Введение.

Данный документ является сокращенным переводом описания стандартного протокола MODBUS фирмы MODICON GOULD. Исходный текст на английском языке можно получить по URL: http://www.modicon.com/techpubs/toc7.html.

Следует особо отметить изменения в описании функции 17. Список параметров прибора, получаемый с помощью данной функции, отличается от стандартного.

Список приборов поддерживающих стандартный протокол MODBUS.

Название прибора	Версия ПО
ВЗЛЕТ-РС(УРСВ-010М)	35.12.02.00
ВЗЛЕТ-ПР	11.13.00.00

Список приборов поддерживающих нестандартный протокол MODBUS.

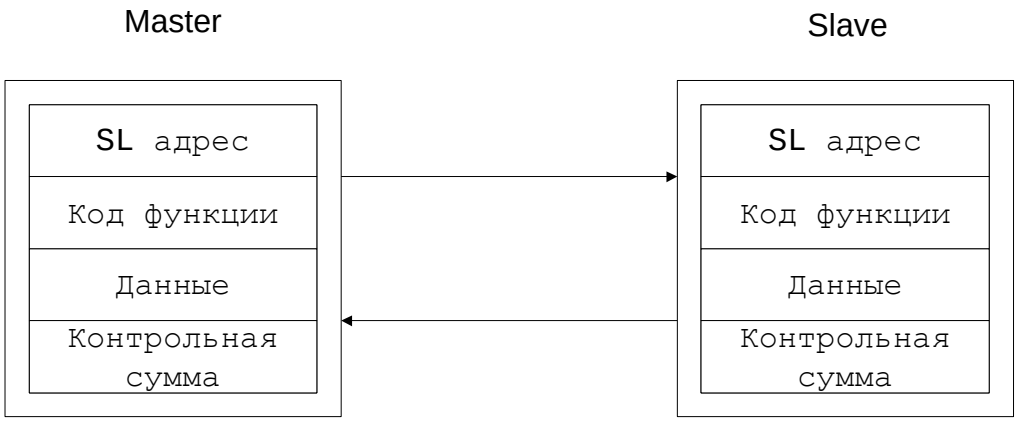
Название прибора	Версия ПО
ВЗЛЕТ-РС(УРСВ-010)	1.13-2.28
ВЗЛЕТ-РС(УРСВ-010М)	1.41-1.43
ВЗЛЕТ-ПР	2.00-2.02
ВЗЛЕТ-РТ	01.00.0 - 01.01.1, 53.12.00.00

Содержание.

Содержание

Протокол MODBUS.

Протокол необходимая часть работы системы. Он определяет как Master (MS) и Slave (SL) устанавливают и прерывают контакт, как идентифицируются отправитель и получатель, каким образом происходит обмен сообщениями, как обнаруживаются ошибки. Протокол управляет циклом запроса и ответа, который происходит между устройствами MS и SL, как показано на рисунке.



Протокол подразумевает на общей шине один MS и до 247 SL. Хотя протокол и поддерживает до 247 SL, некоторые приборы ограничивают число SL, подключаемых к общей шине. Например, драйвер шины расходомера-счетчика УРСВ-10М позволяет подключить к одному сегменту двухпроводной линии RS485 максимум 32 прибора. Каждому SL присвоен уникальный адрес устройства в диапазоне от 1 до 247.

Только MS может инициировать транзакцию. Транзакции бывают либо типа запрос/ответ (адресуется только один SL), либо широковещательные/без ответа (адресуются все SL). Транзакция содержит один кадр запроса и один кадр ответа, либо один кадр широковещательного запроса.

Некоторые характеристики протокола Modbus фиксированы. К относятся формат кадра, последовательность кадров, обработка ошибок коммуникации и исключительных ситуаций, и выполнение функций.

Другие характеристики выбираются пользователем. К ним относятся тип средства связи, скорость обмена, проверка на четность, число стоповых бит, и режим передачи (ASCII или RTU). Параметры, выбираемые пользователем, устанавливаются (аппаратно или программно) на каждой станции. Эти параметры не могут быть изменены во время работы системы.

При передаче по линиям данных, сообщения помещаются в «конверт». «Конверт» покидает устройство через «порт» и «пересылается» по линиям адресуемому устройству. Протокол Modbus описывает «конверт» в форме кадров сообщений. Информация в сообщении представляет адрес требуемого получателя, что получатель должен сделать, данные, необходимые для выполнения этого, и механизм контроля достоверности.

Когда сообщение достигает интерфейса SL, оно попадает в адресуемое устройство через похожий «порт». Адресуемое устройство вскрывает конверт, читает сообщение, и, если не возникло ошибок, выполняет требуемую задачу. Затем оно помещает в конверт ответное сообщение и посылает его «отправителю». Информация в ответном сообщении представляет собой адрес адресуемого устройства, выполненную задачу, данные, полученные в результате выполнения задачи, и механизм контроля достоверности. Если сообщение было широковещательным (сообщение для всех SL), на что указывает адрес 0, то ответное сообщение не передается.

В большинстве случаях, MS посылает следующее сообщение другому SL либо после приема корректного ответного сообщения, либо после прохождения определенного пользователем интервала времени, если ответное сообщение не был получено. Все сообщения могут рассматриваться как запросы, генерирующие ответные сообщения от SL. Широковещательные сообщения могут рассматриваться как запросы, не требующие ответных сообщений от SL.

Содержание

Режимы передачи.

Режим передачи определяет структуру отдельных блоков информации в сообщении и системы счисления, используемую для передачи данных. В системе Modbus существуют два режима передачи. Оба режима обеспечивают одинаковую совместимость при связи с SL. Режим выбирается в зависимости от оборудования, используемого как Master Modbus. Для каждой системы Modbus должен использоваться только один режим. Смешивание режимов не дозволительно. Режимы делятся на ASCII и RTU (Remote Terminal Unit).

Таблица 1.Характеристики режимов ASCII и RTU.

Характеристика	ASCII (7-бит)	RTU(8-бит)
Система кодирования	Используются ASCII символы 0-9,A-F	8-битовая двоичная система
Число бит на символ		
Стартовые биты	1	1
Биты данных (LSB вперед)	7	8
Четность	Вкл./Выкл.	Вкл./Выкл.
Стоповые биты	1 или 2	1 или 2
Контрольная сумма	LRC (Longitudinal Redundancy Check). LRC	CRC (Cyclical Redundancy Check). CRC_16

Символы ASCII удобнее использовать при отладке, поэтому этот режим удобен для компьютеров, программируемых на языке высокого уровня, например, FORTRAN. Режим RTU подходит для компьютеров, программируемых на машинных языках.

В режиме RTU данные передаются в виде 8-ми разрядных двоичных символов. В режиме ASCII каждый RTU символ сначала делится на две 4-х разрядных части (старший и младший), переводится в свой шестнадцатеричный эквивалент и затем используется в создании сообщения. ASCII режим использует в два раза больше символов, чем RTU режим, но декодирование и управление данными - легче. К тому же, в режиме RTU символы сообщения должны передаваться непрерывным потоком. В режиме ASCII допустима задержка до 1 секунды между двумя соседними символами.

Содержание

Обнаружение ошибок.

Существует два типа ошибок, которые могут возникать в системах связи: ошибки передачи и программные или оперативные ошибки. Система Modbus имеет способы определения каждого типа ошибок.

Ошибки связи обычно заключаются в изменении бита или бит сообщения. Например, байт 0001 0100 может измениться на 0001 0110. Ошибки связи выявляются при помощи символа кадра, контроля по четности и избыточным кодированием.

Когда обнаруживается ошибка кадрирования, четности и контрольной суммы, обработка сообщения прекращается. SL не должен генерировать ответное сообщение.(Тот же результат достигается если был использован адрес несуществующего SL).

Если возникает ошибка связи, данные сообщения ненадежны. Устройство SL не может с уверенностью определить, что сообщение было адресовано именно ему. Иначе SL может ответить сообщением, которое не является ответом на исходный запрос. Устройство MS должно программироваться так, чтобы в случае не получения ответного сообщения в течение определенного времени, MS должен фиксировать ошибку связи. Продолжительность этого времени зависит от скорости обмена, типа сообщения, и времени опроса SL. По истечению этого периода, MS должен быть запрограммирован на ретрансляцию сообщения.

Оба режима передачи, RTU и ASCII, могут включать в формат символа дополнительный бит четности. В режиме RTU это девятый бит в поле данных (8 бит данных и бит четности). В режиме ASCII это восьмой бит данных (7 бит данных и бит четности). Если контроль четности не используется, бит четности не передается. Все устройства в системе должны быть сконфигурированы одинаково.

Контроль четности может определить только изменение одного бита в символе. Изменение двух битов в символе контроль четности определить не в состоянии.

Для обеспечения качества передачи данных система Modbus обеспечивает несколько уровней обнаружения ошибок. Для обнаружения множественного изменения битов сообщения система использует избыточный контроль: CRC и LRC. Какой контроль использовать зависит от режима передачи. RTU использует CRC, а ASCII использует LRC. Расчет CRC и LRC описан ниже. Обнаружение ошибок с помощью CRC и LRC выполняется автоматически.

Содержание

CRC-16 (Cyclic Redundancy Check).

Сообщение (только биты данных, без учета старт/стоповых бит и бит четности) рассматриваются как одно последовательное двоичное число, у которого старший значащий бит(MSB) передается первым. Сообщение умножается на X^{16} (сдвигается влево на 16 бит), а затем делится на $X^{16}+X^{15}+X^2+1$, выражаемое как двоичное число (11000000000000101). Целая часть результата игнорируется, а 16-ти битный остаток (предварительно инициализированный единицами для предотвращения случая, когда все сообщение состоит из нулей) добавляется к сообщению (старшим битом вперед) как два байта контрольной суммы. Полученное сообщение, включающее CRC, затем в приемнике делится на тот же полином ($X^{16}+X^{15}+X^2+1$). Если ошибок

не было, остаток от деления должен получится нулевым.(Приемное устройство может рассчитать CRC и сравнить ее с переданной). Вся арифметика выполняется по модулю 2 (без переноса).Пример_расчета_CRC.

Устройство, используемое для подготовки данных для передачи, посылает условно самый правый (LSB) бит каждого символа первым. При расчете CRC, первый передаваемый бит, определен как MSB делимого. Так как арифметика не использует перенос, для удобства расчета CRC можно предположить, что MSB расположен справа. Поэтому порядок бит при расчете полинома должен быть реверсивным. MSB полинома опускается, так как он влияет только на делитель, а не на остаток. В результате получается 1010 0000 0000 0001 (A001H). Заметьте, что эта реверсивность порядка бит, в любом случае, не влияет на интерпретацию или порядок бит байт данных при вычислении CRC.

Пошаговая процедура расчета CRC-16 представлена ниже:

1. Загрузить 16-ти разрядный регистр числом FFFFH.
2. Выполнить операцию XOR над первым байтом данных и старшим байтом регистра. Поместить результат в регистр.
3. Сдвинуть регистр на один разряд вправо.
4. Если выдвинутый вправо бит единица, выполнить операцию XOR между регистром и полиномом 1010 0000 0000 0001 (A001H).
5. Если выдвинутый бит ноль, вернуться в шагу 3.
6. Повторять шаги 3 и 4 до тех пор, пока не будут выполнены 8 сдвигов регистра.
7. Выполнить операцию XOR над следующим байтом данных и регистром.
8. Повторять шаги 3-7 до тех пор, пока не будут выполнена операция XOR над всеми байтами данных и регистром.
9. Содержимое регистра представляет собой два байта CRC и добавляется к исходному сообщению старшим битом вперед.

Рисунок 1. Пример расчета CRC для сообщения - чтение статуса SL с номером 02.

16-ти разрядный регистр				MSB	Флаг
Исключающее ИЛИ	1111	1111	1111	1111	
02			0000	0010	
	1111	1111	1111	1101	
Сдвиг 1	0111	1111	1111	1110	1
Полином	1010	0000	0000	0001	
	1101	1111	1111	1111	
Сдвиг 2	0110	1111	1111	1111	1
Полином	1010	0000	0000	0001	
	1100	1111	1111	1110	
Сдвиг 3	0110	0111	1111	1111	
Сдвиг 4	0011	0011	1111	1111	1
Полином	1010	0000	0000	0001	
	1001	0011	1111	1110	
Сдвиг 5	0100	1001	1111	1111	
Сдвиг 6	0010	0100	1111	1111	1
Полином	1010	0000	0000	0001	
	1000	0100	1111	1110	
Сдвиг 7	0100	0010	0111	1111	
Сдвиг 8	0010	0001	0011	1111	1
Полином	1010	0000	0000	0001	
	1000	0001	0011	1110	
07			0000	0111	

16-ти разрядный регистр				MSB	Флаг
	1000	0001	0011	1001	
Сдвиг 1	0100	0000	1001	1100	1
Полином	1010	0000	0000	0001	
	1110	0000	1001	1101	
Сдвиг 2	0111	0000	0100	1110	1
Полином	1010	0000	0000	0001	
	1101	0000	0100	1111	
Сдвиг 3	0110	1000	0010	0111	1
Полином	1010	0000	0000	0001	
	1100	1000	0010	0110	
Сдвиг 4	0110	0100	0001	0011	
Сдвиг 5	0011	0010	0000	1001	1
Полином	1010	0000	0000	0001	
	1001	0010	0000	1000	
Сдвиг 6	0100	1001	0000	0100	
Сдвиг 7	0010	0100	1000	0010	
Сдвиг 8	0001	0010	0100	0001	
	HEX 12		HEX 41		

Передаваемое сообщение с контрольной суммой CRC-16 (При передаче сообщение выдвигается вправо)			
12	41	07	02
0001 0010	0100 0001	0000 0111	0000 0010
Последний бит	Порядок передачи		Первый бит

Содержание

LRC(Longitudinal Redundancy Check).

Контрольная сумма в режиме ASCII это LRC. Контрольная сумма - это 8-ми разрядное число, передаваемое как два ASCII символа (hex). Контрольная сумма образуется путем конвертирования всех hex символов в двоичные числа, сложением этих чисел без учета переноса, и вычислением дополнительного кода полученного числа. В приемнике LRC заново рассчитывается и сравнивается с полученным LRC. При вычислении LRC двоеточие, CR, LF и любой другой не-ASCII символ отбрасывается.

Рисунок 2. Пример расчета LRC для сообщения - чтение первых 8-ми булевых ячеек SL с номером 02.

Адрес	0	2			0000	0010
Функция	0	1			0000	0001
Адрес первой ячейки (HIGH)	0	0			0000	0000
Адрес первой ячейки(LOW)	0	0			0000	0000
Число ячеек(HIGH)	0	0			0000	0000
Число ячеек(LOW)	0	8		+	0000	1000
					0000	1011
				Инверсия	1111	0100
				+1		1
					1111	0101
Контрольная сумма	F	5			F	5
Приемное устройство складывает все байты данных, включая LRC. Сумма может превышать 8 бит, в расчет принимаются только младшие 8 бит.				 ОК	0000 0000 0000 0000 0000 0000	0010 0001 0000 0000 0000 1000

				К.сумма Сумма	1111 0000	0101 0000

Содержание

Протокол Modbus.

Как было отмечено в введении, протокол описывает правила связи между MS и SL. Протокол Modbus разбит по типу коммуникаций, используемых в промышленных сетях. В общем, интерпретация полей в сообщении идентична для режимов передачи ASCII и RTU.(См. рис. 1-4 и 1-5). Главное отличие заключается в типе проверки контрольной суммы, выполняемой над сообщением, и которое в два раза больше в режиме ASCII. Вместо передачи 80ми разрядного двоичного символа, посылается эквивалент в виде пары 7-ми разрядных ASCII (0-9,A-F) символов.

Содержание

Кадровая синхронизация в режиме ASCII.

В режиме ASCII достигается использованием символа двоеточия ':', указывающего начало кадра, и символов возврата каретки (CR) и перевода строки (LF), указывающих на конец кадра. Символ перевода строки также служит как синхронизирующий символ, который указывает на то, что передающая станция готова для приема ответного сообщения.

Рисунок 3. Формат кадра сообщения в режиме ASCII.

Начало кадра	Адрес	Функция	Данные	Контрольная сумма	EOF	Готовность приема ответного сообщения
:	2 символа 16-бит	2 символа 16 бит	N * 4 символа N * 16 бит	2 символа 16 бит	CR	LF

Содержание

Кадровая синхронизация в режиме RTU.

В режиме RTU может поддерживаться только путем эмулирования синхронного сообщения. Приемное устройство отслеживает время между приемом символов. Если прошло время, равное периоду следования 3.5 символов, а кадр не был завершен или не поступило нового символа, устройство очищает кадр и предполагает, что следующий принимаемый байт - это адрес устройства в новом сообщении.

Рисунок 4. Формат кадра сообщения в режиме RTU.

T1 T2 T3	Адрес	Функция	Данные	Контрольная сумма	T1 T2 T3
	8 бит	8 бит	N * 8 бит	16 бит	

Содержание

Поле адреса.

Поле адреса следует сразу за началом кадра и состоит их одного 8-ми разрядного символа в режиме RTU или 2-х символов в режиме ASCII. Эти биты указывают пользователю адрес SL устройства, которое должно принять сообщение, посланное MS.

Каждый SL должен иметь уникальный адрес и только адресуемое устройство может ответить на запрос, который содержит его адрес. Когда SL посылает ответ, адрес SL информирует MS, с какой SL на связи. В широковещательном режиме используется адрес 0. Все SL интерпретируют такое сообщение как выполнение определенного действия, но без посылки подтверждения.

Содержание

Поле функции

Поле кода функции указывает адресуемому SL какое действие выполнить. Коды функций Modbus специально разработаны для связи ПК и индустриальных коммуникационных систем Modbus.

Старший бит этого поля устанавливается в единицу SL в случае, если он хочет просигналить MS, что ответное сообщение не нормальное. (Смотри). Этот бит остается в нуле, если ответное сообщение повторяет запрос или в случае нормального сообщения.

Таблица 2. Коды функций Modbus.

Код	Название	Действие
01	READ COIL STATUS	Получение текущего состояния (ON/OFF) группы логических ячеек.
02	READ INPUT STATUS	Получение текущего состояния (ON/OFF) группы дискретных входов.
03	READ HOLDING REGISTERS	Получение текущего значения одного или нескольких регистров хранения.
04	READ INPUT REGISTERS	Получение текущего значения одного или нескольких входных регистров.
05	FORCE SINGLE COIL	Изменение логической ячейки в состояние ON или OFF.
06	FORCE SINGLE REGISTER	Запись нового значения в регистр хранения.
07	READ EXCEPTION STATUS	Получение состояния (ON/OFF) восьми внутренних логических ячеек, чье назначение зависит от типа контроллера. Пользователь может использовать эти ячейки по своему выбору.
08	LOOPBACK DIAGNOSTIC TEST	Тестовое сообщение, посылаемое SL для получения данных о связи.
11	FETCH EVENT COUNTER COMMUNICATIONS	Позволяет MS путем последовательной посылки одного сообщения определить выполнение операции.
12	FETCH COMMUNICATIONS EVENT LOG	Позволяет MS получить журнал связи, который содержит информацию о каждой Modbus транзакции данного SL. Если транзакция не выполнена, в журнал заносится информация об ошибки.
13	PROGRAM	Позволяет MS программировать SL.
14	POLL PROGRAM COMPLETE	Позволяет MS связываться с другими SL если один SL выполняет долговременную операцию программирования. SL периодически опрашивается на момент завершения программирования. Данный запрос посылается только в том случае, если предварительно был послан запрос

Код	Название	Действие
		PROGRAM.
15	FORCE MULTIPLE COILS	Изменить состояние (ON/OFF) нескольких последовательных логических ячеек.
16	FORCE MULTIPLE REGISTERS	Установить новые значения нескольких последовательных регистров.
17	REPORT SLAVE I.D.	Позволяет MS определить тип адресуемого SL и его рабочее состояние.
19	RESET COMMUNICATIONS LINK	Сбрасывает SL в известное состояние после неустранимой ошибки. Сбрасывает счетчик принятых байт.
20-64	Зарезервировано под расширения Modbus	
65-72	Зарезервировано под пользовательские функции.	В дальнейшем не будет использоваться в продуктах Modicon.
73-119	ILLEGAL FUNCTION	
120-127	Зарезервировано	Зарезервировано Modicon для внутреннего использования.
128-255	Зарезервировано	Зарезервировано для исключительных ситуаций.

Содержание

Поле данных.

Поле данных содержит информацию, необходимую SL для выполнения указанной функции, или содержит данные собранные SL для ответа на запрос.

Содержание

Поле контрольной суммы.

Это поле позволяет MS и SL проверять сообщение на наличие ошибок. Иногда, вследствие электрических помех или других воздействий, сообщение при пересылке от одного устройства к другому может незначительно измениться. Результат проверки контрольной суммы укажет SL или MS реагировать или не реагировать на такое сообщение. Это увеличивает надежность и эффективность систем MODBUS.

В режиме ASCII в поле контрольной суммы используется LRC, а в режиме RTU – CRC.

Если сообщения запроса и ответа могли бы читаться по-английски, то четыре поля этих сообщений выглядели как на рисунке. (Заметьте, что последовательность посылки полей каждый раз одна и та же – Адрес, Код функции, Данные и Контрольная сумма – независимо от направления.)

MODBUS MS	ERROR CHECK	DATA	FUNCTION CODE (03)	ADDRESS (01)	MODBUS SL
⇒	Информация используется приемным устройством для проверки сообщения	Относительный адрес регистра	Чтение регистра хранения	Запрос для SL с номером 1	⇒
	ADDRESS (01)	FUNCTION CODE (03)	DATA	ERROR CHECK	

⇐	Ответ от SL с номером 1	Чтение регистра хранения	Значение, содержащееся в указанном регистре хранения	Информация, используемая приемным устройством для проверки сообщения	⇐

Содержание

Исключительные ситуации.

Коды исключительных ситуаций приведены в таблице. Когда SL обнаруживает одну их этих ошибок, он посылает ответное сообщение MS, содержащее адрес SL, код функции, код ошибки и контрольную сумму. Для указания на то, что ответное сообщение – это уведомление об ошибке, старший бит поля кода функции устанавливается в 1. На рисунке и представлен пример некорректного запроса и соответствующего ответа с кодом исключительной ситуации.

Код	Название	Смысл
01	ILLEGAL FUNCTION	Функция в принятом сообщении не поддерживается на данном SL. Если тип запроса – POLL PROGRAM COMPLETE, этот код указывает, что предварительный запрос не был командой программирования.
02	ILLEGAL DATA ADDRESS	Адрес, указанный в поле данных, является недопустимым для данного SL.
03	ILLEGAL DATA VALUE	Значения в поле данных недопустимы для данного SL.
04	FAILURE IN ASSOCIATED DEVICE	SL не может ответить на запрос или произошла авария.
05	ACKNOWLEDGE	SL принял запрос и начал выполнять долговременную операцию программирования. Для определения момента завершения операции используйте запрос типа POLL PROGRAM COMPLETE. Если этот запрос был послан до завершения операции программирования, то SL ответит сообщением REJECTED MESSAGE.
06	BUSY, REJECTED MESSAGE	Сообщение было принято без ошибок, но SL в данный момент выполняет долговременную операцию программирования. Запрос необходимо ретранслировать позднее.
07	NAK-NEGATIVE ACKNOWLEDGMENT	Функция программирования не может быть выполнена. Используйте опрос для получения детальной аппаратно-зависимой информации об ошибке.

Адрес SL	Функция	Старший байт адреса	Младший байт адреса	Старший байт числа ячеек	Младший байт числа ячеек	Контрольная сумма
0A	01	04	A1	00	01	4F

Этот запрос требует состояние ячейки с номером 1245 в SL с номером 10, и, если этот контроллер имеет 1К ячеек, то этот адрес является ошибочным. Соответственно, будет сгенерировано следующее ответное сообщение.

Адрес SL	Функция	Код исключительной ситуации	Контрольная сумма
0A	81	02	73

Значение в поле функции равно оригинальному значению с установленным в единицу старшим битом. Код исключительной ситуации 02 указывает на ошибочный адрес данных.

Содержание

Детальное описание функций MODBUS.

Цель данного раздела определить общий формат соответствующих команд, доступных программисту системы MODBUS. В разделе описаны формат каждого запросного сообщения, выполняемая функция и формат нормального ответного сообщения.
Сообщения с номерами функций 1-6, 15 и 16 ссылаются на конкретные доступные переменные программируемого контроллера. Функция 1, 5 и 15 ссылаются на логические ячейки (0XXX(X)), функция 2 на дискретные входы (1XXX(X)), функция 4 на входные регистры (3XXX(X)), функции 3,6 и 16 на внутренние регистры (4XXX(X)). Все адреса ссылок в сообщениях MODBUS индексируются с нуля. Например, первый внутренний регистр в контроллере 584, будучи 40001-ым, имеет адрес ссылки 0. Точно также, ячейка 00127 будет иметь адрес 0126. Примеры в данном разделе демонстрируют протокол независимо от режима RTU или ASCII. Программист может использовать следующий метод для корректировки пакета в зависимости от режима передачи.
Во всем разделе протокол будет представлен по возможности в формате, указанном на Рис.3-1. Числа имеют шестнадцатеричный формат.

Адрес	Функция	Старший байт адреса первого регистра	Младший байт адреса первого регистра	Старший байт числа требуемых регистров	Младший байт числа требуемых регистра	Поле контрольной суммы	
06	03	00	6B	00	03	89	LRC

Данный пример описывает чтение регистров 4108-4110 из SL с адресом 06. Это сообщение при форматировании в RTU и ASCII выглядит следующим образом:

ЗАПРОС		RTU		ASCII	
Заголовок					:
Адрес		0000	0110	0	6
Функция		0000	0011	0	3
Начальный адрес	H.O.	0000	0000	0	0
	L.O.	0110	1011	6	B
Количе	H.O.	0000	0000	0	0

ЗАПРОС		RTU		ASCII	
ство требуемых регистров	L.O.	0000	0011	0	3
Поле контрольной суммы		0111	0101	8	9
		1010	0000		
Trailer				CR	LF
ОТВЕТ		RTU		ASCII	
Заголовок					:
Адрес		0000	0110	0	6
Функция		0000	0011	0	3
Количество байт данных		0000	0110	0	6
Данные	H.O	0000	0010	0	2
	L.O.	0010	1011	2	B
	H.O.	0000	0000	0	0
	L.O.	0000	0000	0	0
	H.O.	0000	0000	0	0
	L.O.	0110	0011	6	3
Контрольная сумма		CRC		6	1
Trailer				CR	LF
Длина пакета		11 байт		23 байта	

ASCII сообщение всегда почти в два раза длинее RTU сообщения.

Содержание

Функция 1. Чтение логических ячеек.

Запрос.
Функция позволяет пользователю получить статус (1/0) логических ячеек. Широковещательный режим не поддерживается. Помимо полей адреса SL и функции, сообщение требует, чтобы информационное поле содержало логический адрес первой ячейки и число ячеек, статус которых необходимо получить.
Адресация позволяет получить за один запрос до 2000 логических ячеек. Однако, некоторые приборы имеют ограничение на максимальное число ячеек, статус которых можно получить за один запрос. Ячейки нумеруются с нуля (ячейка 1 = 0, ячейка 2 = 1 и т.д.).
Ниже представлен запрос на чтение логических ячеек 0020 – 0056 из прибора с адресом 17.

Адрес	Функция	Старший байт адреса первой ячейки	Младший байт адреса первой ячейки	Старший байт число ячеек	Младший байт числа ячеек	Контрольная сумма	
11	01	00	13	00	25	B6	LRC

Ответ.
Ниже представлен пример ответного сообщения на предыдущий запрос.

Адрес	Функция	Количество байт в поле данных	Статус ячеек 20-27	Статус ячеек 28-35	Статус ячеек 36-43	Статус ячеек 44-51	Статус ячеек 52-56	Контрольная сумма
11	01	05	CD	6B	B2	0E	1B	D6

Данные в поле данных упакованы один бит на каждую ячейку. Ответное сообщение включает адрес SL, код функции, число байт в поле данных, данные и контрольную сумму. Младший значащий бит первого байта поля данных содержит первую адресуемую ячейку, за которой следуют остальные. Если число ячеек не кратно 8-ми, то остальные биты заполняются нулями в порядке от старших битов к младшим.

Статус ячеек 20-27 равен CDH = 1101 1101. Читая слева направо, видим, что ячейки 27, 26, 23, 22 и 20 установлены. Остальные данные разбираются так же. Так как было запрошено число ячеек не кратное 8-ми, старшие три бита в последнем байте данных (1BH) заполнены нулями.

Так как запрос обслуживается в конце рабочего цикла прибора, то данные в ответном сообщении отражают состояние ячеек на тот момент.

Содержание

Функция 2. Чтение дискретных входов.

Запрос.

Данная функция позволяет пользователю получить состояние(ВКЛ/ВЫКЛ) входных дискретных линий адресуемого SL. Широковещательный запрос не поддерживается. В дополнение к адресу SL и номеру функции, запрос требует, чтобы информационное поле содержало начальный адрес и количество требуемых линий.

Адресация позволяет получить за один запрос до 2000 линий. Однако, некоторые устройства имеют ограничение на максимальное количество линий, получаемых за один запрос. Входные линии нумеруются с нуля (10001 = 0, 10002 = 1 и т.д.).

На рис.3-2 представлен пример запроса на чтение дискретных входов 10197-10218 из SL с номером 17.

Адрес	Функция	Старший байт номера первой требуемой ячейки	Младший байт номера первой требуемой ячейки	Старший байт количество требуемых ячеек	Младший байт количество требуемых ячеек	Контрольная сумма	
11	02	00	C4	00	16	13	LRC

Ответ.

Пример ответа на данный запрос представлен на рис.3-5.

Ответное сообщение включает адрес SL, код функции, количество байт данных, данные и поле контрольной суммы. Данные упакованы по биту на каждый вход (1 = ON, 0 = OFF). Младший бит первого байта содержит значение первого адресуемого входа, за которым следуют остальные. Если количество запрошенных входов не кратно 8, то остальные биты заполняются нулями. Количество байт данных всегда определяется как количество RTU данных.

Так как SL обслуживает запрос в конце рабочего цикла, данные в ответе отражают состояние входов на данный момент. Некоторые устройства имеют ограничение на максимальное количество входов, запрашиваемых за один запрос.

Адрес	Функция	Количество байт данных	Дискретные входы 10197-10204	Дискретные входы 10205-10212	Дискретные входы 10213-10218	Контрольная сумма	
11	02	03	AC	DB	35	2E	LRC

Статус входов 10197-10204 = ACh = 1010 1100. Читая слева направо, видим, что входы 10204, 10202, 10200 и 10199 в состоянии ON. Все остальные байты данных распаковываются аналогично.

Так как было запрошено 22 линии, последний байт данных (35h = 0011 0101) содержит только 6 входов (10213-10218) вместо 8-ми. Два последних бита заполняются нулями.

Содержание

Функция 3. Чтение регистров.

Запрос.

Данная функция позволяет получить двоичное содержимое 16-ти разрядных регистров адресуемого SL. Адресация позволяет получить за каждый запрос до 125 регистров. Однако, некоторые устройство имеют ограничение на максимальное количество регистров, получаемых за один запрос. Регистры нумеруются с нуля (40001 = 0, 40002 = 1 и т.д.).

Широковещательный режим не допускается.

Ниже представлен пример запроса на чтение регистров 40108-40110 из SL с адресом 17.

Адрес	Функция	Номер первого регистра		Число регистров для чтения (N)		Контрольная сумма	
		Старший байт	Младший байт	Старший байт	Младший байт		
11	03	00	6B	00	03	7E	LRC

Ответ.

Адресуемый SL посылает в ответе свой адрес, код выполненной функции и информационное поле. Информационное поле содержит 2 байта, описывающих количество возвращаемых байт данных. Длина каждого регистра данных – 2 байта. Первый байт данных в посылке является старшим байтом регистра, второй – младшим.

Так как SL обычно обслуживает запрос в конце своего рабочего цикла, данные в ответе отражают содержимое регистров в данный момент. Некоторые SL ограничивают количество регистров, передаваемых за один запрос. В этом случае для получения, большего числа регистров, необходимо выполнить несколько последовательных запросов.

Ниже представлен пример ответного сообщения на чтение регистров 40108-40110, имеющих содержимое, соответственно, 555, 0, 100, из SL с адресом 17.

Адрес	Функция	Количество байт	Старший байт	Младший байт	Старший байт	Младший байт	Старший байт	Младший байт	Контрольная сумма

		данны х	регист ра 40108	регист ра 40108	регист ра 40109	регист ра 40109	регист ра 40110	регист ра 40110		
11	03	06	02	2B	00	00	00	64	55	L R C

Содержание

Функция 5. Запись одной ячейки.

Запрос.

Это сообщение модифицирует одну логическую ячейку. Ячейки нумеруются с нуля (ячейка 1 = 0, ячейка 2 = 1 и т.д.). Число 65280 (FF00H) устанавливает ячейку в 1, а число 0 – в 0. Другие числа не влияют на содержимое ячейки. Данная функция может использоваться в широковещательном режиме.

Ниже приведен пример установки в 1 ячейки 0173 в SL 17.

Адрес	Функция	Старший байт адреса ячейки	Младший байт адреса ячейки	Индикатор установки или сброс ячейки	Всегда 0	Контрольн ая сумма	
11	05	00	AC	FF	00	3F	LRC

Ответ.

Нормальное ответное сообщение полностью совпадает с запросом.

Содержание

Функция 6. Запись одного регистра.

Запрос.

Данная функция позволяет модифицировать содержимое одного регистра. Хотя запрос и является асинхронным, SL изменяет содержимое регистра только в конце рабочего цикла.

Когда в запросе указан адрес равный 0 (широковещательный запрос), все SL, подключенные к шине, загрузят соответствующий регистр указанным значением.

ПРИМЕЧАНИЕ. В широковещательном режиме используются только функции 5, 6, 15 и 16.

Ниже приведен пример записи регистра 40136 значением 926 в SL с номером 17.

Адрес	Функция	Старший байт адреса регистра 40136	Младший байт адреса регистра 40136	Старший байт значения 926	Младший байт значения 926	Контрольн ая сумма	
11	06	00	87	03	9E	C1	LRC

Ответ.

В случае успешного выполнения функции ответное сообщение идентично запросу.

Содержание

Функция 8. Тестовая функция.

Запрос.

Данная функция предназначена для проверки коммуникационной системы и не влияет на данные прибора.

Поле информации содержит 2 байта диагностического кода, указывающего SL выполнить определенное действие, и 2 байта необходимой, для данной диагностики, информации.

Код	Действие
00	Вернуть запрос
01	Сбросить установки связи (без ответа)
02	Вернуть регистр диагностики
03	Изменить символ начала пакета
04	Перевести SL в режим прослушивания линии без посылки ответных сообщений (Listen Only Mode)
05	Сбросить счетчики и регистр диагностики
06	Вернуть счетчик сообщений, полученных с шины MODBUS.
07	Вернуть счетчик сообщений с неправильными контрольными суммами.
08	Вернуть счетчик сообщений, вызвавших исключительную ситуацию.
09	Вернуть счетчик сообщений, адресованных только данному SL.
10	Вернуть счетчик сообщений, адресованных данному SL и оставленных без ответа.
11	Вернуть счетчик сообщений, адресованных данному SL и вызвавшим исключительную ситуацию NACK.
12	Вернуть счетчик сообщений, адресованных данному SL и вызвавшим исключительную ситуацию BUSY.

Ниже дан пример запроса вернуть эхо (диагностический код 0) SL с номером 17.

Адрес	Функция	Старший байт диагностического кода	Младший байт диагностического кода	Старший байт данных[1]	Младший байт данных	Контрольная сумма	
11	08	00	00	00	00	0B	LRC

Ответ.

Адрес	Функция	Старший байт диагностического кода	Младший байт диагностического кода	Старший байт данных[2]	Младший байт данных	Контрольная сумма	
11	08	00	00	00	00	0B	LRC

Содержание

Функция 7. Чтение статуса.

Запрос.

Во многих случаях, для быстрого получения статуса некоторых событий контроллера, желательно иметь в протоколе сообщение, имеющее небольшой размер. Данная функция разработана именно для этой цели.

[1] В поле данных помещается необходимая для данного запроса информация.
[2] В поле данных помещается необходимая для данного ответа информация.

Функция с номером 7 позволяет пользователю опрашивать состояние восьми ячеек контроллера. Эти ячейки могут программироваться для хранения информации состояния контроллера. Широковещательный режим не поддерживается.

Назначение этих ячеек зависит от типа контроллера.

Ниже представлен пример запроса статуса SL с номером 17.

Адрес	Функция	Контрольная сумма	
11	07	E8	LRC

В этой функции не требуется поле данных.

Ответ.

Нормальный ответ содержит статус восьми ячеек, упакованных в один байт данных.

Адрес	Функция	Данные ячеек	Контрольная сумма	
11	07	6D	7B	LRC

В приборах ЗАО "ВЗЛЁТ" постоянно используются два младших разряда регистра статуса, которые отражают состояние прибора во время программирования памяти программ прибора.

Содержание

Функция 16. Запись нескольких регистров.

Запрос.

Данное сообщение меняет содержимое любого регистра опрашиваемого контроллера.

Сообщение позволяет записывать регистры с максимальным логическим адресом до FFFFH.

Неиспользуемые старшие биты адреса регистра должны заполняться нулями. Если используется адрес SL равный 0, то содержимое поля данных записывается во все устройства, подключенные к шине (широковещательный режим).

Ниже дан пример записи в SL с номером 17 двух регистров 40136, 40137 значениями 0x00a0, 0x0102.

Адрес	Функция	Старший байт адреса первого регистра	Младший байт адреса первого регистра	Количество регистров		Количество байт в поле данных	Старший байт регистра 40136	Младший байт регистра 40136	Старший байт регистра 40137	Младший байт регистра 40137	Контрольная сумма	
11	10	00	87	00	02	04	00	0A	01	02	45	LRC

Ответ.

Нормальное ответное сообщение возвращает адрес SL, функцию, адрес первого регистра и количество записанных регистров.

Адрес	Функция	Старший байт адрес первого регистра	Младший байт адреса первого регистра	Количество регистров		Контрольная сумма	
11	10	00	87	00	02	56	LRC

Содержание

Функция 17.Чтение информации об адресуемом устройстве.

Запрос.

Пример запроса прибору с адресом 17.

Адрес	Функция	Контрольная сумма	
11	11	DE	LRC

Ответ.

Общая форма ответного сообщения приведена ниже.

Адрес	Функция	Число байт в поле данных	Поле данных	Контрольная сумма

Информация в поле данных различна для каждого конкретного прибора и указана в протоколе на прибор.

Для приборов фирмы «ВЗЛЕТ» в поле данных обязательно передается следующая информация:

Название параметра	Формат
Версия прибора	ASCII строка, завершающаяся нулем в формате: “VZLJOT AA.BB.CC.DD”
Название прибора	ASCII строка, завершающаяся нулем.
Максимальное число регистров в таблице регистров прибора.	Unsigned (2 байта).

В случае если какая-либо из строк в приборе не существует, на ее месте в ответном сообщении должен передаваться нуль.

Остальная информация зависит от типа прибора, и указывается в описании протокола прибора.

Содержание

Глоссарий.

MS Master. Главное устройство, посылающее запрос.
SL Slave. Адресуемое подчиненное устройство, формирующее ответное сообщение.

Содержание