

ПЮТИК



ПРОТОКОЛ ОБМЕНА
MODBUS RTU

МИКРОПРОЦЕССОРНЫЙ КОМПЛЕКТ ЗАЩИТЫ
ПРИСОЕДИНЕНИЙ 6-20 кВ
ЛЮТИК

ПРОТОКОЛ ОБМЕНА MODBUS RTU

ОПИСАНИЕ ПРОТОКОЛА

МТ.Лютик.01.01.ОП от 11.06.2020

Оглавление

1. Общие сведения	4
2. Физический уровень.....	4
3. Канальный уровень	4
4. Прикладной уровень	4
4.1. Функция 65	5
4.1.1. Чтение текущей даты и времени (подфункция 4).....	7
4.1.2. Установка текущей даты и времени (подфункция 5)	7
4.1.3. Чтение параметров местного времени (подфункция 6)	8
4.1.4. Запись параметров местного времени (подфункция 7).....	9
4.1.5. Чтение текущего состояния дискретных входов (подфункция 16).....	9
4.1.6. Чтение текущего состояния дискретных выходов (подфункция 17)	10
4.1.7. Чтение текущего состояния двоичных входных сигналов (подфункция 18)	10
4.1.8. Чтение текущего состояния двоичных выходных сигналов (подфункция 19)	11
4.1.9. Чтение списка двоичных сигналов (подфункция 20).....	11
4.1.10. Чтение информации о флеш-памяти (подфункция 23)	12
4.1.11. Чтение описателя записи журнала событий (подфункция 24)	13
4.1.12. Чтение записей журнала (подфункция 25).....	15
4.1.13. Чтение краткой информации о записях журнала (подфункция 26)	19
4.1.14. Чтение описателя записи журнала съёма сигнализации (подфункция 41).....	21
4.1.15. Передача длинной команды (подфункция 239)	21
4.1.16. Получение информации о буфере сервера (подфункция 240).....	22
4.1.17. Чтение данных из буфера сервера (подфункция 241).....	23
4.1.18. Уведомление сервера о завершении чтения из буфера (подфункция 242).....	23
4.1.19. Чтение конфигурации интерфейса RS-485 (подфункция 250)	23
4.1.20. Запись конфигурации интерфейса RS-485 (подфункция 251).....	25
4.1.21. Чтение сведений об устройстве (подфункция 254)	25
5. Приложение 1. Пример использования запроса передачи длинной команды (подфункция 239) 30	
6. Литература.....	34

1. Общие сведения

При обмене данными Лютик всегда является сервером, KIWI – клиентом. KIWI может обмениваться данными с Лютиком только напрямую (рис. 1).

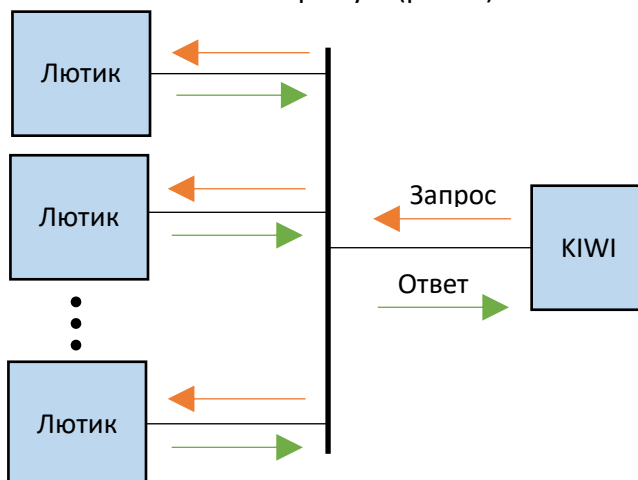


Рисунок 1. Обмен данными между KIWI и Лютиком

2. Физический уровень

На физическом уровне применяется протокол ISO/IEC 8482:1993 (RS-485) [3].

3. Канальный уровень

На канальном уровне используется протокол MODBUS RTU [1], передающий данные в кадрах следующей структуры.

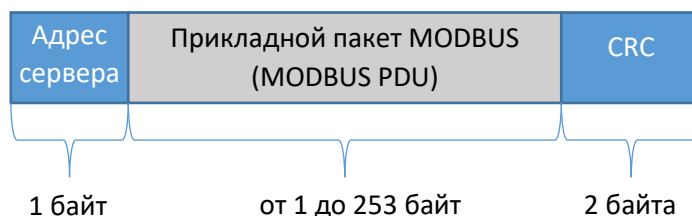


Рисунок 2. Структура кадра MODBUS RTU

4. Прикладной уровень

На прикладном уровне протокола обмена данными используется прикладной уровень протокола MODBUS [2]. Пакеты прикладного уровня подразделяются на запросы, которые клиент посылает серверу и ответы, которые сервер отправляет клиенту.

В дополнение к стандартным функциям MODBUS применяются также прикладные пакеты MODBUS с кодом функции 65 (рис. 3). Кроме кода функции эти пакеты содержат код подфункции, который определяет тип запроса, порядковый номер запроса, длину данных и сами данные. Порядковый номер запроса служит для сопоставления запросов и ответов. Этот номер назначается клиентом при создании запроса и возвращается сервером в ответе на запрос. Поле длины данных указывает количество байтов в следующем за ним поле данных. Если поле данных отсутствует, то длина данных равна 0. Во всех запросах и ответах с кодом функции 65 все многобайтные числа, содержащиеся в MODBUS PDU, передаются, начиная со старшего байта (big-endian).

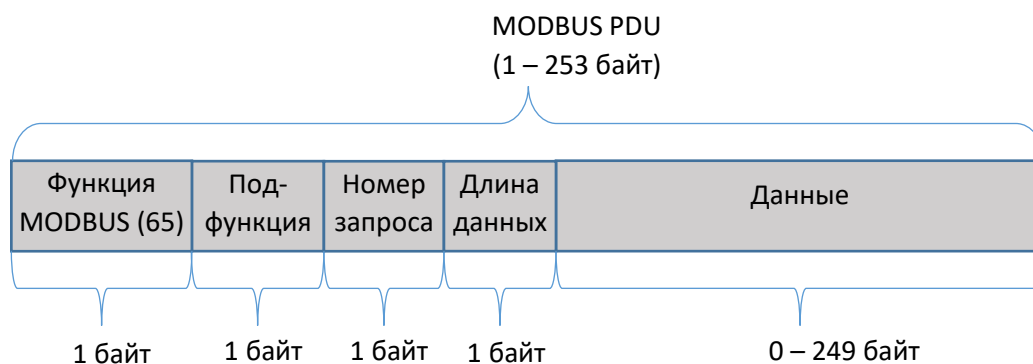


Рисунок 3. Структура прикладного пакета MODBUS (MODBUS PDU)

4.1. Функция 65

Запросы с кодом функции 65 и ответы на эти запросы имеют заголовки одинаковой структуры (рис. 3). Структура поля данных зависит от кода подфункции. Предусмотрена специальная подфункция (239), предназначенная для передачи команд, длина которых превышает объём стандартного прикладного пакета MODBUS (MODBUS PDU). Клиент, посылая серию запросов с этой подфункцией, может частями передать серверу команду такой же структуры, как MODBUS PDU обычного запроса (рис. 3). Единственное отличие команды от MODBUS PDU заключается в том, что её длина может быть больше максимального размера MODBUS PDU, и поэтому поле длины данных в команде указывает действительный объём поля данных команды только тогда, когда он не превышает 255 байт. Сервер, получая запросы с подфункцией 239, накапливает у себя фрагменты команды до тех пор, пока не получит её полностью, а затем исполняет эту команду и внутри ответа на запрос 239 отправляет клиенту результат выполнения команды. Этот результат оформляется как MODBUS PDU, содержащий ответ на запрос с тем кодом функции, который указан в первом байте команды.

Ответ на запрос с кодом функции 65 в поле данных MODBUS PDU содержит тип ответа (0 – короткий, 1 – длинный), код ответа (табл. 1) и, в некоторых случаях, поле данных ответа.

Ответы могут быть положительными и отрицательными. Положительный ответ означает, что запрос выполнен успешно. Отрицательный ответ означает, что сервер не смог выполнить запрос, при этом код ответа указывает причину, по которой запрос не удалось выполнить. Отрицательные ответы отправляются в тех случаях, когда ошибка, возникшая при выполнении запроса, не может быть описана кодами исключений MODBUS. Кроме того, как положительные, так и отрицательные ответы могут быть короткими и длинными. Короткий ответ (рис. 4, 5) целиком помещается в MODBUS PDU.



Рисунок 4. Структура поля данных PDU короткого ответа

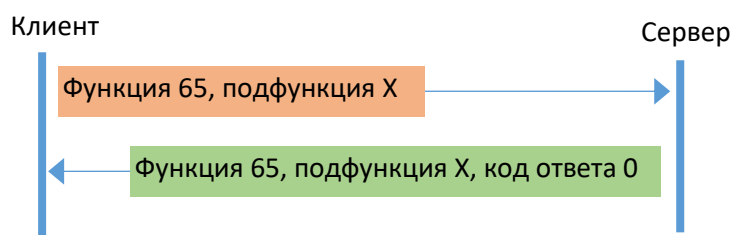


Рисунок 5. Обмен данными при получении короткого ответа

Длинный ответ (рис. 6, 7) указывает общее количество байтов данных ответа, которые передаются через специальный буфер сервера, и содержит первый фрагмент этих данных, начиная со смещения 0. Для чтения данных из буфера сервера используются подфункции [241](#) и [242](#).

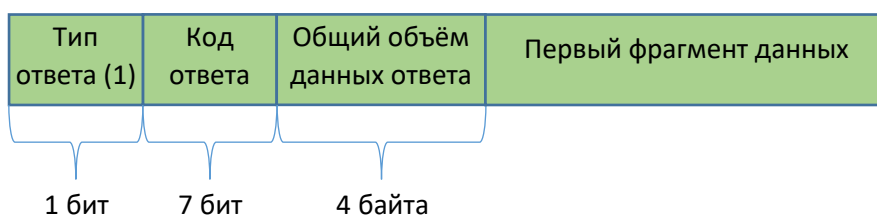


Рисунок 6. Структура поля данных PDU длинного ответа



Рисунок 7. Обмен данными при получении длинного ответа

При ответе на один и тот же запрос длинные и короткие ответы с одним и тем же кодом ответа имеют одинаковую структуру поля данных.

Таблица 1. Коды ответов

Код ответа	Описание
0	Положительный ответ (запрос выполнен успешно).
1	Длина запроса не соответствует типу запроса.
2	Некорректные параметры запроса.
3	Ошибка в сервере при выполнении запроса.
4	Нет прав доступа к ресурсу.
5	Неправильный пароль.
6	Сеанс связи закрыт.
7	Истекло время хранения данных ответа в буфере сервера.
8	Данные ответа не помещаются в буфер сервера.
9	Ошибка при открытии файла.
10	Ошибка при чтении файла.
11	Ошибка при записи файла.
12	Ошибка при записи уставок.
13	Данные запроса не помещаются в буфер сервера.
14	Недопустимый идентификатор файла.
15	Недопустимый режим открытия файла.
16	Операция с файлом не соответствует режиму открытия файла.
17	Неизвестный код подфункции.
18	Длина поля данных запроса не соответствует длине, указанной в заголовке запроса.

4.1.1. Чтение текущей даты и времени (подфункция 4)

В запросе поле данных отсутствует.

Поле данных положительного ответа на запрос текущей даты и времени содержит метку времени с точностью до миллисекунд (уточнённое время UNIX – количество полных миллисекунд, прошедших с 00:00:00 01.01.1970 UTC) и смещение местного времени от UTC в минутах с учётом часового пояса и летнего времени (т.е. разность текущего местного времени и UTC в минутах, представленная как 16-битовое целое число со знаком). Положительное смещение означает, что местное время опережает UTC.



Рисунок 8. Поле данных положительного ответа на запрос текущей даты и времени

4.1.2. Установка текущей даты и времени (подфункция 5)

Поле данных запроса содержит метку времени с точностью до миллисекунд (уточнённое время UNIX – количество полных миллисекунд, прошедших с 00:00:00 01.01.1970 UTC).

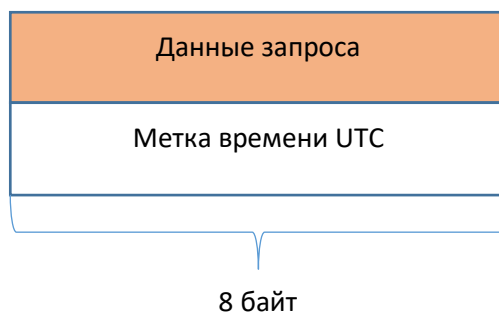


Рисунок 9. Структура запроса установки текущей даты и времени

В положительном ответе на этот запрос поле данных отсутствует.

4.1.3. Чтение параметров местного времени (подфункция 6)

В запросе поле данных отсутствует.

Поле данных положительного ответа на запрос чтения параметров местного времени (рис. 10) содержит местный часовой пояс (разность между стандартным местным временем и UTC в минутах), момент перехода на летнее время, момент возврата к стандартному времени и разность летнего и стандартного времени в минутах. Моменты перехода и возврата указываются по местному времени и имеют структуру, описанную в табл. 2. Разность указывается как 16-битовое целое число со знаком, при этом положительная величина означает, что в момент перехода на летнее время часы переводятся на указанное количество минут вперед, а отрицательная величина – назад. Если разность летнего и стандартного времени равна 0, то летнее время не применяется, и в течение всего года действует стандартное время, соответствующее часовому поясу; в этом случае моменты перехода на летнее время и возврата к стандартному времени не имеют смысла и могут содержать некорректные значения.

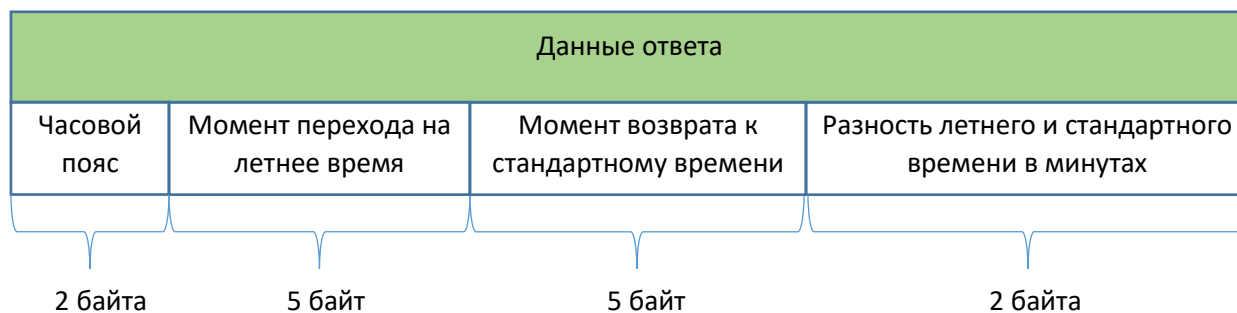


Рисунок 10. Поле данных положительного ответа на запрос чтения параметров летнего времени

Таблица 2. Структура момента времени

Смещение, байт	Размер, байт	Условное обозначение	Описание
0	1	Month	Номер месяца (от 1 до 12).
1	1	DayOfWeek	Код дня недели (табл. 3).
2	1	DayOfMonth	Порядковый номер дня в месяце. Если код дня недели равен 0, то параметр DayOfMonth содержит номер дня месяца, при этом 0 означает последний день месяца. При остальных значениях кода дня недели параметр DayOfMonth содержит порядковый номер указанного дня недели в месяце, при этом 0 означает последний день недели в месяце.
3	1	Hour	Час (от 0 до 23).
4	1	Minute	Минута (от 0 до 59).

Таблица 3. Коды дней недели

Код	Описание
0	Любой день недели
1	Понедельник
2	Вторник
3	Среда
4	Четверг
5	Пятница
6	Суббота
7	Воскресенье

Примеры моментов времени:

1. 2, 0, 21, 1, 15 – 21-е февраля, 1 час 15 минут.
2. 12, 7, 0, 20, 0 – последнее воскресенье декабря, 20 часов 0 минут.
3. 4, 1, 2, 2, 0 – второй понедельник апреля, 2 часа 0 минут.

4.1.4. Запись параметров местного времени (подфункция 7)

Поле данных запроса записи параметров местного времени имеет такую же структуру, как и поле данных ответа на запрос чтения этих параметров (см. п. 4.1.3).

В положительном ответе на этот запрос поле данных отсутствует.

4.1.5. Чтение текущего состояния дискретных входов (подфункция 16)

В запросе поле данных отсутствует.

Поле данных положительного ответа на этот запрос содержит количество дискретных входов и, собственно, состояния дискретных входов. В поле состояния каждый бит отображает состояние соответствующего дискретного входа.

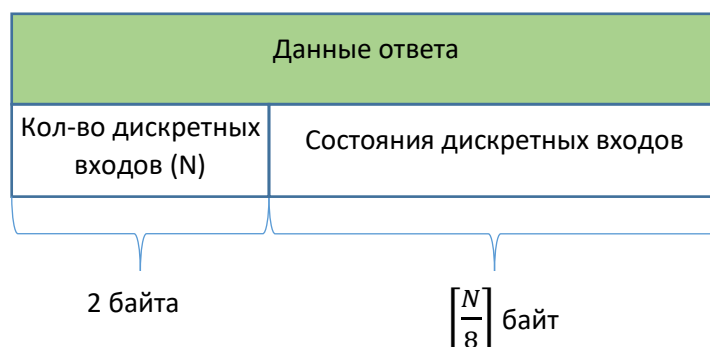


Рисунок 11. Поле данных положительного ответа на запрос чтения состояния дискретных входов

4.1.6. Чтение текущего состояния дискретных выходов (подфункция 17)

В запросе поле данных отсутствует.

Поле данных положительного ответа на этот запрос содержит количество дискретных выходов и, собственно, состояния дискретных выходов. В поле состояния каждый бит отображает состояние соответствующего дискретного выхода.

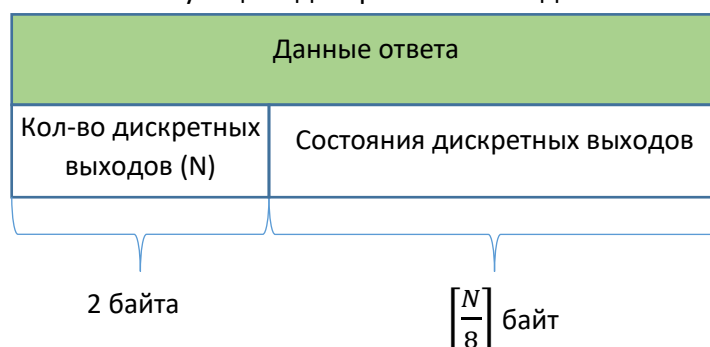


Рисунок 12. Поле данных положительного ответа на запрос чтения состояния дискретных выходов

4.1.7. Чтение текущего состояния двоичных входных сигналов (подфункция 18)

В запросе поле данных отсутствует.

Поле данных положительного ответа на этот запрос содержит количество двоичных входных сигналов и состояния этих сигналов. В поле состояния каждый бит отображает состояние соответствующего сигнала.

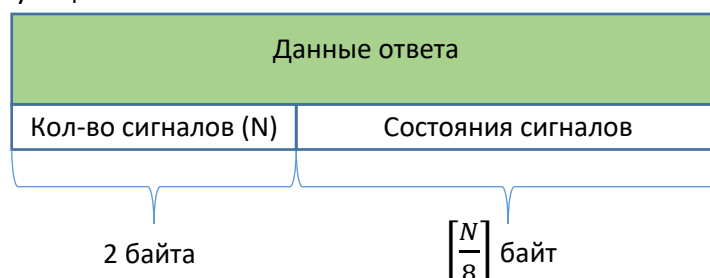


Рисунок 13. Поле данных положительного ответа на запрос чтения состояния двоичных входных сигналов

4.1.8. Чтение текущего состояния двоичных выходных сигналов (подфункция 19)

В запросе поле данных отсутствует.

Поле данных положительного ответа на этот запрос содержит количество двоичных выходных сигналов и состояния этих сигналов. В поле состояния каждый бит отображает состояние соответствующего сигнала.



Рисунок 14. Поле данных положительного ответа на запрос чтения состояния двоичных выходных сигналов

4.1.9. Чтение списка двоичных сигналов (подфункция 20)

Поле данных запроса (рис. 15) содержит код типа двоичного сигнала (табл. 4), состояние запрашиваемых сигналов (0 или 1), и код структуры элемента списка сигналов (табл. 5).



Рисунок 15. Структура запроса списка двоичных сигналов

Поле данных положительного ответа на этот запрос повторяет данные запроса (код типа сигнала, состояние сигналов и код структуры элемента списка), а также содержит количество элементов списка (может быть равно 0) и сам список сигналов (если количество элементов списка не равно 0).

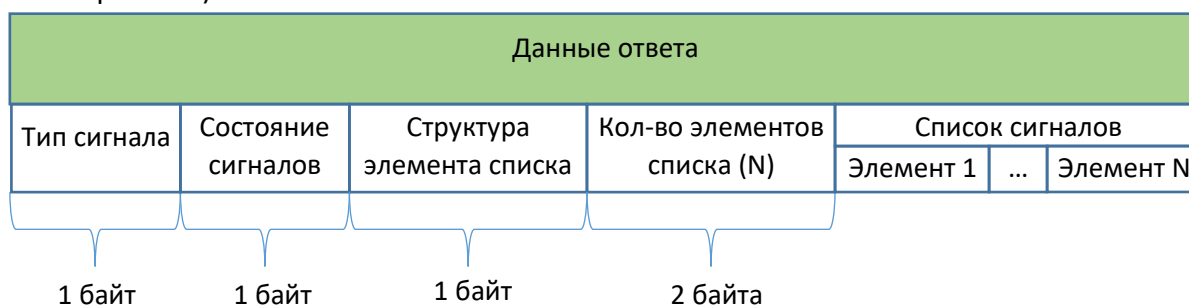


Рисунок 16. Поле данных положительного ответа на запрос чтения списка двоичных сигналов, находящихся в определённом состоянии.

Таблица 4. Типы двоичных сигналов

Код	Название
0	Двоичные входы
1	Двоичные выходы
2	Двоичные входные сигналы
3	Двоичные выходные сигналы
4	Сигналы пусковых органов
5	-

Таблица 5. Типы структур элемента списка сигналов

Код	Название
0	Краткое название сигнала (строка UTF-8, оканчивающаяся нулём)
1	Идентификатор сигнала (16-битовое число)
2	Идентификатор и краткое название сигнала (см. табл. 6)

Таблица 6. Структура элемента списка

Смещение, байт	Размер, байт	Условное обозначение	Описание
0	2	ID	Идентификатор сигнала.
2		ShortName	Краткое название сигнала (строка UTF-8, оканчивающаяся нулём)

4.1.10. Чтение информации о флеш-памяти (подфункция 23)

Поле данных запроса содержит код вида информации (см. табл. 7).

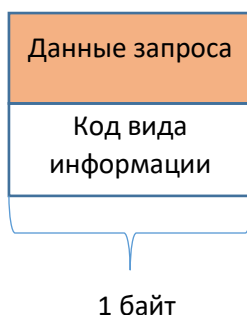


Рисунок 17. Структура запроса чтения информации о флеш-памяти

Таблица 7. Виды информации

Код	Вид информации
0	Вся информация
1	Информация о журнале событий
2	Информация о журнале сообщений
3	Информация о журнале изменений уставок
4	-
5	Информация о заполнении флеш-памяти
6	Информация о журнале съёма сигнализации

Для всех видов информации, относящихся к журналам, поле данных положительного ответа (см. рис. 18) содержит номер версии структуры журнала, номер версии ПО, создавшего журнал, размер записи журнала, максимально допустимое количество записей, текущее количество записей, уникальный идентификатор последней записи (сквозной

порядковый номер записи во всех журналах), метка времени последней записи (уточнённое время UNIX – количество полных миллисекунд, прошедших с 00:00:00 01.01.1970 UTC).

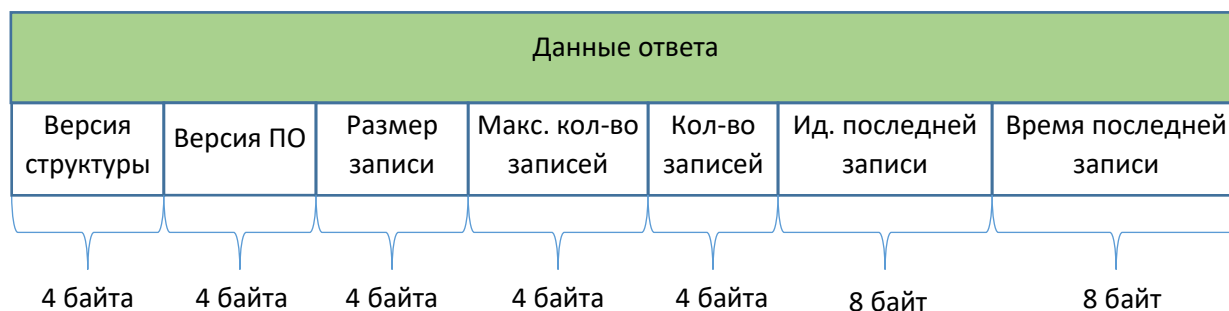


Рисунок 18. Поле данных положительного ответа на запрос чтения информации о конкретном журнале

Для информации о заполнении флеш-памяти поле данных положительного ответа (см. Рисунок 19) содержит общий объём активного раздела флеш-памяти устройства, и объём свободной памяти активного раздела в байтах.

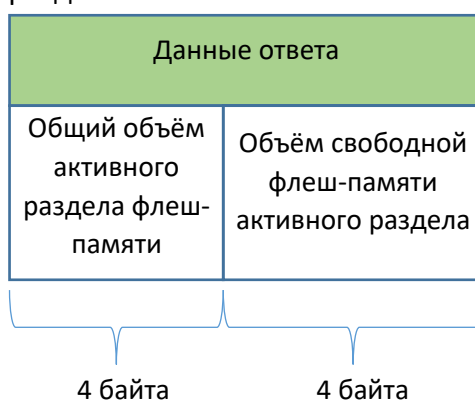


Рисунок 19. Поле данных положительного ответа на запрос информации о заполнении флеш-памяти

Для вида информации, обозначающего всю информацию, поле данных положительного ответа (см. рис. 20) содержит информацию о всех журналах, при этом структура информации о каждом журнале такая же, как в ответе на запрос конкретного вида информации (см. рис. 18 – 19).

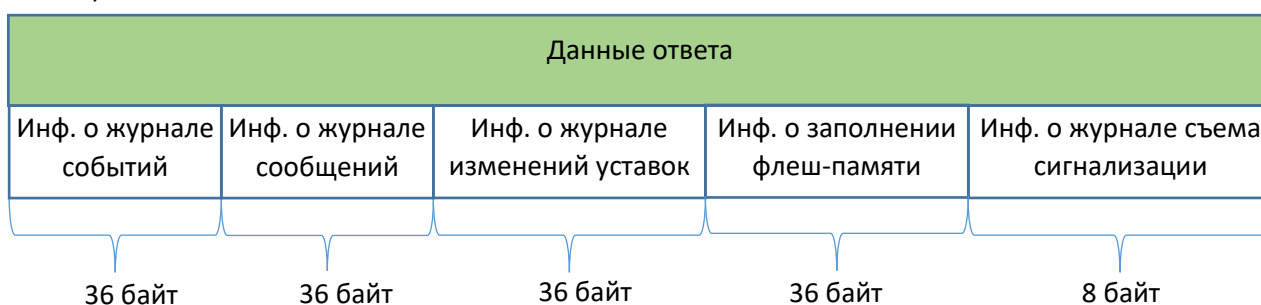


Рисунок 20. Поле данных положительного ответа на запрос чтения всех видов информации

4.1.11. Чтение описателя записи журнала событий (подфункция 24)

В запросе поле данных отсутствует.

Поле данных положительного ответа на этот запрос содержит описатель записи журнала событий (см. табл. 8).

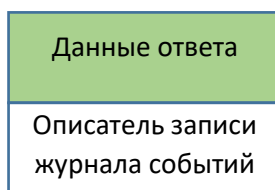


Рисунок 21. Поле данных положительного ответа на запрос чтения описателя записи журнала событий

Таблица 8. Структура описателя записи журнала событий

Смещение, байт	Размер, байт	Условное обозначение	Описание
0	2	NET	Количество типов событий.
2	$(2 + 100) * NET$	EventDescriptors	Массив описателей событий (см. табл. 9).
0	2	NET	Количество типов событий (Для Лютика $NET=200$ (BOS (106), BIS (36), BI (4), резерв события любого типа (54)))
2	$2 * NET$	EventDescriptors	Массив описателей событий (см. табл. 9).
$(2+2*NET)$	2	NBI	Количество двоичных входов (Для Лютика=4)
$(2+2*NET) + 2$	$2 * NBI$	BIIDs	Массив идентификаторов двоичных входов.
$(2+2*NET) + 2*NBI + 2$	2	NBO	Количество двоичных выходов (Для Лютика=4)
$(2+2*NET) + 2*NBI + 4$	$2 * NBO$	BOIDs	Массив идентификаторов двоичных выходов.
$(2+2*NET) + 2*NBI + 2*NBO + 4$	2	NBIS	Количество двоичных входных сигналов (Для Лютика=44)
$(2+2*NET) + 2*NBI + 2*NBO + 6$	$2 * NBIS$	BISIDs	Массив идентификаторов двоичных входных сигналов.
$(2+2*NET) + 2*NBI + 2*NBO + 2*NBIS + 6$	2	NBOS	Количество двоичных выходных сигналов (Для Лютика=106)
$(2+2*NET) + 2*NBI + 2*NBO + 2*NBIS + 8$	$2 * NBOS$	BOSIDs	Массив идентификаторов двоичных выходных сигналов.
$(2+2*NET) + 2*NBI + 2*NBO + 2*NBIS + 2*NBOS + 8$	2	NAOS	Количество аналоговых выходных сигналов (Для Лютика=14)
$(2+2*NET) + 2*NBI + 2*NBO + 2*NBIS + 2*NBOS + 10$	$3 * NAOS$	AOSDescriptors	Массив описателей аналоговых выходных сигналов (см. табл. 10).

Таблица 9. Структура элемента массива описателей событий

Смещение, байт	Размер, байт	Условное обозначение	Описание
0	2	EventID	Идентификатор события. Если идентификатор равен FFFF, то соответствующий бит маски событий в записи журнала не используется.
2	100	EventName	Название события (строка символов UTF-8, оканчивающаяся нулём).

Таблица 10. Структура элемента массива описателей аналоговых выходных сигналов

Смещение, байт	Размер, байт	Условное обозначение	Описание
0	2	AnalogQuantityID	Идентификатор аналогового выходного сигнала. Если идентификатор равен FFFF, то соответствующий элемент массива аналоговых входных сигналов в записи журнала не используется.
2	1	Units	Единицы измерения величины (см. табл. 11)

Таблица 11. Коды единиц измерения для журналов

Код	Описание
1	Безразмерная величина
2	Относительные единицы
3	%
4	мВ
5	В
6	кВ
7	МВ
8	мА
9	А
10	кА
11	МА
12	мГц
13	Гц
14	кГц
15	МГц
16	мГц/с
17	Гц/с
18	кГц/с
19	МГц/с
20	мкс
21	мс
22	с
23	мин

4.1.12. Чтение записей журнала (подфункция 25)

Поле данных запроса содержит код типа журнала (см. табл. 7), номер начальной записи и количество записей. Нумерация записей начинается с нуля, запись с номером 0 – самая новая запись в журнале. В качестве кода типа журнала можно использовать любое значение из табл. 7, кроме того, которое обозначает все журналы.

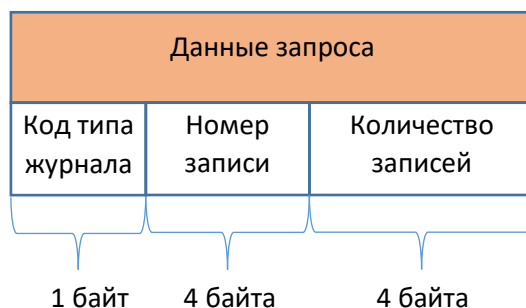


Рисунок 22. Структура запроса чтения записей журнала

Поле данных положительного ответа на этот запрос содержит номер начальной записи, количество записей и последовательность записей. Каждая запись в этой последовательности имеет структуру, определяется типом журнала и версией структуры (см. табл. 12-15).

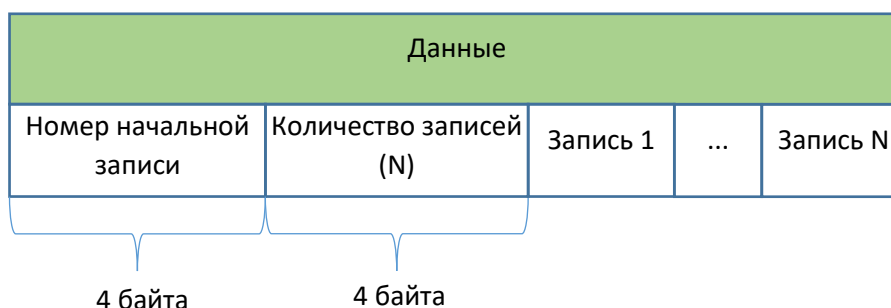


Рисунок 23. Поле данных положительного ответа на запрос чтения записей журнала

Таблица 12. Структура записи журнала событий

Смещение, байт	Размер, байт	Условное обозначение	Описание
0	8	RecordNumber	Сквозной порядковый номер записи во всех журналах (нумерация начинается с 0).
8	8	Timestamp	Метка времени с точностью до миллисекунд (уточнённое время UNIX – количество полных миллисекунд, прошедших с 00:00:00 01.01.1970 UTC). Время 00:00:00 01.01.1970 UTC является признаком невалидной записи (некорректная src).
16	2	Timeoffset	Смещение местного времени относительно UTC в секундах
18	NET/8+1	Events	Битовый массив событий.
$18 + ((NS+NU)/8 + 1)$	NBI/8 + 1	BinaryInputs	Битовый массив состояний двоичных входов.
$18 + ((NS+NU)/8 + 1) + (NBI/8 + 1)$	NBO/8 + 1	BinaryOutputs	Битовый массив состояний двоичных выходов.
$18 + ((NS+NU)/8 + 1) + (NBI/8 + 1) + (NBO/8 + 1)$	NBIS/8 + 1	BinaryInputSignals	Битовый массив состояний двоичных входных сигналов.
$18 + ((NS+NU)/8 + 1) + (NBI/8 + 1) + (NBO/8 + 1)$	NBOS/8 + 1	BinaryOutputSignals	Битовый массив состояний двоичных выходных сигналов.

Смещение, байт	Размер, байт	Условное обозначение	Описание
1) + (NBIS/8 + 1)			
18 + ((NS+NU)/8 + 1) + (NBI/8 + 1) + (NBO/8 + 1) + (NBIS/8 + 1) + (NBOS/8 + 1)	NAOS*4	AnalogOutputSignals	Массив состояний аналоговых выходных сигналов. Элемент массива – число с п. з. в соответствии с IEEE 754-1985 (IEC 60559).

Значения NET, NBI, NBO, NBIS, NBOS, NAOS содержатся в описателе записи журнала событий (см. табл. 8).

Таблица 13. Структура записи журнала сообщений

Смещение, байт	Размер, байт	Условное обозначение	Описание
0	8	RecordNumber	Сквозной порядковый номер записи во всех журналах (нумерация начинается с 0).
8	8	Timestamp	Метка времени с точностью до миллисекунд (уточнённое время UNIX – количество полных миллисекунд, прошедших с 00:00:00 01.01.1970 UTC). Время 00:00:00 01.01.1970 UTC является признаком невалидной записи (некорректная src).
16	2	Timeoffset	Смещение местного времени относительно UTC в секундах
18	1	SCADAAccessLevelID	Идентификатор уровня доступа от АСУ
19	1	KIWIAccessLevelID	Идентификатор уровня доступа от KIWI
20	1	reserve	Резерв
21	2	MsgID	Идентификатор события (таблица 9.1 ТЗ МИНИ-1), см. п.9.2 ТЗ МИНИ-01
23	3*4	MsgData	Данные события (таблица 9.1 ТЗ МИНИ-1). Под данные имеется три поля по 32 бита каждое, если поле не используется, то оно =0. Тип поля интерпретируется в зависимости от типа события (int32, float, маска)

Таблица 14. Идентификаторы уровней доступа пользователя в журналах

Код	Описание
30	Полный доступ
60	Инженер РЗА
90	Наладчик РЗА
120	Инженер АСУ
150	Управление из АСУ
180	Настройка из АСУ

Таблица 15. Структура записи журнала изменений уставок (версия 1)

Смещение, байт	Размер, байт	Условное обозначение	Описание
0	8	RecordNumber	Сквозной порядковый номер записи во всех журналах (нумерация начинается с 0).
8	8	Timestamp	Метка времени с точностью до миллисекунд (уточнённое время UNIX – количество полных миллисекунд, прошедших с 00:00:00 01.01.1970 UTC). Время 00:00:00 01.01.1970 UTC является признаком невалидной записи (некорректная src).
16	2	Timeoffset	Смещение местного времени относительно UTC в секундах
16	2	SettingID	Идентификатор изменённой уставки.
18	1	AccessLevelID	Идентификатор уровня доступа пользователя, изменившего уставку (см. табл. 14).
19	1	SourceID	Идентификатор источника изменения уставки (см. табл. 16).
20	4	OldValue	Старое значение уставки (целое число со знаком).
24	4	NewValue	Новое значение уставки (целое число со знаком).
28	1	Program	Номер программы уставок (для Лютик всегда равен 0)

Таблица 16. Идентификаторы источников изменений уставок и сигналов

Код	Описание
0	Лютик
2	KIWI
3	АСУ
4	Двоичный вход (только для сигналов)

Таблица 17. Структура записи журнала съёма сигнализации

Смещение, байт	Размер, байт	Условное обозначение	Описание
0	8	RecordNumber	Сквозной порядковый номер записи во всех журналах (нумерация начинается с 0).
8	8	Timestamp	Метка времени с точностью до миллисекунд (уточнённое время UNIX – количество полных миллисекунд, прошедших с 00:00:00 01.01.1970 UTC). Время 00:00:00 01.01.1970 UTC является признаком невалидной записи (некорректная src).
16	2	Timeoffset	Смещение местного времени относительно UTC в секундах
18	1	SourceID	Идентификатор источника сигнала «Съём сигнализации» (см. табл. 11).
19	[NBOS/8] + 1	BOSBefore	Состояния двоичных выходных сигналов до съёма сигнализации (битовый массив).
19 + [NBOS/8] + 1	[NBOS/8] + 1	BOSAAfter	Состояния двоичных выходных сигналов после съёма сигнализации (битовый массив).

В записи журнала съёма сигнализации (табл. 17) есть битовые массивы состояний двоичных выходных сигналов до и после возникновения события (поля BOSBefore и BOSAAfter). Соответствие между идентификаторами в подзаголовке и битами в битовых

массивах такое же, как и в журнале событий. Значение NBOS указано в описателе записи журнала съёма сигнализации.

4.1.13. Чтение краткой информации о записях журнала (подфункция 26)

Поле данных запроса содержит код типа журнала (см. табл. 7), номер начальной записи и количество записей. Нумерация записей начинается с нуля, запись с номером 0 – самая новая запись в журнале. В качестве кода типа журнала можно использовать любое значение из табл. 7, кроме того, которое обозначает все журналы.

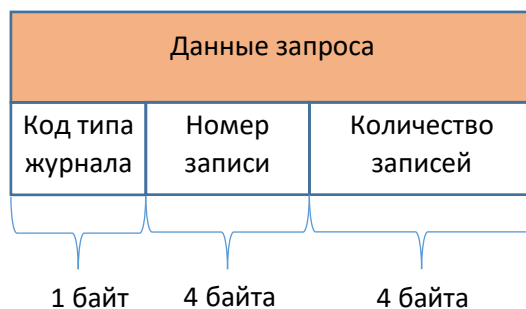


Рисунок 24. Структура запроса чтения краткой информации о записях журнала

Поле данных положительного ответа на этот запрос содержит номер начальной записи (нумерация начинается с 0), количество записей и последовательность записей. Каждая запись в этой последовательности имеет структуру, которая определяется типом журнала и версией структуры (см. табл. 18 – 21).

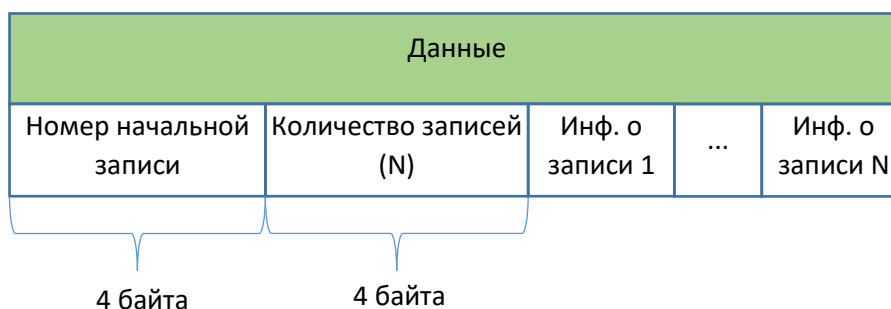


Рисунок 25. Поле данных положительного ответа на запрос чтения информации о записях журнала

Таблица 18. Структура краткой информации о записи журнала событий

Смещение, байт	Размер, байт	Условное обозначение	Описание
0	8	RecordNumber	Сквозной порядковый номер записи во всех журналах (нумерация начинается с 0).
8	8	Timestamp	Метка времени с точностью до миллисекунд (уточнённое время UNIX – количество полных миллисекунд, прошедших с 00:00:00 01.01.1970 UTC). Время 00:00:00 01.01.1970 UTC является признаком невалидной записи (некорректная src).
16	2	Timeoffset	Смещение местного времени относительно UTC в секундах
18	$\lfloor \text{NET}/8 \rfloor + 1$	Events	Битовый массив событий.

Значение NET содержится в описателе записи журнала событий (см. табл. 8).

Таблица 19. Структура краткой информации о записи журнала сообщений

Смещение, байт	Размер, байт	Условное обозначение	Описание
0	8	RecordNumber	Сквозной порядковый номер записи во всех журналах (нумерация начинается с 0).
8	8	Timestamp	Метка времени с точностью до миллисекунд (уточнённое время UNIX – количество полных миллисекунд, прошедших с 00:00:00 01.01.1970 UTC). Время 00:00:00 01.01.1970 UTC является признаком невалидной записи (некорректная src).
16	2	Timeoffset	Смещение местного времени относительно UTC в секундах

Таблица 20. Структура краткой информации о записи журнала изменений уставок

Смещение, байт	Размер, байт	Условное обозначение	Описание
0	8	RecordNumber	Сквозной порядковый номер записи во всех журналах (нумерация начинается с 0).
8	8	Timestamp	Метка времени с точностью до миллисекунд (уточнённое время UNIX – количество полных миллисекунд, прошедших с 00:00:00 01.01.1970 UTC). Время 00:00:00 01.01.1970 UTC является признаком невалидной записи (некорректная src).
16	2	Timeoffset	Смещение местного времени относительно UTC в секундах
18	2	SettingID	Идентификатор изменённой уставки.

Таблица 21. Структура краткой информации о записи журнала съёма сигнализации

Смещение, байт	Размер, байт	Условное обозначение	Описание
0	8	RecordNumber	Сквозной порядковый номер записи во всех журналах (нумерация начинается с 0).
8	8	Timestamp	Метка времени с точностью до миллисекунд (уточнённое время UNIX – количество полных миллисекунд, прошедших с 00:00:00 01.01.1970 UTC). Время 00:00:00 01.01.1970 UTC является признаком невалидной записи (некорректная gcs).
16	2	Timeoffset	Смещение местного времени относительно UTC в секундах
18	1	SourceID	Идентификатор источника сигнала «Съём сигнализации» (см. табл. 16).

4.1.14. Чтение описателя записи журнала съёма сигнализации (подфункция 41)

В запросе поле данных отсутствует.

Поле данных положительного ответа на этот запрос (рис. 26) содержит описатель записи журнала съёма сигнализации (см. табл. 22).

Данные ответа
Описатель записи журнала съёма сигнализации

Рисунок 26. Поле данных положительного ответа на запрос чтения описателя записи журнала съёма сигнализации

Таблица 22. Структура описателя записи журнала съёма сигнализации

Смещение, байт	Размер, байт	Условное обозначение	Описание
0	2	NBOS	Количество двоичных выходных сигналов.
2	2 * NBOS	BOSIDs	Массив идентификаторов двоичных выходных сигналов. Если идентификатор равен FFFF, то соответствующий элемент массива двоичных выходных сигналов в записи журнала не используется.

4.1.15. Передача длинной команды (подфункция 239)

Эта подфункция предназначена для передачи серверу команд, длина которых превышает стандартный размер MODBUS PDU.

Поле данных запроса содержит идентификатор команды, назначенный клиентом, общую длину команды, байтовое смещение очередного фрагмента команды и сам фрагмент команды, начинающийся с указанного смещения.

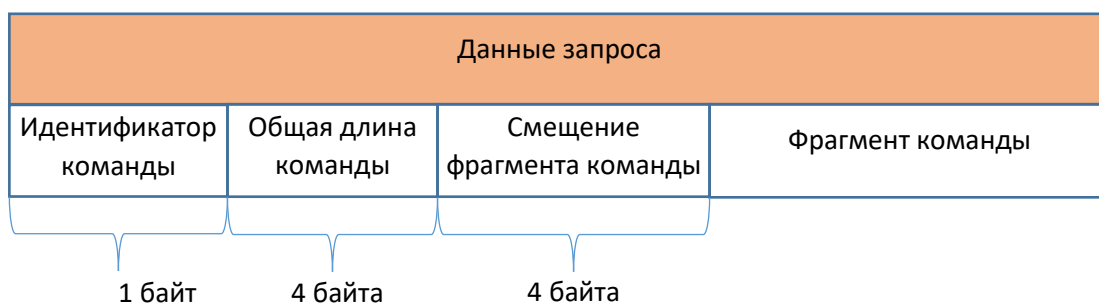


Рисунок 27. Структура запроса передачи длинной команды

В поле данных положительного ответа всегда содержится идентификатор полученной команды и тип результата (0 – подтверждение очередного фрагмента команды, 1 – результат исполнения команды). Если поле «Тип результата» содержит 0 (рис. 28), то за ним следует 32-битовое смещение следующего фрагмента команды. Если же поле «Тип результата» содержит 1 (рис. 29), то за ним следует результат исполнения команды, оформленный как ответ на запрос (MODBUS PDU).

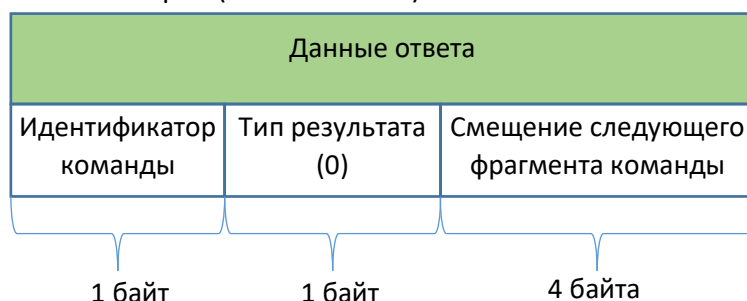


Рисунок 28. Поле данных положительного ответа на запрос передачи длинного запроса с типом результата 0

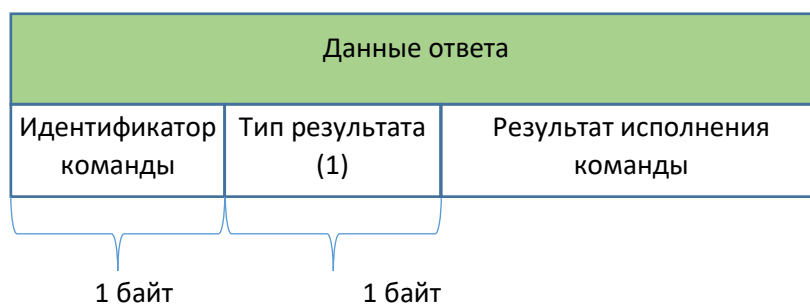


Рисунок 29. Поле данных положительного ответа на запрос передачи длинной команды с типом результата 1

4.1.16. Получение информации о буфере сервера (подфункция 240)

В запросе поле данных отсутствует.

Положительные ответы на этот запрос всегда короткие (т.е. тип ответа равен 0). Поле данных короткого положительного ответа на этот запрос содержит объём данных, находящихся в буфере сервера.

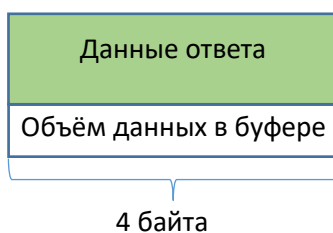


Рисунок 30. Поле данных положительного ответа на запрос информации о буфере сервера

4.1.17. Чтение данных из буфера сервера (подфункция 241)

Поле данных запроса содержит байтовое смещение от начала буфера и количество байтов для чтения.



Рисунок 31. Структура запроса чтения данных из буфера сервера

Положительные ответы на этот запрос всегда короткие (т.е. тип ответа равен 0). Поле данных короткого положительного ответа содержит данные, прочитанные из буфера сервера.



Рисунок 32. Поле данных положительного ответа на запрос чтения данных из буфера сервера

4.1.18. Уведомление сервера о завершении чтения из буфера (подфункция 242)

В запросе поле данных отсутствует.

В положительном ответе на этот запрос поле данных отсутствует.

4.1.19. Чтение конфигурации интерфейса RS-485 (подфункция 250)

Поле данных запроса (рис. 33) содержит порядковый номер интерфейса RS-485 (нумерация начинается с 0). Интерфейс 0 – RS485 на задней плате устройства, интерфейс 1 – передний порт USB (через FTDI). Если FTDI не используется, то интерфейс 1 отсутствует.



Рисунок 33. Структура запроса чтения конфигурации интерфейса RS-485

Поле данных положительного ответа (рис. 34) на этот запрос содержит конфигурацию указанного интерфейса (см. табл. 23 – 26).



Рисунок 34. Структура ответа на запрос чтения конфигурации интерфейса RS-485

Таблица 23. Коды скорости обмена данным через интерфейс RS-485

Код	Скорость, бод
0	По умолчанию
1	2400
2	4800
3	9600
4	14400
5	19200
6	38400
7	57600
8	115200

Таблица 24. Коды количества битов данных

Код	Количество битов данных
0	По умолчанию
1	7
2	8

Таблица 25. Коды количества стоповых битов

Код	Количество битов данных
0	По умолчанию
1	1
2	2

Таблица 26. Коды контроля паритета

Код	Количество битов данных
0	По умолчанию
1	нет контроля
2	чётный
3	нечётный

4.1.20. Запись конфигурации интерфейса RS-485 (подфункция 251)

Поле данных запроса (рис. 35) содержит порядковый номер настраиваемого интерфейса (нумерация начинается с 0) и параметры этого интерфейса (см. табл. 23 – 26). Интерфейс 0 – RS485 на задней плате устройства, интерфейс 1 – передний порт USB (через FTDI). Если FTDI не используется, то интерфейс 1 отсутствует.



Рисунок 35. Структура запроса записи конфигурации интерфейса RS-485

В положительном ответе на этот запрос поле данных отсутствует.

4.1.21. Чтение сведений об устройстве (подфункция 254)

Поле данных запроса содержит список кодов идентификационных параметров устройства, значения которых нужно прочитать (рис. 36). Коды параметров перечислены в табл. 27.

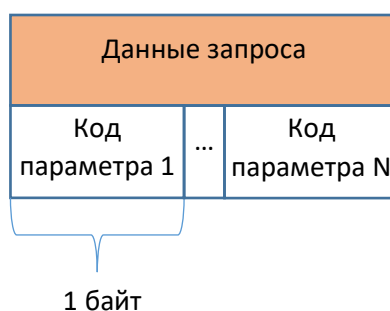


Рисунок 36. Структура запроса чтения сведений об устройстве

Поле данных положительного ответа на этот запрос содержит список значений идентификационных параметров (рис. 37). Каждый элемент этого списка состоит из кода параметра (см. табл. 27) и значения параметра (см. табл. 28 – 34). Количество параметров и порядок их следования в ответе может не соответствовать количеству и порядку следования параметров в запросе.

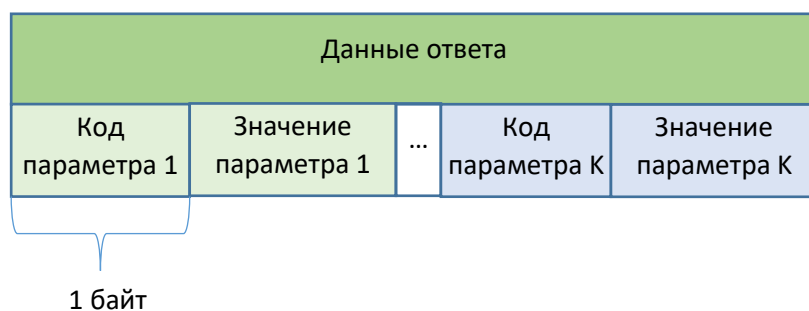


Рисунок 37. Поле данных положительного ответа на запрос чтения сведений об устройстве

Таблица 27. Коды идентификационных параметров устройства

Код	Наименование	Описание
0	Протокол	Версия протокола (табл. 28)
1	Код блока	Уникальный аппаратный код устройства (табл. 29)
2	Наименование блока	Полное условное наименование устройства в соответствии с определённой устройством конфигурацией (табл. 32)
3	Изготовитель	Сведения об изготовителе устройства (табл. 32)
4	Серийный номер блока	Серийный номер устройства (табл. 32)
5	Дата изготовления	Дата и время изготовления устройства (табл. 33)
6	Присоединение	Наименование присоединения (табл. 32)
7	Организация	Наименование организации, эксплуатирующей устройство (табл. 32)
8	Объект	Наименование объекта эксплуатации (табл. 32)
9	Комментарий	Текстовый комментарий (табл. 32)
10	Версия БФПО	Версия базового функционального ПО (БФПО) устройства (табл. 34)
11	Дата БФПО	Дата и время выпуска БФПО устройства (табл. 33)
12	Версия ядра ПО	Версия ПО ядра устройства (табл. 34)
13	Дата ядра ПО	Дата и время выпуска ПО ядра устройства (табл. 33)
14		Резерв
15		Резерв
16		Резерв
17		Резерв
18		Резерв
19		Резерв
20	Версия ПО ЦМ	Версия ПО модуля основного (центрального) процессора (табл. 34)
21	Дата ПО ЦМ	Дата и время выпуска ПО модуля основного (центрального) процессора (табл. 33)
22		Резерв
23		Резерв
24		Резерв
25		Резерв
26		Резерв
27		Резерв
...		...
44		Резерв
45		Резерв
46	Информация о ПО ЦП	Информация о ПО основного (центрального) процессора (см. табл. 35)

Код	Наименование	Описание
47		Резерв
48		Резерв
49		Резерв
...		Резерв
58		Резерв
59		Резерв
...		...
68		Резерв
69	Версия КД ЦМ	Версия конструкторской документации модуля основного (центрального) процессора (табл. 32)
70		Резерв
71		Резерв
72		Резерв
73		Резерв
74		Резерв
75		Резерв
76		Резерв
...		...
85		Резерв
86	Шифр проекта вторичных схем блока	Текст (табл. 32)
87		Резерв
88		Резерв
89		Резерв
90	Версия загрузчика ПО ЦП	Версия загрузчика ПО основного (центрального) процессора (табл. 34)
91	Дата загрузчика ПО ЦП	Дата и время выпуска загрузчика ПО основного (центрального) процессора (табл. 33)
92	Информация о загрузчике ПО ЦП	Информация о загрузчике ПО основного (центрального) процессора (см. табл. 35)
93		Резерв
94		Резерв
95		Резерв
...		...
200		UID основного процессора (Таблица 38)

Таблица 28. Структура версии протокола

Смещение, байт	Размер, байт	Условное обозначение	Описание
0	4	Version	Версия протокола
4	4	Signature	Сигнатура протокола (1297364533)

Таблица 29. Структура уникального аппаратного кода устройства

Смещение, байт	Размер, байт	Условное обозначение	Описание
0	1	DeviceType	Код типа устройства (см. табл. 37)
1	1	DeviceSubtype	Код подтипа устройства
2	1	Voltage	Код напряжения оперативного тока (0 – 110 В, 1 – 220 В)
3	1	SpecFunctionalUnit	Код типа функционального модуля специального назначения (см. табл. 30)
4	1	AuxComUnit	Код типа дополнительного коммуникационного модуля (см. табл. 31)
5	4	Reserved	Зарезервировано
9	1	NXIOU	Количество дополнительных модулей ввода-вывода
10	NXIOU	ExtIOUnits	Массив типов дополнительных модулей ввода-вывода (размер элемента массива – 1 байт)

Таблица 30. Коды типов функциональных модулей специального назначения

Код	Описание
0	Модуль отсутствует

Таблица 31. Коды типов дополнительных коммуникационных модулей

Код	Описание
0	Дополнительный коммуникационный модуль отсутствует

Таблица 32. Структура текстового параметра

Смещение, байт	Размер, байт	Условное обозначение	Описание
0	2	Length	Количество байтов в строке, включая завершающий ноль
2	Length	Text	Строка символов в кодировке UTF-8, оканчивающаяся нулём

Таблица 33. Структура метки времени

Смещение, байт	Размер, байт	Условное обозначение	Описание
0	8	Timestamp	Стандартное время UNIX – количество секунд, прошедших с 1970-01-01 00:00:00 UTC

Таблица 34. Структура версии

Смещение, байт	Размер, байт	Условное обозначение	Описание
0	4	MajorVersion	Старший номер версии
4	4	MinorVersion	Младший номер версии
8	4	Patch	Исправление
12	4	Revision	Редакция в SVN

Таблица 35. Структура информации о ПО

Смещение, байт	Размер, байт	Условное обозначение	Описание
0	2	Platform	Код аппаратной платформы (см. табл. 36)
2	2	DeviceType	Код типа устройства (см. табл. 37)
4	2	DeviceSubtype	Код подтипа устройства
6	1	Modified	Признак наличия незафиксированных изменений ПО (1 – изменения есть, 0 – изменений нет)
7	4	CompilerMajorVersion	Старший номер версии компилятора
11	4	CompilerMinorVersion	Младший номер версии компилятора
15	4	CompilerPatchVersion	Номер редакции компилятора
19	8	BuildTimestamp	Дата и время сборки ПО (табл. 33)
27	8	RevisionTimestamp	Дата и время редакции ПО (табл. 33)

Таблица 36. Коды аппаратных платформ устройства

Код	Описание
3	Лютик

Таблица 37. Коды типов устройства

Код	Описание
3	Лютик

Таблица 38. Структура строкового параметра

Смещение, байт	Размер, байт	Условное обозначение	Описание
0	2	Length	Количество байтов в строке
2	Length	String	Строка символов

5. Приложение 1. Пример использования запроса передачи длинной команды (подфункция 239)

Рассмотрим использование запроса передачи длинной команды на примере записи уставок. Предположим, что нужно записать значения 5 уставок с идентификаторами от 1 до 5 для двух программ. Для записи уставок используется подфункция 11 функции 65, поэтому команда записи уставок будет иметь вид, показанный в табл. 39. Структура поля данных этой команды (байты, начиная со смещения 4 в табл. 39) такая же, как и структура данных запроса записи уставок (см. рис. **Ошибка! Источник ссылки не найден.**). В данном случае размер поля данных команды составляет 66 байт, общая длина команды – 70 байт.

Таблица 39. Пример команды записи уставок

Смещение, байт	Размер, байт	Значение (шестн.)	Описание
0	1	41	Код функции (65)
1	1	0B	Код подфункции (11)
2	1	1E	Номер запроса (30)
3	1	42	Длина данных (66)
4	1	01	Номер программы уставок (1)
5	2	00 05	Количество уставок (5)
7	2	00 01	Идентификатор уставки (1)
9	4	00 00 02 01	Значение уставки (513)
13	2	00 02	Идентификатор уставки (2)
15	4	00 00 00 21	Значение уставки (33)
19	2	00 03	Идентификатор уставки (3)
21	4	00 00 00 22	Значение уставки (34)
25	2	00 04	Идентификатор уставки (4)
27	4	00 00 00 00	Значение уставки (0)
31	2	00 05	Идентификатор уставки (5)
33	4	FF FF FF FE	Значение уставки (-2)
37	1	02	Номер программы уставок (2)
38	2	00 05	Количество уставок (5)
40	2	00 01	Идентификатор уставки (1)
42	4	00 00 02 00	Значение уставки (512)
46	2	00 02	Идентификатор уставки (2)
48	4	00 00 00 20	Значение уставки (32)
52	2	00 03	Идентификатор уставки (3)
54	4	00 00 00 14	Значение уставки (20)
58	2	00 04	Идентификатор уставки (4)
60	4	00 00 00 01	Значение уставки (1)
64	2	00 05	Идентификатор уставки (5)
66	4	FF FF FF FD	Значение уставки (-3)

Для простоты предположим, что максимально допустимый размер MODBUS PDU составляет не 253, а 63 байта, тогда для передачи этой команды записи уставок потребуется 2 запроса с подфункцией 239. В первом запросе (табл. 40) передаются первые 50 байт команды, начиная со смещения 0. Положительный ответ на первый запрос (табл. 41) имеет тип результата 0 и указывает смещение следующего фрагмента команды (50). Во втором запросе (табл. 42) передаются последние 20 байт команды. Если при получении запроса

возникла ошибка, то сервер отправляет отрицательный ответ на подфункцию 239 (табл. 43). Если все фрагменты команды получены успешно, то сразу же после получения последнего фрагмента сервер исполняет команду и, если она исполнена успешно, то сервер отправляет положительный ответ на подфункцию 239 с типом результата 1 и положительным ответом на команду (табл. 44). Если при исполнении команды возникла ошибка, то отправляется положительный ответ на подфункцию 239 с типом результата 1 и отрицательным ответом на команду (табл. 45).

Таблица 40. MODBUS PDU 1-го запроса, передающего команду записи уставок

Смещение, байт	Размер, байт	Значение (шестн.)	Описание
0	1	41	Код функции (65)
1	1	EF	Код подфункции (239)
2	1	11	Номер запроса (17)
3	1	3B	Длина данных (59)
4	1	2B	Идентификатор команды (43)
5	4	00 00 00 46	Общая длина команды (70)
9	4	00 00 00 00	Смещение фрагмента команды (0)
13	1	41	Код функции (65)
14	1	0B	Код подфункции (11)
15	1	1E	Номер запроса (30)
16	1	42	Длина данных (66)
17	1	01	Номер программы уставок (1)
18	2	00 05	Количество уставок (5)
20	2	00 01	Идентификатор уставки (1)
22	4	00 00 02 01	Значение уставки (513)
26	2	00 02	Идентификатор уставки (2)
28	4	00 00 00 21	Значение уставки (33)
32	2	00 03	Идентификатор уставки (3)
34	4	00 00 00 22	Значение уставки (34)
38	2	00 04	Идентификатор уставки (4)
40	4	00 00 00 00	Значение уставки (0)
44	2	00 05	Идентификатор уставки (5)
46	4	FF FF FF FE	Значение уставки (-2)
50	1	02	Номер программы уставок (2)
51	2	00 05	Количество уставок (5)
53	2	00 01	Идентификатор уставки (1)
55	4	00 00 02 00	Значение уставки (512)
59	2	00 02	Идентификатор уставки (2)
61	2	00 00	Два старших байта значения уставки

Таблица 41. MODBUS PDU положительного ответа на 1-й запрос

Смещение, байт	Размер, байт	Значение (шестн.)	Описание
0	1	41	Функция (65)
1	1	EF	Подфункция (239)
2	1	11	Номер запроса (17)
3	1	07	Длина данных (7)
4	1	00	Тип ответа (0), код ответа (0)
5	1	2B	Идентификатор команды (43)
6	1	00	Тип результата (0)
7	4	00 00 00 32	Смещение следующего фрагмента команды (50)

Таблица 42. MODBUS PDU 2-го запроса, передающего команду записи уставок

Смещение, байт	Размер, байт	Значение (шестн.)	Описание
0	1	41	Код функции (65)
1	1	EF	Код подфункции (239)
2	1	12	Номер запроса (18)
3	1	1D	Длина данных (29)
4	1	2B	Идентификатор команды (43)
5	4	00 00 00 46	Общая длина команды (70)
9	4	00 00 00 32	Смещение фрагмента команды (50)
13	2	00 20	Два младших байта значения уставки
15	2	00 03	Идентификатор уставки (3)
17	4	00 00 00 14	Значение уставки (20)
21	2	00 04	Идентификатор уставки (4)
23	4	00 00 00 01	Значение уставки (1)
27	2	00 05	Идентификатор уставки (5)
29	4	FF FF FF FD	Значение уставки (-3)

Таблица 43. MODBUS PDU отрицательного ответа на 2-й запрос

Смещение, байт	Размер, байт	Значение (шестн.)	Описание
0	1	41	Функция (65)
1	1	EF	Подфункция (239)
2	1	11	Номер запроса (18)
3	1	01	Длина данных (1)
4	1	0D	Тип ответа (0), код ответа (13)

Таблица 44. MODBUS PDU положительного ответа на 2-й запрос и положительным ответом на команду

Смещение, байт	Размер, байт	Значение (шестн.)	Описание
0	1	41	Функция (65)
1	1	EF	Подфункция (239)
2	1	11	Номер запроса (18)
3	1	08	Длина данных (8)
4	1	00	Тип ответа (0), код ответа (0)
5	1	2B	Идентификатор команды (43)
6	1	01	Тип результата (1)
7	1	41	Код функции (65)
8	1	0B	Код подфункции (11)
9	1	1E	Номер запроса (30)
10	1	01	Длина данных (1)
11	1	00	Тип ответа (0), код ответа (0)

Таблица 45. MODBUS PDU положительного ответа на 2-й запрос и отрицательным ответом на команду

Смещение, байт	Размер, байт	Значение (шестн.)	Описание
0	1	41	Функция (65)
1	1	EF	Подфункция (239)
2	1	11	Номер запроса (18)
3	1	08	Длина данных (8)
4	1	00	Тип ответа (0), код ответа (0)
5	1	2B	Идентификатор команды (43)
6	1	01	Тип результата (1)
7	1	41	Код функции (65)
8	1	0B	Код подфункции (11)
9	1	1E	Номер запроса (30)
10	1	01	Длина данных (1)
11	1	0C	Тип ответа (0), код ответа (12)

6. Литература

1. MODBUS over Serial Line Specification and Implementation Guide V1.02; www.modbus.org; December 20, 2006.
2. MODBUS Application Protocol Specification V1.1b3; www.modbus.org; April 26, 2012.
3. ISO/IEC 8482:1993. Information technology – Telecommunications and information exchange between systems – Twisted pair multipoint interconnections.