

ООО «НОВАТЕК-ЭЛЕКТРО»



Протокол обмена по интерфейсу RS-485

TP100-MODBUS

Руководство по программированию

WWW.NOVATEK-ELECTRO.COM

1. Общие положения.

Протокол TP100-MODBUS служит для организации обмена данными между TP-100 и персональным компьютером (программируемым логическим контроллером) по интерфейсу EIA/TIA-485. В основу протокола обмена положен протокол MODBUS-RTU. Их отличие заключается в поддержке прибором TP-100 ограниченного набора команд.

При построении сети используется принцип организации ведущий-ведомый (master-slave). В сети может присутствовать только один ведущий узел и несколько ведомых узлов. В качестве ведущего узла выступает персональный компьютер либо программируемый логический контроллер. При данной организации инициатором циклов обмена может выступать исключительно ведущий узел.

Запросы ведущего узла - индивидуальные (адресуемые к конкретному TP-100). Ведомые узлы осуществляют передачу, отвечая на индивидуальные запросы ведущего узла.

При обнаружении ошибок в получении запросов, либо невозможности выполнения полученной команды, ведомый узел, в качестве ответа, генерирует сообщение об ошибке.

2. Форматы сообщений.

Протокол обмена имеет четко определенные форматы сообщений. Описывается формат байт и формат кадров. Соблюдение форматов обеспечивает правильность и устойчивость функционирования сети.

2.1. Формат байта.

TP-100 настраивается на работу с одним из двух форматов байт данных: с контролем паритета (рис. 2.1) и без контроля паритета (рис. 2.2). В режиме работы с контролем паритета указывается также тип контроля: по четности (Even), либо по не четности (Odd). Передача восьми бит данных производится младшими битами вперед.

По умолчанию, при изготовлении, TP-100 настраивается на работу без контроля паритета и 2-я стоповыми битами.

Подробная информация содержится в руководстве по эксплуатации на TP-100.

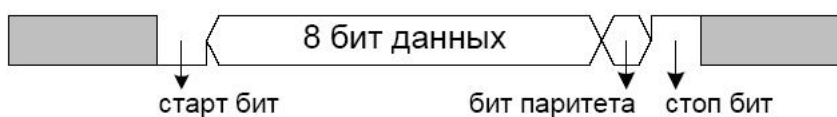


Рис. 2.1. – Формат байта с контролем паритета.



Рис. 2.2. – Формат байта без контроля паритета (2 стоп бита).

Передача байт осуществляется на скоростях (2400, 4800, 9600, 19200). По умолчанию, при изготовлении, TP-100 настраивается на работу со скоростью 9600 бит/с.

2.2. Формат кадра.

Длина кадра не может превышать 256 байт. Контроль начала и окончания кадра осуществляется при помощи интервалов молчания, длиной не менее времени передачи 3.5 байт.

Формат кадра приведен на рис. 2.3.

интервал молчания > 3.5 байта	Адрес	Код функции	Данные	Контрольная сумма	интервал молчания > 3.5 байта
	1 байт	1 байт	до 252 байт	2 байта	

Рис. 2.3. – Формат кадра.

Кадр должен передаваться как непрерывный поток байт. Правильность принятия кадра дополнительно контролируется проверкой контрольной суммы.

На рис. 2.4 приведен пример последовательной передачи 3 кадров. Перед началом передачи и после передачи каждого кадра интервал молчания превышает 3.5 байта.

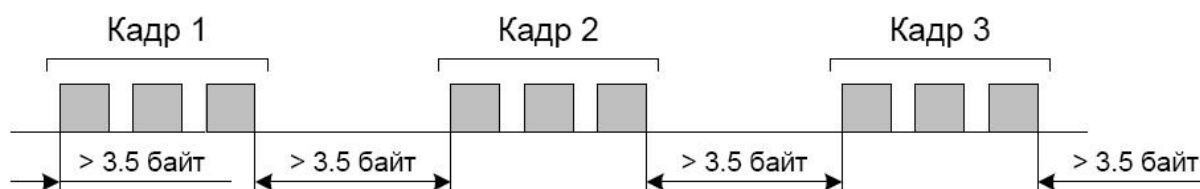


Рис. 2.4. – Пример передачи последовательности кадров

Интервал молчания задается в мс и может быть программно изменен. По умолчанию, при изготовлении TP-100, интервал молчания имеет значение 4 мс.

Поле адреса занимает один байт. Адреса ведомых устройств находятся в диапазоне 1...247.

4. Генерация и проверка контрольной суммы.

Контрольная сумма (CRC16) представляет собой циклический проверочный код на основе неприводимого полинома A001h. Передающее устройство формирует контрольную сумму для всех байт передаваемого сообщения. Принимающее устройство аналогичным образом формирует контрольную сумму для всех байт принятого сообщения и сравнивает ее с контрольной суммой, принятой от передающего устройства. При несовпадении сформированной и принятой контрольных сумм генерируется сообщение об ошибке.

Поле контрольной суммы занимает два байта. Контрольная сумма в сообщении передается младшим байтом вперед.

Можно выделить два способа формирования контрольной суммы: алгоритмический и табличный.

4.1. Формирование контрольной суммы алгоритмическим способом.

Алгоритмический способ формирования контрольной суммы прост в понимании, но имеет более низкое быстродействие по сравнению с табличным. Контрольная сумма формируется по следующему алгоритму:

1. загрузка CRC регистра (16 бит) единицами (FFFFh);
2. исключающее ИЛИ с первыми 8 битами байта сообщения и содержимым CRC регистра;
3. сдвиг результата на один бит вправо;
4. если сдвигаемый бит = 1, исключающее ИЛИ содержимого регистра со значением A001h;
5. если сдвигаемый бит = 0, повторить шаг 3;
6. повторять шаги 3, 4, 5 пока не будут выполнены 8 сдвигов;
7. исключающее ИЛИ со следующими 8 битами байта сообщения и содержимым CRC регистра;
8. повторять шаги 3 – 7 пока все байты сообщения не будут обработаны;
9. конечное содержимое регистра будет содержать контрольную сумму.

Подпрограмма алгоритмического формирования контрольной суммы на языке Паскаль:

```
type
  TsendRecvBuf : array [0..255] of byte;
const
  Polynom = $A001;
function GenerateCRC(Buf: TsendRecvBuf; Count: word): word;
var
  i : word;
  CRC : word;
  bitCounter : byte;
begin
  CRC := $FFFF;
  for i := 0 to Count - 3 do
  begin
    CRC := CRC xor Buf[i];
    for bitCounter := 0 to 7 do
    begin
      if (CRC and $0001) = 0 then
        CRC := CRC shr 1
      else
      begin
        CRC := CRC shr 1;
        CRC := CRC xor Polynom;
      end;
    end;
  end;
  result := CRC;
end;
```

4.2. Формирование контрольной суммы табличным способом.

Подпрограмма табличного формирования контрольной суммы на языке ассемблера микроконтроллера 8051:

```

Table1: ; 00/08 01/09 02/0A 03/0B 04/0C 05/0D 06/0E 07/0F
db 000h, 0C1h, 081h, 040h, 001h, 0C0h, 080h, 041h ;00
db 001h, 0C0h, 080h, 041h, 000h, 0C1h, 081h, 040h
db 001h, 0C0h, 080h, 041h, 000h, 0C1h, 081h, 040h ;01
db 000h, 0C1h, 081h, 040h, 001h, 0C0h, 080h, 041h
db 000h, 0C1h, 081h, 040h, 001h, 0C0h, 080h, 041h ;02
db 000h, 0C1h, 081h, 040h, 001h, 0C0h, 080h, 041h
db 001h, 0C0h, 080h, 041h, 000h, 0C1h, 081h, 040h ;03
db 001h, 0C0h, 080h, 041h, 000h, 0C1h, 081h, 040h
db 000h, 0C1h, 081h, 040h, 001h, 0C0h, 080h, 041h ;04
db 000h, 0C1h, 081h, 040h, 001h, 0C0h, 080h, 041h
db 000h, 0C1h, 081h, 040h, 001h, 0C0h, 080h, 041h ;05
db 001h, 0C0h, 080h, 041h, 000h, 0C1h, 081h, 040h
db 000h, 0C1h, 081h, 040h, 001h, 0C0h, 080h, 041h ;06
db 001h, 0C0h, 080h, 041h, 000h, 0C1h, 081h, 040h
db 000h, 0C1h, 081h, 040h, 001h, 0C0h, 080h, 041h ;07
db 000h, 0C1h, 081h, 040h, 001h, 0C0h, 080h, 041h
db 001h, 0C0h, 080h, 041h, 000h, 0C1h, 081h, 040h ;08
db 000h, 0C1h, 081h, 040h, 001h, 0C0h, 080h, 041h
db 000h, 0C1h, 081h, 040h, 001h, 0C0h, 080h, 041h ;09
db 000h, 0C1h, 081h, 040h, 001h, 0C0h, 080h, 041h
db 001h, 0C0h, 080h, 041h, 000h, 0C1h, 081h, 040h ;0A
db 001h, 0C0h, 080h, 041h, 000h, 0C1h, 081h, 040h
db 000h, 0C1h, 081h, 040h, 001h, 0C0h, 080h, 041h ;0B
db 000h, 0C1h, 081h, 040h, 001h, 0C0h, 080h, 041h
db 001h, 0C0h, 080h, 041h, 000h, 0C1h, 081h, 040h ;0C
db 001h, 0C0h, 080h, 041h, 000h, 0C1h, 081h, 040h
db 000h, 0C1h, 081h, 040h, 001h, 0C0h, 080h, 041h ;0D
db 001h, 0C0h, 080h, 041h, 000h, 0C1h, 081h, 040h
db 000h, 0C1h, 081h, 040h, 001h, 0C0h, 080h, 041h ;0E
db 000h, 0C1h, 081h, 040h, 001h, 0C0h, 080h, 041h
db 001h, 0C0h, 080h, 041h, 000h, 0C1h, 081h, 040h ;0F
db 001h, 0C0h, 080h, 041h, 000h, 0C1h, 081h, 040h

Table2: ; 00/08 01/09 02/0A 03/0B 04/0C 05/0D 06/0E 07/0F
db 000h, 0C0h, 0C1h, 001h, 0C3h, 003h, 002h, 0C2h ;00
db 0C6h, 006h, 007h, 0C7h, 005h, 0C5h, 004h, 004h
db 0CCh, 00Ch, 00Dh, 0CDh, 00Fh, 0CFh, 0CEh, 00Eh ;01
db 00Ah, 0CAh, 0CBh, 00Bh, 0C9h, 009h, 008h, 0C8h
db 0D8h, 018h, 019h, 0D9h, 01Bh, 0DBh, 0DAh, 01Ah ;02
db 01Eh, 0DEh, 0DFh, 01Fh, 0DDh, 01Dh, 01Ch, 0DCh
db 014h, 0D4h, 0D5h, 015h, 0D7h, 017h, 016h, 0D6h ;03
db 0D2h, 012h, 013h, 0D3h, 011h, 0D1h, 0D0h, 010h
db 0F0h, 030h, 031h, 0F1h, 033h, 0F3h, 0F2h, 032h ;04
db 036h, 0F6h, 0F7h, 037h, 0F5h, 035h, 034h, 0F4h
db 03Ch, 0FCh, 0FDh, 03Dh, 0FFh, 03Fh, 03Eh, 0FEh ;05
db 0FAh, 03Ah, 03Bh, 0FBh, 039h, 0F9h, 0F8h, 038h
db 028h, 0E8h, 0E9h, 029h, 0EBh, 02Bh, 02Ah, 0EAh ;06
db 0EEh, 02Eh, 02Fh, 0EFh, 02Dh, 0EDh, 0ECh, 02Ch
db 0E4h, 024h, 025h, 0E5h, 027h, 0E7h, 0E6h, 026h ;07
db 022h, 0E2h, 0E3h, 023h, 0E1h, 021h, 020h, 0E0h
db 0A0h, 060h, 061h, 0A1h, 063h, 0A3h, 0A2h, 062h ;08
db 066h, 0A6h, 0A7h, 067h, 0A5h, 065h, 064h, 0A4h
db 06Ch, 0ACh, 0ADh, 06Dh, 0AFh, 06Fh, 06Eh, 0AEh ;09
db 0AAh, 06Ah, 06Bh, 0ABh, 069h, 0A9h, 0A8h, 068h
db 078h, 0B8h, 0B9h, 079h, 0BBh, 07Bh, 07Ah, 0BAh ;0A
db 0BEh, 07Eh, 07Fh, 0BFh, 07Dh, 0BDh, 0BCh, 07Ch
db 0B4h, 074h, 075h, 0B5h, 077h, 0B7h, 0B6h, 076h ;0B
db 072h, 0B2h, 0B3h, 073h, 0B1h, 071h, 070h, 0B0h
db 050h, 090h, 091h, 051h, 093h, 053h, 052h, 092h ;0C
db 096h, 056h, 057h, 097h, 055h, 095h, 094h, 054h
db 09Ch, 05Ch, 05Dh, 09Dh, 05Fh, 09Fh, 09Eh, 05Eh ;0D
db 05Ah, 09Ah, 09Bh, 05Bh, 099h, 059h, 058h, 098h
db 088h, 048h, 049h, 089h, 04Bh, 08Bh, 08Ah, 04Ah ;0E
db 04Eh, 08Eh, 08Fh, 04Fh, 08Dh, 04Dh, 04Ch, 08Ch
db 044h, 084h, 085h, 045h, 087h, 047h, 046h, 086h ;0F
db 082h, 042h, 043h, 083h, 041h, 081h, 080h, 040h

```

```

;При входе R2 – количество байт в сообщении,
; Addr_buff – начальный адрес буфера сообщения
;При выходе CRC16 – младший байт контрольной суммы (LB),
; CRC16+1 - старший байт контрольной суммы (HB).

```

```

GenerateCRC:
    mov R0,#Addr_buff ;
    mov CRC16,#0FFh ;
    mov CRC16+1,#0FFh ;
$11:
    mov A,@R0 ;
    xrl A,CRC16 ;
    mov B,A ;
    mov DPTR,#Table1 ;
    movc A,@A+DPTR ;
    mov CRC16,CRC16+1 ;
    xrl CRC16,A ;
    mov A,B ;
    mov DPTR,#Table2 ;
    movc A,@A+DPTR ;
    mov CRC16+1,A ;
    inc R0 ;
    djnz R2,$11 ;
    ret

```

5. Описание системы команд.

5.1. Функция 03h – чтение группы регистров.

Функция 03h обеспечивает чтение содержимого регистров ведомого устройства. Широковещательная передача не поддерживается. В запросе ведущего содержится адрес начального регистра, а также количество слов для чтения.

Ответ ведомого содержит количество возвращаемых байт и запрошенные данные. Если количество слов в запросе превышает 120 (240 байт), разбиение ответа на кадры не производится. Количество возвращаемых регистров ограничивается 120.

Пример запроса и ответа приведён на рис. 5.1.

Запрос

Адрес	Функция	Нач. адрес HB	Нач. адрес LB	Кол. слов HB	Кол. слов LB	CRC LB	CRC HB
01h	03h	00h	A0h	00h	02h	C4h	29h

Ответ - значение регистра 00A0h = 1000 (FLOAT)

Адрес	Функция	Кол. байт	Данные HW HB	Данные HW LB	Данные LW HB	Данные LW LB	CRC LB	CRC HB
01h	03h	04h	44h	7Ah	00h	00h	CFh	1Ah

Рис. 5.1. - Пример запроса и ответа функции 03h – чтение группы регистров.

5.2. Функция 06h – установка регистра.

Функция 06h обеспечивает запись в регистр ведомого устройства. Широковещательная передача не поддерживается. В запросе ведущего содержится адрес регистра и данные для записи.

Ответ ведомого совпадает с запросом ведущего и содержит адрес регистра и установленные данные. Пример запроса и ответа приведён на рис. 5.2.

Функция записи имеет аппаратные и программные ограничения, описанные в разделе 6 “Адресное пространство”.

Запрос - регистр 00A0h = 1000 (INT)

Адрес	Функция	Нач. адрес HB	Нач. адрес LB	Данные HB	Данные LB	CRC LB	CRC HB
01h	06h	00h	A0h	03h	E8h	89h	56h

Ответ

Адрес	Функция	Нач. адрес HB	Нач. адрес LB	Данные HB	Данные LB	CRC LB	CRC HB
01h	06h	00h	A0h	03h	E8h	89h	56h

Рис. 5.2. - Пример запроса и ответа функции 06h – установка регистра.

5.3. Функция 08h – диагностика.

Функция 08h обеспечивает ряд тестов для проверки системы связи между ведущим и ведомым, а также для проверки различных внутренних условий ведомого. Широковещательная передача не поддерживается.

Функция использует поле подфункции для конкретизации выполняемого действия (теста). Получение диагностической функции не должно оказывать влияние на работу ведомого устройства.

Подфункция 00h - возврат данных запроса.

Данные, переданные в поле данных запроса, должны быть возвращены в поле данных ответа.

Пример запроса и ответа приведён на рис. 5.3.

Запрос

Адрес	Функция	Подфункция HB	Подфункция LB	Данные HB	Данные LB	CRC LB	CRC HB
01h	08h	00h	00h	A0h	3Ch	98h	1Ah

Ответ

Адрес	Функция	Подфункция HB	Подфункция LB	Данные HB	Данные LB	CRC LB	CRC HB
01h	08h	00h	00h	A0h	3Ch	98h	1Ah

Рис. 5.3. - Пример запроса и ответа подфункции 00h - возврат данных запроса.

Подфункция 01h – рестарт опций связи.

Периферийный порт ведомого устройства должен быть инициализирован и вновь запущен.

Если устройство находится в режиме “только слушать”, ответ не возвращается, а устройство возвращается в нормальный режим работы. Пример запроса и ответа приведён на рис. 5.4.

Запрос

Адрес	Функция	Подфункция HB	Подфункция LB	Данные HB	Данные LB	CRC LB	CRC HB
01h	08h	00h	01h	00h	00h	B1h	CBh

Ответ не возвращается

Рис. 5.4. - Пример запроса и ответа подфункции 01h - рестарт опций связи.

Подфункция 04h – установить режим “только слушать”.

Вынуждает адресованное ведомое устройство перейти в режим “только слушать”. Команда изолирует ведомое устройство от других устройств сети, исключая его влияние на процесс обмена. Ответ не возвращается. Все последующие команды, адресованные данному ведомому, будут приниматься, но ответы возвращаться не будут. Выход из режима “только слушать” возможен лишь при получении команды диагностики с подфункцией 01h - рестарт опций связи. Пример запроса и ответа приведён на рис. 5.5.

Запрос

Адрес	Функция	Подфункция НВ	Подфункция ЛВ	Данные НВ	Данные ЛВ	CRC LB	CRC НВ
01h	08h	00h	04h	00h	00h	A1h	CAh

Ответ не возвращается

Рис. 5.5. - Пример запроса и ответа подфункции 04h - установить режим “только слушать”.

5.4. Обработка ошибок.

В случае возникновения ошибочной ситуации при принятии кадра (ошибка паритета, ошибка кадра, ошибка контрольной суммы) ведомое устройство ответ не возвращает.

В случае возникновения ошибки в формате или значении передаваемых данных (неподдерживаемый код функции и т. д.) ведомое устройство принимает кадр запроса и формирует ответ с признаком и кодом ошибки. Признаком ошибки является установленный в единицу старший бит в поле функции. Под код ошибки отводится отдельное поле в ответе. Пример ответа приведен на рис. 5.7. Коды ошибок приведены в таблице 5.1.

Запрос - функция 30h не поддерживается

Адрес	Функция	Данные	CRC LB	CRC НВ
01h	30h		XXh	XXh

Ответ

Адрес	Функция	Код ошибки	CRC LB	CRC НВ
01h	В0h	01h	94h	00h

Рис. 5.7. - Пример ответа после возникновения ошибки.

Таблица 5.1. - Коды ошибок.

Код ошибки	Название	Описание
01h	ILLEGAL FUNCTION	Принятый код функции не может быть обработан.
02h	ILLEGAL DATA ADDRESS	Адрес данных указанный в запросе не доступен.
03h	ILLEGAL DATA VALUE	Величина, содержащаяся в поле данных запроса, является недопустимой величиной.
04h	SLAVE DEVICE FAILURE	Пока ТР-100 пытался выполнить затребованное действие, произошла не восстанавливаемая ошибка.
07h	NEGATIVE ACKNOWLEDGE	ТР-100 не может выполнить программную функцию, принятую в запросе

6. Адресное пространство.

Все регистры адресного пространства устройства доступны для чтения. Для некоторых регистров запрещена операция записи.

Адреса регистров приведены в Руководстве по эксплуатации устройства.

7. Полезные ссылки.

Дополнительную информацию можно найти по следующим адресам:

1. <http://www.novatek-electro.com/> – официальный сайт ООО «НОВАТЕК-ЭЛЕКТРО»;
2. <http://www.modbus.org/> - сайт, посвященный стандарту MODBUS;
3. <http://www.rs485.com/> - сайт, посвященный последовательным интерфейсам EIA/TIA-232, EIA/TIA-422, EIA/TIA-485;