the CRC 16 checksum does not match the received CRC checksum. (Сгенерированное значение контрольной суммы не равно переданному в сообщении значению CRC)	Проверьте генерацию значения контрольной суммы в системе ведущего ModBus-устройства.
(08)37H	55	Message sequence error: The Modbus master system sent a new request message before the last reply message was transferred by the driver. (Ошибка в последовательности сообщений: Система ведущего ModBus-устройства послала новый запрос-сообщение до того, как было получено ответное сообщение от драйвера).	Увеличьте параметр время ожидания системой ведущего ModBus-устройства ответного сообщения от ведомого устройства.

9.3.3 Номера ошибок в области SYSTAT для "General Processing Errors" ("Общие ошибки обработки")

Класс события 14 (0EH): "Загружаемый драйвер - Общие ошибки обработки" ("Loadable Driver - General Processing Errors")			
Класс / номер события (HEX)	Класс / номер события (десятич.)	Текст события	Действия
(0E)01H	1	Error during initialization of the driver-specific SCC process (Ошибка при инициализации определенного драйвером SCC-процесса)	Переопределите параметры драйвера и перезагрузите его
(0E)02H	2	Error during startup of driver: Wrong SCC process active (SCC driver). The driver cannot function with this SCC driver. (Ошибка при запуске драйвера: активен не тот процесс (SCC-драйвер). Драйвер не работает с этим SCC-драйвером)	Переопределите параметры драйвера и перезагрузите его
(0E)03H	3	Error during startup of driver: Wrong data transfer process active (interface to SFBs). The driver cannot function with this data transfer process. (Ошибка при запуске драйвера: активен не тот процесс передачи (интерфейс с SFC). Драйвер не работает с этим процессом передачи)	Переопределите параметры драйвера и перезагрузите его
(0E)04H	4	Error during startup of driver: Illegal interface submodule. The driver cannot run with the parameterized interface submodule. (Ошибка при запуске драйвера: недопустимый интерфейсный submodule. Драйвер не работает с этим submodule)	Проверьте и исправьте назначенные параметры
(0E)05H	5	Error with driver dongle: No dongle plugged in, or inserted dongle is faulty. The driver is not ready to run. (Ошибка из-за защитного ключа-заглушки драйвера: ключ-заглушка отсутствует или неисправен. Драйвер не готов к работе)	Проверьте наличие защитного ключа-заглушки для драйвера в CP. Замените ключ-заглушку на исправный.
(0E)06H	6	Error with driver dongle: The dongle has no valid contents. The driver is not ready to run. (Ошибка из-за защитного ключа-заглушки драйвера: ключ-заглушка имеет некорректное содержание. Драйвер не готов к работе)	Получите подходящий защитный ключ-заглушку загружаемого драйвера для CP из представительства Siemens.
:	:		

Класс события 14 (0EH): "Загружаемый драйвер - Общие ошибки обработки" ("Loadable Driver - General Processing Errors")			
Класс / номер события (HEX)	Класс / номер события (десятич.)	Текст события	Действия
(0E)10H	16	Internal error procedure: default branch in Send automatic device. (Внутренняя ошибка процедуры: ошибка ветви процедуры в автоматическом передающем устройстве)	Перезапустите CP (включите электропитание)
(0E)11H	17	Internal error procedure: default branch in Receive automatic device. (Внутренняя ошибка процедуры: ошибка ветви процедуры в автоматическом приемном устройстве)	Перезапустите CP (включите электропитание)
(0E)12H	18	Internal error active automatic device: default branch. (Внутренняя ошибка активного автоматич. устройства: ошибка ветви)	Перезапустите CP (включите электропитание)
(0E)13H	19	Internal error passive automatic device: default branch. (Внутренняя ошибка пассивного автоматич. устройства: ошибка ветви)	Перезапустите CP (включите электропитание)
(0E)20H	32	For this data link, the amount of data bits must be set to 8. The driver is not ready to run. (В этих данных связи количество битов данных должно быть 8. Драйвер не готов к работе)	Проверьте и исправьте назначенные параметры драйвера
(0E)21H	33	The multiplication factor set for the character delay time is not within the value range of 1 to 10. The driver is operating with a default setting of 1. (Множитель для времени ожидания символа вне пределов 1...10. Драйвер работает со значением по умолчанию 1)	Проверьте и исправьте назначенные параметры драйвера
(0E)22H	34	The operating mode set for the driver is illegal. "Normal operation" or "Interference Suppression" must be specified. The driver is not ready to run. (Рабочий режим для драйвера недопустим: должен быть задан "нормальный" режим или режим "с подавлением помех". Драйвер не готов к работе)	Проверьте и исправьте назначенные параметры драйвера
(0E)23H	35	An illegal value has been set for the slave address. Slave address 0 is not allowed. The driver is not ready to run. (Задан неверный адрес ведомого устройства. Адрес 0 не допускается. Драйвер не готов к работе)	Проверьте и исправьте назначенные параметры драйвера

Класс события 14 (0EH): "Загружаемый драйвер - Общие ошибки обработки" ("Loadable Driver - General Processing Errors")			
Класс / номер события (HEX)	Класс / номер события (десятич.)	Текст события	Действия
(0E)24H	36	Illegal limitations have been set for write access. The driver is not ready to run. (Заданы неверные границы для записи. Драйвер не готов к работе)	Проверьте и исправьте назначенные параметры драйвера
(0E)25H	37	An illegal "from/to" combination has been set for the input of areas "Conversion of Modbus Addressing for FC 01,05,15." (Areas memory bits, outputs, timers, counters). The driver is not ready to run. (Задана неверно комбинация "от/до" для преобразования адресации для FC01, 05, 15: меркеры, выходы, таймеры, счетчики. Драйвер не готов к работе)	Проверьте и исправьте назначенные параметры драйвера
(0E)26H	38	An illegal "from/to" combination has been set for the input of areas "Conversion of Modbus Addressing for FC 02" (Areas memory bits, inputs). The driver is not ready to run. (Задана неверно комбинация "от/до" для преобразования адресации для FC 02: меркеры, входы. Драйвер не готов к работе).	Проверьте и исправьте назначенные параметры драйвера
(0E)27H	39	An overlap "from/to" combination has been set for the input of areas "Conversion of Modbus Addressing for FC 01, 05, 15." (Areas memory bits, outputs, timers, counters). The driver is not ready to run. (Задана с перекрытием комбинация "от/до" для преобразования адресации для FC 01, 05, 15: меркеры, выходы, таймеры, счетчики. Драйвер не готов к работе)	Проверьте и исправьте назначенные параметры драйвера
(0E)28H	40	An illegal "from/to" combination has been set for the input of areas "Conversion of Modbus Addressing for FC 02" (Areas memory bits, inputs). The driver is not ready to run. (Задана с перекрытием комбинация "от/до" для преобразования адресации для FC 02: меркеры, входы. Драйвер не готов к работе).	Проверьте и исправьте назначенные параметры драйвера
(0E)2EH	46	An error occurred when reading the interface parameter file. The driver is not ready to run. (Ошибка при считывании файла параметров интерфейса. Драйвер не готов к работе)	Перезапустите CP (включите электропитание)

Класс события 14 (0EH): "Загружаемый драйвер - Общие ошибки обработки" ("Loadable Driver - General Processing Errors <CPU - CP>")			
Класс / номер события (HEX)	Класс / номер события (десятич.)	Текст события	Действия
(0E)30H	48	Internal error during data transfer to CPU: Unexpected acknowledgment Passive. (Внутренняя ошибка при передаче в CPU: неожиданный ответ "Пассивное состояние")	Игнорируйте, если это состояние не является постоянным
(0E)31H	49	Timeout during data transfer to CPU. (Прерывание при передаче данных в CPU)	Проверьте интерфейс CP - CPU
(0E)32H	50	Error occurred during data transfer to CPU with RCV: Exact failure reason (detailed error) is in SYSTAT before this entry. (Ошибка при передаче данных в CPU посредством RCV: Точная причина ошибки (подробно) в области SYSTAT перед данным вводом)	Проверьте интерфейс CP - CPU
(0E)33H	51	Internal error during data transfer to CPU: Illegal status of automatic device. (Внутренняя ошибка при передаче данных в CPU: недопустимое состояние автоматического устройства)	Проверьте интерфейс CP - CPU
:	:	:	:
(0E)38H	56	Error occurred when accessing one of the SIMATIC areas "memory bits, outputs, timers, counters, inputs" with function codes FC 01 or FC 02: for example, input does not exist, or read attempt in excess of range end. (Ошибка при обращении к одной из областей: меркеров, выходов, таймеров, счетчиков, входов при кодах FC 01 или FC 02, например, вход не существует или попытка чтения за пределами диапазона)	Проверьте существует ли соответствующая область памяти, проверьте границы областей
(0E)39H	57	Error occurred when accessing SIMATIC area "data block" with function codes FC 03, 04, 06, 16: Data block does not exist or is too short. (Ошибка при обращении к области DB с FC 03, 04, 06, 16: блок DB не существует или слишком короток)	Проверьте, существует ли блок DB и каков его размер.
(0E)3AH	58	Error occurred when executing a write job with function codes FC 05, 15: Instance data block of Modbus FB does not exist or is too short. (Ошибка при записи данных с FC 05, 15: экземплярный DB Modbus-блока FB не существует или слишком короток)	Проверьте, существует ли экземплярный DB Modbus коммуникационного блока FB и каков его размер.

Класс события 14 (0EH): "Загружаемый драйвер - Общие ошибки обработки" ("Loadable Driver - General Processing Errors <CPU - CP>")			
Класс / номер события (HEX)	Класс / номер события (десятич.)	Текст события	Действия
(0E)3BH	59	Timeout during execution of a write job by Modbus communications FB. (Окончание времени на выполнение записи с помощью Modbus коммуникационного блока FB)	Проверьте конфигурацию коммуникационных данных проекта и интерфейс CP - CPU (SFB SEND): возможно перезагрузите коммуникационный FB
(0E)3CH	60	Illegal job with this driver. (Недопустимое задание для драйвера)	Допустимы только задания: SFB BSEND, RCV, STATUS (только для CP 441-2).

Класс события 14 (0EH): "Загружаемый драйвер - Общие ошибки обработки" ("Loadable Driver - General Processing Errors <ошибки приема>")			
Класс / номер события (HEX)	Класс / номер события (десятич.)	Текст события	Действия
(0E)50H	80	The rest bit number resulting from the Modbus address is $\neq 0$ for the word-orientated SIMATIC areas timers / counters. (Оставшееся число битов от деления на 16 не равно 0: для счетчиков/таймеров)	Используйте корректные Modbus-адреса
(0E)51H	81	The received Modbus address is outside the parameterized "from/to" areas. (see section "Assigning Parameters to the Loadable Driver"). (Принятый Modbus-адрес вне области заданной с "от/до"-границами. (См. раздел "Назначение параметров для загружаемого драйвера"))	Используйте только адреса, указанные в сообщении-запросе, предварительно определенные при назначении параметров.
(0E)52H	82	SIMATIC range limitation exceeded during access attempt by Modbus master system: Resulting DB number < 1 , or write access to an area which has not been enabled (parameter assignment), or write access to instance DB of the communications FB. (Границы области SIMATIC нарушены при попытке доступа из Modbus-системы: итоговый номер DB < 1 , или попытка записи в недоступную область, или попытка записи в область экземплярного DB коммуникационного FB)	Обеспечьте ограничение доступа в допустимых областях памяти SIMATIC

Класс события 14 (0EH): "Загружаемый драйвер - Общие ошибки обработки" ("Loadable Driver - General Processing Errors <ошибки приема>")			
Класс / номер события (HEX)	Класс / номер события (десятич.)	Текст события	Действия
(0E)53H	83	SIMATIC range limitation exceeded during access attempt by Modbus master system, for example, overflow when generating the resulting DB number (> 65535). (Границы области SIMATIC нарушены при попытке доступа из Modbus-системы: например, переполнение при генерации итогового номера DB (>65535))	Обеспечьте ограничение доступа в допустимых областях памяти SIMATIC
(0E)54H	84	Access in excess of parameterized range end, or access in excess of SIMATIC range end. (Границы заданной области (границы области SIMATIC) нарушены при попытке доступа)	Обеспечьте ограничение доступа в допустимых областях памяти SIMATIC
(0E)55H	85	Write access to this SIMATIC memory area is not allowed. (Доступ в режиме записи в области SIMATIC недопустим)	Организируйте запись только в области меркеров и выходов SIMATIC.
(0E)56H	86	Data link operation not possible because communications FB not running. (Работа ПО DataLink невозможна, так как коммуникационный FB не запущен)	Организируйте циклический вызов коммуникационного FB в программе пользователя. При необходимости вновь инициализируйте коммуникационный FB.
(0E)57H	87	Error occurred in communications FB during processing of the Modbus function code. (Ошибка коммуникационного FB при обработке функционального кода Modbus)	Сделайте анализ причин ошибки в соответствии с описанием в главе "Диагностика коммуникационных блоков FB"

10 Диагностика коммуникационного FB

Функции диагностики

Modbus коммуникационный FB имеет следующие два выходных параметра, которые индицируют случившиеся ошибки:

- Параметр "ERROR_NR"
- Параметр "ERROR_INFO"

ERROR_NR, ERROR_INFO

В параметре ERROR_NR индицируются обнаруженные ошибки. Более подробная информация об ошибках отображается в выходном параметре ERROR_INFO.

Удаление информации об ошибках

Удаление информации об ошибках производится передним фронтом сигнала на входе CP_START. При необходимости удаление информации об ошибках может быть выполнено пользователем в любое время.

10.1 Диагностика посредством ERROR_NR, ERROR_INFO

ERROR_No 1...9 Ошибки при инициализации FB и CP

Номера ошибок 1...9 показывают ошибки при инициализации FB и CP. При наличии такой ошибки параметр CP_START_ERROR = 1. MODBUS-коммуникации с системой ведущего (master) устройства невозможны.

ERROR_No 10...19 Ошибки при обработке функционального кода

Номера ошибок 10...19 показывают ошибки при обработке функционального кода: CP передал недопустимое для выполнения задание в коммуникационный FB. Информация об ошибке передается также в драйвер. Последующие задания продолжают выполняться.

ERROR_No 90...99 Другие ошибки

Ошибки обработки. Информация о такой ошибке не передается в драйвер. Последующие задания продолжают выполняться.

10.1.1 Ошибки при инициализации

"Ошибки при инициализации"			
ERROR_Nr Номер ошибки (десятич.)	ERROR_ INFO	Текст события	Действия
0	0	нет ошибок	
1	SFC51 -> RET_VAL	Ошибка при считывании SZL с помощью SFC51.	Проанализируйте RET_VAL в ERROR_INFO, устраните причину.
2	SFB12 -> STATUS, SFB22 -> STATUS	Истекло время для инициализации CP или ошибка при инициализации CP. (Ошибка при выполнении задания BSEND).	Проверьте назначение параметров для протокола для ведомого (slave) Modbus-устройства. Проверьте правильность назначенного ID для коммуникационного FB. Проанализируйте ERROR_INFO.

10.1.2 Ошибки при обработке функционального кода

"Ошибки при обработке функционального кода"			
ERROR_Nr Номер ошибки (десятич.)	ERROR_INFO	Текст события	Действия
10	Processing Code (Обработка кода)	Illegal processing function transferred by the driver to the communications FB. (Недопустимый функциональный код передан драйвером коммуникационному FB)	Перезапустите CP. (Выключите, затем включите электропитание)
11	Start Address (Начальный адрес)	Illegal start address transferred by the driver to communications FB. (Недопустимый начальный адрес передан драйвером коммуникационному FB)	Проверьте Modbus-адрес в системе ведущего Modbus-устройства.

"Ошибки при обработке функционального кода"			
ERROR_Nr Номер ошибки (десятич.)	ERROR_INFO	Текст события	Действия
12	Amount of Registers (Количество регистров)	Illegal amount of registers transferred by the driver to communications FB: Amount of registers = 0. (Недопустимое количество регистров передано драйвером коммуникационному FB: количество регистров = 0)	Проверьте количество регистров в системе ведущего Modbus-устройства. (Выключите, затем включите электропитание)
13	Amount of Registers (Количество регистров)	Illegal amount of registers transferred by the driver to communications FB: Amount of registers > 128. (Недопустимое количество регистров передано драйвером коммуникационному FB: количество регистров > 128)	Проверьте количество регистров в системе ведущего Modbus-устройства. (Выключите, затем включите электропитание)
14	Memory bits M - End Address (Меркеры - конец области)	Attempted access to SIMATIC memory area "memory bits" in excess of range end. Attention: Range length in SIMATIC CPU is CPU type-dependent. (Попытка доступа к SIMATIC-области меркеров за ее пределами. Внимание: размер области зависит от типа выбранного CPU)	Уменьшите начальный Modbus-адрес и/или доступный размер области в системе ведущего Modbus-устройства.
15	Outputs Q - End Address (Выходы - конец области)	Attempted access to SIMATIC memory area "outputs" in excess of range end. Attention: Range length in SIMATIC CPU is CPU type-dependent. (Попытка доступа к SIMATIC-области выходов за ее пределами. Внимание: размер области зависит от типа выбранного CPU)	Уменьшите начальный Modbus-адрес и/или доступный размер области в системе ведущего Modbus-устройства.
16	Timers T - End Address (Таймеры - конец области)	Attempted access to SIMATIC memory area "timers" in excess of range end. Attention: Range length in SIMATIC CPU is CPU type-dependent. (Попытка доступа к SIMATIC-области таймеров за ее пределами. Внимание: размер области зависит от типа выбранного CPU)	Уменьшите начальный Modbus-адрес и/или доступный размер области в системе ведущего Modbus-устройства.
17	Counters C - End Address (Счетчики - конец области)	Attempted access to SIMATIC memory area "counters" in excess of range end. Attention: Range length in SIMATIC CPU is CPU type-dependent. (Попытка доступа к SIMATIC-области счетчиков за ее пределами. Внимание: размер области зависит от типа выбранного CPU)	Уменьшите начальный Modbus-адрес и/или доступный размер области в системе ведущего Modbus-устройства.

"Ошибки при обработке функционального кода"			
ERROR_Nr Номер ошибки (десятич.)	ERROR_INFO	Текст события	Действия
18	0	Illegal SIMATIC memory area transferred by the driver to communications FB. (Недопустимый тип области SIMATIC передан драйвером коммуникационному FB)	Перезапустите CP. (Выключите, затем включите электропитание)
19		Error during access to SIMATIC I/Os. (Ошибка доступа к периферийным входам/выходам (I/O) SIMATIC)	Проверьте наличие требуемых периферийных входов/выходов (I/O) и их исправность

10.1.3 Другие ошибки

"Другие ошибки"			
ERROR_Nr Номер ошибки (десятич.)	ERROR_INFO	Текст события	Действия
90	SFB12 -> STATUS	Error during transmission of an acknowledgment message to the driver with SFB12 (BSEND). (Ошибка при передаче подтверждающего сообщения в драйвер с SFB12 (BSEND))	Проанализируйте информацию в параметре STATUS.
91	SFB22 -> STATUS	Error when reading SYSTAT with SFB22 (STATUS). (Ошибка при считывании информации из области SYSTAT с SFB22 (BSEND))	Проанализируйте информацию в параметре STATUS.
92	FB7->STATUS	Error when executing a RECEIVE / FETCH call with FB7 (RCV_RK). (Ошибка при выполнении вызова задания на ПРИЕМ / ВЫБОРКУ с FB7 (RCV_RK))	Проанализируйте информацию в параметре STATUS.

А Технические данные

Значения времени пересылки

Далее в таблицах указываются измеренные значения времени, требуемые для передачи данных, для различных функциональных кодов: Для измерения временных характеристик передачи были использованы станция S7-300 с CPU 315-2DP (6ES7 315-2AF01-0AB0) и CP 341 и как партнер - станция S7-400 с CPU 414 (6ES7 414-1XG01-0AB0) и CP 441-2.

В каждом эксперименте время обработки измерялось:

- от момента инициации задания в пользовательской программе и включая время обработки в ведущем (master) устройстве,
- как время, требуемое для пересылки данных партнеру через последовательный интерфейс,
- как время обработки в ведомом (slave) устройстве,
- как время, требуемое для пересылки квитирования партнеру через последовательный интерфейс.

При вычислении полного времени передачи все указанные выше компоненты складываются. Если в качестве партнерского устройства Вы используете другое (ведущее (master) или ведомое (slave)), при вычислении полного времени передачи Вы должны использовать те компоненты времени, которые соответствуют варианту вместо табличных значений. Значения времени для выполнения задания и квитирования не изменяются; они зависят только от используемой скорости передачи.

Ведомое (slave) устройство CP 341
Функциональный код 1 (Read) - считывание выхода (в мс)

Скорость передачи (бит/с)	300			
Пользовательские данные	Ведущий CP 441-2	Задание (Job)	Ведомый CP 341	Квитиро- вание
1 байт	229	257	179	184
10 байтов	229	257	179	514
20 байтов	229	257	180	882
50 байтов	232	257	182	1986
100 байтов	236	257	192	3824
200 байтов	243	257	208	7501
255 байтов	251	257	214	9487

Скорость передачи (бит/с)	9600			
Пользовательские данные	Ведущий CP 441-2	Задание (Job)	Ведомый CP 341	Квитиро- вание
1 байт	74	8	18	6
10 байтов	75	8	19	16
20 байтов	77	8	19	27
50 байтов	83	8	24	62
100 байтов	90	8	34	119
200 байтов	92	8	48	235
255 байтов	95	8	56	296

Скорость передачи (бит/с)	76800			
Пользовательские данные	Ведущий CP 441-2	Задание (Job)	Ведомый CP 341	Квитиро- вание
1 байт	73	1	13	1
10 байтов	74	1	13	2
20 байтов	76	1	13	3
50 байтов	86	1	20	8
100 байтов	93	1	29	15
200 байтов	95	1	45	29
255 байтов	97	1	50	37

Ведомое (slave) устройство CP 341
Функциональный код 15 (Write) - запись в несколько выходов (в мс)

Скорость передачи (бит/с)	300			
Пользовательские данные	Ведущий CP 441-2	Задание (Job)	Ведомый CP 341	Квитиро- вание
1 байт	205	331	199	257
10 байтов	206	662	200	257
20 байтов	206	1028	201	257
50 байтов	208	2132	212	257
100 байтов	211	3971	223	257
200 байтов	217	7648	238	257
255 байтов	221	9634	243	257

Скорость передачи (бит/с)	9600			
Пользовательские данные	Ведущий CP 441-2	Задание (Job)	Ведомый CP 341	Квити- рование
1 байт	48	10	41	8
10 байтов	48	20	41	8
20 байтов	50	32	43	8
50 байтов	52	67	48	8
100 байтов	55	124	56	8
200 байтов	63	239	74	8
255 байтов	67	301	88	8

Скорость передачи (бит/с)	76800			
Пользовательские данные	Ведущий CP 441-2	Задание (Job)	Ведомый CP 341	Квити- рование
1 байт	58	1	40	1
10 байтов	61	3	43	1
20 байтов	62	4	43	1
50 байтов	63	8	44	1
100 байтов	64	15	50	1
200 байтов	66	30	69	1
255 байтов	68	38	85	1

Ведомое (slave) устройство CP 441-2
Функциональный код 1 (Read) - считывание выхода (в мс)

Скорость передачи (бит/с)	300			
Пользовательские данные	Ведущий CP 341	Задание (Job)	Ведомый CP 441-2	Квитирование
1 байт	236	257	188	184
10 байтов	236	257	190	515
20 байтов	238	257	190	882
50 байтов	244	257	193	1986
100 байтов	280	257	199	3824
200 байтов	286	257	207	7502
255 байтов	288	257	216	9487

Скорость передачи (бит/с)	9600			
Пользовательские данные	Ведущий CP 341	Задание (Job)	Ведомый CP 441-2	Квитирование
1 байт	33	8	40	6
10 байтов	33	8	43	16
20 байтов	35	8	44	28
50 байтов	42	8	45	62
100 байтов	56	8	56	120
200 байтов	75	8	64	235
255 байтов	82	8	77	296

Скорость передачи (бит/с)	76800			
Пользовательские данные	Ведущий CP 341	Задание (Job)	Ведомый CP 441-2	Квитиро- вание
1 байт	35	1	23	1
10 байтов	36	1	25	2
20 байтов	37	1	26	3
50 байтов	46	1	27	8
100 байтов	61	1	30	15
200 байтов	82	1	39	29
255 байтов	92	1	48	37

Ведомое (slave) устройство CP 441

Функциональный код 15 (Write) - запись в несколько выходов (в мс)

Скорость передачи (бит/с)	300			
Пользовательские данные	Ведущий CP 341	Задание (Job)	Ведомый CP 441-2	Квитиро- вание
1 байт	225	331	223	257
10 байтов	227	662	224	257
20 байтов	227	1030	228	257
50 байтов	227	2132	232	257
100 байтов	229	3971	236	257
200 байтов	230	7648	243	257
255 байтов	237	9634	255	257

Скорость передачи (бит/с)	9600			
Пользовательские данные	Ведущий CP 341	Задание (Job)	Ведомый CP 441-2	Квитиро- вание
1 байт	64	11	62	8
10 байтов	64	21	63	8
20 байтов	69	32	64	8
50 байтов	69	67	68	8
100 байтов	72	124	70	8
200 байтов	75	239	76	8
255 байтов	75	301	86	8

Скорость передачи (бит/с)	76800			
Пользовательские данные	Ведущий CP 341	Задание (Job)	Ведомый CP 441-2	Квитиро- вание
1 байт	60	1	56	1
10 байтов	60	3	58	1
20 байтов	62	4	58	1
50 байтов	64	9	60	1
100 байтов	67	16	67	1
200 байтов	72	30	77	1
255 байтов	77	38	84	1

Требования к памяти

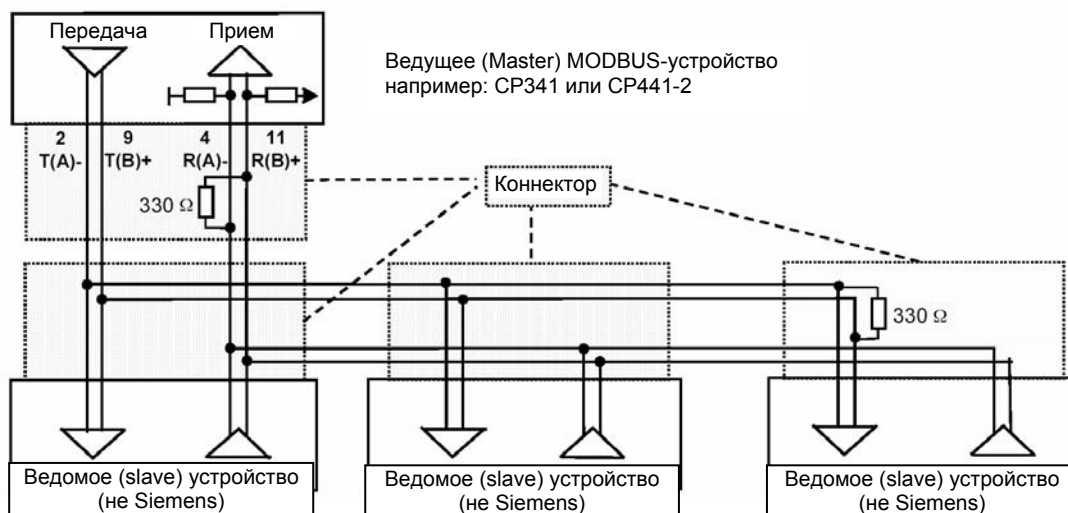
В следующей таблице отражены требования к памяти для функционального блока FB80 для коммуникационного процессора CP 341 и для функционального блока FB180 для коммуникационного процессора CP 441 в байтах.

Требования к памяти для функциональных блоков FB 7 и FB 8 могут быть найдены в руководстве по CP 341.

Блок	Имя	Версия	Загрузочная (load) память, (байтов)	Рабочая (work) память, (байтов)	Локальные данные, (байтов)
FB 80	MODB_341	1.0	2770	2034	104
FB 180	MODB_441	1.0	2982	2360	102

В Схемы многоточечных соединений

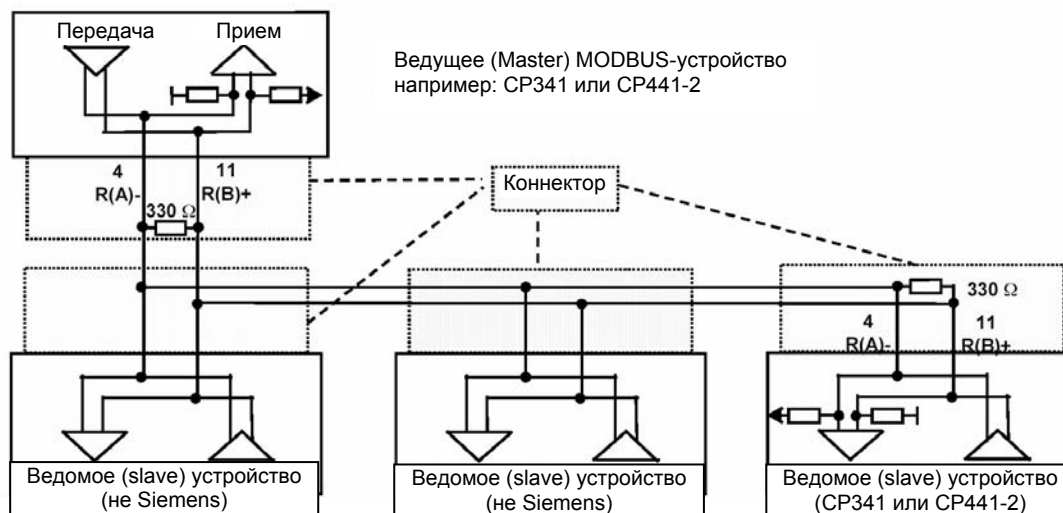
Схема многоточечного соединения RS422 (MODBUS)



Внимание

В режиме RS422 модули CP341 и CP441-2 могут использоваться только как ведущие (Master) устройства, потому что входные (receive) линии могут иметь три состояния ("Tri State")!

Схема многоточечного соединения RS485 (MODBUS)



Следующее касается модулей обоих типов:

- "Земляной" контакт (GND) (контакт 8 для CP341 / CP441-2) должен быть всегда подключен на обоих концах кабеля.
- Везде должны быть использованы экранированные кобуха.
- На коннекторе должен быть распаян оконечный резистор (примерно 330 Ом) в последнем приемном устройстве (receiver) всей последовательности узлов.
- Рекомендуется использовать следующий тип кабеля:
LIYCY 3 x 2 x 0,14 R(A)/R(B) и T(A)/T(B) (витая пара).
- Схема с фиктивным модулем ("Stub") не допускается.

С Справочная литература

Протокол Modbus (Modbus Protocol)

/1/ Gould Modbus Protocol Reference Guide PI-MBUS-300 Rev B GOULD Electronics

Справочное руководство *Протокол Gould Modbus PI-MBUS-300 выпуск B*
(GOULD Electronics)

Словарь терминов

А

Автономный режим (Офлайн)

Термин обозначающий режим работы с данными. Режим офлайн (или автономный режим) означает отсутствие обмена данными между PLC и программатором.

Адрес

Адрес определяет физическое место расположения данных. Если адрес известен, то возможен прямой доступ к операнду, на который адрес указывает.

Б

Блок

Блоки - это элементы пользовательской программы, которые характеризуются своей функцией, структурой или назначением. В STEP 7 различают

- кодовые блоки (FB, FC, OB, SFB, SFC)
- блоки данных (DB, SDB)
- типы данных, определенные пользователем (UDT)

Блок данных (DB)

Блоки данных - это блоки, содержащие данные и параметры, которые используются в пользовательской программе. В отличие от других блоков, блоки данных не содержат инструкций. Они подразделяются на глобальные (общие) блоки данных и экземплярные блоки данных. Доступ к данным, содержащимся в этих блоках, может быть абсолютным или символьным. Сложные данные могут храниться в структурированных формах.

В

Время цикла (Cycle Time)

Время цикла (cycle time) - это время, необходимое CPU для однократного сканирования пользовательской программы.

Вызов блока

Вызов блока происходит, когда ход выполнения основной программы временно оставляется для того, чтобы выполнить программу вызванного блока.

Д

Диагностические события (Diagnostic Events)

Диагностические события - это, например, ошибки в модуле или системные ошибки в CPU, которые вызваны, скажем, ошибкой в программе или изменением рабочего режима.

Диагностические функции

Диагностические функции охватывают целую систему средств диагностики, включающую в себя функции детектирования, анализа и генерации отчетов об ошибках в PLC.

Диагностический буфер

Каждый CPU имеет диагностический буфер, в котором хранится подробная информация о диагностических событиях в порядке их возникновения. Модуль CP 341 имеет свой диагностический буфер, в который вводятся все диагностические события CP 341 (ошибки оборудования и микропрограммы, ошибки инициализации и параметризации, ошибки при приеме и при передаче).

З

Загрузка (Download)

Термин загрузка означает загрузку загружаемых объектов (например, кодовых блоков) из программатора в загружаемую память (load memory) CPU.

И

Интерактивная справка (On-line Help)

STEP 7 обеспечивает пользователя контекстно отображаемой справочной информацией при работе с программным обеспечением.

Интерактивный режим (On-line)

Термин обозначающий режим работы с данными. Режим онлайн (или интерактивный режим) означает наличие обмена данными между PLC и программатором.

Интерфейсный submodule

Интерфейсный submodule модуля CP 441-2 отвечает за физическое преобразование сигналов. Заменяя интерфейсный submodule на другой делает возможным совместимость модуля CP с коммуникационным партнером.

Загружаемый драйвер для PtP-соединений CP
(S7 - ведомое устройство)
A5E00218418-04

Интерфейс параметризации CP: *Point-to-Point Communication, Parameter Assignment*

Интерфейс параметризации CP: *Point-to-Point Communication, Parameter Assignment* используется для параметризации submodule коммуникационного процессора CP.

К

Коммуникационный процессор

Коммуникационные процессоры - это модули, обеспечивающие PtP-соединения и соединения по шине.

Конфигурация связанных данных проекта (Project Configuration of Data Link)

Конфигурация связанных данных проекта - это термин относящийся к размещению ID соединений в системном функциональном блоке. ID соединений позволяет отдельным системным функциональным блокам обеспечивать связь между двумя коммуникационными узлами.

Конфигурирование

Конфигурирование - это установка отдельных модулей PLC в таблице конфигурации (configuration table).

М

Модули

Модули - это вставляемые в PLC печатные платы, расширяющие его функциональность.

О

Образ процесса (Process Image)

Образ процесса - это специальная область памяти в PLC. В начале выполнения циклической программы состояния сигналов входных модулей пересылается в таблицу входов области отображения процесса (process image input table). В конце выполнения циклической программы значения из таблицы выходов области отображения процесса (output table) пересылаются в выходные модули и формируют состояния выходов модулей вывода сигналов.

Онлайн/Офлайн (On-line/Off-line)

Термины обозначающие режим работы с данными. Режим онлайн (или интерактивный режим) означает наличие обмена данными между PLC и программатором. Режим офлайн (или автономный режим) означает отсутствие обмена данными.

Операнд

Операнд - это часть инструкции и выражений в STEP 7, с которой процессор выполняет действия. Операнд может быть адресован символьным или абсолютным способом.

Операционная система CPU

Операционная система CPU организует все функции и операции CPU, которые не связаны с пользовательской задачей управления.

П

Параметр

Параметры - это значения, которые могут быть назначены. Различают параметры блоков и параметры модулей.

Параметризация

Термин означает процесс задания определенного поведения модуля.

Параметр блока

Параметры блока - это переменные внутри блоков многократного использования, которые заменяются текущими значениями, когда вызывается соответствующий блок.

Параметр модуля

Параметры модулей устанавливаются для задания их определенных реакций. Различают статические и динамические параметры.

Перезапуск (RESTART)

PLC при переходе от режима STOP к режиму RUN проходит режим перезапуска (RESTART).

Переменная

Переменная - это операнд (например, E 1.0), который может иметь символьное имя и может, следовательно, быть адресован с помощью этого символьного имени.

Прерывание (Interrupt)

Прерывание происходит, когда обработка программы в процессоре PLC прерывается внешним сигналом.

Программируемый логический контроллер (PLC)

Программируемый логический контроллер (PLC) - это электронное устройство управления с по крайней мере одним центральным процессором, различными модулями ввода/вывода и устройствами операторского управления и мониторинга.

Протокол

Коммуникационные партнеры, участвующие в обмене данными, должны руководствоваться определенными правилами для управления и обработки потока данных. Такие правила называются протоколами.

Процедура

Выполнение операции обмена данными в соответствии с определенным протоколом называется процедурой.

Р

Рабочая память (Work Memory)

Рабочая память - это RAM-память в CPU, к которой процессор обращается, когда обрабатывает программу пользователя.

Рабочий режим

Программируемые контроллеры SIMATIC S7 имеют три рабочих режима: STOP (Стоп), RESTART (Перезапуск) и RUN (Выполнение). Функции CPU изменяются в зависимости от выбранного рабочего режима.

С

Системный блок

Системные блоки отличаются от других блоков тем, что они уже интегрированы в систему S7-300 и уже доступны для уже определенных системных функций. Системные блоки подразделяются на системные блоки данных, системные функции (SFC) и системные функциональные блоки (SFB).

Системные функции (SFC)

Системные функции (SFC) - это модули, не имеющие собственной памяти, которые уже интегрированы в операционную систему CPU и могут быть вызваны из пользовательской программы.

Системные функциональные блоки (SFB)

Системные функциональные блоки (SFB) - это модули, имеющие свою память, которые уже интегрированы в операционную систему CPU и могут быть вызваны из пользовательской программы.

Стойка

Стойка (rack) - это базовый конструктив, имеющий слоты для монтажа модулей контроллера.

Т**Тип данных (Data Type)**

Типы данных позволяют пользователям определять, как значение переменной или константы должно использоваться в пользовательской программе. Типы данных подразделяются на простые и структурированные типы.

У**Установки по умолчанию (Default Setting)**

Установки по умолчанию - практические базовые установки для параметров, которые всегда используются, если для параметров не заданы другие значения.

Ф**Функциональный блок (FB)**

Функциональные блоки - это компоненты пользовательской программы, являющиеся в соответствии со стандартом IEC "блоками с памятью". Памятью для функционального блока является назначенный экземплярный блок данных ("instance data block"). Функциональные блоки могут иметь, а могут и не иметь параметров.

Ц**Циклическое выполнение программы**

При циклическом выполнении пользовательская программа обрабатывается в постоянно повторяющемся цикле.

Э**Экземпляр DB (Instance Data Block)**

Экземпляр DB - блок данных, назначенный функциональному блоку, содержащий данные для этого специального блока.

C

CPU

Центральный процессор программируемого контроллера S7, имеющий в своем составе АЛУ, память, операционную систему и интерфейсы для модулей ввода/вывода (I/O).

CRC

Cyclic Redundancy Check - циклическая проверка целостности данных методом сверки контрольной суммы с гарантированной точностью распознавания ошибки.

Cycle Time (Время цикла)

Время цикла - это время, необходимое CPU для однократного сканирования пользовательской программы.

D

Data Block (DB) (Блок данных)

Блоки данных - это блоки, содержащие данные и параметры, которые используются в пользовательской программе. В отличие от других блоков блоки данных не содержат инструкций. Они подразделяются на глобальные (общие) блоки данных и экземплярные блоки данных. Доступ к данным, содержащимся в этих блоках может быть абсолютным или символьным. Сложные данные могут храниться в структурированных формах.

Data Type (Тип данных)

Типы данных позволяют пользователям определять, как значение переменной или константы должно использоваться в пользовательской программе. Типы данных подразделяются на простые и структурированные типы.

Default Setting (Установки по умолчанию)

Установки по умолчанию - практические базовые установки для параметров, которые всегда используются, когда для параметров не заданы другие значения.

Diagnostic Events (Диагностические события)

Диагностические события - это, например, ошибки в модуле или системные ошибки в CPU, которые вызваны, скажем, ошибкой в программе или изменением рабочего режима.

Download (Загрузка)

Термин загрузка означает загрузку загружаемых объектов (например, кодовых блоков) из программатора в загружаемую память (load memory) CPU.

F**Function Block (FB) (Функциональный блок)**

Функциональные блоки - это компоненты пользовательской программы, являющиеся в соответствии со стандартом IEC "блоками с памятью". Памятью для функционального блока является назначенный экземплярный блок данных ("instance data block"). Функциональные блоки могут иметь, а могут и не иметь параметров.

H**Hardware (Оборудование)**

Термин обозначающий все физическое и техническое оборудование (аппаратная часть) PLC.

I**Instance Data Block (Экземпляр DB)**

Экземпляр DB - блок данных, назначенный функциональному блоку, содержащий данные для этого специального функционального блока.

Interrupt (Прерывание)

Прерывание происходит, когда обработка программы в процессоре PLC прерывается внешним сигналом.

O**On-line/Off-line (Онлайн/Офлайн)**

Термины обозначающие режим работы с данными. Режим онлайн (или интерактивный режим) означает наличие обмена данными между PLC и программатором. Режим офлайн (или автономный режим) означает отсутствие обмена данными.

On-line Help (Интерактивная справка)

STEP 7 обеспечивает пользователя контекстно отображаемой справочной информацией при работе с программным обеспечением.

P**PtP-соединение**

Используя PtP-соединения (point-to-point - "точка к точке"), модуль CP формирует интерфейс между PLC и коммуникационным партнером.

Process Image (Образ процесса)

Образ процесса - это специальная область памяти в PLC. В начале выполнения циклической программы состояния сигналов входных модулей пересылается в таблицу входов области отображения процесса (process image input table). В конце выполнения циклической программы значения из таблицы выходов области отображения процесса (output table) пересылаются в выходные модули и формируют состояния выходов модулей вывода сигналов.

Project Configuration of Data Link

Project Configuration of Data Link (конфигурация связанных данных проекта) - это термин относящийся к размещению ID соединений в системном функциональном блоке. ID соединений позволяет отдельным системным функциональным блокам обеспечивать связь между двумя коммуникационными узлами.

PLC

Программируемый логический контроллер (PLC) - это электронное устройство управления с по крайней мере одним центральным процессором, различными модулями ввода/вывода и устройствами операторского управления и мониторинга.

R

RESTART (Перезапуск)

PLC при переходе от режима STOP к режиму RUN проходит режим перезапуска (RESTART).

S

Software (Программное обеспечение)

Программное обеспечение (Software) - это термин, объединяющий все программы, используемые в компьютерной системе. Сюда входят также операционная система и пользовательские программы.

STEP 7

STEP 7 - это программное обеспечение для программируемых контроллеров SIMATIC S7.

U

Upload (Выгрузка)

Выгрузка означает выгрузку загружаемых объектов (например, кодовых блоков) из загружаемой памяти (load memory) CPU в программатор. Программа пользователя содержит все инструкции и объявления, касающиеся обработки сигналов, с помощью которых системой или процессом можно управлять. Пользовательская программа для системы SIMATIC S7 структурируется и подразделяется на более мелкие части, называемые блоками.

W

Work Memory (Рабочая память)

Рабочая память - это RAM-память в CPU, к которой процессор обращается, когда обрабатывает программу пользователя.

Предметный указатель

А

- Автоматическое использование вспомогательных сигналов 7-6
- Адрес ведомого устройства (Slave Address) 7-2
- Адрес ведомого устройства (Slave Address) 4-12
- Адрес модуля (LADDR) 6-1

Б

- Блокировка / ограничение функций записи 3-8, 4-19

В

- Ведомое (slave) Modbus-устройство 6-2, 6-5
- Возможная конфигурация системы 1-2
- Временная диаграмма запроса на передачу 7-7
- Время реакции 5-8
- Вспомогательные сигналы RS 232C (Secondary Signals) 7-5
- Выбор протокола 4-9
- Вызов коммуникационного FB 5-5
- Вычисление начального адреса регистра Start_register 8-11

Д

- Деинсталляция драйвера 4-2
- Диагностика драйвера 9-1
- Диагностика коммуникационного FB 10-1
- Диагностика посредством выходного параметра "состояний" STATUS функциональных блоков 9-1
- Диагностика посредством диагностического буфера CP 9-1
- Диагностика посредством сообщений об ошибках в области SYSTAT 9-1
- Диагностика посредством элементов индикации модуля 9-1
- Диагностика с использованием элементов индикации CP 341 9-2
- Диагностика с использованием элементов индикации CP 441-2 9-4
- Диагностические сообщения системных функциональных блоков CP 441-2 9-5
- Диагностические сообщения функциональных блоков CP 341 9-3
- Дискета с дистрибутивом 5-1
- Допустимые области для записи 3-8
- Доступ с функциональными кодами для битов 3-5
- Доступ с функциональными кодами для регистров 3-6

3

- Загрузка MODBUS-драйвера для ведомого устройства 3-2
- Загрузка данных конфигурации и параметризации для CP 341 4-23
- Загрузка данных конфигурации и параметризации для CP 441-2 4-24
- Загрузочная (load) память A-8
- Загрузочная память (Loading Memory) CPU 1-2
- Запуск, инициализация коммуникационного FB 5-5
- Защитный ключ-заглушка (dongle) 1-2,2-1
- Значение времени для считывания области SYSTAT 5-6
- Значения времени пересылки A-1

И

- Индикатор групповых сообщений SF 9-2
- Инициализация (Initialization) 4-25
- Инициализация окончания контрольного времени (Timeout Initialization) 5-6
- Инсталляция FB 5-1
- Инсталляция драйвера в STEP 7 в программаторе/ПК 4-1
- Интерпретация MODBUS-адресов 1-1
- Интерфейс CPU - CP 6-1
- Интерфейс CPU - CP для модуля CP 341 6-1
- Интерфейс CPU - CP для модуля CP 441-2 6-3
- Интерфейс RS232C / TTY 2-2
- Интерфейс X27 (RS 422/485) 4-20
- Интерфейс X27 / RS422 (четырёхпроводный режим) 2-2
- Интерфейс X27 / RS485 (двухпроводный режим) 2-2
- Интерфейсы 2-2

Использование защитного ключа-заглушки 2-1

Используемые MODBUS функциональные коды 3-3

Используемые модули 1-2

"Итоговый" номер блока DB 3-7

К

- Карта памяти 1-2
- Классы событий 9-8
- Коды ошибок в области SYSTAT для "CPU Job Errors" ("Ошибки в работе CPU") 9-9
- Коды ошибок в области SYSTAT для "General Processing Errors" ("Общие ошибки обработки") 9-12
- Коды ошибок в области SYSTAT для "Receive Errors" ("Ошибки при приеме") 9-9
- Коды ошибок в сообщениях 7-4
- Количество регистров 8-12
- Компоненты ПО Data Link 3-1
- Компоненты проекта 4-2
- Консистентность данных 6-2, 6-4
- Конфигурация оборудования 4-3
- Конфигурирование данных коммуникаций Data Link 4-2
- Конфигурирование данных коммуникаций для CP 341 4-2
- Конфигурирование данных коммуникаций для CP 441-2 4-3
- Конфигурирование данных коммуникаций проекта 4-78

М

Множитель для времени ожидания символа (Multiplier Character Delay Time) 4-12

Загружаемый драйвер для PtP-соединений CP (S7 - ведомое устройство) A5E00218418-04

Н

- Назначение параметров (Parameter Assignment) 4-25
- Назначение параметров для CP 341 4-4
- Назначение параметров для CP 4-4
- Назначение параметров для CP 441-2 4-5
- Назначение параметров для коммуникационного партнера 4-6
- Назначение параметров запуска ("Startup of the CPU"- "Запуск CPU") 4-26
- Настройка драйвера 4-1
- Начало SIMATIC-области 4-14
- Номер слова в DB 3-7
- Нормальный режим ("Normal Operation") 4-12

О

- Обзор параметров для обмена данными 4-10
- Области данных 1-4, 3-4
- Область SYSTAT 5-8
- Обмен данными 1-4
- Обслуживание и поддержка с помощью Интернета vi
- Общий обзор функциональных кодов 01, 05, 15 4-13
- Ответное сообщение для функции 01 8-5
- Ответное сообщение для функции 02 8-8
- Ответное сообщение для функции 03 8-11
- Ответное сообщение для функции 04 8-15
- Ответное сообщение для функции 05 8-18
- Ответное сообщение для функции 06 8-20

Загружаемый драйвер для PtP-соединений CP (S7 - ведомое устройство)
A5E00218418-04

- Ответное сообщение для функции 08 8-22
- Ответное сообщение для функции 15 8-25
- Ответное сообщение для функции 16 8-28
- Отладка коммуникационного FB 5-1
- Очистка (Deleting) области SYSTAT 9-7

П

- Параметр "ERROR_INFO" 10-1
- Параметр "ERROR_NR" 10-1
- Параметр длины LENGTH при передаче данных между CP и CPU 6-1, 6-3
- Параметры FB 180 (CP 441-2) 5-4
- Параметры FB 80 (CP 341) 5-3
- Параметры и рабочие режимы для загружаемого драйвера 3-2, 4-9
- Параметры, зависящие от драйвера 4-9
- ПО MODBUS Slave Data Link 3-1
- Полный дуплексный режим (RS422) 4-21
- Полудуплексный режим (RS485) 4-22
- Представление адресов 1-5
- Предустановка "R(A) 0V, R(B) 5V" (High) ("Высокий уровень") 4-22
- Предустановка "R(A) 5V, R(B) 0V" (BREAK) ("Обрыв") 4-22
- Предустановки для входных линий 4-22
- Преобразование MODBUS-адреса для битовых функций 4-13
- Преобразование Modbus-адресов для функций для регистров 4-17
- Пригодные для использования интерфейсы и протоколы 1-2
- Применимость данного руководства iii
- Пример OB1 для CP 341 5-7
- Пример OB1 для CP 441-2 5-7

- Пример OB100/101 5-7
- Проверка четности (CRC Check) 7-3
- Проект в S7 4-2
- Протокол для ведомого устройства Modbus 4-10
- Протокол обмена GOULD - MODBUS 1-1,1-4
- Протокол обмена данными 7-1
- Р**
- Рабочая (work) память A-8
- Размер блока 6-2, 6-4
- Распределение задач 3-3
- Рассылка сообщений (Broadcast Messages) 7-2
- Режим работы "нормальный" ("Normal Operation") 4-12
- Режим работы "с подавлением помех" ("Interference Suppression") 4-12
- Режим работы Data Link 3-1
- Рекомендуемый базовый уровень подготовки пользователей iii
- С**
- Связь ведущего и ведомого устройств 7-1
- Скорость передачи (Transmission Rate) 4-10
- Содержание файла проекта Modsl для CP 341 5-2
- Содержание файла проекта Modsl для CP 441-2 5-2
- Сообщение-запрос для функции 01 8-4
- Сообщение-запрос для функции 02 8-8
- Сообщение-запрос для функции 03 8-11
- Сообщение-запрос для функции 04 8-14
- Сообщение-запрос для функции 05 8-18
- Сообщение-запрос для функции 06 8-20
- Сообщение-запрос для функции 08 8-22
- Сообщение-запрос для функции 15 8-25
- Сообщение-запрос для функции 16 8-28
- Сообщения в выходном параметре STATUS 9-5
- Сообщения об отказах (Exception Responses) 7-3
- Специальный индикатор (Special) RXD 9-2
- Специальный индикатор (Special) TXD 9-2
- Средства диагностики модуля CP 341 9-2
- Средства диагностики модуля CP 441-2 9-4
- Стоповые биты (Stop Bits) 4-11
- Структура области SYSTAT 9-7
- Структура сообщения 7-1
- Схема многоточечного соединения RS422 (MODBUS) B-1
- Схема многоточечного соединения RS485 (MODBUS) B-2
- Схема нумерации 9-7
- Схема нумерации классов и номеров событий 9-3
- Схемы многоточечных соединений B-1
- Считывание сообщений об ошибках из области SYSTAT 9-6
- Т**
- Таблицы событий (Events) / ошибок (Errors) 9-8
- Техническая поддержка A&D vii
- Технические данные A-1
- Требования к памяти A-8
- Требования к программному и аппаратному обеспечению 1-2, 1-3
- Загружаемый драйвер для PtP-соединений CP (S7 - ведомое устройство) A5E00218418-04

У

Удаление информации об ошибках 10-1

Установка защитного ключа-заглушки 2-1

Учебные курсы vi

Ф

Функции диагностики 10-1

Функции диагностики 9-1

Функции драйвера 1-1

Функциональные возможности 1-1

Функциональные коды FC 05, 06, 15, 16 3-8

Функциональные коды 01, 05, 15 3-5

Функциональные коды FC 03, 06, 16 3-6

Функциональный код (Function Code) 7-2

Функциональный код 01 (Read Coil (Output) Status) (Считывание состояния выхода (катушки)) 8-2

Функциональный код 02 (Read Input Status) (Считывание состояния входа) 8-6

Функциональный код 02 3-6

Функциональный код 03 (Read Output Registers) (Считывание выходных регистров) 8-9

Функциональный код 04 3-7

Функциональный код 04 (Read Input Registers) (Считывание входных регистров) 8-13

Функциональный код 05 (Force Single Coil) (Назначение сигнала для отдельного выхода) 8-16

Функциональный код 06 (Preset Single Register) (Назначение для отдельного регистра) 8-19

Функциональный код 08 (Loop Back Diagnostic Test) (Тестирование соединительных линий) 8-22

Загружаемый драйвер для PtP-соединений CP (S7 - ведомое устройство) A5E00218418-04

Функциональный код 15 (Force Multiple Coil) (Назначение для нескольких выходов) 8-23

Функциональный код 16 (Preset Multiple Register) (Назначение для нескольких регистров) 8-26

Х

Характеристики запуска CP 4-25

Ц

Циклическая обработка программы 5-8

Ч

Четность (Parity) 4-11

Э

Экземплярный DB 5-6

A

A&D Technical Support vii

C

CTS 7-5

D

DCD 7-5

DSR 7-5

DTR 7-5

DTR-управление 7-5

EERROR_No 1...9 (Ошибки при
инициализации FB и CP) 10-1, 10-2ERROR_No 10...19 (Ошибки при
обработке функционального кода)
10-1, 10-2ERROR_No 90...99 (Другие ошибки)
10-1, 10-4Exception Responses (Сообщения об
отказах) 7-3

EXTF 9-4

F

FAULT 9-4

FB V24_SET 7-5

FB V24_STAT 7-5

Force Multiple Coil (Функциональный
код 15) (Назначение для нескольких
выходов) 8-23Force Single Coil (Функциональный код
05) (Назначение сигнала для
отдельного выхода) 8-16**G**GOULD - MODBUS (протокол обмена)
1-1,1-4**I**Interference Suppression (Режим работы
"с подавлением помех") 4-12

INTF 9-4

LLoading Memory (Загрузочная память)
CPU 1-2Loop Back Diagnostic Test
(Функциональный код 08)
(Тестирование соединительных
линий) 8-22**M**"MIN" / "MAX" в области памяти
SIMATIC 4-19MODBUS Communications Function
Block (коммуникационный
функциональный блок) 3-1MODBUS Slave Driver (драйвер для
ведомого устройства) 3-2

Modbus-адрес - "от" / "до" 4-14

MODBUS-драйвер для ведомого
устройства 3-2MODBUS-коммуникационный
функциональный блок 3-1Multiplier Character Delay Time
(Множитель для времени ожидания
символа) 4-12**N**Normal Operation (Режим работы
"нормальный") 4-12

"None" (плавающее значение) 4-22

PPreset Multiple Register
(Функциональный код 16)
(Назначение для нескольких
регистров) 8-26Preset Single Register (Функциональный
код 06) (Назначение для
отдельного регистра) 8-19Загружаемый драйвер для PtP-соединений CP
(S7 - ведомое устройство)
A5E00218418-04

R

- Read Coil (Output) Status
(Функциональный код 01)
(Считывание состояния выхода
(катушки)) 8-2
- Read Input Registers (Функциональный
код 04) (Считывание входных
регистров) 8-13
- Read Input Status (Функциональный код
02) (Считывание состояния входа)
8-6
- Read Output Registers
(Функциональный код 03)
(Считывание выходных регистров)
8-9
- RI 7-5
- RTS 7-5
- RTS-управление 7-5
- RTU-формат (протокол обмена
GOULD - MODBUS) 1-1
- RXD 9-2, 9-4

S

- S7wfpb1a.dll 4-1
- S7wfpb1x.cod 4-1
- S7wfpb2x.cod 4-1
- SF 9-2
- Slave Address (Адрес ведомого
устройства) 7-2
- Slave Address (Адрес ведомого
устройства) 4-12

T

- Timeout Initialization (Инициализация
окончания контрольного времени) 5-
6
- TXD 9-2, 9-4

