

Глава 5 Ввод в действие PtP соединения

Введение

В главе описывается применение PtP соединения на основе интерфейса RS485.

Вы также найдете всю информацию о протоколах, включении и проектировании интерфейса связи, которая необходима для организации последовательного соединения с использованием интерфейса RS485.

Содержание

Раздел	Страница
Глава 5 Ввод в действие PtP соединения	5-1
Быстрый старт	5-2
Протоколы.....	5-3
Перевод интерфейса RS485 в режим PtP.....	5-7
Механизм передачи данных	5-9
Параметрирование.....	5-10
Обмен данными.....	5-13

Быстрый старт

Общее	В конфигураторе оборудования Вы можете отключить в ЦПУ встроенное Profibus устройство и перевести интерфейс RS485 в режим соединения точка-точка (PtP - Point to Point). Использование соединения точка-точка позволяет обеспечить соединение с различными системами по последовательному протоколу.
Протоколы	В этом режиме поддерживаются ASCII, STX/ETX, 3964R, USS и Modbus протоколы.
Переключение RS485 в режим PtP	По умолчанию интерфейс RS485 ЦПУ установлен в режим Profibus master. В конфигураторе оборудования имеется возможность перевести интерфейс RS485 в режим соединения точка-точка, установив требуемое значение параметра "Function RS485 X3" в диалоговом окне свойств ЦПУ.
Параметрирование	Дальнейшее параметрирование последовательного интерфейса выполняется во время выполнения программы с помощью функций SFC 216 (SER_CFG). Для этого Вы должны сохранить параметры в блоке данных для всех протоколов, за исключением ASCII.
Обмен данными	Для обмена данными по последовательному интерфейсу применяются SFC. Для передачи данных используется SFC 217 (SER_SND), для приема данных SFC 218 (SER_RCV). Выходная переменная RetVal системной функции SFC217 SER_SND в случае применения протоколов 3964R, USS или Modbus в следующем вызове содержит, среди прочего, информацию о подтверждении от партнерской станции. Протоколы USS и Modbus позволяют оценивать полученную телеграмму при вызове SFC218 SER_RCV после вызова SER_SND. Функции SFC поставляются вместе с ЦПУ.
SFC функции для последовательного соединения	Для последовательного соединения используются следующие SFC:

SFC		Описание
SFC 216	SER_CFG	RS485 параметрирование
SFC 217	SER_SND	RS485 передача данных
SFC 218	SER_RCV	RS485 прием данных

Протоколы

Введение

ЦПУ поддерживает следующие протоколы

- ASCII
- STX/ETX
- 3964R
- USS
- Modbus

ASCII

Протокол ASCII – это один из самых простых способов обмена данными. Поступающие от ЦПУ символы передаются в линию без изменений.

В протоколе ASCII на каждом цикле системная функция чтения передает данные, имеющиеся в буфере на момент запроса, в блок данных, указанный в качестве параметра. Если телеграмма передается в нескольких циклах, то данные переписываются. Подтверждение не требуется. Процедура обмена должна управляться из программы пользователя.

Функциональный блок для приема данных по протоколу ASCII (Receive_ASCII-FB) можно найти по адресу <ftp://ftp.vipa.de/>.

STX/ETX

STX/ETX – это простой протокол с начальным и конечным ID (идентификаторами), где STX означает **Start of Text** (Начало строки) и ETX означает **End of Text** (конец строки).

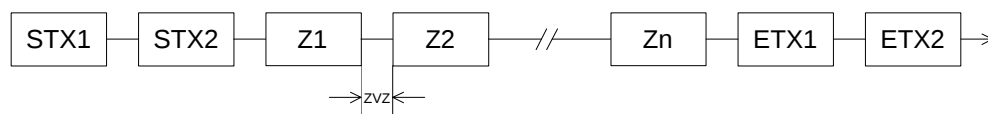
Протокол STX/ETX удобен для передачи ASCII символов. В нем не используется контрольная сумма. Start-ID должен предшествовать любым передаваемым данным, которые, в свою очередь, должны завершаться End-ID.

В зависимости от числа битов данных имеется возможность передать следующие ASCII символы: 5 бит: не разрешено; 6 бит: 20...3Fh, 7 бит: 20...7Fh, 8 бит: 20...FFh.

Полезными данными являются данные, полученные ПЛК после Start-ID до End-ID.

Данные, посылаемые из ПЛК в периферийное устройство, предварительно обрабатываются в SFC217 (SER_SND) и передаются коммуникационному партнеру с добавленными Start-ID и End-ID.

Структура сообщения:



Вы можете определить до 2 Start и End ID.

Вы можете работать с 1, 2 или без Start-ID и с 1, 2 или без End-ID. Для управляющих Start-ID и End-ID разрешено использование шестнадцатеричных символов в диапазоне 01h ...1Fh. Коды, превышающие 1Fh, игнорируются. В данных пользователя символы с

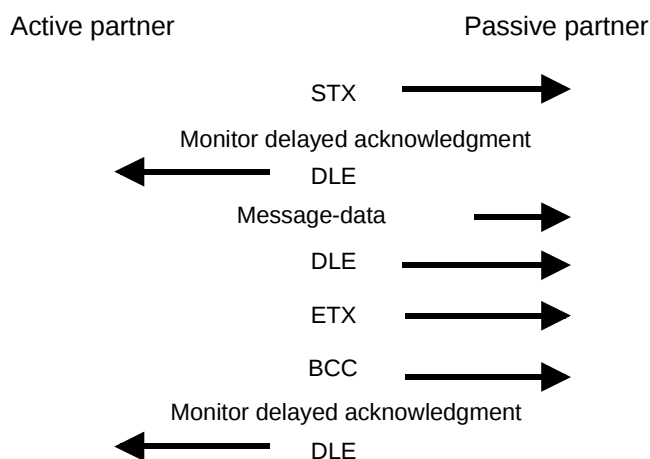
3964R

кодом меньше 20h не разрешены и могут вызвать ошибку. Количество Start-ID и End-ID может отличаться (1 Start, 2 End или 2 Start, 1 End или другие комбинации). Если End-ID не используется, то все полученные в интерфейсе символы передаются в ПЛК по истечении заданной временной задержки между символами (межбайтового таймаута).

Протокол 3964R управляет передачей данных по PtP-соединению между ЦПУ и коммуникационным партнером. Во время передачи данных протокол добавляет управляющие символы к информационным данным. Эти управляющие символы позволяют коммуникационному партнеру проверить полноту и целостность полученных данных.

В протоколе используются следующие управляющие символы:

- STX **S**tart of **T**ext (Начало передаваемой строки)
- DLE **D**ata **L**ink **E**scape (Отключение передачи данных)
- ETX **E**nd of **T**ext (Конец передаваемой строки)
- BCC **B**lock **C**heck **C**haracter (Символ контроля блока)
- NAK **N**egative **A**cknowledge (Негативное квитирование)

Процедура связи

Вы можете передать до 255 байт за одно сообщение.

**Внимание!**

Когда символ DLE передается как часть информационных данных, то он должен повториться дважды, чтобы отличить его от управляющего символа DLE, который, в свою очередь, используется для установления и завершения соединения (DLE дублирование). Дублирование DLE устраняется в принимающей станции автоматически.

Протокол 3964R требует, чтобы одному из коммуникационных партнеров был назначен низкий приоритет. Когда коммуникационные партнеры одновременно посылают команды передачи, партнер с более низким приоритетом приостанавливает свою передачу.

USS

Протокол USS (**U**niverselle **s**erielle **S**chnittstelle - универсальный последовательный интерфейс) – это последовательный протокол, разработанный фирмой Siemens для связи с приводами и другими системными компонентами. Он позволяет подключать несколько slave-устройств к master-устройству по последовательной шине

Протокол USS позволяет передавать данные с постоянным циклом, указывая фиксированную длину телеграммы.

Основные особенности, характеризующие протокол USS:

- Многоточечное соединение
- Master-Slave процедура доступа
- Единственное Master-устройство
- Подключение до 32 участников сети
- Простая и безопасная структура телеграммы

Вы можете подключить до 31 slave-устройства к 1 master-устройству по шине. Обращение к slave-устройству происходит по его адресу, указываемому в телеграмме. Протокол USS работает в полудуплексном режиме.

После отправки телеграммы ответная телеграмма считывается с помощью SFC218 SER_RCV.

Посылаемые и получаемые телеграммы имеют следующую структуру:

Телеграмма Master->Slave

STX	LGE	ADR	PKE		IND		PWE		STW		HSW		BCC
02h			H	L	H	L	H	L	H	L	H	L	

Телеграмма Slave->Master

STX	LGE	ADR	PKE		IND		PWE		ZSW		HIW		BCC
02h			H	L	H	L	H	L	H	L	H	L	

где	STX:	Стартовый символ	STW:	Слово управления
	LGE:	Длина телеграммы	ZSW:	Слово состояния
	ADR:	Адрес	HSW:	Задаваемое значение
	PKE:	ID параметра	HIW:	Текущая величина
	IND:	Индекс	BCC:	Контрольная сумма
	PWE:	Значение параметра		

Modbus

Протокол Modbus – это коммуникационный протокол для обмена данными между одним master-устройством и несколькими slave-устройствами.

Физически Modbus работает на последовательном соединении в полудуплексном режиме.

Поскольку master-устройство за один сеанс устанавливает соединение только с одним slave-устройством в сети, то в сети отсутствуют конфликты передачи. После того, как запрос был послан master-устройством, оно ожидает ответа от slave-устройства заданное время. Во время ожидания обмен с другими slave-устройствами не происходит.

После того, как master-устройство отправило телеграмму, ответная телеграмма считывается с помощью SFC218 SER_RCV.

Телеграмма, посланная master-устройством, и ответная телеграмма от slave-устройства должны иметь следующую структуру:

Стартовый символ	Адрес Slave-устройства	Код функции	Данные	Контрольная сумма	Символ окончания
------------------	------------------------	-------------	--------	-------------------	------------------

Широковещательный запрос с адресом slave-устройства = 0

Запрос может быть отправлен к одному slave-устройству или отправлен всем slave-устройствам как широковещательное сообщение. Для идентификации широковещательного сообщения Вы должны назначить адрес slave-устройства 0.

В отличие от "обычной" телеграммы, широковещательное сообщение не требует получения ответа с помощью SFC218 SER_RCV. Только команда записи может быть послана как широковещательное сообщение.

режимы
ASCII, RTU

Modbus предлагает 2 различных режима:

- режим ASCII: Каждый байт передается двумя символами ASCII кода. Передаваемые данные ограничиваются символом начала и символом завершения. Режим обеспечивает прозрачную, но медленную передачу.
- режим RTU: Каждый байт передается одним символом. Это обеспечивает более быструю передачу данных по сравнению с режимом ASCII. Вместо символов начала и завершения используются временные задержки.

Режим работы интерфейса задается при помощи SFC216 SER_CFG.

Поддерживаемые протоколы Modbus

На интерфейсе RS485 поддерживаются следующие режимы протоколов Modbus

- Modbus RTU Master
- Modbus ASCII Master

Перевод интерфейса RS485 в режим PtP

Переключение в режим PtP

По умолчанию интерфейс RS485 субмодуля DP ЦПУ используется как Profibus DP master-устройство. В конфигураторе оборудования интерфейс RS485 может быть переведен в режим соединения точка-точка (PtP) через параметр *Function RS485 X3*.

Конфигурирование оборудования

Конфигурирование оборудования выполняется в соответствии с описанием "Установка специфических параметров ЦПУ фирмы VIPA":

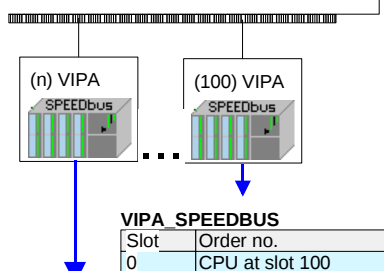
Standard bus

Slot	Module
1	
2	CPU 318-2
X2	DP
X1	MPI/DP
3	

Modules at the bus

	343-1EX11 (Ethernet PG/OP)
	343-1EX11 (internal CP)
CPs resp. DP master at SPEED-Bus as 343-1EX11 resp. 342-5DA02	
	342-5DA02 V5.0

virtual DP master for CPU
and every SPEEDbus module



Slot	Order no.
0	Module at slot n

- Запустите конфигуратор оборудования и подключите VIPA GSD файл SPEEDBUS.GSD.
- Установите в стойку Siemens CPU 318-2AJ00 (6ES7 318-2AJ00-0AB0/V3.0).
- Установите модули Серии 300 в стойку последовательно, начиная со слота 4.
- Вставьте CP 343-1 (343-1EX11) фирмы Siemens для описания встроенного интерфейса Ethernet PG/OP сразу после всех установленных модулей.
- Вставьте CP 343 как CP 343-1 (343-1EX11) сразу после интерфейса PG/OP.
- Далее описываются коммуникационные процессоры (CP) SPEED-Bus. Для описания CP Ethernet используйте Siemens CP 343-1 (343-1EX11), для CP Profibus DP используйте Siemens CP 342-5DA02 V5.0.
- SPEED-Bus модули устанавливаются на виртуальную шину Profibus. Для создания виртуальной сети Profibus обязательно вставьте в конфигурацию последним модуль CP 342-5 (342-5DA02 V5.0) фирмы Siemens и переведите его в режим master.
- Установите на виртуальную шину Profibus slave-устройство "VIPA_SPEEDBUS". После подключения файла SPEEDBUS.GSD это устройство находится в каталоге оборудования *Profibus-DP / Additional field devices / I/O / VIPA / VIPA_SPEEDBUS*.
- Установите для этого устройства Profibus адрес 100.
- Поместите CPU 317-4NE12 из каталога оборудования VIPA_SPEEDbus в слот 0.
- Двойным нажатием на CPU 317-4NE12 вызовите диалоговое окно свойств ЦПУ.
- Задайте параметру *Function RS485* режим "PtP".
- Для каждого SPEED-Bus модуля установите на виртуальную шину Profibus slave-устройство "VIPA_SPEEDBUS". Установите адреса устройств, начиная с адреса 101. Адрес slave-устройства равен номеру слота на *SPEED-BUS* шине +100. Поместите в слот 0 соответствующий модуль, и, если необходимо, измените параметры.

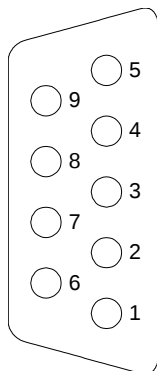
После переноса проекта интерфейс RS485 ЦПУ перейдет в режим PtP после запуска.

Свойства интерфейса RS485

- Логическое значение сигнала определяется по разности потенциалов между двумя проводами витой пары
- Подключение к последовательной шине в двухпроводном полудуплексном режиме
- Передача данных до 500 м
- Скорость обмена до 115.2кбод

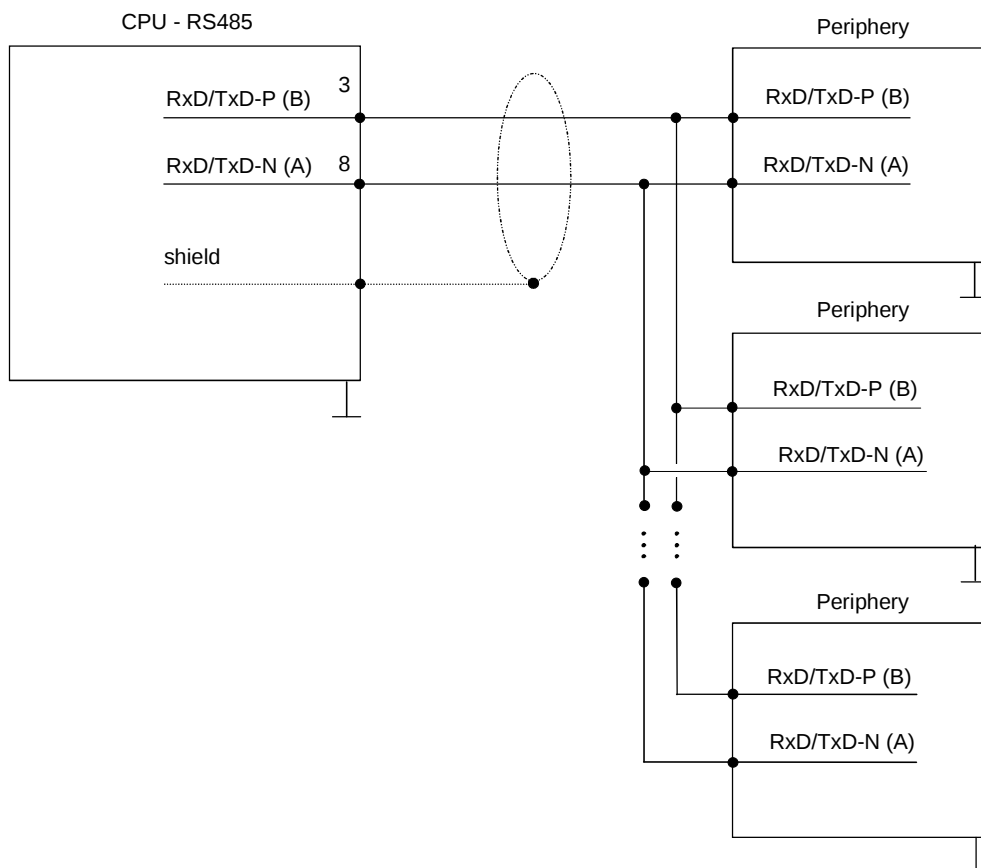
Контакты RS485

9-ти контактный разъем D-типа



Pin	Assignment
1	Зарезервировано
2	-24В
3	RxD/TxD-P (Линия В)
4	RTS
5	-5В
6	+5В
7	+24В
8	RxD/TxD-N (Линия А)
9	не используется

Подключение



Механизм передачи данных

Введение

Передача данных во время выполнения программ происходит с использованием системных функций SFC. Механизмы передачи данных, одинаковые для всех протоколов, иллюстрируются ниже.

Механизм

Данные, которые записываются в интерфейс ПЛК для передачи, попадают сначала в передающий буфер FIFO с размером 2x1024байт и затем выдаются в интерфейс.

Когда интерфейс получает данные, они сначала записываются в приемный буфер FIFO размером 2x1024байт и затем считываются ПЛК.

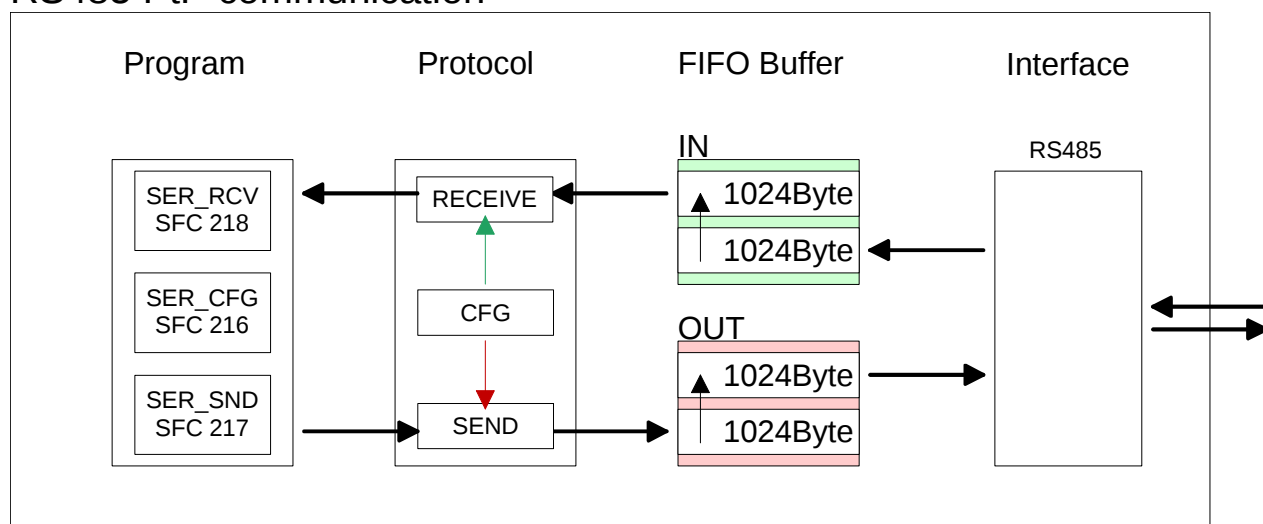
При передаче данных по выбранному программному протоколу преобразование данных к формату этого протокола происходит автоматически в функциях SFC.

В отличие от протоколов ASCII и STX/ETX, протоколы 3964R, USS и Modbus требуют подтверждения от партнерской станции.

Выходная переменная RetVal системной функции SFC217 SER_SND в случае применения протоколов 3964R, USS и Modbus содержит, среди прочего, информацию о подтверждении от партнерской станции.

Протоколы USS и Modbus позволяют оценивать полученную телеграмму при вызове SFC218 SER_RCV после вызова SER_SND.

RS485 PtP communication



Параметрирование

SFC216 (SER_CFG)

Параметрирование интерфейса происходит во время выполнения программы с использованием SFC216 (SER_CFG). Вы должны сохранить в блоке данных параметры для протоколов STX/ETX, 3964R, USS и Modbus.

Name	Declaration	Тип	Комментарии
Protocol	IN	BYTE	1=ASCII, 2=STX/ETX, 3=3964R
Parameter	IN	ANY	Указатель на параметры протокола
Baudrate	IN	BYTE	Скорость передачи данных
CharLen	IN	BYTE	0=5бит, 1=6бит, 2=7бит, 3=8бит
Parity	IN	BYTE	0=без паритета, 1=не чет, 2=чет
StopBits	IN	BYTE	1=1бит, 2=1.5бит, 3=2бит
FlowControl	IN	BYTE	1 (не меняется)
RetVal	OUT	WORD	Код ошибки (0 = OK)

Описание параметров

Все таймауты должны задаваться в шестнадцатеричной форме. Эта величина получается умножением требуемого времени на скорость обмена.

Пример: Требуемое время 8мс и скорость обмена 19200бод

Вычисление: $19200\text{бит/с} \times 0,008\text{с} \approx 154\text{бит} \rightarrow (9Ah)$

Шестнадцатеричная величина 9Ah.

Протоколы

Предусмотрена поддержка следующего набора протоколов:

- 1: ASCII
- 2: STX/ETX
- 3: 3964R
- 4: USS Master
- 5: Modbus RTU Master
- 6: Modbus ASCII Master

Parameter (в DB)

Для ASCII протокола параметры не используются.

Для протоколов STX/ETX, 3964R, USS и Modbus Вы определяете блок данных, в котором указываются коммуникационные параметры с указанной ниже структурой:

Блок данных для STX/ETX

DBB0:	STX1	BYTE	(1. Start-ID в шестнадцатеричной виде)
DBB1:	STX2	BYTE	(2. Start-ID в шестнадцатеричной виде)
DBB2:	ETX1	BYTE	(1. End-ID в шестнадцатеричной виде)
DBB3:	ETX2	BYTE	(2. End-ID в шестнадцатеричной виде)
DBW4:	TIMEOUT	WORD	(максимальное время между двумя символами)

**Внимание!**

Для управляющих Start-ID и End-ID разрешено использование шестнадцатеричных символов < 20h, в противном случае значения игнорируются!

Блок данных для 3964R

DBB0:	Prio	BYTE	(Должен быть определен для обоих партнеров)
DBB1:	ConnAtmptNr	BYTE	(Количество телеграмм для установления соединения)
DBB2:	SendAtmptNr	BYTE	(Количество повторов телеграммы)
DBW4:	CharTimeout	WORD	(Задержка для приема символа)
DBW6:	ConfTimeout	WORD	(Время ожидания подтверждения)

Блок данных для USS

DBW0:	Timeout	WORD	(Время задержки)
-------	---------	------	------------------

Блок данных для Modbus-Master

DBW0:	Timeout	WORD	(Время ожидания ответа)
-------	---------	------	-------------------------

Baud rate

Скорость передачи данных в бит/с (baud).

04h: 1200	05h: 1800	06h: 2400	07h: 4800
08h: 7200	09h: 9600	0Ah: 14400	0Bh: 19200
0Ch: 38400	0Dh: 57600	0Eh: 115200	

CharLen

Количество бит данных на символ.

0: 5	1: 6	2: 7	3: 8
------	------	------	------

Parity Паритет – это контроль четности. Если паритет задан, то информационные биты дополняются битом паритета ("0" или "1") до заданного значения. Если паритет не задан, бит паритета устанавливается в "1", но не используется.
0: не используется 1: нечет 2: чет

StopBits Стоповые биты устанавливаются в конце каждого символа и завершают передачу символа.
1: 1 2: 1.5 3: 2

FlowControl Управление потоком не используется. При передаче RTS=0, при приеме RTS=1.

RetVal SFC 216
(Сообщение об
ошибках
SER_CFG) Возвращаемое значение RetVal при конфигурации интерфейса

Код ошибки	Описание
0000h	нет ошибок
809Ah	интерфейс не обнаружен
8x24h	Ошибка x параметра, где x: 1: Ошибка в "Protocol" 2: Ошибка в "Parameter" 3: Ошибка в "Baudrate" 4: Ошибка в "CharLength" 5: Ошибка в "Parity" 6: Ошибка в "StopBits" 7: Ошибка в "FlowControl"
809xh	Ошибка x значения параметра, где x: 1: Ошибка в "Protocol" 2: Ошибка в "Parameter" 3: Ошибка в "Baudrate" 4: Ошибка в "CharLength" 5: Ошибка в "Parity" 6: Ошибка в "StopBits" 7: Ошибка в "FlowControl"
8092h	Ошибка доступа к параметрам (размер DB мал)
828xh	Ошибка x параметра DB, где x: 1: Ошибка параметра 1 2: Ошибка параметра 2 ...

Обмен данными

Введение Для обмена данными используются системные функции SFC217 (SER_SND) и SFC218 (SER_RCV).
Эти системные функции поставляются вместе с ЦПУ.

SFC 217 (SER_SND)

SFC217 предназначена для отправки данных по последовательному интерфейсу.

Выходная переменная RetVal системной функции SFC217 SER_SND в случае применения протоколов 3964R, USS и Modbus в следующем вызове содержит, среди прочего, информацию о подтверждении от партнерской станции.

Протоколы USS и Modbus позволяют оценивать полученную телеграмму при вызове SFC218 SER_RCV после вызова SER_SND.

.

Параметры

Имя	Описание	Тип	Комментарии
DataPtr	IN	ANY	Указатель на буфер посылаемых данных
DataLen	OUT	WORD	Длина посылаемых данных
RetVal	OUT	WORD	Код ошибки (0 = OK)

DataPtr Описывается указатель на буфер, где хранятся передаваемые данные. Вы должны указать тип, начало и длину данных.

Пример: Данные хранятся в блоке данных DB5, начиная с адреса 0.0 длиной 124байта.

DataPtr:=P#DB5.DBX0.0 BYTE 124

DataLen Количество переданных байтов данных.

Для протокола **ASCII**, если данные посылаются с помощью функции SFC217 быстрее, чем последовательный интерфейс может отправить в линию, длина данных для передачи может отличаться от длины, указанной в *DataLen* из-за переполнения буфера. Данную возможность необходимо учесть при разработке программы.

Для протоколов **STX/ETX**, **3964R**, **Modbus** и **USS** всегда возвращается длина, указанная в DataPtr, или 0.

**RetVal SFC 217
(Сообщение об
ошибках
SER_SND)**

Возвращаемое значение RetVal:

Код ошибки	Описание
0000h	Готовность к отправке данных
1000h	Данные не отправлены (длина данных 0)
20xxh	Протокол выполнен без ошибок с битовой маской xx для диагностики
7001h	Данные сохранены во внутреннем буфере – активность (занято)
7002h	Идет передача
80xxh	Протокол выполнен с ошибками с битовой маской xx для диагностики (нет подтверждения от партнера)
90xxh	Протокол не выполнен с битовой маской xx для диагностики (нет подтверждения от партнера)
8x24h	Ошибка x в параметре вызова функции SFC, где x: 1: Ошибка в "DataPtr" 2: Ошибка в "DataLen"
8122h	Ошибка в параметре "DataPtr" (размер DB мал)
807Fh	Внутренняя ошибка
809Ah	Интерфейс не найден или используется как Profibus устройство
809Bh	Интерфейс не сконфигурирован

**Протоколзави-
симые значения
RetVal***ASCII*

Значение	Описание
9000h	Переполнение буфера (данные не отправлены)
9002h	Размер данных мал (0 байт)

STX/ETX

Значение	Описание
9000h	Переполнение буфера (данные не отправлены)
9001h	Размер данных велик (>1024 байт)
9002h	Размер данных мал (0 байт)
9004h	Не разрешенный символ

3964R

Значение	Описание
2000h	Готовность к передаче, нет ошибок
80FFh	Получено NAK – ошибка в соединении
80FEh	Передача данных без подтверждения партнера или ошибка подтверждения
9000h	Переполнение буфера (данные не отправлены)
9001h	Размер данных велик (>1024 байт)
9002h	Размер данных мал (0 байт)

... Продолжение
RetVal SFC 217
SER_SND

USS

Значение	Описание
2000h	Готовность к передаче, нет ошибок
8080h	Переполнение приемного буфера (нет места для приема данных)
8090h	Превышено время подтверждения
80F0h	Ошибка контрольной суммы в ответе
80FEh	Ошибочный стартовый символ в ответе
80FFh	Ошибочный адрес slave-устройства в ответе
9000h	Переполнение буфера (данные не отправлены)
9001h	Размер данных велик (>1024 байт)
9002h	Размер данных мал (0 байт)

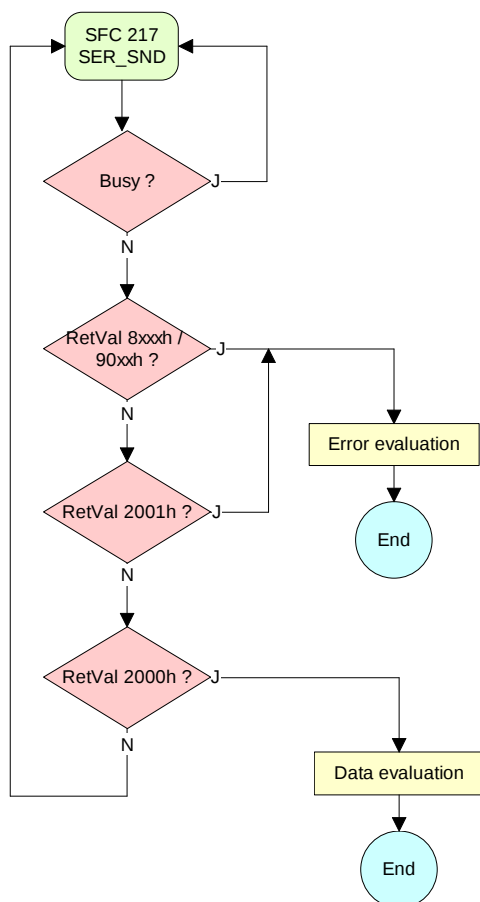
Modbus RTU/ASCII Master

Error code	Description
2000h	Готовность к передаче, нет ошибок
2001h	Готовность к передаче, есть ошибка
8080h	Переполнение приемного буфера (нет места для приема данных)
8090h	Превышено время подтверждения
80F0h	Ошибка контрольной суммы в ответе
80FDh	Длина ответа большая
80FEh	Ошибка кода функции в ответе
80FFh	Ошибочный адрес slave-устройства в ответе
9000h	Переполнение буфера (данные не отправлены)
9001h	Размер данных велик (>1024 байт)
9002h	Размер данных мал (0 байт)

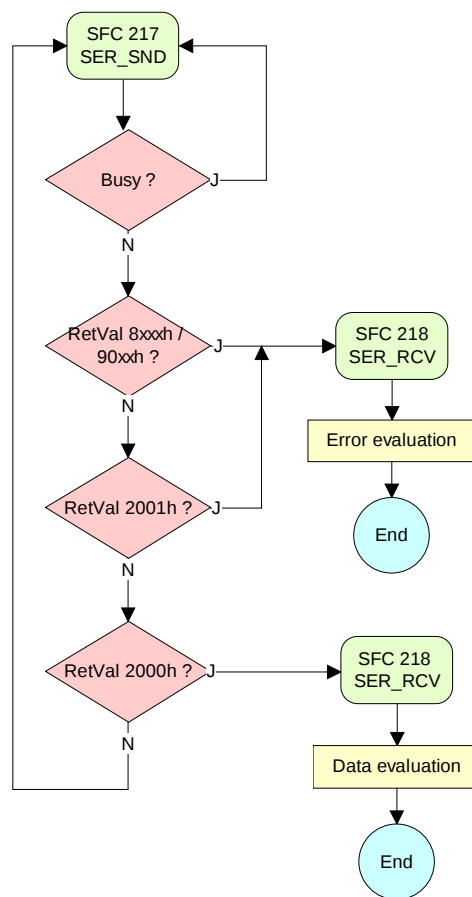
Принципы программирования

Приведенные ниже фрагменты иллюстрируют порядок отправки данных для различных протоколов.

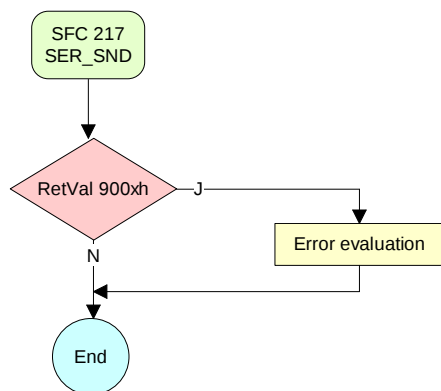
3964R



USS / Modbus



ASCII / STX/ETX



**SFC218
(SER_RCV)**

SFC218 предназначена для получения данных от последовательного интерфейса. Для протоколов USS и Modbus вызов SFC218 SER_RCV после вызова SER_SND позволяет получить ответ от партнера.

Параметры

Имя	Описание	Тип	Комментарии
DataPtr	IN	ANY	Указатель на буфер посылаемых данных
DataLen	OUT	WORD	Длина полученных данных
Error	OUT	WORD	Номер ошибки
RetVal	OUT	WORD	Код ошибки (0 = ОК)

DataPtr

Задается указатель на приемный буфер, где сохраняются получаемые данные. Вы должны указать тип, начало и длину данных.

Пример: Данные хранятся в блоке данных DB5, начиная с адреса 0.0 длиной 124 байта.

DataPtr:=P#DB5.DBX0.0 BYTE 124

DataLen

Возвращается количество полученных байт данных.

Для протоколов **STX/ETX** и **3964R** указывается количество полученных байт данных пользователя.

Для протокола **ASCII** указывается количество полученных символов. Это количество может отличаться от длины телеграммы

Error

В зависимости от протокола могут появляться следующие ошибки:

ASCII

Бит	Ошибка	Описание
0	overrun	Переполнение, символ не может быть прочитан из интерфейса достаточно быстро
1	framing error	Выбран неправильный битовый формат данных: превышена длина слова, ошибка последовательности бит (ошибка стоповых битов)
2	parity	Ошибка паритета
3	overflow	Буфер переполнен

STX/ETX

Бит	Ошибка	Описание
0	overflow	Полученная телеграмма превышает размер приемного буфера.
1	char	Полученный код символа вне диапазона 20h...7Fh.
3	overflow	Буфер переполнен

3964R / Modbus RTU/ASCII Master

Бит	Ошибка	Описание
0	overflow	Полученная телеграмма превышает размер приемного буфера.

RetVal SFC 218
(Сообщение об
ошибках
SER_RCV)

Возвращаемое значение RetVal:
:

Код ошибки	Описание
0000h	Нет ошибок
1000h	Приемный буфер слишком мал (данные потеряны)
8x24h	Ошибка x в параметре вызова, где x: 1: Ошибка в "DataPtr" 2: Ошибка в "DataLen" 3: Ошибка в "Error"
8122h	Ошибка в параметре "DataPtr" (размер DB мал)
809Ah	Интерфейс не найден или используется как Profibus устройство
809Bh	Интерфейс не сконфигурирован

Принципы
программирова-
ния

Приведенный ниже рисунок иллюстрируют структуру программы получения данных. Эту структуру можно использовать для всех протоколов

