



**НАУЧНО-ПРОИЗВОДСТВЕННОЕ ПРЕДПРИЯТИЕ
“МИКРОТЕРМ”**

Протокол обмена по интерфейсу RS-485

MTM-MODBUS

Руководство по программированию

г. Северодонецк
2004

Содержание

1. Общие положения.....	3
2. Форматы сообщений.....	4
2.1. Формат байта.....	4
2.2. Формат кадра.....	4
3. Форматы данных.....	6
3.1. Формат FLOAT.....	6
3.2. Формат INT.....	6
3.3. Формат WORD.....	7
3.4. Формат BOOL.....	7
4. Генерация и проверка контрольной суммы.....	8
4.1. Формирование контрольной суммы алгоритмическим способом.....	8
4.2. Формирование контрольной суммы табличным способом.....	8
5. Описание системы команд.....	11
5.1. Функция 03h – чтение группы регистров.....	11
5.2. Функция 06h – установка регистра.....	11
5.3. Функция 08h – диагностика.....	11
5.4. Функция 10h – установка группы регистров.....	12
5.5. Обработка ошибок.....	13
6. Адресное пространство.....	15
6.1. Регистры, общие для всех устройств.....	15
6.2. Регистры прибора MTM 120.....	17
6.3. Регистры прибора MTM 292.....	18
6.4. Регистры прибора MTM 900.....	19
7. Полезные ссылки.....	20

1. Общие положения.

Протокол MTM-MODBUS служит для организации обмена данными между приборами ООО НПП “Микротерм” и персональным компьютером (программируемым логическим контроллером) по интерфейсу EIA/TIA-485. В основу протокола обмена положен протокол MODBUS-RTU. Их отличие заключается в поддержке приборами ООО НПП “Микротерм” 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 (в частности MTM4000CU), в качестве ведомых узлов – приборы ООО НПП “Микротерм”. При данной организации инициатором циклов обмена может выступать исключительно ведущий узел.

Запросы ведущего узла - индивидуальные (адресуемые к конкретному прибору). 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.

Приборы настраиваются на работу с одним из двух форматов байт данных: с контролем паритета (рис. 2.1) и без контроля паритета (рис. 2.2, 2.3). В режиме работы с контролем паритета указывается также тип контроля: по четности (Odd), либо по не четности (Even). В режиме работы без контроля паритета указывается количество стоп битов (1 или 2). Передача восьми бит данных производится младшими битами вперед.

По умолчанию, при изготовлении, приборы настраиваются на работу без контроля паритета. Некоторые приборы могут не поддерживать режимы работы с контролем паритета. Подробная информация содержится в руководстве по эксплуатации на прибор.

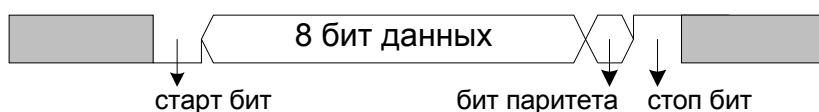


Рис. 2.1. – Формат байта с контролем паритета.



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



Рис. 2.3. – Формат байта без контроля паритета (1 стоп бит).

Передача байт осуществляется на скоростях, кратных 1200 бит/с (1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200). По умолчанию, при изготовлении, приборы настраиваются на работу со скоростью 19200 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.

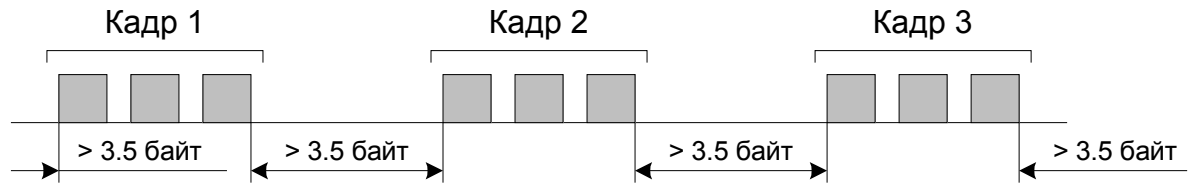


Рис. 2.4. – Пример передачи последовательности кадров.

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

Поле адреса занимает один байт. В связи с тем, что нагрузочная способность сетевого оборудования составляет 32 устройства, реальные адреса ведомых устройств находятся в диапазоне 1...31.

3. Форматы данных.

Приборы ООО НПП “Микроterm” имеют четыре формата программно-доступных регистров (таблица 3.1). Протокол обмена не имеет средств указания типа передаваемых данных. Тип конкретного регистра определяется исключительно описанием адресного пространства прибора. Вся ответственность по толкованию типа ложиться на программное обеспечение верхнего уровня.

Таблица 3.1 - Форматы программно-доступных регистров.

Тип	Размерность	Диапазон	Примечание
FLOAT	4 байта	-1,175,494,35E-38.....3,402,823,47E+39	Вещественное с плавающей точкой
INT	2 байта	-32,768.....+32,767	Знаковое целое
WORD	2 байта	0.....FFFFh	Беззнаковое целое
BOOL	2 байта	0000h.....FFFFh	Битовый массив – 16 бит

3.1. Формат FLOAT.

Формат вещественное с плавающей точкой (стандарт IEEE) представлен на рис. 3.1. Данные передаются старшим байтом и старшим словом вперед.

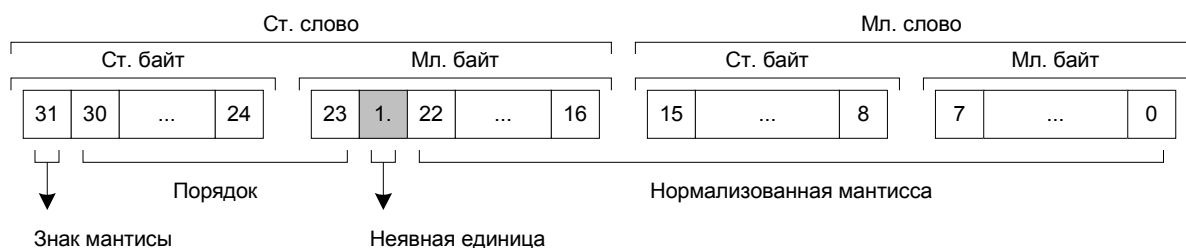


Рис. 3.1. - Формат FLOAT.

Порядок представлен с положительным смещением. Т. е. для получения реального значения порядка из него необходимо вычесть 127. Мантисса нормализованная, т. е. не имеет незначащих разрядов. В числе всегда присутствует неявная целая единица, но в коде числа место под нее не отводится.

На рис. 3.2. приведен пример кодирования числа 1000 в формате FLOAT.

31	30	29	28	27	26	25	24	23	1.	22	21	20	19	18	17	16	15	...	0
0	1	0	0	0	1	0	0	0	1.	1	1	1	1	0	1	0	0	0	0
+	88h - 7Fh = 136 - 127 = 9								1.111101										
1.111101 * 2^9 = 1111101000 = 03E8h = 1000																			

Рис. 3.2. - Пример кодирования числа 1000 в формате FLOAT.

3.2. Формат INT.

Формат целое знаковое представлен на рис. 3.3. Данные передаются старшим байтом вперед. На рис. 3.4. приведен пример кодирования числа -1000 в формате INT.

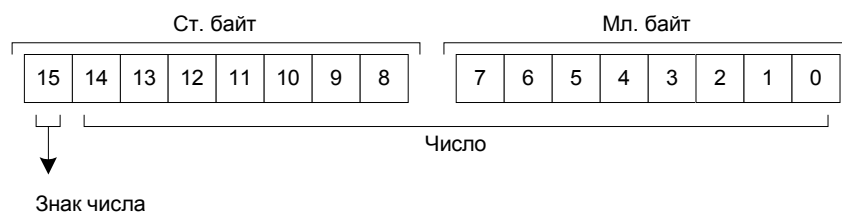


Рис. 3.3. - Формат INT.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	1	1	1	1	1	0	0	0	0	0	1	1	0	0	0
-	FC18h = -1000														

Рис. 3.4. - Пример кодирования числа -1000 в формате INT.

3.3. Формат WORD.

Формат целое беззнаковое представлен на рис. 3.5. Данные передаются старшим байтом вперед. На рис. 3.6. приведен пример кодирования числа 1000 в формате WORD.

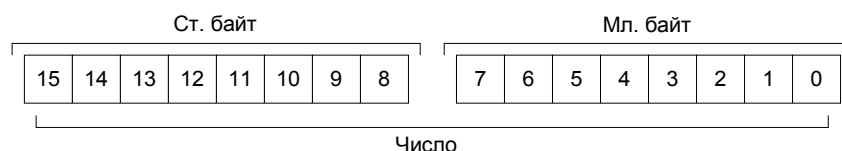


Рис. 3.5. - Формат WORD.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	1	1	1	1	1	0	1	0	0	0
03E8h = 1000															

Рис. 3.6. - Пример кодирования числа 1000 в формате WORD.

3.4. Формат BOOL.

Формат BOOL предназначен для кодирования дискретных величин. Один регистр типа BOOL содержит состояние шестнадцати логических переменных. Адресация переменных начинается с младших разрядов. Значение 1 соответствует TRUE (истина), 0 – FALSE (ложь).

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;
      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	001h,	0C0h,	080h,	041h,	000h,	0C1h,	081h,	040h	;02
db	000h,	0C1h,	081h,	040h,	001h,	0C0h,	080h,	041h	
db	000h,	0C1h,	081h,	040h,	001h,	0C0h,	080h,	041h	;03
db	001h,	0C0h,	080h,	041h,	000h,	0C1h,	081h,	040h	
db	001h,	0C0h,	080h,	041h,	000h,	0C1h,	081h,	040h	;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	001h,	0C0h,	080h,	041h,	000h,	0C1h,	081h,	040h	;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	001h,	0C0h,	080h,	041h,	000h,	0C1h,	081h,	040h	
db	000h,	0C1h,	081h,	040h,	001h,	0C0h,	080h,	041h	;0A
db	001h,	0C0h,	080h,	041h,	000h,	0C1h,	081h,	040h	
db	001h,	0C0h,	080h,	041h,	000h,	0C1h,	081h,	040h	;0B
db	000h,	0C1h,	081h,	040h,	001h,	0C0h,	080h,	041h	
db	000h,	0C1h,	081h,	040h,	001h,	0C0h,	080h,	041h	;0C
db	001h,	0C0h,	080h,	041h,	000h,	0C1h,	081h,	040h	
db	001h,	0C0h,	080h,	041h,	000h,	0C1h,	081h,	040h	;0D
db	000h,	0C1h,	081h,	040h,	001h,	0C0h,	080h,	041h	
db	001h,	0C0h,	080h,	041h,	000h,	0C1h,	081h,	040h	;0E
db	000h,	0C1h,	081h,	040h,	001h,	0C0h,	080h,	041h	
db	000h,	0C1h,	081h,	040h,	001h,	0C0h,	080h,	041h	;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,	0C4h,	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. Адресация регистров начинается с нуля. Различия между запросом чтения регистров различных типов не существует. К примеру, запрос на чтение двух регистров типа INT совпадает с запросом на чтение одного регистра типа FLOAT.

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. Для некоторых приборов количество возвращаемых регистров ограничивается величиной менее 120 слов. Подробная информация содержится в руководстве по эксплуатации на прибор.

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 устанавливает указанные регистры во всех ведомых устройствах. В запросе ведущего содержится адрес регистра и данные для записи. Адресация регистров начинается с нуля.

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 HB	Подфункция LB	Данные HB	Данные LB	CRC LB	CRC HB
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. Функция 10h – установка группы регистров.

Функция 10h обеспечивает запись в регистры ведомого устройства. Широковещательная передача устанавливает указанные регистры во всех ведомых устройствах. В запросе ведущего содержится адрес начального регистра, количество слов для записи и количество байт данных. Адресация регистров начинается с нуля. Различия между запросом записи регистров различных

типов не существует. К примеру, запрос на запись двух регистров типа INT совпадает с запросом на запись одного регистра типа FLOAT.

Ответ ведомого содержит адрес начального регистра и количество установленных регистров. Наибольшее количество регистров для записи не должно превышать 120 (240 байт). Для некоторых приборов количество записываемых регистров ограничивается величиной менее 120 слов. Подробная информация содержится в руководстве по эксплуатации на прибор.

Пример запроса и ответа приведён на рис. 5.6.

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	Кол. слов HB	Кол. слов LB	Кол. байт	Данные HB	Данные LB	CRC LB	CRC HB
01h	10h	00h	A0h	00h	01h	02h	03h	E8h	BEh	4Eh

Ответ

Адрес	Функция	Нач. адрес HB	Нач. адрес LB	Кол. слов HB	Кол. слов LB	CRC LB	CRC HB
01h	10h	00h	A0h	00h	01h	01h	EBh

Рис. 5.6. - Пример запроса и ответа функции 10h – установка группы регистров.

5.5. 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 должно принять кадр запроса и сформировать 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 HB
01h	30h		XXh	XXh

Ответ

Адрес	Функция	Код ошибки	CRC LB	CRC HB
01h	B0h	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 на подчиненном
02h	ILLEGAL DATA ADDRESS	Адрес данных указанный в запросе не доступен данному подчиненному
03h	ILLEGAL DATA VALUE	Величина содержащаяся в поле данных запроса является не допустимой величиной для ведомого
04h	SLAVE DEVICE FAILURE	Пока ведомый пытался выполнить затребованное действие произошла не восстанавливаемая ошибка
05h	ACKNOWLEDGE	Ведомый принял запрос и обрабатывает его, но это требует много времени. Этот ответ предохраняет ведущего от генерации ошибки таймаута
06h	SLAVE DEVICE BUSY	Ведомый занят обработкой команды. Ведущий должен повторить сообщение позже, когда ведомый освободится
07h	NEGATIVE ACKNOWLEDGE	Ведомый не может выполнить программную функцию, принятую в запросе
08h	MEMORY PARITY ERROR	Ведомый пытается читать расширенную память, но обнаружил ошибку паритета. Ведущий может повторить запрос, но обычно в таких случаях требуется ремонт

6. Адресное пространство.

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

Адресное пространство можно разделить на две части:

1. регистры общие для всех устройств;
2. регистры, специфичные для конкретного устройства.

6.1. Регистры, общие для всех устройств.

Общие регистры приведены в таблице 6.1.

Таблица 6.1. - Общие регистры адресного пространства.

№	Функция	Группа	Адрес регистра	Формат	Наименование параметра	Диапазон значений
1	03	“нач. настр.”	0000h	WORD	Идентификация изделия: Ст. байт – версия программы Мл. байт – код и модель изделия	
2	03, 06, 16	“нач. настр.”	0001h	WORD	Мл. байт - Сетевой адрес	1 - 31
3	03, 06, 16	“нач. настр.”	0002h	WORD	Ст. байт – Скорость обмена	03-1200, 04-2400 05-4800, 06-9600 07-19200, 08-38400 09-57600, 0A-115200
					Мл. байт – Режим работы последовательного порта	0 – NO PARITY ONE STOP BIT, 1 – NO PARITY TWO STOP BIT, 2 – ODD, 3 – EVEN,
4	03, 06, 16	“нач. настр.”	0003h	WORD	Мл. байт – Интервал молчания, мс	2 - 255

Идентификаторы формируются по следующей схеме (рис. 6.1). Перечень идентификаторов изделий приведен в таблице 6.2.



Рис. 6.1. - Система идентификаторов изделий.

Таблица 6.2. - Перечень идентификаторов изделий.

№	Прибор	Версия ПО	Идентификатор	Пояснения
1	МТМ103		0118h	
2	МТМ120		0114h	
3	МТМ201		011Dh	
4	МТМ310		0115h	
5	МТМ620		011Ch	
6	МТМ160		0090h	
7		01	0190h	
8		02	0290h	
9		02-C	0390h	
10	ПЦ12Р		0194h	
11	МТМ292		019Ch	
12	МТМ900		015Ch	
13	БПС-2		13CCh	
14	МТМ4000AI		01B4h	
15	МТМ4000AIR		01B5h	
16	МТМ4000АО		01B8h	
17	МТМ4000DI		01A4h	
18	МТМ4000DO		01A8h	
19	МТМ4000Р		017Ch	

6.2. Регистры прибора МТМ 120.

Регистры прибора МТМ 120 приведены в таблице 6.7.

Внимание! Прибор не имеет общих регистров.

Таблица 6.7. - Регистры прибора МТМ 120.

№	Функция	Группа	Адрес регистра	Формат	Наименование параметра	Диапазон значений
1	03		00A8h	FLOAT	Мгновенный расход	
2	03		00AAh	FLOAT	Интегральный расход	
3	03		00ACh	FLOAT	Часовой расход	
4	03		00AEh	WORD	Время вычисления часового расхода	0 - 24
5	03		00A0h	FLOAT	Диапазон min	
6	03		00A2h	FLOAT	Диапазон max	

6.3. Регистры прибора МТМ 292.

Регистры прибора МТМ 292 приведены в таблице 6.8.

Таблица 6.8. - Регистры прибора МТМ 292.

№	Функция	Группа	Адрес регистра	Формат	Наименование параметра	Диапазон значений
1	03		0100h	WORD	16 Уст.1 – 1 Уст.1	
2	03		0101h	WORD	16 Уст.2 – 1 Уст.2	
3	03		0102h	WORD	Мл. байт – Групповые уставки	
4	03		0103h	WORD	Зарезервировано	
5	03		0104h	FLOAT	Входной сигнал, канал 1	
6	03		0106h	FLOAT	Входной сигнал, канал 2	
7	03		0108h	FLOAT	Входной сигнал, канал 3	
8	03		010Ah	FLOAT	Входной сигнал, канал 4	
9	03		010Ch	FLOAT	Входной сигнал, канал 5	
10	03		010Eh	FLOAT	Входной сигнал, канал 6	
11	03		0120h	FLOAT	Входной сигнал, канал 7	
12	03		0122h	FLOAT	Входной сигнал, канал 8	
13	03		0124h	FLOAT	Входной сигнал, канал 9	
14	03		0126h	FLOAT	Входной сигнал, канал 10	
15	03		0128h	FLOAT	Входной сигнал, канал 11	
16	03		012Ah	FLOAT	Входной сигнал, канал 12	
17	03		012Ch	FLOAT	Входной сигнал, канал 13	
18	03		012Eh	FLOAT	Входной сигнал, канал 14	
19	03		0130h	FLOAT	Входной сигнал, канал 15	
20	03		0132h	FLOAT	Входной сигнал, канал 16	
21	03,06		0200h	FLOAT	Диапазон min, канал 1	
22	03,06		0202h	FLOAT	Диапазон max, канал 1	
23	03,06		0204h	FLOAT	Уставка 1, канал 1	
24	03,06		0206h	FLOAT	Уставка 2, канал 1	
25	03,06		0208h	FLOAT	Гистерезис, канал 1	
26	03,06		020Ah	WORD	Ст. байт - Признак группы уставка 1 Мл. байт - Признак группы уставка 2	
27	03,06		020Bh	WORD	Ст. байт - Тип датчика Мл. байт 6,7 биты – тип уставки 1 4,5 биты – тип уставки 2 0,1,2 биты – тип токового выхода	
28	03,06		020Ch	WORD	Зарезервировано	
29	03,06		020Bh	WORD	Зарезервировано	
30	03,06		020Eh	WORD	Зарезервировано	
31	03,06		020Fh	WORD	Зарезервировано	

6.4. Регистры прибора МТМ 900.

Регистры прибора МТМ 900 приведены в таблице 6.9.

Таблица 6.9. - Регистры прибора МТМ 900.

№	Функция	Группа	Адрес регистра	Формат	Наименование параметра	Диапазон значений
1	03		0100h	WORD	Расстояние до поверхности (Lx), мм	600-4600
2	03		0101h	WORD	Уровень жидкости в емкости (Hx), мм	0-4000
3	03		0102h	FLOAT	Объем жидкости в емкости (Vx), м ³	0,0-999,9
4	03		0104h	INT	Температура в емкости (T), 0,1°C	-40,0...+80,0
5	03,06		0200h	WORD	Уставка 1, мм	0-9999
6	03,06		0201h	WORD	Уставка 2, мм	0-9999
7	03,06		0202h	WORD	Тип Уставок (мл. байт)	0 - Up/Dn, 1 - Up/Up, 2 - Dn/Dn
8	03,06		0203h	WORD	Звонки Уставок: Мл. байт – Звонок Уставки 1; Ст. байт – Звонок Уставки 2.	0-Выкл., 1-Вкл.
9	03,06		0204h	WORD	Токовый выходной сигнал: Мл. байт – привязка токового выхода;	0-по уровню, 1-по объему.
					Ст. байт – тип токового выхода.	0-(0-5)мА, 1-(0-20)мА, 2-(4-20)мА.
10	03,06		0205h	WORD	Максимальная высота емкости, мм	600-9999
11	03,06		0206h	WORD	Высота емкости, мм	600-9999
12	03,06		0207h	WORD	Параметры емкости: Мл. байт – счет объема жидкости;	0 - по формулам, 1 - по таблице
					Ст. байт – Тип емкости.	0 – вертикальная, 1 – горизонтальная
13	03,06		0208h	FLOAT	Площадь поверхности для вертикальной емкости, м ² или длина для горизонтальной, м	0-999,9
14	03,06		0300h 03EFh	FLOAT	Градуировочная таблица емкости: 120 значений (240 слов)	0-999,9
15	03,06		03F0h	WORD	Параметры градуировочной таблицы емкости: Мл. байт – количество Float ячеек;	0 - 120
					Ст. байт – шаг приращения, см.	1 - 255

7. Полезные ссылки.

Дополнительную информацию можно найти по следующим адресам:

1. <http://www.mikroterm.lg.ua> – официальный сайт ООО НПП «Микротерм»;
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;