



# Контроллеры многофункциональные «Интеллектуальный контроллер SM160» Протокол обмена

*(редакция 3)*

Документ содержит описание протокола обмена Интеллектуального контроллера SM160 и ПО верхнего уровня в локальном и удалённом режиме доступа к конфигурации и данным

## Оглавление

1.	Общие положения .....	3
1.1.	Применение.....	3
1.2.	Типы протоколов.....	3
1.3.	Стандартные параметры .....	3
2.	Описание протокола .....	3
2.1.	Канальный уровень .....	3
2.2.	Прикладной уровень .....	4
2.2.1.	Коды функций.....	4
2.2.1.1.	Аутентификация (0x46) .....	5
2.2.1.2.	Чтение регистров (0x03) .....	7
2.2.1.3.	Запись регистров (0x10).....	7
2.2.1.4.	Диагностика подключения (0x08).....	7
2.2.1.5.	Чтение файла (0x45) .....	8
2.2.2.	Коды ошибок Modbus .....	8
2.2.3.	Порядок взаимодействия .....	8
2.3.	Подпротокол чтения файлов .....	10
2.3.1.	Формат кадра .....	11
2.3.2.	Формат дейтаграмм .....	13
2.3.3.	Запрос. Согласование размера кадра (0x0C).....	14
2.3.4.	Ответ. Согласование размера кадра (0x8C) .....	15
2.3.5.	Запрос. Чтение файла (0x07) .....	16
2.3.6.	Ответ. Чтение файла (0x87).....	17
2.3.7.	Пример чтения файла по протоколу Modbus/TCP .....	18
3.	Форматы данных.....	28
3.1.	Карта регистров .....	28
3.1.1.	Чтение и запись даты и времени .....	29
3.1.2.	Запись значения плавной коррекции времени.....	32
3.2.	Форматы файлов.....	33
3.2.1.	Общий формат .....	33
3.2.2.	Формат файла списка устройств («serial.xml») .....	34
3.2.3.	Формат файлов архивов (arh_*.xml).....	34
3.2.4.	Кодировка типов приборов.....	37
3.3.	Имена параметров .....	37
3.3.1.	Пример интерпретации значения параметра .....	41
3.3.2.	Кодировка результата сбора, параметр «EV».....	41
3.3.3.	Кодировка событий, параметр «EVD» .....	42



## 1. Общие положения

### 1.1. Применение

Документ содержит описание протокола обмена Интеллектуального контроллера SM160 (ИК SM160) и программного обеспечения (ПО) верхнего уровня в локальном и удалённом режиме доступа к конфигурации контроллера, накопленным в базе данных (БД) контроллера данным потребления энергоресурсов и журналам событий.

Прямое взаимодействие с подключенным к ИК SM160 оборудованием возможно при соответствующей настройке контроллера в туннельном режиме с использованием протокола обмена оконечного оборудования без использования дополнительных протоколов.

### 1.2. Типы протоколов

В качестве базового протокола для взаимодействия с ИК SM160 используется стандартный протокол:

- Modbus TCP – для взаимодействия по локальной сети Ethernet и GPRS;
- Modbus RTU – для взаимодействия по RS485.

### 1.3. Стандартные параметры

Стандартные параметры связи для взаимодействия с ИК SM160 по протоколу обмена:

- Собственный адрес (сетевой номер) – «255»;
- Имя пользователя – «root»;
- Пароль – «12345».

## 2. Описание протокола

### 2.1. Канальный уровень

Структура пакета для протокола Modbus TCP:

Id транзакции	Id протокола	Длина пакета	Адрес ведомого устройства	Код функции	Данные
2 байта	2 байта	2 байта	1 байт	1 байт	Данные (размер зависит от кода функции), либо код ошибки (1 байт)

Назначение полей пакета для протокола Modbus TCP:

- «Id транзакции» – уникальный
- «Id протокола» – всегда 0;
- «Длина пакета» – длина пакета, включая поля «Адрес ведомого устройства», «Код функции», «Данные»;
- «Адрес ведомого устройства» – собственный адрес ИК SM160;
- «Код функции» – код функции;
- «Данные» – блок данных в соответствии с кодом функции.

Структура пакета для протокола Modbus RTU:

Адрес ведомого устройства	Код функции	Данные	Блок обнаружения ошибок
1 байт	1 байт	Данные (размер зависит от кода функции), либо код ошибки (1 байт)	2 байта

Назначение полей пакета для протокола Modbus RTU:

- «Адрес ведомого устройства» – собственный адрес ИК SM160;
- «Код функции» – код функции;
- «Данные» – блок данных в соответствии с кодом функции;
- «Блок обнаружения ошибок» – контрольная сумма пакета, включая все поля.

Структура пакета для запросов и ответов идентична. На один пакет запроса возможен один пакет ответа.

Поля «Id транзакции», «Id протокола», «Адрес ведомого устройства», «Код функции» транслируются из пакета запроса в пакет ответа за исключением случая возникновения ошибки.

В случае возникновения ошибки при обработке запроса:

- Поле «Код функции» = «Код функции» OR 0x80 (старший бит устанавливается в значение «1»);
- Поле «Данные» = Код ошибки (1 байт).

## 2.2. Прикладной уровень

### 2.2.1. Коды функций

Коды функций:

Код функции	Назначение
0x46	Функция аутентификации используется для проверки подлинности пользователя, обращающегося к ИК SM160 по протоколу обмена
0x03	Функция чтения регистра используется для чтения содержимого указанных регистров из ИК SM160
0x10	Функция чтения регистра используется для записи содержимого указанных регистров в ИК SM160
0x08	Функция диагностики подключения используется для проверки связи с ИК SM160
0x45	Функция чтения файлов используется для получения файлов с ИК SM160

**2.2.1.1. Аутентификация (0x46)**

Для аутентификации используется специализированный подпротокол аутентификации, встроенный в данные запросов и ответов основного протокола.

Общий список типов пакетов специализированного подпротокола аутентификации в типовом порядке использования:

Тип пакета	Инициатор	Назначение
0x03	ЭВМ	Запрос аутентификации (передача имени пользователя)
0x04	ИК SM160	Передача ключа аутентификации
0x05	ЭВМ	Передача зашифрованного ключа аутентификации
0x01	ИК SM160	Подтверждение успешной аутентификации
0x02	ИК SM160	Уведомление об отказе в аутентификации (неверное имя пользователя и/или пароль)

Формат пакета запроса аутентификации (передачи имени пользователя):

Тип пакета	Метод аутентификации	Имя пользователя
0x03	0x00	16 байт (имя не более 15 байт)

Формат пакета передачи ключа аутентификации:

Тип пакета	Ключ аутентификации
0x04	16 байт

Формат пакета передачи зашифрованного ключа аутентификации:

Тип пакета	Зашифрованный ключ аутентификации
0x04	16 байт

Формат пакета подтверждения успешной аутентификации:

Тип пакета
0x01

Формат пакета уведомления об отказе в аутентификации (неверное имя пользователя и/или пароль):

Тип пакета
0x02

Описание форматов:

- Логин – имя пользователя, длина имени фиксирована - 16 байт. Имя представляет собой латинские буквы в нижнем регистре, не более 15 символов (pchar). Не использованные байты имени заполняются нулевым байтом.
- Ключ аутентификации – последовательность из 16 байт, формируемая контроллером;
- Зашифрованный ключ аутентификации – ключ аутентификации, зашифрованный по алгоритму AES с длиной ключа шифрования 128 бит. При этом в качестве ключа шифрования используется хэш пароля аутентификации, полученный по алгоритму MD5.

Пример процедуры аутентификации при использовании имени пользователя «root» и пароля «12345».

1)  $\exists \text{BM} \rightarrow \text{SM160}$ : 00 02 00 00 00 14 FF 46 03 00 72 6F 6F 74 00 00 00 00  
00 00 00 00 00 00 00 00

2) SM160→ $\Theta$ BM: 00 02 00 00 00 13 FF 46 04 01 01 01 01 01 01 01 01 01  
01 01 01 01 01 01 01

3)  $\exists \text{BM} \rightarrow \text{SM160}$ : 00 03 00 00 00 13 FF 46 05 B0 98 6D 6B F3 EF D3 19 FF  
D4 8F 0E A1 07 9F FC

4) SM160 → ЭВМ: 00 03 00 00 00 03 FF 46 01

Уровень	Трафик	Комментарий
Modbus	00 03 00 00 00 03 FF 46	Id транзакции = 3 Id протокола Длина Modbus пакета, 3 байта. Адрес устройства, SM160. Функция 0x46, подпротокол аутентификации.
Аутентификация	01	Тип пакета аутентификации «Успешная аутентификация».

**2.2.1.2. Чтение регистров (0x03)**

Формат данных запроса:

Адрес регистра	Количество регистров
2 байта	2 байта

Формат данных ответа:

Размер блока данных	Данные регистров
1 байт	2 байта на каждый регистр

Байты регистров должны записываться в порядке «Big-Endian»: сначала старший байт, потом младший. Перечень доступных регистров приведен в разделе 3.1.

**2.2.1.3. Запись регистров (0x10)**

Формат данных запроса:

Адрес регистра	Количество регистров	Количество байт	Данные регистров
2 байта	2 байта	1 байт	2 байта на каждый регистр

Формат данных ответа:

Адрес регистра	Количество регистров
2 байта	2 байта

Байты регистров должны записываться в порядке «Big-Endian»: сначала старший байт, потом младший. Перечень доступных регистров приведен в разделе 3.1.

**2.2.1.4. Диагностика подключения (0x08)**

Формат данных запроса:

Диагностическая последовательность
Произвольный размер

Формат данных ответа:

Диагностическая последовательность
Произвольный размер

В нормальном режиме работы диагностическая последовательность ответа должна повторять диагностическую последовательность запроса.



**2.2.1.5. Чтение файла (0x45)**

Для чтения файлов используется специализированный подпротокол чтения файлов.

Формат данных запроса и ответа:

Кадр подпротокола чтения файлов
Произвольный размер

Описание формата кадров подпротокола чтения файлов см в п. 2.3.

Формат данных «Modbus Next», пакет о готовности принять следующий кадр.

Кадр «Modbus Next»
2 нулевых байта: 0x0, 0x0

**2.2.2. Коды ошибок Modbus**

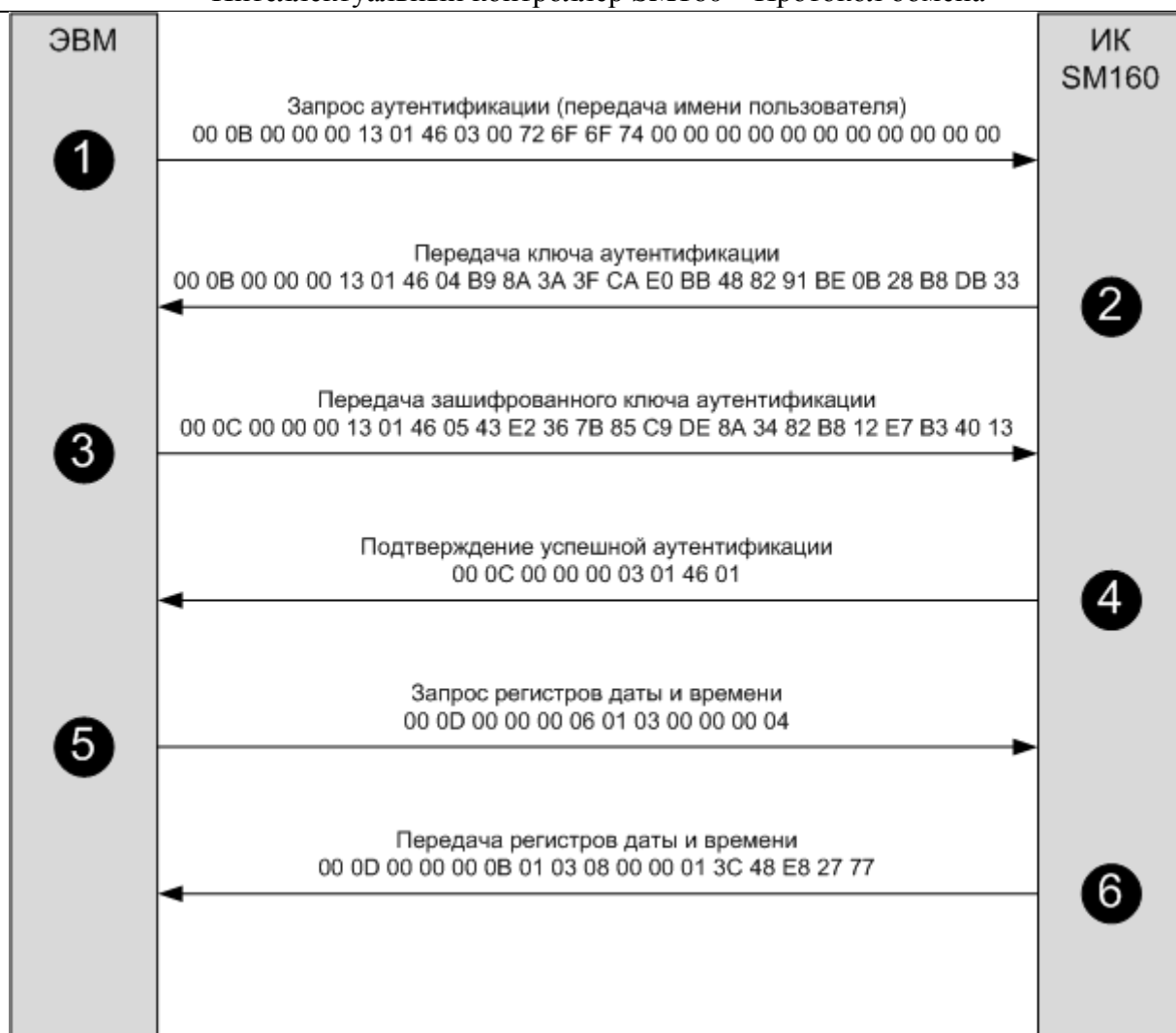
Коды ошибок:

Код ошибки	Назначение
0x01	Неверная функция
0x02	Неверный адрес регистра
0x03	Неверное значение регистра
0x0C	Клиент не прошёл аутентификацию

**2.2.3. Порядок взаимодействия**

Типовой порядок взаимодействия с ИК SM160 по протоколу обмена, начинается с процедуры аутентификации. В случае обращения без аутентификации, возвращается соответствующий код ошибки.

Типовой порядок взаимодействия с ИК SM160:



### **2.3. Подпротокол чтения файлов**

Подпротокол чтения файлов построен по принципу «запрос-ответ» и сам по себе имеет двухуровневую структуру:

1. Дейтаграммный уровень. Дейтаграммы оперируют действиями над файлами: запросить часть файла, передать часть файла и т.п.
  - 1.1. Дейтаграмма «запрос», передается от ЭВМ в ИК SM160.
  - 1.2. Дейтаграмма «ответ», передается в ответ от ИК SM160 в ЭВМ.
2. Кадровый уровень. Один кадр целиком помещается в одном пакете Modbus и транспортирует дейтаграммы по частям, помещающимся в один кадр. Таким образом, одна дейтаграмма может передаваться несколькими кадрами меньшей длины.
  - 2.1. «Мастер-кадр», передается от ЭВМ в ИК SM160.
  - 2.2. «Слэйв-кадр», передается в ответ от ИК SM160 в ЭВМ.

Перед непосредственным чтением файлов рекомендуется выполнить процедуру согласования размера части дейтаграмм файлового подпротокола. Если процедуру согласования не выполнить, то допустимый размер части дейтаграммы, передающейся внутри одного пакета по протоколу Modbus, равен 20480 байт. Размер может быть изменен в процессе согласования размеров, в диапазоне от 58 до 20480 байт.

При этом полный размер пакета Modbus/TCP будет больше, чем выбранный размер кадров, на величину служебных полей пакета Modbus/TCP (больше на 8 байт).

После согласования размеров перейти к обмену дейтаграммами запросами и ответами.

**2.3.1. Формат кадра**

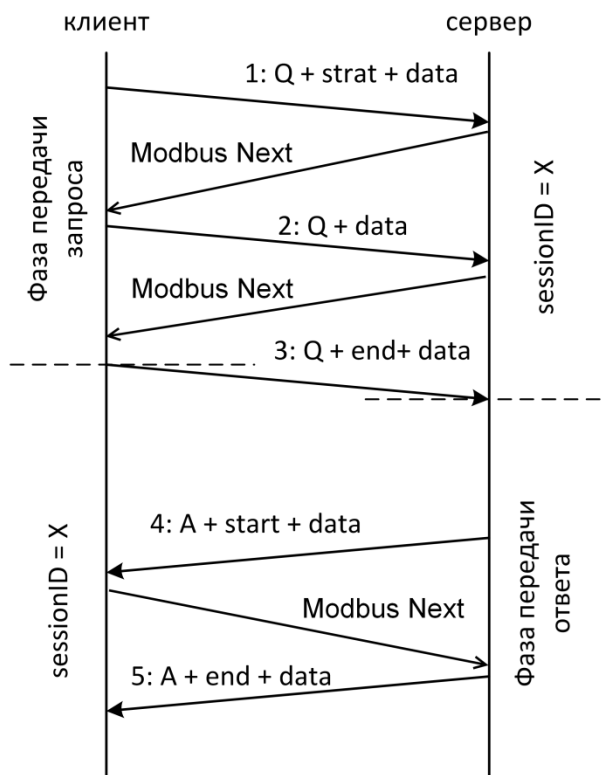
Форматы «мастер-кадров» и «слэйв-кадров»:

№ п/п	Название поля	Размер, байт	Описание
1.	SessionID	2	Идентификатор сессии. Назначается, стороной передающей первый «мастер-кадр». Отвечающая сторона возвращает этот же номер в «слэйв-кадрах». Внутри сессии может быть несколько мастер и слэйв кадров, все они должны иметь один и тот же SessionID. Формат: целое число (uint32).
2.	DatagramOffset	4	Номер байта в дейтаграммах в рамках одной сессии. Контрольное поле, начинается с нуля, заканчивается числом, равным сумме длин дейтаграммы запроса и дейтаграммы ответа. Формат: целое число (uint64).
3.	DatagramPartSize	2	Размер части дейтаграммы (поле DatagramPart), байт. Формат: целое число (uint64).
4.	CRC16	2	Контрольная сумма рассчитывается по полиному CRC-16-CCITT $x^{16} + x^{12} + x^5 + 1$ , для всего кадра: все поля с №1 и по №6 включительно. В момент вычисления контрольной суммы само поле №5 (CRC16) обнулено. Формат: целое число (uint16). <a href="http://ru.wikipedia.org/wiki/Циклический_избыточный_код">http://ru.wikipedia.org/wiki/Циклический_избыточный_код</a>
5.	Flags	2	Флаги (см. ниже). Формат: целое число (uint32).
6.	DatagramPart	Переменный	Часть дейтаграммы (или дейтаграмма целиком, если помещается в пакеты Modbus, которые могут быть больше 255 байт).

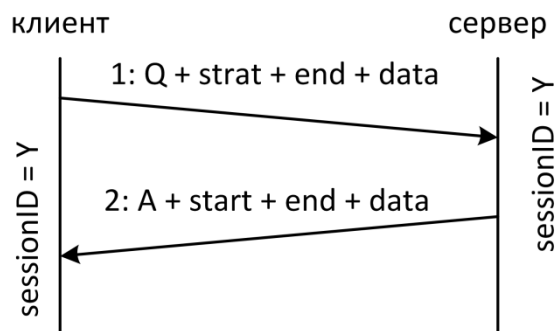
Кодировка поля Flags.

№ п/п	Название	Номера бит	Описание
1.	QA	0	0 – «слэйв-кадр» (от контроллера), 1 – «мастер-кадр» (от ЭВМ)
2.	START	1	Признак первого кадра.
3.	END	2	Признак последнего кадра.
4.		3...15	Не используются и должны быть заполнены нулями.

Примеры обмена кадрами:



а) Множество мастер и слэйв кадров.



б) Минимально возможное количество мастер и слэйв кадров.

Оба примера передают одну дейтаграмму «запрос» и одну дейтаграмму «ответ». В варианте «а» обе дейтаграммы передаются несколькими кадрами, в варианте «б» каждая дейтаграмма поместилась в один кадр.

Примечания:

Q – обозначает QA=1, «мастер-кадр»;

A – обозначает QA=0, «слэйв-кадр»;

start – Флаг START=1, первый кадр;

end – Флаг END=1, последний кадр;

data – обозначает часть дейтаграммы;

клиент – ЭВМ;

сервер – SM160;

Modbus Next – Modbus пакет о готовности принять следующий кадр.

### 2.3.2. Формат дейтаграмм

Форматы дейтаграмм «запросов» и «ответов»:

№ п/п	Название поля	Размер, байт	Описание
1.	DatagrammType	1	Тип дейтаграммы. Формат: целое число (uint8).
2.	CompressionType	1	Флаг использования сжатия. 0x00 – не используется, 0x01 - используется. Формат: целое число (uint8).
3.	DatagrammOptionsLength	4	Размер опций, байт. Формат: целое число (int32).
4.	DatagrammOptions	Переменный	Набор опций. Зависит от типа дейтаграммы
5.	DatagrammData	Переменный	Данные, передаваемые в дейтаграмме. Могут отсутствовать.
6.	DatagrammDataLength	8	Размер области данных дейтаграммы, байт. Формат: целое число (int64).
7.	DatagrammCRC32	4	Контрольная сумма рассчитывается по полиному CRC-32-IEEE 802.3, с поля №1 по поле №5 включительно. Формат: целое число (uint32). <a href="http://ru.wikipedia.org/wiki/Циклический_избыточный_код">http://ru.wikipedia.org/wiki/Циклический_избыточный_код</a>

Поврежденные дейтаграммы отбрасываются. Никакой реакции на них нет. Отсутствие ответа на поврежденные дейтаграммы определяется по тайм-ауту ответа протокола Modbus/TCP.

Типы дейтаграмм, указываемых в поле Type:

№ п/п	Тип дейтаграммы (hex)	Описание
1.	0x0C	Запрос. Согласование размера кадра.
2.	0x8C	Ответ на запрос согласования размера кадра.
3.	0x07	Запрос. Чтение файла из SM160.
4.	0x87	Ответ на запрос чтения файла. Содержит либо часть файла, либо информацию об ошибке (например, если файла нет).

**2.3.3. Запрос. Согласование размера кадра (0x0C)**

Формат дейтаграммы запроса о согласовании размера части дейтаграммы (DatagramPartSize) в кадре:

№ п/п	Название поля	Размер, байт	Значение	Примечание
1.	DatagrammType	1	0x0C	Тип дейтаграммы.
2.	CompressionType	1	0x00	Флаг использования сжатия. Формат: целое число (uint8).
3.	DatagrammOptionsLength	4	4	Размер опций.
4.	DatagrammOptions.FrameSizeNeed	4	Желаемая максимальная длина поля DatagramPart кадра, байт.	Информацию о поле DatagramPart см. в пункте «2.3.1. Формат кадров». Формат: целое (int32).
5.	DatagrammData	0		Поле отсутствует.
6.	DatagrammDataLength	8	0	
7.	DatagrammCRC32	4	CRC32	

Внимание! Дейтаграммы согласования размеров кадров всегда передаются одним кадром.

**2.3.4. Ответ. Согласование размера кадра (0x8C)**

Формат дейтаграммы ответа на запрос о согласовании размера части дейтаграммы (DatagramPartSize) в кадре:

№ п/п	Название поля	Размер, байт	Значение	Примечание
1.	DatagrammType	1	0x8C	Тип дейтаграммы.
2.	CompressionType	1	0x00	Флаг использования сжатия. Формат: целое число (uint8).
3.	DatagrammOptionsLength	4	Например, 13	Размер опций. В примере 13 байт – согласование прошло успешно.
4.	DatagrammOptions.StatusCode	4	0	Код выполнения. = 0 – ОК, значение поля DatagrammOptions.FrameSizeLimit равно размеру кадра. Формат: целое (int32).
5.	DatagrammOptions.StatusMsgLen	2	Например, 2	Длина текстового статусного сообщения, без символа конца строки. Формат: целое число (int16). В примере 2 байта – согласование прошло успешно.
6.	DatagrammOptions.StatusMsgStr	3	“ОК”+0	Текстовое статусное сообщение. Формат: строка (pchar), заканчивающаяся символом конца строки 0. В примере строка равна «ОК» – согласование прошло успешно.
7.	DatagrammOptions.FrameSizeLimit	4	Возможный размер кадра.	Размер кадра определяется техническими возможностями ИК SM160. Например, если желаемый размер больше, чем технически возможный, то вернется меньший размер кадра, а ЭВМ обязана передавать кадры не более технически возможного размера кадра.
8.	DatagrammData	0		Поле отсутствует.
9.	DatagrammDataLength	8	0	
10.	DatagrammCRC32	4	CRC32	

Внимание! Дейтаграммы согласования размеров кадров всегда передаются одним кадром.



**2.3.5. Запрос. Чтение файла (0x07)**

Формат дейтаграммы запроса чтения файла целиком.

№ п/п	Название поля	Размер, байт	Значение	Примечание
1.	DatagrammType	1	0x07	Тип дейтаграммы.
2.	CompressionType	1	0x00 – не используется, 0x01 - используется	Флаг использования сжатия. Формат: целое число (uint8).
3.	DatagrammOptionsLength	4	Например, 55	Размер опций. В примере 55 – это сумма длин полей с №4 по №7 включительно.
4.	DatagrammOptions.Offset	8	0	Формат: целое (uint64). 0 = с начала файла.
5.	DatagrammOptions.FileSize	8	0xFFFFFFFFFFFFFFFF	Формат: целое (uint64). 0xFFFFFFFFFFFFFFFF = весь файл.
6.	DatagrammOptions.NameLen	2	Например, 36	Длина имени файла, без символа конца строки. Формат: целое (int32). В примере 36 – это длина строки с именем файла в следующем поле №7.
7.	DatagrammOptions.NameStr	37	Например, “/usr/piramida/store/arch_20121212.xml”+0	Полное имя файла, включая путь. Формат: строка (pchar), заканчивающаяся символом конца строки 0.
8.	DatagrammData	0		Поле отсутствует.
9.	DatagrammDataLength	8	0	
10.	DatagrammCRC32	4	CRC32	

**2.3.6. Ответ. Чтение файла (0x87)**

Формат дейтаграммы ответа на запрос чтения файла целиком.

№ п/п	Название поля	Размер, байт	Значение	Примечание
1.	DatagrammType	1	0x87	Тип дейтаграммы.
2.	CompressionType	1	0x00 – не используется, 0x01 - используется	Флаг использования сжатия. Формат: целое число (uint8).
3.	DatagrammOptionsLength	4	Например, 9	Размер опций. В примере 9 – это сумма длин полей с №4 по №6 включительно.
4.	DatagrammOptions.StatusCode	4	0	Код выполнения. = 0 – ОК, прочитан фрагмент файла. Формат: целое (int32).
5.	DatagrammOptions.StatusMsgLen	2	Например, 2	Длина текстового статусного сообщения, без символа конца строки. Формат: целое число (int16). В примере 2 байта – длина строки «ОК».
6.	DatagrammOptions.StatusMsgStr	3	“ОК”+0	Текстовое статусное сообщение. Формат: строка (pchar), заканчивающаяся символом конца строки 0. В примере строка равна «ОК» – фрагмент файла прочитан успешно.
7.	DatagrammData.FileChunks	Переменный	Набор структур FileChunk.	Содержимое файла, по-фрагментно (FileChunk). Набор фрагментов заканчивается структурой FileChunk размером 0.
8.	DatagrammDataLength	8	Сумма данных всех структур FileChunk.	Длина поля №7, DatagrammData. При этом, все фрагменты файла, FileChunkLength, образуют файл целиком.
9.	DatagrammCRC32	4	CRC32	

## Формат структуры DataChunk (фрагмент файла).

№ п/п	Название поля	Размер, байт	Описание
1.	FileChunkLength	2	Длина фрагмента. Формат (uint16). 1) От 1 до 65535. Если файл больше чем размер 65535, то файл фрагментируется по 65535 байт, и последний фрагмент может быть меньше 65535 (остаток от деления). 2) 0 – фрагмент с длиной ноль, является признаком окончания файла.
2.	FileChunkData	FileChunkLength	Фрагмент файла.

## 2.3.7. Пример чтения файла по протоколу Modbus/TCP

В примере показан трафик при выполнении согласования размеров блока файла (по 80 байт) и чтения файла «/usr/piramida/store/arh\_20121212.xml». Считается, что аутентификация выполнена, сетевой адрес ИК SM160 = 255 (0xFF).

Для удобства восприятия траффика, поля файлового подпротокола подчеркнуты и выделены жирным шрифтом.

## 1) Запрос. Согласование размера кадра.

Трафик ЭВМ→SM160: 00 01 00 00 00 24 FF 45 01 00 00 00 00 00 16  
00 80 A6 07 00 0C 00 04 00 00 00 50 00 00 00 00 00 00 00 00 00  
00 A0 89 F1 B6

Т.к. дейтаграммы согласования размеров кадров всегда передаются одним кадром, то приводится расшифровка уже отправленного траффика.

Уровень	Трафик	Комментарий
Modbus	00 01 00 00 00 24 FF 45	Id транзакции Id протокола Длина Modbus пакета, 36 байт. Адрес устройства, SM160. Функция 0x45, подпротокол передачи файлов.
Мастер кадр	01 00 00 00 00 00 16 00 80 A6 07 00	SessionID = 1 DatagrammOffset = 0 DatagrammPartSize = 22 байта CRC16 = 0xA680 Flags, набор признаков: QA=1 – «мастер кадр»; START=1 – первый кадр; END=1 – он же и последний кадр.
Дейтаграмма	0C 00 04 00 00 00 50 00 00 00  00 00 00 00 00 00 00 00 A0 89 F1 B6	DatagrammType = 0x0C – Запрос. Согласование размера кадра. Reserved = 0. DatagrammOptionsLength = 4 байта. DatagrammOptions.FrameSizeNeeded = 80 байт, желаемый размер кадра DatagrammPart.  DatagrammDataLength = 0 CRC32

2) Ответ на запрос согласования размера кадра.

Трафик ЭВМ ← SM160: 00 01 00 00 00 2D FF 45 01 00 16 00 00 00 1F  
00 89 CA 06 00 8C 00 0D 00 00 00 00 00 00 00 02 00 4F 4B 00 50 00  
00 00 00 00 00 00 00 00 00 00 51 7A 49 80

Т.к. дейтаграммы согласования размеров кадров всегда передаются одним кадром, то приводится расшифровка уже отправленного трафика.

Уровень	Трафик	Комментарий
Modbus	00 01 00 00 00 2D FF 45	Id транзакции Id протокола Длина Modbus пакета, 45 байт. Адрес устройства, SM160. Функция 0x45, подпротокол передачи файлов.
Слэйв кадр	01 00	SesionID = 1.
	16 00 00 00	DatagrammOffset = 22 (нумерация продолжается с длины дейтаграммы запроса).
	1F 00	DatagrammPartSize = 31 байт.
	89 CA	CRC16 = 0xCA89
	06 00	Flags, набор признаков: QA=0 – «слэйв кадр»; START=1 – первый кадр; END=1 – он же и последний кадр.
Дейтаграмма	8C	DatagrammType = 0x8C – Ответ на запрос согласования размера кадра.
	00	Reserved = 0.
	0D 00 00 00	DatagrammOptionsLength = 4 байта.
	00 00 00 00	DatagrammOptions.StatusCode = 0 (OK).
	02 00	DatagrammOptions.StatusMsgLen = 2.
	4F 4B 00	DatagrammOptions.StatusMsgStr = “OK”+0.
	50 00 00 00	DatagrammOptions.FrameSizeLimit = 80 байт.
	00 00 00 00 00 00 00 00	DatagrammDataLength = 0 байт.
	51 7A 49 80	DatagrammCRC32

Внимание! Дейтаграммы согласования размеров кадров всегда передаются одним кадром.

## 3) Запрос. Чтение файла.

Дейтаграмма запроса на чтение файла, перед разбиением на кадры.

Бинарное представление дейтаграммы	Комментарий
07	DatagrammType = 0x07 – Запрос. Чтение файла из SM160.
00	Reserved.
37 00 00 00	DatagrammOptionsLength = 55 байт.
00 00 00 00 00 00 00 00	DatagrammOptions.Offset = 0.
FF FF FF FF FF FF FF FF	DatagrammOptions.FileSize = весь файл.
24 00	DatagrammOptions.NameLen = 36 bytes
2F 75 73 72 2F 70 69 72 61 6D 69 64 61 2F 73 74 6F 72 65 2F 61 72 68 5F 32 30 31 33 30 31 32 33 2E 78 6D 6C 00	DatagrammOptions.NameStr = “/usr/piramida/store/arch_20121212.xml”+0
00 00 00 00 00 00 00 00 00	DatagrammDataLength = 0
A3 56 58 77	CRC32

Следует учесть, что длина дейтаграммы более 80 байт (ранее выполненного согласования размера кадров), и она будет передана двумя кадрами. Для удобства восприятия, в таблице выделено серым цветом часть дейтаграммы, которая будет отправлена вторым кадром.

4) Первый кадр запроса.

Трафик ЭВМ→SM160: 00 02 00 00 00 52 FF 45 02 00 00 00 00 00 44  
00 2E 2D 03 00 07 00 37 00 00 00 00 00 00 00 00 00 00 00 FF FF FF  
FF FF FF FF FF 24 00 2F 75 73 72 2F 70 69 72 61 6D 69 64 61 2F 73  
74 6F 72 65 2F 61 72 68 5F 32 30 31 33 30 31 32 33 2E 78 6D 6C 00  
00 00 00 00 00 00 00 00

Уровень	Трафик	Комментарий
Modbus	00 02 00 00 00 52 FF 45	Id транзакции = 2 Id протокола Длина Modbus пакета, 82 байт. Адрес устройства, SM160. Функция 0x45, подпротокол передачи файлов.
Мастер кадр	02 00 00 00 00 00 44 00 2E 2D 03 00	SesionID = 2 DatagrammOffset = 0 DatagrammPartSize = 68 байт. (сумма длин всех полей уровней «мастер-кадр» и «Дейтаграмма» = 80 байт, что было выбрано при согласовании размера кадра файлового подпротокола) CRC16 = 0x2D2E Flags, набор признаков: QA=1 – «мастер кадр»; START=1 – первый кадр; END=0 – НЕ последний кадр.
Дейтаграмма (начало дейтаграммы, окончание отправится в следующем кадре)	07 00 37 00 00 00 00 00 00 00 00 00 00 FF FF FF FF FF FF FF 24 00 2F 75 73 72 2F 70 69 72 61 6D 69 64 61 2F 73 74 6F 72 65 2F 61 72 68 5F 32 30 31 33 30 31 32 33 2E 78 6D 6C 00 00 00 00 00 00 00 00	DatagrammType = 0x07 – Запрос. Чтение файла из SM160. Reserved. DatagrammOptionsLength = 55 байт. DatagrammOptions.Offset = 0. DatagrammOptions.FileSize = весь файл. DatagrammOptions.NameLen = 36 bytes DatagrammOptions.NameStr = “/usr/piramida/store/arch_20121212.xml”+0 DatagrammDataLength = 0 (частично в этом кадре, остальная часть в следующем кадре).

5) «Modbus Next», пакет о готовности принять следующий кадр.

Трафик ЭВМ←SM160: 00 02 00 00 00 04 FF 45 00 00

Уровень	Трафик	Комментарий
Modbus	00 02 00 00 00 04 FF 45 00 00	Id транзакции = 2 Id протокола Длина Modbus пакета, 4 байта. Адрес устройства, SM160. Функция 0x45, подпротокол передачи файлов. Пакет «Modbus Next».

6) Второй кадр запроса.

Трафик PC→SM160: 00 03 00 00 00 13 FF 45 02 00 44 00 00 00 05 00  
18 2D 05 00 00 A3 56 58 77

Уровень	Трафик	Комментарий
Modbus	00 03 00 00 00 13 FF 45	Id транзакции =3 Id протокола Длина Modbus пакета, 4 байта. Адрес устройства, SM160. Функция 0x45, подпротокол передачи файлов.
Мастер кадр	02 00	SesionID = 2
	44 00 00 00	DatagrammOffset = 68
	05 00	DatagrammPartSize = 5 байт
	18 2D	CRC16 = 0x2D18
	05 00	Flags, набор признаков: QA=1 – «мастер кадр»; START=0 – НЕ первый кадр; END=1 – последний кадр.
Дейтаграмма	00	DatagrammDataLength = 0 (частично в этом кадре, начало было в предыдущем кадре).
(окончание дейтаграммы)	A3 56 58 77	CRC32

7) Ответ на запрос чтения файла.

В этом случае, сначала из отдельных кадров собирается дейтаграмма ответа, затем она расшифровывается. Т.к. фрагменты большие, частичную расшифровку дейтаграмм в кадрах не приводится, а после приема последнего кадра, приводится расшифровка всей дейтаграммы ответа.

8) Первый кадр ответа.

Трафик PC←SM160: 00 03 00 00 00 52 FF 45 02 00 49 00 00 00 44 00  
35 7A 02 00 87 00 09 00 00 00 00 00 00 00 00 00 00 00 00 00 02 00 4F 4B 00 D6 00 3C  
72 20 53 3D 22 32 30 31 33 30 31 32 32 30 30 30 30 35 31 35 32 34  
32 37 39 22 20 4E 3D 22 73 65 6C 66 5C 47 50 52 53 5C 53 74 61 74  
65 5C 44 49 22 20

Уровень	Трафик	Комментарий
Modbus	00 03 00 00 00 52 FF 45	Id транзакции = 3 Id протокола Длина Modbus пакета, 82 байта. Адрес устройства, SM160. Функция 0x45, подпротокол передачи файлов.
Слэйв кадр	02 00	SesionID = 2.
	49 00 00 00	DatagrammOffset = 73 (нумерация продолжается с длины дейтаграммы запроса).
	44 00	DatagrammPartSize = 68 байт.
	35 7A	CRC16 = 0x2D18
	02 00	Flags, набор признаков: QA=0 – «слэйв кадр»; START=1 – первый кадр; END=0 – НЕ последний кадр.
Дейтаграмма	87 00 09 00 00 00 00 00 00 00 00 02 00 4F 4B 00 D6 00 3C 72 20 53 3D 22 32 30 31 33 30 31 32 32 30 30 30 30 35 31 35 32 34 32 37 39 22 20 4E 3D 22 73 65 6C 66 5C 47 50 52 53 5C 53 74 61 74 65 5C 44 49 22 20	



9) «Modbus Next», пакет о готовности принять следующий кадр.

Трафик ЭВМ→SM160: 00 04 00 00 00 04 FF 45 00 00

Уровень	Трафик	Комментарий
Modbus	00 04 00 00 00 04 FF 45 00 00	Id транзакции = 4 Id протокола Длина Modbus пакета, 4 байта. Адрес устройства, SM160. Функция 0x45, подпротокол передачи файлов. Пакет «Modbus Next».

10) Второй кадр ответа.

Трафик ЭВМ←SM160: 00 04 00 00 00 52 FF 45 02 00 8D 00 00 00 44  
00 69 D3 00 00 56 3D 22 30 22 20 45 3D 22 34 31 36 22 20 2F 3E 0A  
3C 72 20 53 3D 22 32 30 31 33 30 31 32 32 30 30 30 30 35 36 35 38  
31 37 31 36 22 20 4E 3D 22 73 65 6C 66 5C 47 50 52 53 5C 54 72 61  
66 66 69 63 5C 52 58

Уровень	Трафик	Комментарий
Modbus	00 04 00 00 00 52 FF 45	Id транзакции = 4 Id протокола Длина Modbus пакета, 82 байта. Адрес устройства, SM160. Функция 0x45, подпротокол передачи файлов.
Слэйв кадр	02 00	SesionID = 2.
	8D 00 00 00	DatagrammOffset = 141.
	44 00	DatagrammPartSize = 68 байт.
	69 D3	CRC16 = 0xD369
	00 00	Flags, набор признаков: QA=0 – «слэйв кадр»; START=0 – НЕ первый кадр; END=0 – НЕ последний кадр.
Дейтаграмма	56 3D 22 30 22 20 45 3D 22 34 31 36 22 20 2F 3E 0A 3C 72 20 53 3D 22 32 30 31 33 30 31 32 32 30 30 30 30 35 36 35 38 31 37 31 36 22 20 4E 3D 22 73 65 6C 66 5C 47 50 52 53 5C 54 72 61 66 66 69 63 5C 52 58	

11) «Modbus Next», пакет о готовности принять следующий кадр.

Трафик ЭВМ→SM160: 00 05 00 00 00 04 FF 45 00 00

Уровень	Трафик	Комментарий
Modbus	00 05 00 00 00 04 FF 45 00 00	Id транзакции = 5 Id протокола Длина Modbus пакета, 4 байта. Адрес устройства, SM160. Функция 0x45, подпротокол передачи файлов. Пакет «Modbus Next».

12) Третий кадр ответа.

Трафик ЭВМ←SM160: 00 05 00 00 00 52 FF 45 02 00 D1 00 00 00 44  
00 9C BD 00 00 5C 41 49 22 20 56 3D 22 30 22 20 45 3D 22 34 31 36  
22 20 2F 3E 0A 3C 72 20 53 3D 22 32 30 31 33 30 31 32 32 30 30 30  
30 35 36 35 38 31 38 33 33 22 20 4E 3D 22 73 65 6C 66 5C 47 50 52  
53 5C 54 72 61 66 66

Уровень	Трафик	Комментарий
Modbus	00 05 00 00 00 52 FF 45	Id транзакции = 5 Id протокола Длина Modbus пакета, 82 байта. Адрес устройства, SM160. Функция 0x45, подпротокол передачи файлов.
Слэйв кадр	02 00	SesionID = 2.
	D1 00 00 00	DatagrammOffset = 209.
	44 00	DatagrammPartSize = 68 байт.
	9C BD	CRC16 = 0xBD9C
	00 00	Flags, набор признаков: QA=0 – «слэйв кадр»; START=0 – НЕ первый кадр; END=0 – НЕ последний кадр.
Дейтаграмма	5C 41 49 22 20 56 3D 22 30 22 20 45 3D 22 34 31 36 22 20 2F 3E 0A 3C 72 20 53 3D 22 32 30 31 33 30 31 32 32 30 30 30 30 35 36 35 38 31 38 33 33 22 20 4E 3D 22 73 65 6C 66 5C 47 50 52 53 5C 54 72 61 66 66	

13) «Modbus Next», пакет о готовности принять следующий кадр.

Трафик ЭВМ→SM160: 00 06 00 00 00 04 FF 45 00 00

Уровень	Трафик	Комментарий
Modbus	00 06 00 00 00 04 FF 45 00 00	Id транзакции = 6 Id протокола Длина Modbus пакета, 4 байта. Адрес устройства, SM160. Функция 0x45, подпротокол передачи файлов. Пакет «Modbus Next».

14) Четвертый кадр ответа, последний.

Трафик ЭВМ←SM160: 00 06 00 00 00 00 37 FF 45 02 00 15 01 00 00 29  
00 60 55 04 00 69 63 5C 54 58 5C 41 49 22 20 56 3D 22 30 22 20 45  
3D 22 34 31 36 22 20 2F 3E 0A 00 00 DA 00 00 00 00 00 00 01 86  
40 21

Уровень	Трафик	Комментарий
Modbus	00 06 00 00 00 52 FF 45	Id транзакции = 6 Id протокола Длина Modbus пакета, 82 байта. Адрес устройства, SM160. Функция 0x45, подпротокол передачи файлов.
Слэйв кадр	02 00	SesionID = 2.
	15 01 00 00	DatagrammOffset = 277.
	29 00	DatagrammPartSize = 41 байт.
	60 55	CRC16 = 0x5560
	04 00	Flags, набор признаков: QA=0 – «слэйв кадр»; START=0 – НЕ первый кадр; END=1 – последний кадр.
Дейтаграмма	69 63 5C 54 58 5C 41 49 22 20 56 3D 22 30 22 20 45 3D 22 34 31 36 22 20 2F 3E 0A 00 00 DA 00 00 00 00 00 00 00 01 86 40 21	

Т.к. пришел последний кадр, запрос и ответ на чтение файла считается завершенным, необходимо собрать все части дейтаграммы воедино и расшифровать.

## 15) Расшифровка дейтаграммы ответа.

Массив	Комментарий
87	Type = 0x87 – Ответ на запрос чтения файла.
00	Reserved
09 00 00 00	DatagramOptionsLength = 9 байт
00 00 00 00	DatagramOptions.StatusCode = 0, «ОК»
02 00	DatagramOptions.StatusMsgLen = 2 байта.
4F 4B 00	Текстовое статусное сообщение, DatagramOptions.StatusMsgStr = “OK”+0
D6 00	Фрагмент файла, FileChunkLength = 214 байт.
3C 72 20 53 3D 22 32 30 31 33 30 31 32 32 30 30 30 30 35 31 35 32 34 32 37 39 22 20 4E 3D 22 73 65 6C 66 5C 47 50 52 53 5C 53 74 61 74 65 5C 44 49 22 20 56 3D 22 30 22 20 45 3D 22 34 31 36 22 20 2F 3E 0A 3C 72 20 53 3D 22 32 30 31 33 30 31 32 32 30 30 30 30 35 36 35 38 31 37 31 36 22 20 4E 3D 22 73 65 6C 66 5C 47 50 52 53 5C 54 72 61 66 66 69 63 5C 52 58 5C 41 49 22 20 56 3D 22 30 22 20 45 3D 22 34 31 36 22 20 2F 3E 0A 3C 72 20 53 3D 22 32 30 31 33 30 31 32 32 30 30 30 30 35 36 35 38 31 38 33 33 22 20 4E 3D 22 73 65 6C 66 5C 47 50 52 53 5C 54 72 61 66 66 69 63 5C 54 58 5C 41 49 22 20 56 3D 22 30 22 20 45 3D 22 34 31 36 22 20 2F 3E 0A	Содержимое фрагмента файла, FileChunkData = «<r S="20130122000051524279" N="self\GPRS\State\DI" » «V="0" E="416" /> <r S="20130122000056581716" N="self\GPRS\Traffic\RX»  «\AI" V="0" E="416" /> <r S="20130122000056581833" N="self\GPRS\Traff»  «ic\TX\AI" V="0" E="416" />»
00 00	Фрагмент файла длиной ноль, FileChunkLength = 0 байт, означает, что все фрагменты файла переданы.
DA 00 00 00 00 00 00 00	DatagramDataLength = 218 байт. Длина дейтаграммы.
01 86 40 21	DatagramCRC32

### 3. Форматы данных

#### 3.1. Карта регистров

Ниже приведен перечень регистров доступных для стандартных функций чтения и записи протоколов Modbus TCP и Modbus RTU:

Адреса	Кол-во регистров	Доступ	Параметр	Тип данных
0x0000 – 0x0003	4	чтение / запись	Текущая дата и время контроллера.	TimeStamp
0x0004 – 0x0007	4	чтение/ запись	Текущая дата и время контроллера без проверки на допустимые значения. Используется для первоначальной установки даты и времени.	TimeStamp
0x0008 – 0x0009	2	чтение / запись	Плавная коррекция времени.	TimeCorrection
0x000A	1	чтение	Телесигнализация (ТС), текущее состояние сухих контактов.	TSbit

Типы данных:

№	Тип данных	Описание
1.	TimeStamp	64-разрядное целое число со знаком. Содержит количество миллисекунд, прошедшее с полуночи (00:00:00 UTC) 1 января 1970 года. <b>Внимание!</b> Разрешены операции чтения и записи только всех регистров сразу. Обращение к части регистров или за пределы регистров считается ошибкой адреса (02 – Неверный адрес регистра).
2.	TimeCorrection	32-разрядное целое число со знаком. Содержит количество микросекунд, на которое необходимо скорректировать текущую дату и время. Допустимый диапазон значений +/- 2100000000 мкс (+/- 35 мин). Запись времени отменяет плавную коррекцию. <b>Внимание!</b> Разрешены операции чтения и записи только всех регистров сразу. Обращение к части регистров или за пределы регистров считается ошибкой адреса (02 – Неверный адрес регистра).
3.	TSbit	16 бит, текущее состояние сухих контактов. Кодировка разрядов регистра: бит №0 соответствует TC№1, №1 – TC№2 и т.д, биты с №6 по №15 не используются, их значения должны игнорироваться. Значение бита равное 1 соответствует состоянию «замкнуто», значение 0 – состоянию «разомкнуто».

### 3.1.1. Чтение и запись даты и времени

Пример трафика чтения регистров текущей даты и времени контроллера по протоколу Modbus/TCP.

Трафик ЭВМ→SM160: 00\_02\_00\_00\_00\_06\_00\_03\_00\_00\_00\_04, где

Уровень	Трафик	Комментарий
Modbus	00 02 00 00 00 06 00 03 00 00 00 04	Id транзакции = 2 Id протокола Длина Modbus пакета, 6 байт Адрес устройства, SM160 Функция 0x03, чтение регистров Адрес первого регистра группы времени-даты Количество регистров на чтение

Трафик ЭВМ←SM160:

00\_04\_00\_00\_00\_0B\_00\_03\_08\_00\_00\_01\_42\_E1\_E0\_3F\_06, где

Уровень	Трафик	Комментарий
Modbus	00 02 00 00 00 0B 00 03 08 00 00 01 42 E1 E0 3F 06	Id транзакции = 2 Id протокола Длина Modbus пакета, 11 байтов Адрес устройства, SM160 Функция 0x03, чтение регистров Количество байтов данных Данные, текущее время контроллера, формат указан в пункте 3.1. Число 00000142E1E03F06h, в десятичном виде равно 1 386 769 039 110, является количеством миллисекунд, прошедших с полуночи (00:00:00 UTC) 1 января 1970 года. Дата 11.12.2013 Время 13:37:19.110

Пример трафика удачной записи регистров текущей даты и времени контроллера по протоколу Modbus/TCP.

Трафик ЭВМ→SM160:

00\_03\_00\_00\_00\_0F\_00\_10\_00\_00\_00\_04\_08\_00\_00\_01\_42\_E1\_AD\_1E\_B4 , где

Уровень	Трафик	Комментарий
Modbus	00 03 00 00 00 0F 00 10 00 00 00 04 08 00 00 01 42 E1 AD 1E B4	Id транзакции = 3 Id протокола Длина Modbus пакета, 15 байт Адрес устройства, SM160 Функция 0x10, запись регистров Адрес первого регистра группы времени-даты Количество регистров на запись Количество байтов данных на запись Данные на запись, формат указан в пункте 3.1. Число 00000142E1AD1EB4h, в десятичном виде равно 1 386 765 688 500, является количеством миллисекунд, прошедших с полуночи (00:00:00 UTC) 1 января 1970 года. Дата 11.12.2013 Время 12:41:28.500

Трафик ЭВМ←SM160:

00\_03\_00\_00\_00\_06\_00\_10\_00\_00\_00\_04 , где

Уровень	Трафик	Комментарий
Modbus	00 03 00 00 00 06 00 10 00 00 00 04	Id транзакции = 3 Id протокола Длина Modbus пакета, 6 байт Адрес устройства, SM160 Функция 0x10, запись регистров Адрес первого регистра группы времени-даты Количество успешно записанных регистров

Пример трафика неудачной записи регистров текущей даты и времени контроллера по протоколу Modbus/TCP.

Трафик ЭВМ→SM160:

00\_07\_00\_00\_00\_0F\_00\_10\_00\_04\_00\_04\_08\_00\_00\_02\_52\_BC\_D7\_FF\_2F, где

Уровень	Трафик	Комментарий
Modbus	00 03 00 00 00 0F 00 10 00 00 00 04 08 00 00 02 52 BC D7 FF 2F	Id транзакции = 3 Id протокола Длина Modbus пакета, 15 байт Адрес устройства, SM160 Функция 0x10, запись регистров Адрес первого регистра группы времени-даты Количество регистров на запись Количество байтов данных на запись Данные на запись, формат указан в пункте 3.1. Число 00000252BCD7FF2Fh, в десятичном виде равно 2 554 378 845 999, является количеством миллисекунд, прошедших с полуночи (00:00:00 UTC) 1 января 1970 года. Дата 11.12.2050 Время 13:40:45.999

Трафик ЭВМ←SM160:

00\_03\_00\_00\_00\_03\_00\_90\_03, где

Уровень	Трафик	Комментарий
Modbus	00 03 00 00 00 03 00 90 03	Id транзакции = 3 Id протокола Длина Modbus пакета, 3 байта Адрес устройства, SM160 При ошибке к функциональному коду Modbus добавляется 0x80 (0x10 + 0x80 = 0x90) Код ошибки «неверное значение данных»



### 3.1.2. Запись значения плавной коррекции времени

Пример трафика удачной записи плавной коррекции времени контроллера по протоколу Modbus/TCP.

Трафик ЭВМ→SM160:

00\_1B\_00\_00\_00\_0B\_00\_10\_00\_08\_00\_02\_04\_FB\_09\_18\_70, где

Уровень	Трафик	Комментарий
Modbus	00 1B 00 00 00 0B 00 10 00 08 00 02 04 FB 09 18 70	Id транзакции = 27 Id протокола Длина Modbus пакета, 11 байт Адрес устройства, SM160 Функция 0x10, запись регистров Адрес первого регистра группы коррекции Количество регистров на запись Количество байтов данных на запись Данные на запись, формат указан в пункте 3.1. Коррекция времени назад, число FB091870h в десятичном виде является отрицательным, равно –83290000 микросекунд,

Трафик ЭВМ←SM160:

00\_1B\_00\_00\_00\_06\_00\_10\_00\_08\_00\_02, где

Уровень	Трафик	Комментарий
Modbus	00 1B 00 00 00 06 00 10 00 08 00 02	Id транзакции = 27 Id протокола Длина Modbus пакета, 6 байт Адрес устройства, SM160 Функция 0x10, запись регистров Адрес первого регистра группы коррекции Количество успешно записанных регистров

Пример трафика неудачной записи плавной коррекции времени контроллера по протоколу Modbus/TCP.

Трафик ЭВМ→SM160:

00\_1C\_00\_00\_00\_0B\_00\_10\_00\_08\_00\_02\_04\_7D\_2B\_75\_32, где

Уровень	Трафик	Комментарий
Modbus	00 1C 00 00 00 0B 00 10 00 08 00 02 04 7D 2B 75 32	Id транзакции = 28 Id протокола Длина Modbus пакета, 11 байт Адрес устройства, SM160 Функция 0x10, запись регистров Адрес первого регистра группы коррекции Количество регистров на запись Количество байтов данных на запись Данные на запись, формат указан в пункте 3.1. Коррекция времени вперед, число 7D2B7532h, в десятичном виде является положительным числом, равно 2100000050 микросекунд.

Трафик ЭВМ←SM160:

00\_03\_00\_00\_00\_03\_00\_90\_03, где

Уровень	Трафик	Комментарий
Modbus	00 1C 00 00 00 03 00 90 03	Id транзакции = 3 Id протокола Длина Modbus пакета, 3 байта Адрес устройства, SM160 При ошибке к функциональному коду Modbus добавляется 0x80 (0x10 + 0x80 = 0x90) Код ошибки «неверное значение данных»

## 3.2. Форматы файлов

### 3.2.1. Общий формат

Данные об энергопотреблении и событиях располагаются в файлах отчетов.

Файл представляет собой текстовый XML-подобный документ в кодировке ANSI. Отличием от стандарта XML 1.0 является отсутствие корневого тэга. Для разбора файлов существующими средствами работы с XML (например, MS XML) рекомендуется производить подстановку открывающего корневого тэга <body> в начале документа и закрывающего корневого тэга </body> в конце документа.

Каждый документ содержит строки, где одной сущности соответствует одна строка файла, оканчивающаяся символом перевода строки.

### 3.2.2. Формат файла списка устройств («serial.xml»)

Файл «serial.xml» содержит описание списка приборов, подключенных к ИК SM160. Каждому прибору соответствует одна строка (запись) файла.

Общий вид записи файла:

```
<r D="Префикс данных" K="Тип прибора" S="Серийный номер" />
```

Назначение полей записи файла:

- «r» – имя тэга (одинаковое для всех записей файла);
- «D» – префикс в имени параметров, в архиве, для данного прибора;
- «K» – тип прибора;
- «S» – серийный (заводской) номер прибора.

Пара значений - «K» и «S» - однозначно идентифицирует любой прибор, подключенный к ИК SM160, в том числе и сам контроллер.

Пример файла «serial.xml»:

```
<r D="dev$1" K="24" S="310879" />
```

```
<r D="dev$2" K="101" S="5947911" />
```

```
<r D="self" K="128" S="101" />
```

### 3.2.3. Формат файлов архивов (arh\_\*.xml)

Имена архивных файлов всегда начинаются с префикса «arh\_» и имеют расширение «.xml». После префикса в имени файла следует 8 цифровых символов вида ГГГГММДД, определяющих сутки архива, где:

- «ГГГГ» – номер года по календарю (например, 2013);
- «ММ» – номер месяца (в диапазоне 01...12);
- «ДД» – номер дня в месяце (в диапазоне 01...31).

На каждые сутки создается свой файл архива.

Каждый файл архива содержит строки (записи), где одному измерению или событию соответствует одна запись файла.

Общий вид записи файла:

```
<r [R="Штамп времени"] S="Штамп времени" N="Имя параметра" V="Число" E="Число"
[C="Целое число"] [T="Целое число"] />.
```

В квадратных скобках [...] указаны необязательные атрибуты, которые могут отсутствовать в конкретных записях.

Назначение полей записи файла:

- «r» – имя тэга (одинаковое для всех записей файла);
- «R» – дополнительный штамп времени (необязательное поле);
- «S» – штамп времени;
- «N» – имя параметра, состоящее из «префикса» и «наименования параметра»;

- «V» – значение измерения;
- «E» – дополнительное значение измерения;
- «C» – код события (необязательное поле);
- «T» – код статуса (необязательное поле).

Значения отсутствующих полей «C» и «T» считаются равными нулю. Если отсутствует поле «R», его значение совпадает со значением поля «S».

Значениями полей «R» и «S» является штамп времени в формате ГГГГММДДччммссxxxxxx, где:

- «ГГГГ» – номер года по календарю (например 2013);
- «ММ» – номер месяца (в диапазоне 01...12);
- «ДД» – номер дня в месяце (в диапазоне 01...31);
- «чч» – номер часа (в диапазоне 00...24);
- «мм» – номер минуты (в диапазоне 00..59);
- «сс» – номер секунды (в диапазоне 00..59);
- «xxxxxx» – номер микросекунды в секунде (в диапазоне 000000..999999).

Значениями атрибутов E и V является дробное десятичное число со знаком, например, «42», «-23.456». Целая и дробная части разделяются символом «.» (точка).

Атрибуты C и T могут принимать только целочисленные значения со знаком.

Пример файла «arh\_\*.xml»:

```
<r S="20130111000000000000" N="dev$1\r$0\fix\import\a\e\AI" V="931.3975" E="931.3975" T="0" />
<r S="20130111000000000000" N="dev$1\r$0\fix\export\a\e\AI" V="0.153" E="0.153" T="0" />
<r S="20130111000000000000" N="dev$1\r$0\fix\import\r\e\AI" V="144.045" E="144.045" T="0" />
<r S="20130111000000000000" N="dev$1\r$0\fix\export\r\e\AI" V="1336.7315" E="1336.7315" T="0" />
<r S="20130111000000000000" N="dev$1\r$1\fix\import\a\e\AI" V="618.277" E="618.277" T="0" />
<r S="20130111000000000000" N="dev$1\r$1\fix\export\a\e\AI" V="0.1515" E="0.1515" T="0" />
<r S="20130111000000000000" N="dev$1\r$1\fix\import\r\e\AI" V="89.206" E="89.206" T="0" />
<r S="20130111000000000000" N="dev$1\r$1\fix\export\r\e\AI" V="896.0705" E="896.0705" T="0" />
<r S="20130111000000000000" N="dev$1\r$2\fix\import\a\e\AI" V="313.1205" E="313.1205" T="0" />
<r S="20130111000000000000" N="dev$1\r$2\fix\export\a\e\AI" V="0.0015" E="0.0015" T="0" />
<r S="20130111000000000000" N="dev$1\r$2\fix\import\r\e\AI" V="54.839" E="54.839" T="0" />
<r S="20130111000000000000" N="dev$1\r$2\fix\export\r\e\AI" V="440.661" E="440.661" T="0" />
<r S="20130111000000000000" N="dev$1\r$3\fix\import\a\e\AI" V="0" E="0" T="0" />
<r S="20130111000000000000" N="dev$1\r$3\fix\export\a\e\AI" V="0" E="0" T="0" />
<r S="20130111000000000000" N="dev$1\r$3\fix\import\r\e\AI" V="0" E="0" T="0" />
<r S="20130111000000000000" N="dev$1\r$3\fix\export\r\e\AI" V="0" E="0" T="0" />
<r S="20130111000000000000" N="dev$1\r$4\fix\import\a\e\AI" V="0" E="0" T="0" />
<r S="20130111000000000000" N="dev$1\r$4\fix\export\a\e\AI" V="0" E="0" T="0" />
<r S="20130111000000000000" N="dev$1\r$4\fix\import\r\e\AI" V="0" E="0" T="0" />
<r S="20130111000000000000" N="dev$1\r$4\fix\export\r\e\AI" V="0" E="0" T="0" />
<r S="20130111000000000000" N="dev$1\dt$30\import\a\e\AI" V="0.049" E="0.049" />
```

```
<r S="20130111000000000000" N="dev$1\dt$30\export\a\e\AI" V="0" E="0" />
<r S="20130111000000000000" N="dev$1\dt$30\import\r\e\AI" V="0" E="0" />
<r S="20130111000000000000" N="dev$1\dt$30\export\r\e\AI" V="0.068" E="0.068" />
<r S="20130111000747048391" N="self\GPRS\State\DI" V="0" E="1170" />
<r S="20130111000752107153" N="self\GPRS\Traffic\RX\AI" V="0" E="1170" />
<r S="20130111000752107275" N="self\GPRS\Traffic\TX\AI" V="0" E="1170" />
<r S="20130111000754486338" N="self\GPRS\SimN\AI" V="2" E="1171" />
<r S="20130111000945008383" N="self\GPRS\State\DI" V="0" E="1171" />
<r S="20130111000950067046" N="self\GPRS\Traffic\RX\AI" V="0" E="1171" />
<r S="20130111000950067169" N="self\GPRS\Traffic\TX\AI" V="0" E="1171" />
<r S="20130111000952326238" N="self\GPRS\SimN\AI" V="2" E="1172" />
<r S="20130111001140837884" N="self\GPRS\State\DI" V="0" E="1172" />
<r S="20130111001145896670" N="self\GPRS\Traffic\RX\AI" V="0" E="1172" />
<r S="20130111001145896793" N="self\GPRS\Traffic\TX\AI" V="0" E="1172" />
<r S="20130111001148236238" N="self\GPRS\SimN\AI" V="2" E="1173" />
<r S="20130111001336697910" N="self\GPRS\State\DI" V="0" E="1173" />
<r S="20130111001341756574" N="self\GPRS\Traffic\RX\AI" V="0" E="1173" />
<r S="20130111001341756696" N="self\GPRS\Traffic\TX\AI" V="0" E="1173" />
<r S="20130111001344096244" N="self\GPRS\SimN\AI" V="2" E="1174" />
<r S="20130111001532499796" N="self\GPRS\State\DI" V="0" E="1174" />
<r S="20130111001537557354" N="self\GPRS\Traffic\RX\AI" V="0" E="1174" />
<r S="20130111001537557479" N="self\GPRS\Traffic\TX\AI" V="0" E="1174" />
<r S="20130111001539806221" N="self\GPRS\SimN\AI" V="2" E="1175" />
<r S="20130111001716776356" N="dev$2\EVD" V="1357863436" E="1357863436" C="2006" />
```

### 3.2.4. Кодировка типов приборов

Коды типов приборов используются в файле «serial.xml» для формирования значений поля «К».

Типы приборов:

Тип прибора	Наименование прибора	Про
24	Счётчик электроэнергии Меркурий 230	ООО НПК «Инкотекс»
101	Счётчик электроэнергии Меркурий 203	ООО НПК «Инкотекс»
126	Счётчик электроэнергии Меркурий 200	ООО НПК «Инкотекс»
127	Счётчик электроэнергии Меркурий 233	ООО НПК «Инкотекс»
128	ИК SM160	ЗАО ИТФ «Системы и технологии», ООО Завод «Промприбор»

### 3.3. Имена параметров

Имена параметров используются в файле (arch\_\*.xml) для формирования значений поля «N».

Составляющие части в имени параметра:

- Префикс – внутреннее обозначение прибора в ИК SM160, уникальное для прибора в пределах контроллера, назначается автоматически при подключении. Префикс представляет из себя запись вида «dev\$п», где изменяется только «п» – автоматически присваиваемый номер при подключении прибора к ИК SM160. Для самого ИК SM160 как источника информации существует специальный префикс, который всегда равен «self». Префикс используется также в файле «serial.xml» для формирования значений поля «D».
- Наименование параметра – внутреннее обозначение типа параметра измерения или журнала событий.

Имена параметров:

№	Имя параметра	Единицы измерения	Описание
1	<префикс>\r\$x\total\import\a\e\AI	КВт*ч	Текущие показания нарастающим итогом (total), активная (a) энергия (e) в прямом направлении (import), по тарифу (r) с номером (x). Суммарные показания по всем тарифам, x = 0. Номера тарифов (x), начиная с 1.
2	<префикс>\r\$x\total\export\a\e\AI	КВт*ч	Текущие показания нарастающим итогом (total), активная (a) энергия (e) в обратном направлении (export), по тарифу (r) с номером (x). Кодировка тарифов аналогична строке №1.

Интеллектуальный контроллер SM160 – Протокол обмена

№	Имя параметра	Единицы измерения	Описание
3	<префикс>\r\$\total\import\r\e\AI <префикс>\r\$\total\export\r\e\AI	кВар*ч	Текущие показания нарастающим итогом, реактивная энергия в прямом и обратном направлении. Кодировка тарифов аналогична строке №1.
4	<префикс>\r\$\fix\import\а\е\AI <префикс>\r\$\fix\export\а\е\AI	КВт*ч	Зафиксированное значение (fix) текущих показаний нарастающим итогом, активная энергия в прямом и обратном направлениях. Остальная кодировка аналогична строке №1.  Метка времени определяет момент фиксации показаний в приборе учёта.
5	<префикс>\r\$\fix\import\r\e\AI <префикс>\r\$\fix\export\r\e\AI	кВар*ч	Зафиксированное значение (fix) текущих показаний нарастающим итогом, реактивная энергия в прямом и обратном направлениях. Остальная кодировка аналогична строке №3.

№	Имя параметра	Единицы измерения	Описание
6	<pre>&lt;префикс&gt;\dt\$n\import\a\p\AI</pre> <pre>&lt;префикс&gt;\dt\$n\export\a\p\AI</pre>	кВт	<p>Активная (a) мощность (p) в прямом (import) и обратном (export) направлении, усредненная за период (dt), n</p> <p>Вместо n подставляется период усреднения в минутах из ряда: 1, 2, 3, 5, 6, 10, 15, 30, 60, а так же специальные значения: day, month, year.</p> <p>Метка времени выровнена на конец интервала измерения.</p> <p>Например, мощность усредненная за 30 минут на интервале [31.11.2012 23:30 – 01.12.2012 00:00) будет иметь метку времени 01.12.2012 00:00:00, и все такие мощности содержат метки времени кратные 30 минутам.</p> <p>Для периода day, за период [31.11.2012 00:00 – 01.12.2012 00:00), метка времени будет равна 01.12.2012 00:00:00, и все метки с периодом day кратны дням.</p> <p>Аналогично с периодом month и year, метки времени кратны месяцу и году.</p> <p>Следует заметить, что для мощностей периоды day, month и year не применяются. Такие периоды обычно применяются для энергий.</p>
7	<pre>&lt;префикс&gt;\dt\$n\import\r\p\AI</pre> <pre>&lt;префикс&gt;\dt\$n\export\r\p\AI</pre>	кВар	<p>Реактивная (r) мощность (p) в прямом (import) и обратном (export) направлении, усредненная за период (dt), n, по тарифу x. Кодировка периода n аналогична строке №6.</p>
8	<pre>&lt;префикс&gt;\r\$x\dt\$n\import\a\e\AI</pre> <pre>&lt;префикс&gt;\r\$x\dt\$n\export\a\e\AI</pre>	кВт*ч	<p>Активная (a) энергия (e) в прямом (import) и обратном (export) направлении, за период (dt), n, по тарифу x. Кодировка периода n аналогична строке №6, кодировка тарифов аналогична строке №1.</p>



Интеллектуальный контроллер SM160 – Протокол обмена

№	Имя параметра	Единицы измерения	Описание
9	<префикс>\r\$x\dt\$n\import\r\e\AI <префикс>\r\$x\dt\$n\export\r\e\AI	кВар*ч	Реактивная (а) энергия (е) в прямом (import) и обратном (export) направлении, за период (dt), n, по тарифу x. Кодировка периода n аналогична строке №6, кодировка тарифов аналогична строке №1.
10	<префикс>\l\$n\phase\v\AI	КВольт	Напряжение (phase\v) фазное или линейное (между фазами), номер n. Номер фазы из ряда: 1, 2, 3, 12, 23, 31. 1, 2, 3 – номера фаз: А, В, С соответственно. 12, 23, 31 – межфазное обозначение, АВ, ВС и АС соответственно.
11	<префикс>\l\$n\i\AI	КАмпер	Ток (i) в фазе (l), n. Кодировка номера фазы аналогична строке №10.
12	<префикс>\l\$n\ap\AI	кВт	Активная (а) мощность (р) в фазе (l), n. Кодировка номера фазы аналогична строке №10.
13	<префикс>\l\$n\r\p\AI	кВар	Реактивная (r) мощность (р) в фазе (l), n. Кодировка номера фазы аналогична строке №10.
14	<префикс>\l\$n\appar\p\AI	кВА	Полная (appar) мощность (р) в фазе (l), n. Кодировка номера фазы аналогична строке №10.
15	<префикс>\l\$n\cosfi\AI	Число с плавающей точкой.	Коэффициент мощности (cosfi) в фазе (l), n. Кодировка номера фазы аналогична строке №10.
16	<префикс>\f\AI	Гц	Частота (f).
17	<префикс>\inside\t\AI	Градус Цельсия	Температура (t) внутри (inside) счётчика.
18	<префикс>\EV	-	Результат сбора данных, сгенерированный программой сбора, равен коду канальной ошибки, плюс несколько специальных кодов, например, смена счётчика (изменен серийный номер). Результат помещается в поле «V».
19	<префикс>\EVD	-	Регистратор событий счётчика, полученный с самого счётчика. Код события определяется полем «С».

### 3.3.1. Пример интерпретации значения параметра

Исходные данные:

Запись в файле архива (arch\_\*.xml):

```
<r S="20130111000000000000" N="dev$1\r$2\fix\import\a\e\AI" V="313.1205" E="313.1205" T="0" />
```

Запись в файле серийных номеров (serial.xml)

```
<r D="dev$1" K="24" S="310879" />
```

Интерпретация параметра и его значения:

«\r\$2\fix\import\a\e\AI» – Зафиксированное значение текущих показаний нарастающим итогом (fix), активная энергия (a\e) в прямом направлении (import), по тарифу (r\$2) = 2, единицы измерения – кВт\*ч.

- «20130111000000000000» – момент фиксации 00:00:00 11.01.2013.
- «V="313.1205"» – значение, 313.1202 кВт\*ч.

Идентификация прибора, к которому относится значение по префиксу «dev\$1» в файле «serial.xml»:

- «S="310879"» – с серийным номером 310879;
- «K="24"» – Счётчик электроэнергии Меркурий 230.

### 3.3.2. Кодировка результата сбора, параметр «EV»

Для диагностических целей, по каждому подключенному прибору программой сбора в ИК SM160 генерируется и записывается в архив результат сбора данных.

В архив записываются только изменения, произошедшие с последнего сеанса связи, а не результат каждого сеанса связи с прибором.

Таблица результата сбора:

Код результата	Код результата (hex)	Описание
0	0	Сбор данных прошел успешно
40961	A001h	Неверный пароль
40962	A002h	Нет ответа от счётчика, таймаут
40963	A003h	Ошибка в запросе к счётчику
40964	A004h	Ответ от счётчика с неправильной контрольной суммой
40965	A005h	Слишком большой ответ от счётчика
40966	A006h	Слишком большой запрос к счётчику
40969	A009h	Не поддерживается протокол данного счётчика, сбор невозможен
40972	A00Ch	Изменился серийный номер счётчика, сбор остановлен
40973	A00Dh	Не задан связной адрес счётчика, сбор невозможен
40974	A00Eh	Зафиксирован ответ от счётчика, на другой запрос
40975	A00Fh	Технологическая ошибка, обратитесь к разработчику ИК SM160

### 3.3.3. Кодировка событий, параметр «EVD»

Коды событий используются для маркировки событий, возникающих в приборах и самом ИК SM160. Коды событий используются в файле архивов (arh\_\*.xml), в поле «С» для параметров, имена которых заканчиваются словом «\EVD».

Коды событий:

Код события	Описание
<b>События ИК SM160</b>	
1008	Коррекция времени. Поле «V» содержит величину коррекции со знаком: целая и дробная часть секунд, как разницу: «новое время» – «старое время».
1009	Установка времени. Поле «S» содержит время, которое было установлено. Поле «V» содержит величину коррекции со знаком: целая и дробная часть секунд, как разницу: «новое время» – «старое время».
<b>События в приборах, подключенных к ИК SM160</b>	
2001	Включение прибора
2002	Выключение прибора
2003	Перезагрузка прибора
2004	Изменение конфигурации прибора
2005	Сброс журнала событий прибора
2006	Установка времени прибора. Поле «S» содержит время, которое было установлено. Поле «V» содержит величину коррекции со знаком: целая и дробная часть секунд, как разницу: «новое время» – «старое время».
2007	Коррекция времени прибору. Поле «V» содержит величину коррекции со знаком: целая и дробная часть секунд, как разницу: «новое время» – «старое время».