

Протокол обмена информацией для декодеров T32, T35, T36, T37

1.1. Общий формат запросов и ответов

Все запросы и ответы имеют формат:

Заголовок	Данные	Контрольная сумма
-----------	--------	-------------------

- Заголовок;
- Данные;
- Контрольная сумма.

1.2. Формат заголовка и контрольной суммы для декодеров T32 и T36

Для декодеров T32 и T36 заголовки запроса и ответа имеют одинаковый формат:

Адрес устройства	Код команды	длина данных
------------------	-------------	--------------

- адрес устройства (1 байт) – адрес устройства, к которому адресован запрос (имеет смысл только для декодера T36). Адрес устройства может изменяться от 1 до 247. Адреса в диапазоне 248...255 — зарезервированы. Для декодера T32 - ноль;
- код команды (1 байт) – говорит ведомому устройству, какие данные или выполнение какого действия требует от него ведущее устройство. Старшая единица (0x80) в коде команды используется в ответе подчиненного, чтобы указать, что операция закончилась с ошибкой;
- длина данных (1 байт).

Контрольная сумма (2 байта)

Контрольная сумма

Контрольная сумма служит для проверки отсутствия ошибок в запросе и ответе. Это циклическая контрольная сумма CRC-16 Modbus.

1.3. Формат заголовка и контрольной суммы для декодеров T35 и T37

Для декодеров T35 и T37 заголовки запроса и ответа имеют разный формат.

Заголовок запроса:

Код команды

- код команды (1 байт);

Заголовок ответа:

Длина данных

- длина данных (Целое число - 2 байта);

Контрольная сумма – отсутствует.

1.4. Коды команд декодеров

Все декодеры имеют одинаковый набор команд:

- 101 – START_MEASURING – Старт измерений
- 102 – STOP_MEASURING – Стоп измерений
- 103 – GET_ID – Читать служебную информацию датчика
- 104 – READ_BASE – Читать основную изм. величину
- 105 – READ_SPEED – Читать скорость и мощность
- 106 – READ_TEMPER – Читать температуру
- 107 – READ_COMPLEX – Читать все величины (крутящий момент, скор , мощность, температура)
- 108 – READ_BASE2 – Читать основную изм. величину потоком

- 109 – SET_DECODER_PARAM – Установить параметры (поправка, диапазон измерения скорости, коэффициент усреднения)
- 67 – GET_CURRENT_TIME – Получить текущее время
- 68 – SET_CURRENT_TIME – Установить текущее время (необходима для измерения скорости и мощности);
- 69 – GET_MESSAGE – Получить сообщение от декодера

1.5. Формат поля данных запросов

- **Формат данных запроса START_MEASURING**
 - Режим передачи (1 байт);
 - Коэффициент усреднения (целое число 2 байта);
 - Поправка (Число с плавающей точкой - 4 байта).
 - Период измерения скорости (Целое число - 4 байта).
 - Признак наличия внешнего датчика скорости (1 байт).
- **Формат данных запроса SET_DECODER_PARAM**
 - Коэффициент усреднения (целое число 2 байта);
 - Период измерения скорости (Целое число - 2 байта).
 - Поправка (Число с плавающей точкой - 4 байта).
- **Формат данных запроса SET_CURRENT_TIME**
 - Время старта в тиках часов декодера (Целое число 64 разрядное число - 8 байт). Тик часов декодера составляет 0.0000000125 сек. Можно указать 0 - время старта будет равно времени получения декодером команды SET_CURRENT_TIME.
- **Формат данных остальных запросов**
 - Формат данных остальных команд одинаковый - это код команды (1 байт).

1.6. Формат поля данных ответов

Если во время выполнения команды произошла ошибка, то все декодеры возвращают в поле данных код завершения (1 байт). Код завершения может принимать значения:

- 101 – RET_CODE_KOMMAND_ERR – Неверная команда
- 102 – RET_CODE_CHECKSUM_ERR – Неверная контрольная сумма
- 103 – RET_CODE_NODATA – Нет данных

Признаком того, что во время выполнения команды произошла ошибка, может служить длина данных в заголовке равная 1 и ненулевое значение кода завершения. Для декодеров T32 и T36 дополнительно устанавливается 1 в старшем бите кода команды в заголовке ответа.

Ниже приведены форматы данных ответов на различные команды в случае успешного завершения операции.

- **Формат данных ответа на команду GET_ID**
 - Идентификатор датчика (3 байта);
 - Температура (целое число 1 байт);
 - Коррекция чувствительности (целое число 1 байт);
 - Количество зубьев (Целое число - 2 байта);
 - Максимальная скорость вращения (целое число 1 байт);
 - Дата поверки (3 байта);
 - Дополнительная текстовая информация (49 байт);

Примечание. Более подробно формат служебной информации датчика приведен в разделе 1.8.

- **Формат данных ответа на команду READ_BASE**

- Время измерения (целое число 8 байт);
- Основная измеряемая величина (число с плавающей точкой 4 байта);
- **Формат данных ответа на команду *READ_SPEED***
 - Время измерения (целое число 8 байт);
 - Частота вращения (число с плавающей точкой 4 байта);
 - Мощность (число с плавающей точкой 4 байта);
- **Формат данных ответа на команду *READ_TEMPER***
 - Время измерения (целое число 8 байт);
 - Температура (число с плавающей точкой 4 байта);
- **Формат данных ответа на команду *READ_COMPLEX***
 - Время измерения (целое число 8 байт);
 - Основная измеряемая величина (число с плавающей точкой 4 байта);
 - Температура (число с плавающей точкой 4 байта);
 - Частота вращения (число с плавающей точкой 4 байта);
 - Мощность (число с плавающей точкой 4 байта);
- **Формат данных ответа на команду *READ_BASE2***
 - Тип данных (резерв - 1 байт);
 - Время измерения (целое число 8 байт);
 - Массив значений основной измеряемой величины (N чисел с плавающей точкой, длиной 4 байта);

Примечание. N принимает следующие значения:

- 60 - для декодеров T32 и T36;
- 48 - для декодеров T37;
- 12 - для декодеров T35;
- **Формат данных ответа на команду *GET_CURRENT_TIME***
 - Время измерения (целое число 8 байт);
- **Формат данных ответа на команду *GET_MESSAGE***
 - Тип данных (резерв 1 байт);
 - Время измерения (целое число 8 байт);
 - Массив кодов сообщений (от 1 до 50 байтов). Коды сообщений приведены в разделе 1.9.
- **Формат данных ответов на другие команды**
 - Формат данных ответов на другие команды - код завершения (1 байт).

1.7. Последовательность команд для работы с декодером

- Первой командой всегда должна быть команда **Начать измерения (*START_MEASURING*)**;
- Второй командой должна быть – **Установить текущее время (*SET_CURRENT_TIME*)**;
- Далее могут следовать любые команды чтения данных или изменения параметров. Команду ***READ_BASE2*** следует применять только в том случае, если вы хотите прочитать весь поток данных от датчика (5000 измерений в секунду), потому что декодер буферизирует эти данные и время доступа увеличивается. Особенно при больших коэффициентах усреднения;
- Завершается работа командой **Остановить измерения (*STOP_MEASURING*)**.

1.8. Примеры запросов и ответов декодера T36

1) Начать измерения (START_MEASURING).

Параметры запроса:

- Режим передачи - 0;
- Коэффициент усреднения – 1;
- Поправка 0.0.
- Период измерения скорости – 1000.
- Признак наличия внешнего датчика скорости – 0.

Запрос (в 16 системе счисления):

Заголовок			Данные											Контрольная сумма		
Сетевой номер	Код команды	Длина	Режим работы	Коэффициент усреднения		Поправка				Период измерения скорости				Признак	Контрольная сумма	
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
01	65	0C	00	01	00	00	00	00	00	E8	03	00	00	00	91	B9

Ответ при успешном завершении:

Заголовок			Данные	Контрольная сумма	
Сетевой номер	Код команды	Длина	Код завершения	Контрольная сумма	
1	2	3	4	5	6
01	65	01	00	10	57

2) Установить текущее время (SET_CURRENT_TIME).

Запрос (в 16 системе счисления):

Заголовок			Данные										Контрольная сумма
Сетевой номер	Код команды	Длина	Точное время измерения										Контрольная сумма
1	2	3	4	5	6	7	8	9	10	11	12	13	
01	44	08	00	00	00	00	00	00	00	00	50	A0	

Ответ при успешном завершении:

Заголовок			Данные	Контрольная сумма	
Сетевой номер	Код команды	Длина	Код завершения	Контрольная сумма	
1	2	3	4	5	6
01	44	01	00	40	5D

3) Прочитать служебные данные (GET_ID)

Параметры запроса: нет.

Запрос (в 16 системе счисления):

Заголовок			Контрольная сумма	
Сетевой номер	Код команды	Длина	Контрольная сумма	
1	2	3	4	5
01	67	00	0A	30

Ответ при успешном завершении:

Заголовок			Данные											Контрольная сумма
-----------	--	--	--------	--	--	--	--	--	--	--	--	--	--	-------------------

01	6A	0C	35	32	34	AB	04	00	00	00	00	00	B8	41	13	33
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

7) Прочитать все величины (READ_COMPLEX).

Параметры запроса: нет.

Запрос (в 16 системе счисления):

Заголовок			Контрольная сумма	
Сетевой номер	Код команды	Длина	Контрольная сумма	
1	2	3	4	5
01	6B	00	0F	30

Ответ при успешном завершении:

Заголовок			Данные																								Контрольная сумма		
Сет. номер	Код ком.	Длина	Точное время измерения									Основная измеряемая величина				Температура				Частота вращения				Мощность				Контрольная сумма	
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	
01	6B	18	41	34	8C	4A	05	00	00	00	08	28	C8	3E	00	00	D	C	41	00	00	00	00	00	00	00	00	F7	C3

8) Читать основную изм. величину потоком (READ_BASE2).

Параметры запроса: нет.

Запрос (в 16 системе счисления):

Заголовок			Контрольная сумма	
Сетевой номер	Код команды	Длина	Контрольная сумма	
1	2	3	4	5
01	6C	00	0D	00

Ответ при успешном завершении:

Заголовок			Данные													Контрольная сумма									
Сетевой номер	Код команды	Длина	тип данных	Точное время измерения									Основная измеряемая величина (60 шт.)				Контрольная сумма								
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	253	254
01	6C	FB	64	4A	1F	C9	9C	04	00	00	00	07	20	A0	3E	07	20	A0	3E	07	20	A0	3E	50	A0
												-----	-----	-----	-----										
												249	250	251	252	0B	30	70	3E						

Ответ при отсутствии данных (код 103):

Заголовок			Данные	Контрольная сумма	
Сетевой номер	Код команды	Длина	Код заверш.	Контрольная сумма	
1	2	3	4	5	6
01	EC	01	67	81	9B

9) Остановить измерения (STOP_MEASURING).

Параметры запроса: нет.

Запрос (в 16 системе счисления):

Заголовок			Контрольная сумма	
Сетевой номер	Код команды	Длина	Контрольная сумма	
1	2	3	4	5
01	66	00	0B	0A

Ответ при успешном завершении:

Заголовок	Данные	Контрольная сумма

Сетевой номер	Код команды	Длина	Код завершения	Контрольная сумма	
1	2	3	4	5	6
01	66	01	00	E0	57

1.9. Служебная информация

Служебный канал имеет следующую структуру:

```
struct _SK {
    unsigned char ID_Datchik[3]; // Идентификатор датчика
    char Temperature; // Температура
    char Korrekt; // Коррекция чувствительности
    short int K_Zub; // Количество зубьев
    char MaxSkorVr; // Макс. Скорость вращения
    char Date_Poverki[3]; // Дата поверки
    char Slug_Inf[49]; // Служебная информация
};
```

Рассмотрим составляющие элементы этой структуры:

1) **Идентификатор датчика (заводской номер). Состоит из шести шестнадцатеричных цифр:**

- **Старшая цифра идентификатора определяет назначение датчика:**
 - 0 – датчики момента. Показания в мкНм; мНм; Нм; кНм; МНм;
 - 1 – датчики силы. Показания в мкН; мН; Н; кН; МН;
 - 2 – датчики массы. Показания в мкг; мг; г; кг; Т;
 - 3 – датчики давления. Показания в мкПа; мПа; Па; кПа; МПа;
 - 4 – датчики перемещения. Показания в мкм; мм; м; км; Мм;
 - 5 – датчики угла. Показания в градусах;
 - 6 – скорости. Показания в мкм/с; мм/с; м/с; км/с; Мм/с
 - 7 – иные датчики, требующие специальной программы.
- **Вторая цифра идентификатора определяет тип датчика:**
 - Для датчиков момента (первая цифра идентификатора 0):**
 - 0 – не определено
 - 1 – датчики типа MA20
 - 2 – датчики типа M20C
 - 3 – датчики типа M40E
 - 4 – датчики типа M40
 - Для датчиков силы (первая цифра идентификатора 1):**
 - 0 – не определено
 - 1 – датчики СТ1
 - 2 – датчики СТ2
 - 3 – датчики СТ3
 - 4 – датчики СТ4
- **Третья цифра идентификатора определяет степень (размерность) и формат отображения основной измеряемой величины:**
 - 0 – 10^{-6} Например, для крутящего момента – XXXX мкНм;
 - 1 – 10^{-5} – XX,XX мНм;
 - 2 – 10^{-4} – XXX,X мНм;
 - 3 – 10^{-3} – XXXX мНм;
 - 4 – 10^{-2} – XX,XX Нм;
 - 5 – 10^{-1} – XXX,X Нм;
 - 6 – 10^0 – XXXX Нм;
 - 7 – 10^1 – XX,XX кНм;
 - 8 – 10^2 – XXX,X кНм;
 - 9 – 10^3 – XXXX кНм;
 - A – 10^4 – XX,XX МНм;
 - B – 10^5 – XXX,X МНм;
 - C – 10^6 – XXXX МНм;
 - D – 10^{-7} – XXX,X мкНм;
 - E – 10^{-8} – XX,XX мкНм;
 - F – 10^{-9} – X,XXX мкНм.

- **Четвертая цифра идентификатора определяет множитель измеряемой величины, К:**
 - 0 – 1 – датчики, имеющие множитель измеряемого диапазона 1
 - 1 – 1,5 – датчики, имеющие множитель измеряемого диапазона 1,5
 - 2 – 2 – датчики, имеющие множитель измеряемого диапазона 2
 - 3 – 2,5 – датчики, имеющие множитель измеряемого диапазона 2,5
 - 4 – 3 – датчики, имеющие множитель измеряемого диапазона 3
 - 5 – 4 – датчики, имеющие множитель измеряемого диапазона 4
 - 6 – 5 – датчики, имеющие множитель измеряемого диапазона 5
 - 7 – 6 – датчики, имеющие множитель измеряемого диапазона 6
 - 8 – 8 – датчики, имеющие множитель измеряемого диапазона 8
 - **Две младшие цифры** (третий байт) идентификатора задает порядковый номер датчика.
- 2) **Температура датчика.** Температура датчика один байт. 0x00 соответствует -50 °С, а 0xFF – +77,5 °С.
 - 3) **Коррекция чувствительности основного датчика.** Байт коррекции чувствительности основного датчика может принимать значения от 00 до FF, причем FF соответствует минимальная чувствительность, а 00 максимальная.
 - 4) **Количество зубьев** датчика вращения от 0 до 65535. 0 – признак невращающегося датчика
 - 5) **Максимальная скорость** вращения × 100 об/мин
 - 6) **Дата поверки.** Дата поверки три байта в десятичном представлении дата, месяц, год (две последние цифры года).
 - 7) **Служебная информация** 49 байт, часть этих байт может использоваться для дополнительных параметров.

1.10. Коды сообщений декодера

- | | |
|---------------------------------|---|
| • 001 – MESSAGE_SBOI_SYNCHRO | Сбой синхронизации |
| • 002 – MESSAGE_POLUCHEN_ID | Получена служебная информация |
| • 003 – MESSAGE_OSHIBKA_ID | Служебная информация не совпадает со старой |
| • 004 – MESSAGE_DATCHIK_OFF | Датчик отключился |
| • 005 – MESSAGE_DATCHIK_ON | Датчик подключился |
| • 006 – MESSAGE_DECODER_OFF | Декодер отключился |
| • 007 – MESSAGE_DECODER_ON | Декодер подключился |
| • 008 – MESSAGE_ISKAJENIA | Искажения приема-передачи |
| • 009 – MESSAGE_OSHIBKA_IO | Ошибка ввода-вывода |
| • 010 – MESSAGE_NET_PAUSY | Нет паузы в приеме |
| • 011 – MESSAGE_NETWORK_OFF | Нет доступа к сети |
| • 013 – MESSAGE_VOSST_SYNCHRO | Восстановление синхронизации |
| • 015 – MESSAGE_BUFFER_LOSS | Потеря буфера данных |
| • 019 – MESSAGE_MESSAGEQ_OVERFL | Была переполнена очередь сообщений |
| • 020 – MESSAGE_INPUT_TIMEOUT | Нет ответа на запрос |