

Использование библиотеки протокола Modbus

12

Библиотека команд STEP 7-Micro/WIN облегчает обмен данными с master-устройствами Modbus путем включения заранее спроектированных подпрограмм и программ обработки прерываний, которые специально разработаны для обмена данными с Modbus. С помощью команд протокола Modbus Slave можно настроить S7-200 так, чтобы он действовал как slave-устройство Modbus RTU и обменивался данными с master-устройствами Modbus.

Вы найдете эти команды в папке Libraries [Библиотеки] дерева команд STEP 7-Micro/WIN. С помощью этих новых команд вы можете заставить S7-200 действовать как slave-устройство Modbus. Когда вы выбираете команду Modbus Slave, к вашему проекту автоматически присоединяется одна или более соответствующих подпрограмм.

Библиотеки фирмы Siemens продаются на отдельном компакт-диске с дополнением к STEP 7-Micro/WIN: библиотека команд, номер для заказа 6ES7 830-2BC00-0YX0. Если вы купили и установили версию 1.1 библиотеки фирмы Siemens, то любое последующее обновление STEP 7-Micro/WIN V3.2x и V4.0, которое вы устанавливаете, автоматически заменит новой версией и ваши библиотеки без дополнительной оплаты (если библиотеки были дополнены или изменены).

В этой главе

Требования для использования протокола Modbus	350
Инициализация и время исполнения протокола Modbus	350
Адресация Modbus	351
Использование команд протокола Modbus Slave	352
Команды протокола Modbus Slave	353

Требования для использования протокола Modbus

Команды протокола Modbus Slave используют следующие ресурсы S7-200:

- Инициализация протокола Modbus Slave подготавливает порт 0 к обмену данными с помощью протокола Modbus Slave.
 Когда порт 0 используется для обмена данными с помощью протокола Modbus Slave, он не может использоваться для других целей, включая обмен данными со STEP 7-Micro/WIN. Команда MBUS_INIT управляет назначением порта 0 протоколу Modbus Slave или PPI.
- Команды протокола Modbus Slave влияют на все адреса специальной памяти (SM), связанные со свободно программируемым обменом данными через порт 0.
- Команды протокола Modbus Slave используют 3 подпрограммы и 2 программы обработки прерываний.
- Команды протокола Modbus Slave требуют 1857 байтов программной памяти для двух команд Modbus Slave и поддерживающих программ.
- Переменные для команды протокола Modbus Slave требуют в памяти переменных блока размером 779 байтов. Начальный адрес этого блока назначается пользователем и резервируется для переменных Modbus.



Совет

Если вы снова хотите настроить порт 0 на работу с PPI, чтобы он мог обмениваться данными со STEP 7-Micro/WIN, еще раз используйте команду MBUS_INIT, чтобы переназначит порт 0.

Вы можете также установить переключатель на S7-200 в положение STOP. Это сбрасывает параметры порта 0.

Инициализация и время исполнения протокола Modbus

Обмен данными с помощью протокола Modbus включает в себя контроль с помощью избыточного циклического кода (cyclic redundancy check, CRC), обеспечивающий целостность передаваемых сообщений. Протокол Modbus Slave использует таблицу заранее рассчитанных значений для уменьшения времени, необходимого для обработки сообщения. Инициализация таблицы CRC требует около 425 миллисекунд. Эта инициализация выполняется внутри подпрограммы MBUS_INIT, и обычно это делается в первом цикле программы пользователя после перехода в режим RUN. Вы отвечаете за сброс контроля времени и сохранение активности выходов (если это необходимо для модулей расширения), если время, необходимое подпрограмме MBUS_INIT и любой другой пользовательской инициализации, превышает 500 миллисекунд контроля времени. Контроль времени модуля вывода сбрасывается путем записи выходы модуля. См. описание команды сброса контроля времени в главе 6.

Время цикла увеличивается, когда подпрограмма MBUS_SLAVE обрабатывает запрос. Так как наибольшая часть времени тратится на расчет CRC Modbus, то время цикла увеличивается примерно на 650 миллисекунд на каждый байт, содержащийся в запросе и в ответе. Максимальное увеличение времени цикла на один запрос или ответ (чтение или запись 120 слов) составляет примерно 165 миллисекунд.

Адресация Modbus

Адреса Modbus обычно записываются как 5- или 6-символьные значения, содержащие тип данных и смещение. Один или два первых символа определяют тип данных, а четыре последних символа указывают надлежащее значение внутри типа данных. Затем master-устройство Modbus отображает эти адреса в соответствующие функции. Командами Modbus Slave поддерживаются следующие адреса:

- от 000001 до 000128 – цифровые выходы, отображаемые в Q0.0 – Q15.7
- от 010001 до 010128 – цифровые входы, отображаемые в I0.0 – I15.7
- от 030001 до 030032 – регистры аналоговых входов, отображаемые в AIW0 – AIW62
- от 040001 до 04xxxx – регистры временного хранения информации, отображаемые в память переменных.

Все адреса Modbus имеют базис, равный 1. В таблице 12–1 показано отображение адресов Modbus в адреса S7–200.

Протокол Modbus Slave позволяет ограничить количество входов, выходов, аналоговых входов и регистров временного хранения информации (памяти переменных), к которым может обращаться master-устройство Modbus. Параметр MaxIQ команды MBUS_INIT указывает максимальное количество цифровых входов или выходов (I или Q), к которым разрешен доступ master-устройству Modbus.

Параметр MaxIQ команды MBUS_INIT указывает максимальное количество цифровых входов или выходов (I или Q), к которым разрешен доступ master-устройству Modbus.

Параметр MaxHold команды MBUS_INIT указывает максимальное количество регистров временного хранения информации (памяти переменных), к которым Modbus разрешен доступ master-устройству Modbus.

Параметр MaxHold команды MBUS_INIT указывает максимальное количество регистров временного хранения информации (памяти переменных), к которым Modbus разрешен доступ master-устройству Modbus.

Таблица 12–1. Отображение адресов Modbus в S7–200

Адрес Modbus	Адрес S7–200
000001	Q0.0
000002	Q0.1
000003	Q0.2
...	...
000127	Q15.6
000128	Q15.7
010001	I0.0
010002	I0.1
010003	I0.2
...	...
010127	I15.6
010128	I15.7
030001	AIW0
030002	AIW2
030003	AIW4
...	...
030032	AIW62
040001	HoldStart
040002	HoldStart+2
040003	HoldStart+4
...	...
04xxxx	HoldStart+2 x (xxxx-1)

Конфигурирование таблицы символов

После ввода адреса для первого символа таблица автоматически рассчитывает остальные символы и размещает их в таблице.

Вы должны назначить для таблицы, которая занимает 779 байтов, начальный адрес в памяти переменных. Обратите внимание на то, чтобы назначение символов Modbus Slave не пересекалось с памятью переменных, назначенной регистрам временного хранения информации Modbus с помощью параметров HoldStart и MaxHold команды MBUS_INIT. При пересечении областей памяти команда MBUS_INIT выдает ошибку.

Использование команд протокола Modbus Slave

Для использования команд протокола Modbus Slave в своей программе для S7–200 действуйте следующим образом:

1. Вставьте команду MBUS_INIT в свою программу и выполните эту команду только в одном цикле. Команду MBUS_INIT можно использовать для инициализации или для изменения параметров обмена данными протокола Modbus.
Когда вы вставляете команду MBUS_INIT, к вашей программе автоматически добавляется несколько скрытых подпрограмм и программ обработки прерываний.
2. Назначьте начальный адрес для 779 последовательных байтов в памяти переменных для команд протокола Modbus Slave.
3. Поместите в своей программе только одну команду MBUS_SLAVE. Эта команда вызывается в каждом цикле для обслуживания поступающих запросов.
4. Подключите коммуникационный кабель к порту 0 на S7–200 и к master-устройствам Modbus.

Предостережение

Соединение друг с другом оборудования с различными опорными потенциалами может вызвать протекание нежелательных токов через соединительный кабель. Эти нежелательные токи могут вызвать ошибки в передаче данных или повреждение оборудования.

Чтобы предотвратить протекание нежелательных токов, убедитесь, что все оборудование, которое соединено при помощи коммуникационного кабеля, либо совместно использует общую опорную точку для электрических цепей, либо гальванически развязано.

Аккумуляторы (AC0, AC1, AC2, AC3) используются командами Modbus Slave и появляются в списке перекрестных ссылок. Перед выполнением команды Modbus Slave значения, находящиеся в аккумуляторах, сохраняются, а перед завершением этой команды восстанавливаются в аккумуляторах, обеспечивая сохранение всех пользовательских данных в аккумуляторах, когда выполняется команда Modbus Slave.

Команды протокола Modbus Slave поддерживают протокол Modbus RTU. Эти команды используют свободно программируемый обмен данными S7–200 для поддержки наиболее часто используемых функций Modbus. Поддерживаются следующие функции Modbus:

Таблица 12–2. Поддерживаемые функции протокола Modbus Slave

Функция	Описание
1	Чтение состояния одного или нескольких цифровых выходов. Функция 1 возвращает состояние (вкл/выкл) любого количества выходов (Q).
2	Чтение состояния одного или нескольких цифровых входов. Функция 2 возвращает состояние (вкл/выкл) любого количества входов (I).
3	Чтение одного или нескольких регистров временного хранения информации. Функция 3 возвращает содержимое памяти переменных. Регистры временного хранения информации в Modbus – это значения в формате слова, служащие для чтения до 120 слов в одном запросе.
4	Чтение одного или нескольких входных регистров. Функция 4 возвращает значения на аналоговых входах.
5	Запись одного цифрового выхода. Функция 5 устанавливает на указанное значение один цифровой выход. Этот выход не является принудительно установленным, и программа может перезаписать значение, записанное запросом Modbus.
6	Запись одного регистра временного хранения информации. Функция 6 записывает одно значение регистра временного хранения информации в память переменных S7–200.
15	Запись нескольких цифровых выходов. Функция 15 записывает значения нескольких цифровых выходов в регистр образа процесса на выходах Q модуля S7–200. Первый выход должен начинаться на границе байта (например, Q0.0 или Q2.0), а количество записываемых выходов должно быть кратно восьми. Это ограничение для команд протокола Modbus Slave. Эти выходы не являются принудительно установленными, и программа может перезаписать значения, записанные запросом Modbus.
16	Запись нескольких регистров временного хранения информации. Функция 16 записывает значения нескольких регистров временного хранения информации в память переменных S7–200. В одном запросе можно записать до 120 слов.

Команды протокола Modbus Slave

Команда MBUS_INIT

Команда MBUS_INIT используется для активизации и инициализации или деактивизации обмена данных через протокол Modbus. Прежде чем сможет быть применена команда MBUS_SLAVE, должна быть без ошибок выполнена команда MBUS_INIT. При завершении этой команды немедленно устанавливается бит Done [Готово], прежде чем будет выполняться следующая команда.

Команда выполняется в каждом цикле, когда включен вход EN.

Команда MBUS_INIT должна выполняться ровно один раз для каждого изменения состояния обмена данными. Поэтому вход EN должен включаться импульсно с помощью элемента обнаружения фронта или исполняться только в первом цикле.

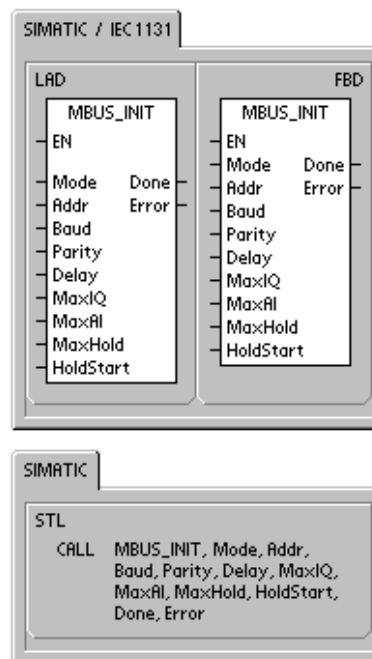
Значение входа Mode [Режим] определяет протокол обмена данными: значение входа, равное 1, назначает порт 0 протоколу Modbus и активизирует этот протокол, а значение входа, равное 0, назначает порт 0 протоколу PPI и деактивизирует протокол Modbus.

Параметр Baud устанавливает скорость передачи на 1200, 2400, 4800, 9600, 19200, 38400, 57600 или 115200 Бод. Скорости передачи 57600 и 115200 поддерживаются CPU S7-200, начиная с версии 1.2.

Параметр Addr устанавливает для адреса значения от 1 до 247 (включительно).

Таблица 12-3. Параметры для команды MBUS_INIT

Входы/выходы	Тип данных	Операнды
Mode, Addr, Parity	BYTE	VB, IB, QB, MB, SB, SMB, LB, AC, константа, *VD, *AC, *LD
Baud, HoldStart	DWORD	VD, ID, QD, MD, SD, SMD, LD, AC, константа, *VD, *AC, *LD
Delay, MaxIQ, MaxAI, MaxHold	WORD	VW, IW, QW, MW, SW, SMW, LW, AC, константа, *VD, *AC, *LD
Done	BOOL	I, Q, M, S, SM, T, C, V, L
Error	BYTE	VB, IB, QB, MB, SB, SMB, LB, AC, *VD, *AC, *LD



Параметр Parity соответствует контролю четности, используемому master-устройством Modbus. Допустимы следующие значения:

- 0—проверка четности отсутствует
- 1—проверка на нечетность
- 2— проверка на четность

Параметр Delay [Задержка] продлевает обычное для Modbus условие контроля времени ожидания конца сообщения на указанное количество миллисекунд. Обычное значение для этого параметра при работе в подключенной сети должно быть равно 0. Если вы используете модемы с исправлением ошибок, установите задержку на значение от 50 до 100 миллисекунд. Если вы используете радиомодем с расширенной полосой частот, установите задержку на значение от 10 до 100 миллисекунд. Значение параметра Delay может находиться в пределах от 0 до 32767 миллисекунд.

Параметр MaxIQ устанавливает количество входов и выходов, предоставляемых в распоряжение адресам Modbus 00xxxx и 01xxxx, в диапазоне от 0 до 128. Значение 0 блокирует все операции чтения и записи для входов и выходов. Предлагаемое значение для MaxIQ равно 128, что делает возможным обращение ко всем входам и выходам S7-200.

Параметр MaxAI устанавливает количество входных регистров, имеющих размер слова (AI), предоставляемых в распоряжение адресу Modbus 03xxx, в диапазоне от 0 до 32. Значение 0 блокирует чтение аналоговых входов. Предлагаемое значение для MaxAI, допускающее обращение ко всем аналоговым входам S7-200:

- 0 для CPU 221
- 16 для CPU 222
- 32 для CPU 224, CPU 224XP и CPU 226

Параметр MaxHold устанавливает количество имеющих размер слова регистров для временного хранения информации в памяти переменных, предоставляемых в распоряжение адресу Modbus 04xxx. Например, чтобы дать master-устройству возможность обращаться к 2000 байтов памяти переменных, установите MaxHold на значение 1000 слов (регистры временного хранения информации).

Параметр HoldStart – это адрес начала регистров для временного хранения информации в памяти переменных. Это значение обычно устанавливается на VB0, так что параметр HoldStart устанавливается на &VB0 (адрес VB0). В качестве начального адреса для регистров временного хранения информации могут быть указаны и другие адреса в памяти переменных, чтобы дать возможность использования VB0 в каком-нибудь другом месте проекта. Master-устройство Modbus имеет доступ к количеству слов в памяти переменных, определяемому параметром MaxHold, начиная с адреса, указанного в HoldStart.

Когда команда MBUS_INIT завершается, то включается выход Done [Готово]. Выходной байт Error [Ошибка] содержит результат выполнения команды. Таблица 12-5 определяет ошибочные состояния, которые могут возникнуть в результате выполнения команды.

Команда MBUS_SLAVE

Команда MBUS_SLAVE используется для обслуживания запроса от master-устройства Modbus и должна выполняться в каждом цикле, чтобы дать ей возможность контролировать и реагировать на запросы Modbus.

Команда выполняется в каждом цикле, когда включен вход EN.

Команда MBUS_SLAVE не имеет входных параметров.

Выход Done [Готово] включен, когда команда MBUS_SLAVE отвечает на запрос Modbus. Выход Done выключен, если запросы не обрабатывались.

Выход Error [Ошибка] содержит результат выполнения команды. Этот выход действителен, если включен выход Done. Если Done выключен, то параметр Error не изменяется. Таблица 12–5 определяет ошибочные состояния, которые могут возникнуть в результате выполнения команды.

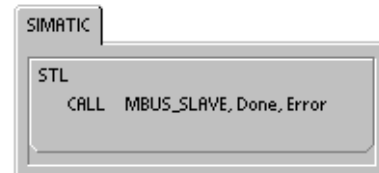
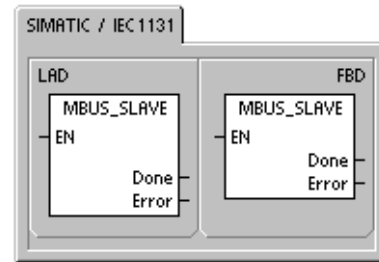


Таблица 12–4. Параметры для команды MBUS_SLAVE

Параметр	Тип данных	Операнды
Done	BOOL	I, Q, M, S, SM, T, C, V, L
Error	BYTE	VB, IB, QB, MB, SB, SMB, LB, AC, *VD, *AC, *LD

Таблица 12–5. Коды ошибок исполнения протокола Modbus Slave

Коды ошибок	Описание
0	Нет ошибки
1	Ошибка области памяти
2	Недопустимая скорость передачи или четность
3	Недопустимый адрес slave-устройства
4	Недопустимое значение для параметра Modbus
5	Регистры для временного хранения информации пересекаются с символами Modbus Slave
6	Ошибка проверки четности при приеме
7	Ошибка CRC при приеме
8	Недопустимый запрос функции/функция не поддерживается
9	Недопустимый адрес памяти в запросе
10	Функция slave-устройства не активизирована

