

УТВЕРЖДАЮ

АО « ПСК Миландр »

« _____ » _____ 20__ г

**Протокол взаимодействия с управляющим терминалом
для изделия Милур 304 / 305 / 306 / 307
(с изменениями 3.18)**

2016

Изделие «Милур 305 / 306 / 307»
Протокол взаимодействия с управляющим терминалом.
(С изменениями 304-307.3.18)

1. ОБЩИЕ ПОЛОЖЕНИЯ.

Протокол взаимодействия (в дальнейшем «протокол») основан на уровневой модели в соответствии с рекомендациями OSI.

2. КАНАЛЬНЫЙ УРОВЕНЬ

Канальный уровень соответствует спецификациям протокола MODBUS.

3. ПРИКЛАДНОЙ УРОВЕНЬ

3.1. Общие положения.

Прикладной протокол использует основные подходы протокола MODBUS. Протокол предполагает одно активное (запрашивающее) устройство в линии (master), которое может обращаться к нескольким пассивным устройствам (slave), обращаясь к ним по уникальному в линии адресу. Синтаксис команд протокола позволяет адресовать 254 устройства.

Инициатива проведения обмена всегда исходит от ведущего устройства. Ведомые устройства прослушивают линию связи. Мастер подает запрос (посылка, последовательность байт) в линию и переходит в состояние прослушивания линии связи. Ведомое устройство отвечает на запрос, пришедший в его адрес.

3.1.1. RTU фрейм

В RTU режиме сообщение должно начинаться и заканчиваться интервалом тишины — временем передачи не менее 3,5 символов при данной скорости в сети. Первым полем затем передаётся адрес устройства.

Вслед за последним передаваемым символом также следует интервал тишины продолжительностью не менее 3,5 символов. Новое сообщение может начинаться после этого интервала.

Фрейм сообщения передаётся непрерывно. Если интервал тишины продолжительностью 1,5 возник во время передачи фрейма, принимающее устройство должно игнорировать этот фрейм как неполный.

Таким образом, новое сообщение должно начинаться не раньше 3,5 интервала, так как в этом случае устанавливается ошибка.

При скорости 9600 и 10 битах в кадре (стартовый бит + 8 бит данных + стоп-бит): $3.5 * 10 / 9600 = 0,00365$ с; $1.5 * 10 / 9600 = 0,0015625$ с, то есть более 1 мс. Для скоростей более 19200 бод допускается использовать интервалы 1,75 и 0,75 мс соответственно.

3.1.2. Разграничение доступа к объектам.

Для обеспечения безопасности предусматривается механизм разграничения доступа к интерфейсным объектам. Объектами разграничения доступа являются классы, объекты классов, методы и атрибуты объектов.

Используется следующий механизм разграничения доступа. Вводится понятия уровня клиента, права уровня клиента.

Уровень клиента это параметр клиента, определяющий права доступа клиента.

Определяются клиенты 3-х уровней: обычный, административный и уровень разработчика.

Для объектов определяются права на чтение и полный доступ.

Для атрибутов определяются права на чтение и полный доступ.

Для методов определяются права на выполнение.

Используется наследование прав доступа.

Если право доступа определено для некоторого уровня клиента, то оно определено и для вышестоящих уровней клиента.

Для записи уровней клиента введены следующие обозначения:

CL_ORD - обычный;
 CL_ADM- административный;
 CL_DEV - уровень разработчика.

Для обозначения прав доступа для разных уровней клиента введена следующая нотация:

CL_ADM/CL_ORD- означает, что клиенту с уровнем CL_ORD разрешено только чтение атрибута, клиенту с уровнем CL_ADM, CL_DEV разрешены запись и чтение атрибута.

CL_DEV/CL_ORD- означает, что клиенту с уровнем CL_ORD и CL_ADM разрешено только чтение атрибута, клиенту с уровнем CL_DEV разрешены запись и чтение атрибута.

CL_ADM/CL_ADM- означает, что клиенту с уровнем CL_ORD доступ запрещён, клиенту с уровнем CL_ADM, CL_DEV разрешены запись атрибута и чтение атрибута.

3.2. Интерфейсные классы

3.2.1. При описании интерфейсных классов будут использоваться обозначения predefined типов, числовые типы по умолчанию предполагаются беззнаковыми:

BYTE- байт

WORD- 2-х байтовое число

3.2.2. При описании методов будет использоваться «C»- подобная нотация. При этом указывается возвращаемое значение. Возвращаемое значение доставляется клиенту сервисом ARESPONSE.

3.2.3. Перечень интерфейсных классов

Обозначение	УИД	Наименование	Примечание
DATA	1	Данные	Данные общего вида

3.2.4. Класс Data

3.2.4.1. В класс **Data** входят данные объектов устройства. Этот класс порождён от базового класса. Тип данных определяется атрибутом value_type. Этот класс может использоваться для хранения определяемых производителем конфигурационных данных и параметров, а также результатов измерения.

<i>Имя класса</i>	<i>Объём</i>	<i>Ид</i>	<i>Версия</i>		<i>Права доступа</i>
Data	0..n	1	0		
<i>Атрибуты</i>	<i>Тип</i>	<i>Мин</i>	<i>Макс</i>	<i>Знач.</i>	
1. value_type (static)	BYTE				-/CL_ORD
2. value	Instance specific				CL_ORD/CL_ORD
3. value_len	BYTE	0	255		-/CL_ORD
<i>Методы</i>	<i>m/o</i>				
1. get	M				CL_ORD
2. set	M				CL_ORD
3. getbyte	M				CL_ORD
4. setbyte	M				CL_ORD
5. listinit	M				CL_ORD
6. getlistne	M				CL_ORD
7. getlistrecpwi	M				CL_ORD
8. aopen	M				CL_ORD
9. arelease	M				CL_ORD
10. setrtc	M				CL_ADM
11. get_evtlist	M				CL_ORD
12. get_entalist	M				CL_ORD

13. getlistrecpwi_par	M					CL_ORD
14. pwilist_search	M					CL_ORD
15. getcurindex	M					CL_ORD
16. list_search	M					CL_ORD
17. get_collection	M					CL_ORD

3.2.4.2. Атрибуты

- value_type - тип данных (UID атрибута равен 1).

Параметр имеет следующий формат (символами * отмечены неиспользуемые биты)

Тип данных	7	6	5	4	3	2	1	0
Битовый	0	0	0	*	*	Номер бита		
Целочисленный	0	0	1	*	*	Длина в байтах - 1		
Вещественный с фиксированной запятой	0	1	1	Признак Знакового	Масштаб (степень 10)		Длина в байтах - 1	
Статический байтовый массив	1	0	0	Длина в байтах - 1				

- value_len- длина данного в байтах. Этот атрибут является статическим.

3.2.4.3. Методы

- **GET (BYTE ObjUID)-** получить значение объекта,
ObjUID – UID объекта
Возвращает значение объекта
- **SET(BYTE ObjUID, BYTE DataB0, BYTE DataB1...)-** Установить значение объекта,
ObjUID – UID объекта
DataB0... DataBn - байт значения объекта, начиная с младшего
- **GETBYTE(BYTE ObjUID, BYTE Ind)-** Получить байт объекта,
ObjUID – UID атрибута
Ind- индекс байта объекта, принимает значение [0 .. Длина_объекта-1]
Возвращает байт объекта.
- **SETBYTE (BYTE ObjUID, BYTE Ind, BYTE DataB)-** Установить байт объекта,
ObjUID – UID объекта;
Ind- индекс байта объекта, принимает значение [0 .. Длина_объекта-1]
DataB- байт значения объекта.
- **LISTINIT (BYTE ObjUID)-** Инициализация списка
ObjUID – UID объекта;
- **GETLISTNE (BYTE ObjUID)-** Получить число элементов в списке
ObjUID – UID объекта;
Метод возвращает число элементов списка, 2 байта, младший байт вперёд.
- **GETLISTRECPWI (BYTE ObjUID, WORD Index)-** получить элемент списка профиля мощности, где:
ObjUID – UID объекта;
Ind- двухбайтовый индекс элемента списка, принимает значение [0...Длина_списка-1],
[0]- первый элемент списка, [1]- следующий элемент списка....[Длина списка-1]- текущий элемент списка. Младший байт вперёд.
Метод возвращает структуру данных. Младший байт вперёд.
Более подробно возвращаемая структура данные описана в п.п. 4.1.12

Значения энергий, накопленных за интервал передаются в упакованном BCD формате. Младший байт вперёд.

- **AOPEN (BYTE GrCode, BYTE Psw[6]) – открытие сеанса связи.**

GrCode – код группы доступа:

- 0 – уровень пользователя;
- 1 – уровень администратора;
- 2 – уровень разработчика;

Psw – пароль;

- **ARELEASE () – закрытие сеанса связи.**

- **SETRTC (BYTE ObjUID, BYTE Seconds, BYTE Minutes, BYTE Hours, BYTE DayInWeek, BYTE Day, BYTE Month, BYTE Year)- установить часы реального времени,**

ObjUID – UID объекта;

Seconds- значение секунд, принимает значение [0 .. 59],

Minutes- значение минут, принимает значение [0 .. 59],

Hours- значение часов, принимает значение [0 .. 23],

DayInWeek- день недели, принимает значение [1 .. 7], 1- Воскресенье

Day- день месяца, принимает значение [1 .. 31],

Month- значение месяца, принимает значение [1 .. 12],

Year- год, принимает значение [00 .. 99].

Если актуален переход на сезонное (летнее/зимнее) время, то при установке времени необходимо обратить внимание на то, чтобы перед установкой времени, был установлен флаг перехода на летнее/зимнее время.

При установке времени в счётчик, анализируется флаг перехода на летнее/зимнее время. В часы контроллера должно быть установлено поясное время. Если флаг сброшен, в часы контроллера записывается время, которое было передано от верхнего уровня с командой установки времени. Если флаг установлен, счётчик анализирует месяц, день и часы в переданных данных от верхнего уровня. Если счётчик определил, что устанавливаемое время зимнее, коррекция часов не производится и к контроллер записывается переданное время. Если счётчик определил, что устанавливаемое время летнее, в часы контроллера записывается поясное время на час меньше, чем было передано от верхнего уровня. См описание объекта «Управление автоматическим переходом на летнее/зимнее время», п.п. 4.1.18.

- **GET_EVTLIST (BYTE ObjUID, WORD Index)- получить элемент списка событий.**

ObjUID – UID объекта (объекты «errors», «messages», «warnings»);

Index- двухбайтовый индекс элемента списка, принимает значение [0...Длина_списка-1].

[0]- первый элемент списка, [1]- следующий элемент списка... [Длина списка-1]- последний элемент списка. Младший байт вперёд.

Метод возвращает структуру данных. Младший байт вперёд.

struct EvtRecord_t

```
{
    BYTE seconds;      // Секунды
    BYTE minute;      // Минута
    BYTE hour;         // Час
    BYTE day;          // День
    BYTE month;        // Месяц
    BYTE year;         // Год
    WORD events;       // Элемент списка событий
}
```

- **GET_ENTALIST (BYTE ObjUID, WORD Index)- получить элемент списка энергий в интервалах.**

ObjUID – УИД объекта (объекты «энергия в месячных интервалах», «энергия в суточных интервалах»);

Index- двухбайтовый индекс элемента списка, принимает значение [0...Длина_списка-1]. [0]- первый элемент списка, [1]- следующий элемент списка... [Длина списка-1]- последний элемент списка. Младший байт вперёд.

Метод возвращает структуру данных. Младший байт вперёд.

Подробно возвращаемая структура данные описана в п.п. 4.1.26

- **GETLISTRECPWI_PAR (BYTE ObjUID, WORD Index)- получить элемент списка профиля мощности с признаком неполного интервала,**

ObjUID – УИД объекта;

Ind- двухбайтовый индекс элемента списка, принимает значение [0...Длина_списка-1],

[0]- первый элемент списка, [1]- следующий элемент списка... [Длина списка-1]- текущий элемент списка. Младший байт вперёд.

Метод возвращает структуру данных. Младший байт вперёд.

Подробно возвращаемая структура данные описана в п.п. 4.1.12

Значения энергий, накопленных за интервал передаются в упакованном BCD формате. Младший байт вперёд.

- **PWILIST_SEARCH (BYTE ObjUID, BYTE minutes, BYTE hour, BYTE mday, BYTE month, BYTE year)- найти запись в списке профиля мощности по заданным дате и времени записи,**

- ObjUID – УИД объекта;

- minutes- значение минут в записи списка,

- hour – значение часов в записи списка,

- mday – значение дня месяца в записи списка,

- month – значение месяца в записи списка,

- year – значение года в записи списка.

Метод возвращает структуру данных. Младший байт вперёд.

Возвращаемая структура данных описана в п.п. 4.1.30.

- **GETCURINDEX (BYTE ObjUID)- получить индекс последнего заполненного элемента списка.**

ObjUID – УИД объекта (объекты «Энергия в месячных интервалах», «Энергия в суточных интервалах», «Срезы мощности», «Списки событий»);

Метод возвращает двухбайтовый индекс последнего заполненного элемента списка. Младший байт вперёд.

- **LIST_SEARCH (BYTE ObjUID, BYTE start_minutes, BYTE start_hour, BYTE start_mday, BYTE start_month, BYTE start_year, BYTE end_minutes, BYTE end_hour, BYTE end_mday, BYTE end_month, BYTE end_year) - найти запись в списке по заданным начальным и конечным дате и времени,**

ObjUID – УИД объекта (только для списков);

start_minutes- начальное значение минут диапазона поиска,

start_hour – начальное значение часов диапазона поиска,

start_mday – начальное значение дня месяца диапазона поиска,

start_month – начальное значение месяца диапазона поиска,

start_year – начальное значение года диапазона поиска,

end_minutes- конечное значение минут диапазона поиска,

end_hour – конечное значение часов диапазона поиска,

end_mday – конечное значение дня месяца диапазона поиска,

end_month – конечное значение месяца диапазона поиска,
end_year – конечное значение года диапазона поиска.

Метод возвращает структуру данных. Младший байт вперёд.

```
typedef struct {  
    BYTE search_result;    // Состояние поиска  
    BYTE start_index[2];  // Начальный индекс диапазона найденных записей, 2 байта  
    BYTE end_index[2];    // Начальный индекс диапазона найденных записей, 2 байта  
} ListSearch_t;
```

Более подробно метод и особенности его использования описаны в п.п. 4.1.35.

- **GET_COLLECTION (BYTE CollectionUID, WORD Index)** – получить коллекцию данных.
CollectionUID – UID коллекции (1 байт);
Index – индекс пакета данных в коллекции (2 байта);

Метод возвращает структуру данных, различающуюся в зависимости от параметров метода. Младший байт вперёд. Более подробно метод и возвращаемые структуры данных описаны в п.п. 4.1.36.

3.3. Вид взаимодействия.

Используется асимметричный способ взаимодействия типа «запрос от клиента-ответ от сервера». В качестве клиента выступает управляющий компьютер.

В дальнейшем прикладной пакет будет изображаться в виде последовательности байтов в угловых скобках. Структура байта будет изображаться в виде совокупности групп бит с указанием номера бита (нумерация начинается с нуля).

3.4. Адресация

Адресация устройства осуществляется на канальном уровне. Адрес счетчика может принимать значения 1..FF. Адрес 0xFF задаётся по умолчанию при производстве счётчика.

Начиная с версии 2.00 включительно, возможно обращение к счётчику не только по адресу счетчика, но и по его серийному номеру. Для обращения по серийному номеру используется 4-байтовый адрес, формируемый из части серийного номера. Серийный номер счётчика задаётся при производстве. Алгоритм вычисления 4-байтового адреса по серийному номеру описан в п.п. 4.1.27. Структура фреймов для 1-байтовой и 4-байтовой адресации описана в п.п. 3.6.6, 3.6.7, 3.6.8, 3.6.9, 3.6.10.

Выбор типа адресации (1-байтовой или 4-байтовой) производится программно при помощи объекта «Тип адресации». Структура объекта «Тип адресации» рассмотрена в п.п. 4.1, 4.1.32.

3.4.1. Групповая адресация

Адрес 0 является групповым адресом и используется для широковещательного доступа. Значение адреса 0 соответствует любому устройству. В этом случае выбранный метод будет выполнен одновременно для всех устройств, подключённых к интерфейсу.

Все обращения по групповому адресу производятся в рамках сеанса работы со счётчиками.

При посылке счётчику пакета с групповым адресом анализ ответа от счётчика производится не должен.

3.5. Соединение

Соединение определяет совокупность параметров взаимодействия на время взаимодействия. При открытии соединения производится согласование параметров взаимодействия, при закрытии соединения производится отказ от параметров. Любое взаимодействие должно предваряться

открытием соединения, при окончании взаимодействия должно производиться завершение соединения.

Параметрами взаимодействия являются уровень доступа к данным устройства. Уровень доступа подтверждается паролем.

3.6. Кодирование прикладных данных в pdu канального уровня.

3.6.1. Данные взаимодействия прикладного уровня кодируются в pdu канального уровня следующим образом:

<ID_Command, ID_Object [, Data]>

3.6.2. Для сервисов AOPEN и ARELEASE кодировка описана в п.п.3.6.9,3.6.10.

3.6.3. Числовые данные в пакете размещаются так: сначала младшие байты, затем старшие. В дальнейшем такая последовательность байтов будет предполагаться по умолчанию, если явно не указана иная последовательность.

3.6.4. Идентификаторы команды ID_Command:

ID	Наименование
1	GET (BYTE ObjUID)- получить значение объекта
2	SET (BYTE ObjUID, BYTE DataB0, BYTE DataB1...) - Установить значение объекта
3	GETBYTE (BYTE ObjUID, BYTE Ind)- Получить байт объекта
4	SETBYTE (BYTE ObjUID, BYTE Ind, BYTE DataB) - Установить байт объекта
5	LISTINIT (BYTE ObjUID) - Инициализация списка
6	GETLISTNE (BYTE ObjUID) - Получить число элементов в списке
7	GETLISTRECPWI (BYTE ObjUID, WORD Index)- получить элемент списка профиля мощности
8	AOPEN (BYTE GrCode, BYTE Psw[6]) – открытие сеанса связи. Вызывает сервис AOPEN.
9	ARELEASE () – закрытие сеанса связи. Вызывает сервис ARELEASE.
10	SETRTC (BYTE ObjUID, BYTE Seconds, BYTE Minutes, BYTE Hour, BYTE DayInWeek, BYTE Day, BYTE Month, BYTE Year)
11	GET_EVTLIST (BYTE ObjUID, WORD Index) - получить элемент списка событий
12	GET_ENTALIST (BYTE ObjUID, WORD Index) - получить элемент списка энергий в интервалах
13	GETLISTRECPWI_PAR (BYTE ObjUID, WORD Index) - получить элемент списка профиля мощности с признаком неполного интервала
14	PWILIST_SEARCH (BYTE ObjUID, BYTE minutes, BYTE hour, BYTE mday, BYTE month, BYTE year)- найти запись в списке профиля мощности по заданным дате и времени записи
15	GETCURINDEX (BYTE ObjUID)- получить индекс последнего заполненного элемента списка. Данный метод доступен, начиная с версии 1.03 включительно.
16	LIST_SEARCH (BYTE ObjUID, BYTE start_minutes, BYTE start_hour, BYTE start_mday, BYTE start_month, BYTE start_year, BYTE end_minutes, BYTE end_hour, BYTE end_mday, BYTE end_month, BYTE end_year) - найти запись в списке по заданным начальным и конечным дате и времени. Данный метод доступен, начиная с версии 1.06 включительно.
17	GET_COLLECTION (BYTE CollectionUID, WORD Index) – получить коллекцию данных. Метод возвращает данные из нескольких

	объектов. Данный метод доступен, начиная с версии 1.06 включительно.
--	----------------------------------------------------------------------

3.6.5. Идентификаторы объекта указана в п.4

3.6.6. Для сервиса **AREQUEST** структура фрейма при 1-байтовой адресации имеет вид:

Байт	Структура фрейма	
0	Address	Адрес устройства
1	ID_Command	Идентификатор команды
2	ID_Object	Идентификатор объекта
3	DataB0	Данные, байт 0
...	...	
n+2	DataBn	Данные, байт n
n+3	CRC0	Контрольная сумма, байт 0
n+4	CRC1	Контрольная сумма, байт 1

Для сервиса **AREQUEST** структура фрейма при 4-байтовой адресации имеет вид:

Байт	Структура фрейма	
0	AddressB0	Адрес 4-байтовый, байт 0
1	AddressB1	Адрес 4-байтовый, байт 1
2	AddressB2	Адрес 4-байтовый, байт 2
3	AddressB3	Адрес 4-байтовый, байт 3
4	ID_Command	Идентификатор команды
5	ID_Object	Идентификатор объекта
6	DataB0	Данные, байт 0
...	...	
n+5	DataBn	Данные, байт n
n+6	CRC0	Контрольная сумма, байт 0
n+7	CRC1	Контрольная сумма, байт 1

Количество данных DataB0... DataBn зависит от идентификаторов ID_Command, ID_Object и определяется данным протоколом.

3.6.7. Для сервиса **ARESPONSE** структура фрейма при 1-байтовой адресации имеет вид:

Байт	Структура фрейма	
0	Address	Адрес устройства
1	ID_Command	Идентификатор команды
2	ID_Object	Идентификатор объекта
3	NBytes	Число байт данных во фрейме, n
4	DataB0	Данные, байт 0
...	...	
n+3	DataBn	Данные, байт n
n+4	CRC0	Контрольная сумма, байт 0
n+5	CRC1	Контрольная сумма, байт 1

Для сервиса **ARESPONSE** структура фрейма при 4-байтовой адресации имеет вид:

Байт	Структура фрейма	
0	AddressB0	Адрес 4-байтовый, байт 0
1	AddressB1	Адрес 4-байтовый, байт 1
2	AddressB2	Адрес 4-байтовый, байт 2
3	AddressB3	Адрес 4-байтовый, байт 3
4	ID_Command	Идентификатор команды
5	ID_Object	Идентификатор объекта

6	NBytes	Число байт данных во фрейме, n
7	DataB0	Данные, байт 0
...	...	
n+6	DataBn	Данные, байт n
n+7	CRC0	Контрольная сумма, байт 0
n+8	CRC1	Контрольная сумма, байт 1

Количество данных DataB0... DataBn для сервиса **ARESPONSE** определяется параметром NBytes.

3.6.8. Обработка исключений для сервиса **ARESPONSE**.

Для сервиса **ARESPONSE** структура фрейма исключений при 1-байтовой адресации имеет вид:

Байт	Структура фрейма	
0	Address	Адрес счетчика
1	ID_Command	Идентификатор команды с установленным старшим битом
2	ErrCode	Код ошибки
3	ServiceData	Данные, байт 0
4	CRC0	Контрольная сумма, байт 0
5	CRC1	Контрольная сумма, байт 1

Для сервиса **ARESPONSE** структура фрейма исключений при 4-байтовой адресации имеет вид:

Байт	Структура фрейма	
0	AddressB0	Адрес 4-байтовый, байт 0
1	AddressB1	Адрес 4-байтовый, байт 1
2	AddressB2	Адрес 4-байтовый, байт 2
3	AddressB3	Адрес 4-байтовый, байт 3
4	ID_Command	Идентификатор команды с установленным старшим битом
5	ErrCode	Код ошибки
6	ServiceData	Данные, байт 0
7	CRC0	Контрольная сумма, байт 0
8	CRC1	Контрольная сумма, байт 1

Параметр ServiceData не анализируется.

Ниже приведен набор исключений, возвращаемых SLAVE устройством.

enum ErrCode

```
{
    NO_ERROR = 0,
    ILLEGAL_FUNCTION,           // 01, Некорректный идентификатор функции
    ILLEGAL_DATA_ADDRESS,      // 02, Некорректный идентификатор объекта
    ILLEGAL_DATA_VALUE,        // 03, Некорректное значение данных
    SLAVE_DEVICE_FAILURE,      // 04, Невозможно выполнить команду
    ACKNOWLEDGE,               // 05, Запрос принят, начата обработка.
    SLAVE_DEVICE_BUSY,          // 06, Устройство занято
    EEPROM_ACCESS_ERROR,        // 07, Ошибка доступа к памяти EEPROM
    SESSION_CLOSED,             // 08, Сеанс связи закрыт
    ACCESS_DENIED,              // 09, Доступ с указанным уровнем запрещён
    ERROR_CRC,                  // 0A, Ошибка контрольной суммы
    FRAME_INCORRECT,            // 0B, Некорректный фрейм
    JUMPER_ABSENT,              // 0C, Не установлена защитная перемычка
    PASSW_INCORRECT             // 0D, Неверный пароль
};
```

3.6.9. Для сервиса **AOPEN** структура фрейма при 1-байтовой адресации имеет вид:

Байт	Структура фрейма	
0	Address	Адрес счетчика
1	0x08	Код сервиса AOPEN
2	GR_Code	Код уровня доступа
3	PswB0	Пароль для выбранного уровня доступа, байт 0
...	...	
8	PswB6	Пароль для выбранного уровня доступа, байт 5
9	CRC0	Контрольная сумма, байт 0
10	CRC1	Контрольная сумма, байт 1

Для сервиса **AOPEN** структура фрейма при 4-байтовой адресации имеет вид:

Байт	Структура фрейма	
0	AddressB0	Адрес 4-байтовый, байт 0
1	AddressB1	Адрес 4-байтовый, байт 1
2	AddressB2	Адрес 4-байтовый, байт 2
3	AddressB3	Адрес 4-байтовый, байт 3
4	0x08	Код сервиса AOPEN
5	GR_Code	Код уровня доступа
6	PswB0	Пароль для выбранного уровня доступа, байт 0
...	...	
11	PswB6	Пароль для выбранного уровня доступа, байт 5
12	CRC0	Контрольная сумма, байт 0
13	CRC1	Контрольная сумма, байт 1

3.6.10. Для сервиса **ARELEASE** структура фрейма при 1-байтовой адресации имеет вид

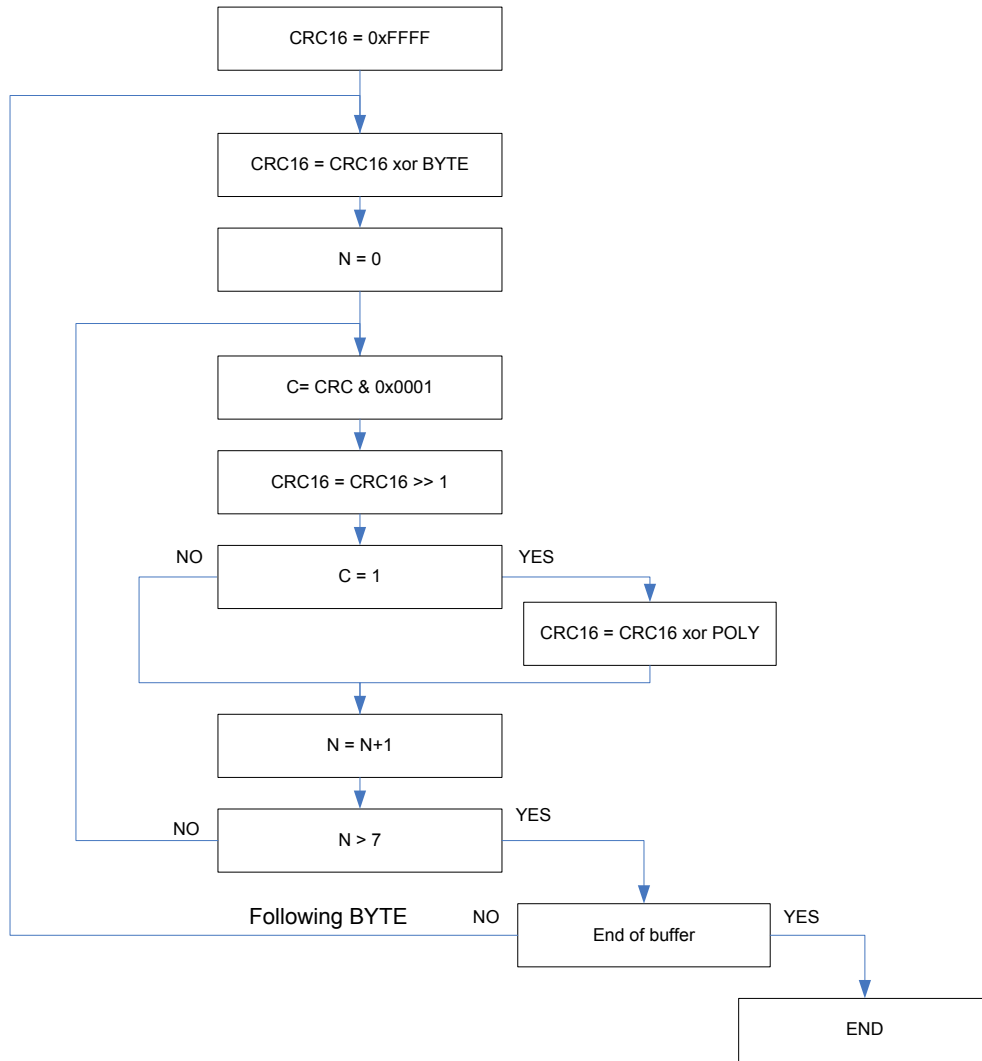
Байт	Структура фрейма	
0	Address	Адрес устройства
1	0x09	Код сервиса ARELEASE
2	0x01	Байт-заглушка
3	CRC0	Контрольная сумма, байт 0
4	CRC1	Контрольная сумма, байт 1

Для сервиса **ARELEASE** структура фрейма при 4-байтовой адресации имеет вид

Байт	Структура фрейма	
0	AddressB0	Адрес 4-байтовый, байт 0
1	AddressB1	Адрес 4-байтовый, байт 1
2	AddressB2	Адрес 4-байтовый, байт 2
3	AddressB3	Адрес 4-байтовый, байт 3
4	0x09	Код сервиса ARELEASE
5	0x01	Байт-заглушка
6	CRC0	Контрольная сумма, байт 0
7	CRC1	Контрольная сумма, байт 1

3.6.11. Алгоритм вычисления контрольной суммы следующий:

Calculation algorithm of the CRC16



XOR = exclusive OR
 N = number of information bits in the BYTE
 POLY = calculation polynomial = 0xA001
 In the CRC16, the 1st byte transmitted is the least significant one

4. ИНТЕРФЕЙСНЫЕ ОБЪЕКТЫ УСТРОЙСТВА.

4.1. Объекты типа Data.

Ид	Наименование	Длина объекта, байт	Знак	Размерность	Права доступа (запись / чтение)
Результаты измерения, первичные оценки (в нестандартных единицах)					
1	Reserved				
2	Reserved				
3	Reserved				
4	Reserved				
5	Reserved				
6	Reserved				
7	Reserved				
8	Reserved				
9	Частота сети	2	Нет	мГц	-/CL_ORD
10	Текущий тариф	1	-	-	-/CL_ORD
11	Параметры индикации	3	-	-	CL_ADM/CL_ADM
12	Reserved				

13	Идентификатор управляющей процедуры	1	-	-	CL_ADM/-
14	Часы реального времени	7	-	-	-/CL_ORD
15	Список праздничных дней	40	-	-	CL_ADM/CL_ORD
16	Срезы мощности	13/14*5904 21/22*5904	Нет	См п.п. 4.1.12	-/CL_ORD
17	Буфер событий	6	-	-	- /CL_ORD
18	Список событий (errors)	8*200	-	-	CL_ADM/CL_ORD
19	Максимальное число тарифов	1	-	-	CL_ADM/ CL_ADM
20	Reserved				
21	Reserved				
22	Reserved				
23	Reserved				
24	Reserved				
25	Reserved				
26	Reserved				
27	Reserved				
28	Reserved				
29	Reserved				
30	Reserved				
31	Reserved				
32	Информация об устройстве (наименование, производитель)	16	-	-	-/CL_ORD
33	Версия встроенного программного обеспечения	4	-	-	-/CL_ORD
34	Калибровка часов реального времени	2	-	-	CL_DEV/CL_ADM
35	Управление автоматическим переходом на летнее/зимнее время	1	-	-	CL_ADM/CL_ORD
36	Нижний предел по напряжению	3	Нет	мВольт	CL_ADM/CL_ORD
37	Верхний предел по напряжению	3	Нет	мВольт	CL_ADM/CL_ORD
38	Нижний предел по частоте	2	Нет	мГц	CL_ADM/CL_ORD
39	Верхний предел по частоте	2	Нет	мГц	CL_ADM/CL_ORD
40	Верхний предел по активной мощности	4	Нет	мВт	CL_ADM/CL_ORD
41	Параметры сеанса связи	5	-	-	CL_ADM/CL_ADM
42	Пароль 1-го уровня	6	-	-	CL_ADM/CL_ORD
43	Пароль 2-го уровня	6	-	-	CL_ADM/CL_ADM
44	Пароль 3-го уровня	6	-	-	CL_DEV/CL_DEV
45	Список событий (messages1)	8*40 или 8*400	-	См п.п. 4.1.13	CL_ADM/CL_ORD
46	Список событий (warnings1)	8*400	-	См п.п. 4.1.13	CL_ADM/CL_ORD
47	Список событий (warnings2)	8*400		См п.п. 4.1.13	CL_ADM/CL_ORD
57	Напряжение батареи резервного питания	2	Нет	мВольт	-/CL_ORD
58	Технологический объект	4	-	-	CL_DEV/CL_DEV
59	Список событий (messages)	8*400	-	-	CL_ADM/CL_ORD

60	Список событий (warnings)	8*400	-	-	CL_ADM/CL_ORD
61	Время интегрирования профиля мощности	1	-	Минут	CL_ADM/CL_ORD
62	Цифровой идентификатор ПО	2	-	-	-/CL_ORD
63	Режим импульсного выхода счётчика	1	-	-	CL_ADM/CL_ORD
64	Тип выхода управления нагрузкой	1	-	-	CL_ADM/CL_ORD
65	Порог автоматического отключения нагрузки	2	Нет	мВт	CL_ADM/CL_ORD
66	Энергия в суточных интервалах	78*35/123 150*123	Нет	См п.п. 4.1.26	-/CL_ORD
67	Энергия в месячных интервалах	78*12 150*12	Нет	См п.п. 4.1.26	-/CL_ORD
68	Серийный номер счетчика	15	-	-	CL_DEV/CL_ORD
69	Таймаут ответа счетчика	1	-	-	CL_ADM/CL_ORD
70	Reserved				
71	Серийный номер печатного узла	15	-	-	CL_DEV/CL_ORD
83	Версия метрологически значимой части ПО	4	-	-	-/CL_ORD
84	Управление встроенным реле отключения нагрузки. Объект используется в ПО счетчика, начиная с версии 1.06 включительно. Доступ к объекту разрешён ТОЛЬКО для модели счетчика со встроенным реле отключения нагрузки.	1	-	-	CL_ADM/CL_ORD
85	Тип адресации	1	-	-	CL_ADM/CL_ORD
86	Ключ сети ZigBee	2	Нет	-	CL_ADM/CL_ORD
98	Reserved for some needs				
99	Reserved for some needs				
100	Напряжение. Фаза А	3	Нет	мВольт	-/CL_ORD
101	Напряжение. Фаза В	3	Нет	мВольт	-/CL_ORD
102	Напряжение. Фаза С	3	Нет	мВольт	-/CL_ORD
103	Ток. Фаза А	3	Да	мАмпер	-/CL_ORD
104	Ток. Фаза В	3	Да	мАмпер	-/CL_ORD
105	Ток. Фаза С	3	Да	мАмпер	-/CL_ORD
106	Активная мощность. Фаза А	4	Да	0,01 Вт	-/CL_ORD
107	Активная мощность. Фаза В	4	Да	0,01 Вт	-/CL_ORD
108	Активная мощность. Фаза С	4	Да	0,01 Вт	-/CL_ORD
109	Активная мощность. Сумма	4	Да	0,01 Вт	-/CL_ORD
110	Реактивная мощность. Фаза А	4	Да	0,01 квар	-/CL_ORD
111	Реактивная мощность. Фаза В	4	Да	0,01 квар	-/CL_ORD
112	Реактивная мощность. Фаза С	4	Да	0,01 квар	-/CL_ORD
113	Реактивная мощность. Сумма	4	Да	0,01 квар	-/CL_ORD
114	Полная мощность. Фаза А	4	Нет	0,01 кВА	-/CL_ORD
115	Полная мощность. Фаза В	4	Нет	0,01 кВА	-/CL_ORD
116	Полная мощность. Фаза С	4	Нет	0,01 кВА	-/CL_ORD

117	Полная мощность. Сумма	4	Нет	0,01 кВА	-/CL_ORD
118	Активная импортируемая энергия суммарная	4	Нет	См п.п. 4.1.1	-/CL_ORD
119	Активная импортируемая энергия по тарифу 1	4	Нет	См п.п. 4.1.1	-/CL_ORD
120	Активная импортируемая энергия по тарифу 2	4	Нет	См п.п. 4.1.1	-/CL_ORD
121	Активная импортируемая энергия по тарифу 3	4	Нет	См п.п. 4.1.1	-/CL_ORD
122	Активная импортируемая энергия по тарифу 4	4	Нет	См п.п. 4.1.1	-/CL_ORD
123	Активная импортируемая энергия по тарифу 5	4	Нет	См п.п. 4.1.1	-/CL_ORD
124	Активная импортируемая энергия по тарифу 6	4	Нет	См п.п. 4.1.1	-/CL_ORD
125	Активная импортируемая энергия по тарифу 7	4	Нет	См п.п. 4.1.1	-/CL_ORD
126	Активная импортируемая энергия по тарифу 8	4	Нет	См п.п. 4.1.1	-/CL_ORD
127	Реактивная импортируемая энергия суммарная	4	Нет	См п.п. 4.1.1	-/CL_ORD
128	Реактивная импортируемая энергия по тарифу 1	4	Нет	См п.п. 4.1.1	-/CL_ORD
129	Реактивная импортируемая энергия по тарифу 2	4	Нет	См п.п. 4.1.1	-/CL_ORD
130	Реактивная импортируемая энергия по тарифу 3	4	Нет	См п.п. 4.1.1	-/CL_ORD
131	Реактивная импортируемая энергия по тарифу 4	4	Нет	См п.п. 4.1.1	-/CL_ORD
132	Реактивная импортируемая энергия по тарифу 5	4	Нет	См п.п. 4.1.1	-/CL_ORD
133	Реактивная импортируемая энергия по тарифу 6	4	Нет	См п.п. 4.1.1	-/CL_ORD
134	Реактивная импортируемая энергия по тарифу 7	4	Нет	См п.п. 4.1.1	-/CL_ORD
135	Реактивная импортируемая энергия по тарифу 8	4	Нет	См п.п. 4.1.1	-/CL_ORD
136	Параметры калибровки	136	Да	-	CL_DEV/CL_DEV
137	Тарифное расписание на январь	64	-	-	CL_ADM/CL_ORD
138	Тарифное расписание на февраль	64	-	-	CL_ADM/CL_ORD
139	Тарифное расписание на март	64	-	-	CL_ADM/CL_ORD
140	Тарифное расписание на апрель	64	-	-	CL_ADM/CL_ORD
141	Тарифное расписание на май	64	-	-	CL_ADM/CL_ORD
142	Тарифное расписание на июнь	64	-	-	CL_ADM/CL_ORD
143	Тарифное расписание на июль	64	-	-	CL_ADM/CL_ORD
144	Тарифное расписание на август	64	-	-	CL_ADM/CL_ORD
145	Тарифное расписание на сентябрь	64	-	-	CL_ADM/CL_ORD
146	Тарифное расписание на	64	-	-	CL_ADM/CL_ORD

	октябрь				
147	Тарифное расписание на ноябрь	64	-	-	CL_ADM/CL_ORD
148	Тарифное расписание на декабрь	64	-	-	CL_ADM/CL_ORD
149	Активная экспортируемая энергия суммарная	4	Нет	См п.п. 4.1.1	-/CL_ORD
150	Активная экспортируемая энергия по тарифу 1	4	Нет	См п.п. 4.1.1	-/CL_ORD
151	Активная экспортируемая энергия по тарифу 2	4	Нет	См п.п. 4.1.1	-/CL_ORD
152	Активная экспортируемая энергия по тарифу 3	4	Нет	См п.п. 4.1.1	-/CL_ORD
153	Активная экспортируемая энергия по тарифу 4	4	Нет	См п.п. 4.1.1	-/CL_ORD
154	Активная экспортируемая энергия по тарифу 5	4	Нет	См п.п. 4.1.1	-/CL_ORD
155	Активная экспортируемая энергия по тарифу 6	4	Нет	См п.п. 4.1.1	-/CL_ORD
156	Активная экспортируемая энергия по тарифу 7	4	Нет	См п.п. 4.1.1	-/CL_ORD
157	Активная экспортируемая энергия по тарифу 8	4	Нет	См п.п. 4.1.1	-/CL_ORD
158	Реактивная экспортируемая энергия суммарная	4	Нет	См п.п. 4.1.1	-/CL_ORD
159	Реактивная экспортируемая энергия по тарифу 1	4	Нет	См п.п. 4.1.1	-/CL_ORD
160	Реактивная экспортируемая энергия по тарифу 2	4	Нет	См п.п. 4.1.1	-/CL_ORD
161	Реактивная экспортируемая энергия по тарифу 3	4	Нет	См п.п. 4.1.1	-/CL_ORD
162	Реактивная экспортируемая энергия по тарифу 4	4	Нет	См п.п. 4.1.1	-/CL_ORD
163	Реактивная экспортируемая энергия по тарифу 5	4	Нет	См п.п. 4.1.1	-/CL_ORD
164	Реактивная экспортируемая энергия по тарифу 6	4	Нет	См п.п. 4.1.1	-/CL_ORD
165	Реактивная экспортируемая энергия по тарифу 7	4	Нет	См п.п. 4.1.1	-/CL_ORD
166	Реактивная экспортируемая энергия по тарифу 8	4	Нет	См п.п. 4.1.1	-/CL_ORD
167	Конфигурация учета энергий и импульсных выходов	1	Нет	См п.п. 4.1.37	CL_ADM/CL_ORD
168	ПКЭ	44	Нет	См п.п. 4.1.38	CL_ADM/CL_ORD
169	Защитное отключение нагрузки	25	Нет	См п.п. 4.1.39	CL_ADM/CL_ORD
170	Причина защитного отключения нагрузки	2	Нет	См п.п. 4.1.40.	CL_INT/CL_ORD

4.1.1. Интерфейсные объекты «Активная импортируемая энергия суммарная», «Активная импортируемая энергия по тарифу 1», «Активная импортируемая энергия по тарифу 2», «Активная импортируемая энергия по тарифу 3», «Активная импортируемая энергия по

тарифу 4», «Активная импортируемая энергия по тарифу 5», «Активная импортируемая энергия по тарифу 6», «Активная импортируемая энергия по тарифу 7», «Активная импортируемая энергия по тарифу 8», «Реактивная импортируемая энергия суммарная», «Реактивная импортируемая энергия по тарифу 1», «Реактивная импортируемая энергия по тарифу 2», «Реактивная импортируемая энергия по тарифу 3», «Реактивная импортируемая энергия по тарифу 4», «Реактивная импортируемая энергия по тарифу 5», «Реактивная импортируемая энергия по тарифу 6», «Реактивная импортируемая энергия по тарифу 7», «Реактивная импортируемая энергия по тарифу 8», «Активная экспортируемая энергия суммарная», «Активная экспортируемая энергия по тарифу 1», «Активная экспортируемая энергия по тарифу 2», «Активная экспортируемая энергия по тарифу 3», «Активная экспортируемая энергия по тарифу 4», «Активная экспортируемая энергия по тарифу 5», «Активная экспортируемая энергия по тарифу 6», «Активная экспортируемая энергия по тарифу 7», «Активная экспортируемая энергия по тарифу 8», «Реактивная экспортируемая энергия суммарная», «Реактивная экспортируемая энергия по тарифу 1», «Реактивная экспортируемая энергия по тарифу 2», «Реактивная экспортируемая энергия по тарифу 3», «Реактивная экспортируемая энергия по тарифу 4», «Реактивная экспортируемая энергия по тарифу 5», «Реактивная экспортируемая энергия по тарифу 6», «Реактивная экспортируемая энергия по тарифу 7», «Реактивная экспортируемая энергия по тарифу 8».

Интерфейсные объекты имеют размерность 4 байта. Значение энергий выводится в упакованном BCD-формате, младший байт вперёд, младшая тетрада вперёд. Каждая значащая цифра энергии располагается в своей тетраде. Размерность значения энергии зависит от модели счётчика. Модель счётчика указывается в объекте «Информация об устройстве» (ID 32).

Для модификаций счётчиков «Милур 305.11», «Милур 305.12» размерность значений энергии составляет 0,001 кВт*ч (квар*ч). Например, значение 85 10 00 00 означает значение энергии 0,158 кВт*ч (квар*ч).

Для модификаций счётчиков «Милур 305.32» размерность значений энергии составляет 0,01 кВт*ч (квар*ч). Например, значение 35 10 00 00 означает значение энергии 1,53 кВт*ч (квар*ч).

Версия ПО счетчика	Интерфейсные объекты 118-135	Интерфейсные объекты 149 – 166
0100 - 0199	Учёт активной и реактивной энергии по модулю	недоступны
0200 - 0299	Учёт импортируемой активной и реактивной энергии (энергия прямого направления) *	Учёт экспортируемой активной и реактивной энергии (энергия обратного направления) *

* - конфигурирование учета активной и реактивной электроэнергии указывается в интерфейсном объекте «Конфигурация учета электроэнергии», см п.п. 4.1.37.

4.1.2. Интерфейсные объекты «Активная мощность. Фаза А», «Активная мощность. Фаза В», «Активная мощность. Фаза С», «Активная мощность. Сумма», «Реактивная мощность. Фаза А», «Реактивная мощность. Фаза В», «Реактивная мощность. Фаза С», «Реактивная мощность. Сумма», «Полная мощность. Фаза А», «Полная мощность. Фаза В», «Полная мощность. Фаза С», «Полная мощность. Сумма».

Интерфейсные объекты «Активная мощность», «Реактивная мощность», «Полная мощность» имеют размерность 4 байта. Значение мощности выводится в шестнадцатеричном формате, младший байт вперёд. Отрицательное значение мощности выводится в дополнительном коде. Например, мощность -1150 Вт будет соответствовать коду 0xFFFE3EC8.

4.1.3. Интерфейсные объекты «Напряжение. Фаза А», «Напряжение. Фаза В», «Напряжение. Фаза С», «Ток. Фаза А», «Ток. Фаза В», «Ток. Фаза С».

Интерфейсные объекты «Напряжение» и «Ток» имеют размерность 3 байта. Значение напряжения и тока выводятся в шестнадцатеричном формате, младший байт вперёд.

Отрицательное значение тока выводится в дополнительном коде. Например, Ток -0,4 Ампера будет соответствовать коду: 0xFFFE70.

4.1.4. Интерфейсный объект «Управляющая процедура»

Интерфейсный объект «Управляющая процедура» служит для выполнения определенных действий по отношению к другим объектам. Вызов управляющей процедуры производится следующей командой:

SET(BYTE ObjUID, BYTE DataB0)- Установить значение объекта,

ObjUID – UID объекта «Управляющая процедура»;

DataB0 – байт атрибута управляющей процедуры (AttrID).

Управляющая процедура имеет права доступа на выполнение.

Для выполнения требуемой управляющей процедуры необходимо в качестве байта атрибута указать нужный идентификатор AttrID.

Интерфейсный объект «Управляющая процедура» имеет следующие атрибуты:

AttrID	Назначение	Значение	Права доступа (выполнение)
1	Сброс на заводские установки		CL_ADM
2	Сброс буфера событий		CL_ADM
3	Сброс накопленной энергии по всем тарифам		CL_ADM
4	Перезагрузка калибровочных коэффициентов в измерительный модуль		CL_DEV
5	Применить изменения в интерфейсном объекте «Параметры сеанса связи».		CL_ADM
6	Однократный запуск измерения напряжения батареи резервного питания		CL_ADM
7	Заводская инициализация. Для выполнения процедуры должна быть установлена защитная перемычка на плате счетчика.		CL_DEV
8	Показать адрес счетчика		CL_ADM

При выполнении управляющей процедуры с кодом 1 (Сброс на заводские установки) выполняются следующие настройки:

- Установка праздничного расписания по умолчанию;
- Установка тарифного расписания по умолчанию;
- Установка параметров обмена с хостом по умолчанию;
- Установка типа адресации в 1-байтовый тип.
- Сброс флага разрешения перехода на летнее/зимнее время;
- Установка пороговых значений по умолчанию;
- Установка времени интегрирования срезов мощности на 30 минут;
- Установка порога автоматического управления нагрузкой;
- Установка импульсного выхода в режим телеметрии (импульсный выход).

При выполнении управляющей процедуры с кодом 7 (Заводская инициализация счетчика) выполняются все настройки, указанные для кода 1 и дополнительно следующие настройки:

- Сброс накопленной энергии по всем тарифам;
- Установка калибровочных коэффициентов в начальное значение;
- Сброс ключа сети ZigBee;
- Инициализация профиля мощности;
- Инициализация всех списков событий;
- Инициализация списков энергий на начало суток и месяца;
- Сброс регистра смещения указателя резервной копии накопленной энергии.

При выполнении управляющей процедуры с кодом 8 (Показать адрес счетчика), на ЖК индикаторе счетчика в течение 5 секунд индицируется адрес счетчика в десятичном виде. Во время индикации адреса, счетчик продолжает учитывать энергию и остается полностью работоспособен.

4.1.5. Интерфейсный объект «Параметры индикации»

Интерфейсный объект «Параметры индикации» имеет размер 3 байта. Структура объекта «Параметры индикации» представлена ниже:

Параметр	Значение по умолчанию	Размер	Маска параметра
Время индикации параметра в меню по умолчанию, секунд	10	1 байт	0xFFFF
Таймаут перехода на пользовательское меню, минут	1	1 байт	0xFF
Активная энергия суммарная	1	1 бит	0x1
Активная энергия по тарифу 1	1	1 бит	0x2
Активная энергия по тарифу 2	1	1 бит	0x4
Активная энергия по тарифу 3	1	1 бит	0x8
Активная энергия по тарифу 4	1	1 бит	0x1
Активная мощность	1	1 бит	0x2
Дата	1	1 бит	0x4
Время	1	1 бит	0x8

Все данные, перечисленные в таблице, отображаются на индикаторе циклически.

Время индикации параметра – это время, в секундах, в течение которого каждый из включённых параметров показывается на индикаторе.

Таймаут перехода на пользовательское меню – это время, в минутах, по истечении которого автоматический возврат на пользовательское меню по умолчанию.

Битовые значения определяют, включён или выключен данный параметр для индикации. Если бит установлен в 1, данный параметр будет отображаться на индикаторе. Если бит сброшен в 0, данный параметр отображаться не будет.

4.1.6. Интерфейсный объект «Параметры сеанса связи»

Интерфейсный объект «Параметры сеанса связи» имеет длину 5 байт. Объект имеет следующую структуру:

Атрибут	Смещение	Длина атрибута, байт	Значение по умолчанию
Адрес счетчика	0	1	0xFF
Кол-во попыток открытия сеанса	1	1	3
Скорость обмена	2	1	0x05 (BRATE_9600)
Время удержания сеанса, мин	3	1	0x0A
Время блокировки сеанса, мин	4	1	0x1E

- **Bitrate**- скорость обмена

enum *typeBRate*

```
{
  BRATE_300      = 0
  BRATE_600,
  BRATE_1200,
  BRATE_2400,
  BRATE_4800,
  BRATE_9600,
```

```

BRATE_19200,
BRATE_38400,
}

```

- Адрес счётчика- адрес устройства, значение 0- соответствует любому устройству (для широковещательного доступа). Адрес устройства указывается в формуляре. При попытке установить адрес 0, счетчик вернет исключение ILLEGAL_DATA_VALUE.
- Кол-во попыток открытия сеанса связи ограничивает число попыток установить связь со счётчиком в целях защиты от подбора пароля.
- Время блокировки сеанса – время в течение которого установка связи со счётчиком невозможна, даже при верно набранном пароле. Таймер блокировки взводится при исчерпании числа попыток открытия сеанса. После выключения и повторного включения счётчика таймер блокировки сбрасывается.
- Время удержания сеанса – время, по истечении которого сеанс автоматически закрывается, если в течение этого времени отсутствует прием/передача данных.

После изменения параметров объекта «Параметры сеанса связи», необходимо применить эти изменения, запустив управляющую процедуру с кодом 0x05. См описание интерфейсного объекта «Управляющая процедура». После вызова управляющей процедуры, дальнейшие команды должны подаваться с учетом выполненных изменений. Изменения также можно применить, выключив и повторно включив счётчик.

4.1.7. Интерфейсный объект «Часы реального времени»

Интерфейсный объект «ЧРВ» имеет длину 7 байт. Объект ЧРВ имеет следующую структуру:

Атрибут	Смещение	Длина атрибута, байт	Значение по умолчанию
Секунды	0	1	0
Минуты	1	1	0
Часы	2	1	0
День недели	3	1	07
День	4	1	01
Месяц	5	1	01
Год	6	1	11

Дни недели и месяцы имеют следующую нотацию:

```
enum eDAY { SUN = 1, MON, TUE, WED, THU, FRI, SAT };
```

```
enum eMONTH { JAN = 1, FEB, MAR, APR, MAY, JUN, JUL, AUG, SEP, OCT, NOV, DEC };
```

4.1.8. Интерфейсный объект «Тарифное расписание»

Дискрет установки интервала действия тарифа равняется 1 мин.

Тарифное расписание задаётся на рабочий день, праздничный день, субботу, воскресенье. В тарифном расписании предусматривается 16 переключений.

Каждое переключение задаёт время (часы:минуты) переключения и код тарифа, на который происходит переключение.

Запись переключения имеет длину 4 байта. Формат записи переключения следующий:

Тарифная зона	Время	Рабочий	Праздничный	Суббота	Воскресенье
	2 байта 07:00 hh:mm	4 бита 0x0	4 бита 0xF	4 бита 0x0	4 бита 0x1
		2 байта			

Пример записи тарифного расписания:

Тарифная зона	Время	Рабочий	Праздничный	Суббота	Воскресенье
	hh:mm 07:00	4 бита 0x0	4 бита 0xF	4 бита 0x0	4 бита 0x1

Ниже приведен пример тарифного расписания на месяц:

Тарифная зона	Время	Рабочий	Праздничный	Суббота	Воскресенье
1	00:00	1	1	1	1
2	07:00	0	0xf	0	0xf
3	15:00	0xf	0xf	1	0xf
4	22:00	1	0xf	0xf	0xf
5	0xffff	0xf	0xf	0xf	0xf
6	0xffff	0xf	0xf	0xf	0xf
7	0xffff	0xf	0xf	0xf	0xf
8	0xffff	0xf	0xf	0xf	0xf
9	0xffff	0xf	0xf	0xf	0xf
10	0xffff	0xf	0xf	0xf	0xf
11	0xffff	0xf	0xf	0xf	0xf
12	0xffff	0xf	0xf	0xf	0xf
13	0xffff	0xf	0xf	0xf	0xf
14	0xffff	0xf	0xf	0xf	0xf
15	0xffff	0xf	0xf	0xf	0xf
16	0xffff	0xf	0xf	0xf	0xf

В колонке «Время» задаётся время переключения на выбранный тариф. Время задается в часах и минутах.

В колонках «Рабочий», «Праздничный», «Суббота», «Воскресенье» задаётся код тарифа.

Значения 0xffff и 0xf означают отсутствие переключения тарифа.

Объём расписания на один месяц- $4*8 = 32$ байта.

Объём расписания на один год- $32*12 = 384$ байта.

В тарифном расписании задаётся код тарифа. Тарифу 1 соответствует код 0. Тарифу 2 соответствует код 1. Тарифу 3 соответствует код 2. Тарифу 4 соответствует код 3.

Для удобства предусмотрены отдельные интерфейсные объекты для каждого месяца.

Таким образом, в примере тарифного расписания, приведённом выше, заданы следующие переключения:

Дни	Время переключения
Рабочие дни:	07:00 – 22:00 – тариф №1. 22:00 – 07:00 – тариф №2.
Праздничные дни:	00:00 – 24:00 – тариф №2.
Суббота:	00:00 – 07:00 – тариф №2. 07:00 – 15:00 – тариф №1. 15:00 – 24:00 – тариф №2.
Воскресенье:	00:00 – 24:00 – тариф №2.

Записи тарифного расписания на сутки должны начинаться с начала суток. В первой записи тарифного расписания на сутки должно быть установлено время 00:00. Счётчик производит проверку этого условия и при его невыполнении устанавливает ошибку «Неверные данные в тарифном расписании». Переключения тарифного расписания должны быть записаны последовательно, без пропусков. Время переключения на следующий тариф должно задаваться строго последовательно по увеличению времени. Если время переключения в текущей записи окажется меньше, чем в предыдущей записи, то будет установлен тариф текущей записи. Установка времени переключения 00:00 в любую запись кроме первой, приведёт к установке тарифа 1 независимо от времени переключения в предыдущих записях.

4.1.9. Интерфейсный объект «Список праздничных дней»

Список праздничных дней содержит 20 записей. Каждая запись имеет длину 2 байта, содержит параметры праздничного дня и имеет формат:

Индекс	День	Месяц
	1 байт	1 байт

По умолчанию (заводские установки) установлен следующий список праздничных дней:

Индекс	День	Месяц
0	1	1
1	2	1
2	3	1
3	4	1
4	5	1
5	7	1
6	23	2
7	8	3
8	1	5
9	9	5
10	12	6
11	4	11
12	0xff	0xff
13	0xff	0xff
14	0xff	0xff
15	0xff	0xff
16	0xff	0xff
17	0xff	0xff
18	0xff	0xff
19	0xff	0xff

В незаполненные дни праздничного расписания должно быть записано значение 0xff означающее отсутствие записи.

4.1.10. Интерфейсный объект «Информация об устройстве»

Интерфейсный объект представляет собой строку из 16 байт. Каждый байт соответствует символу в соответствии со стандартной Win-кодировкой. Строка "Milandr-120" будет представлена в виде последовательности: 4D 69 6C 61 6E 64 72 2D 31 32 30.

4.1.11. Интерфейсный объект «Версия встроенного ПО»

Интерфейсный объект «Версия встроенного ПО» представляет собой строку из 4 байт. Каждый байт соответствует символу в соответствии со стандартной Win-кодировкой. Строка «1.00» будет представлена в виде последовательности: 31 2E 30 30.

4.1.12. Интерфейсный объект «Срезы мощности»

Объект «Срезы мощности» представляет собой линейный список. Функциональность «Срезом мощности» реализует хранение записей в памяти EEPROM счетчика, их чтение посредством последовательного интерфейса по запросу с ПК.

Каждая запись содержит информацию о потребленной энергии за полчаса период времени, время и дату начала интервала измерения.

Для версий ПО счетчика с 0100 по 0199 включительно, ответ на команду **GETLISTRECPWI** описывается следующей структурой:

```
typedef struct{
    BYTE minutes;    // Минуты (0..59)
    BYTE hours;      // Часы (0..12)
    BYTE day;        // День месяца (1..31)
```

```

BYTE month;      // Месяц (1...12)
BYTE year;       // Год (00...99)
BYTE pwrP[4];    // Значение суммарной активной энергии, накопленной за период
BYTE pwrQ[4];    // Значение суммарной реактивной энергии, накопленной за период
} PWIRRecord_t;

```

Ответ на команду **GETLISTRECPWI_PAR** описывается следующей структурой:

```

typedef struct{
  BYTE minutes;  // Минуты (0..59)
  BYTE hours;    // Часы (0...12)
  BYTE day;      // День месяца (1...31)
  BYTE month;    // Месяц (1...12)
  BYTE year;     // Год (00...99)
  BYTE pwrP[4];  // Значение суммарной активной энергии, накопленной за период
  BYTE pwrQ[4];  // Значение суммарной реактивной энергии, накопленной за период
  BYTE wpar;     // Признак неполного среза
} PWIRRecordw_t;

```

Для версий ПО счетчика с 0200 по 0299 включительно, ответ на команду **GETLISTRECPWI** описывается следующей структурой:

```

typedef struct{
  BYTE minutes;  // Минуты (0..59)
  BYTE hours;    // Часы (0...12)
  BYTE day;      // День месяца (1...31)
  BYTE month;    // Месяц (1...12)
  BYTE year;     // Год (00...99)
  BYTE pwrPI[4]; // Значение суммарной импортируемой активной энергии
  BYTE pwrQI[4]; // Значение суммарной импортируемой реактивной энергии
  BYTE pwrPE[4]; // Значение суммарной экспортируемой активной энергии
  BYTE pwrQE[4]; // Значение суммарной экспортируемой реактивной энергии
} PWIRRecord_t;

```

Ответ на команду **GETLISTRECPWI_PAR** описывается следующей структурой:

```

typedef struct{
  BYTE minutes;  // Минуты (0..59)
  BYTE hours;    // Часы (0...12)
  BYTE day;      // День месяца (1...31)
  BYTE month;    // Месяц (1...12)
  BYTE year;     // Год (00...99)
  BYTE pwrPI[4]; // Значение суммарной импортируемой активной энергии
  BYTE pwrPE[4]; // Значение суммарной экспортируемой активной энергии
  BYTE pwrQI[4]; // Значение суммарной импортируемой реактивной энергии
  BYTE pwrQE[4]; // Значение суммарной экспортируемой реактивной энергии
  BYTE wpar;     // Признак неполного среза
} PWIRRecordw_t;

```

В данной структуре значение энергии, выводится в упакованном BCD-формате, младший байт вперёд, младшая тетрада вперёд. Каждая значащая цифра располагается в своей тетраде.

Размерность значения энергии зависит от модели счётчика. Модель счётчика указывается в объекте «Информация об устройстве» (ID 32).

Для модификаций счётчиков «Милур 305.11», «Милур 305.12» размерность значений энергии составляет 0,001 кВт*ч (квар*ч). Например, значение 85 10 00 00 означает значение энергии 0,158 кВт*ч (квар*ч).

Для модификаций счётчиков «Милур 305.32» размерность значений энергии составляет 0,01 кВт*ч (квар*ч). Например, значение 35 10 00 00 означает значение энергии 1,53 кВт*ч (квар*ч).

Управление функциональностью «Срезов мощности» осуществляется посредством команд **LISTINIT**, **GETLISTNE**, **GETCURINDEX** и **GETLISTRECPWI**.

Для работы со списком его необходимо сначала выставить время в счетчике, а затем проинициализировать его командой **LISTINIT**.

После успешной инициализации в список автоматически добавляется одна запись.

Командой **GETLISTRECPWI** с параметром порядкового номера записи можно получить её содержимое одной записи списка (13 байт) в формате **PWIRRecord_t**. Командой **GETLISTRECPWI_PAR** с параметром порядкового номера записи можно получить её содержимое одной записи списка (14 байт) в формате **PWIRRecordw_t**.

Максимальное количество записей в списке – 5904 (за последние 123 суток). Номер запрашиваемой записи в команде **GETLISTRECPWI** должен быть в диапазоне от 0 до 5903, в противном случае будет возвращен код ошибки.

Командой **GETLISTNE** запрашивается общее количество имеющихся в списке записей. Количество записей для заполненного списка всегда будет возвращаться всегда одно и то же – 5904.

Командой **GETCURINDEX** запрашивается индекс последнего заполненного элемента списка срезов мощности.

При установке времени счетчика, переводе часов, потере времени из-за разряженной батареи резервного питания в профиле мощности могут появляться пропуски или появляться дублирующие записи с одинаковым временем. Дата и время в записях срезов мощности соответствуют поясному времени счётчика. Это означает, что при переходе на летнее/зимнее время, для срезов мощности время всегда соответствует поясному (зимнему) времени. См. описание объекта «Управление автоматическим переходом на летнее/зимнее время», п.п. 4.1.18.

В выключенном состоянии новые записи в профиле мощности не формируются. При включении счётчика производится проверка времени. При истечении времени накопления энергии в профиле, текущий интервал закрывается и открывается следующая запись в списке. При этом никаких пропусков в профиле мощности не формируется.

4.1.13. Интерфейсный объект «Буфер событий»

Буфер событий представляет собой битовый буфер длиной 6 байт. Каждое событие устанавливается независимо. Структурно буфер событий разделён на три области: errors, messages, warnings.

WORD errors

WORD messages

WORD warnings

Содержимое буфера событий доступно для чтения методами **GET** и **GETBYTE**.

- **Область errors содержит следующие коды (двоичный вид):**

Код	Наименование
0000000000000000	Нет ошибок
errors	
1000000000000000	Разряжена батарея
0100000000000000	Ошибка записи в EEPROM
0010000000000000	Некорректная дата/время
0001000000000000	Неверное тарифное расписание

- **Область messages содержит следующие коды (двоичный вид):**

Код	Наименование

0000000000000000	Нет сообщений
messages	
1000000000000000	Сброс на заводские установки
0100000000000000	Изменение тарифного расписания
0010000000000000	Изменение праздничного расписания
0001000000000000	Сброс энергий
0000100000000000	Инициализация профиля мощности
0000010000000000	Инициализация списков событий
0000001000000000	Изменение адреса счетчика
0000000100000000	Изменение пароля уровня пользователя
0000000010000000	Изменение пароля уровня администратора
0000000001000000	Изменение пароля уровня разработчика
0000000000100000	Блокирование пароля
0000000000010000	Включение режима поверки
0000000000001000	Выключение режима поверки
0000000000000100	Включение нагрузки
0000000000000010	Выключение нагрузки
0000000000000001	Коррекция времени

- **Область warnings содержит следующие коды (двоичный вид):**

Код	Наименование
0000000000000000	Нет предупреждений
warnings	
1000000000000000	Выключение счетчика
0100000000000000	Включение счетчика
0010000000000000	Реверсивное потребление энергии по фазе А
0001000000000000	Реверсивное потребление энергии по фазе В
0000100000000000	Реверсивное потребление энергии по фазе С
0000010000000000	Пропадание напряжения по фазе А
0000001000000000	Пропадание напряжения по фазе В
0000000100000000	Пропадание напряжения по фазе С
0000000010000000	Восстановление напряжения по фазе А
0000000001000000	Восстановление напряжения по фазе В
0000000000100000	Восстановление напряжения по фазе С
0000000000010000	Выход за пределы порога по напряжению (по фазе А*)
0000000000001000	Выход за пределы порога по частоте (по фазе А*)
0000000000000100	Превышен порог по мощности (по фазе А*)
0000000000000010	Вскрытие крышки счетчика
0000000000000001	Воздействие магнитным полем

* Для версий ПО счетчика с 0100 по 0199 включительно, в области **warnings** выход за пределы порога по напряжению, частоте и мощности фиксируется для любой фазы.

Для версий ПО счетчика с 0200 включительно:

- в области **warnings** выход за пределы порога по напряжению, частоте и мощности фиксируется для фазы А;
- вводится объект **message1**, расширяющий список сообщений;
- вводится объект **warnings1**, расширяющий список предупреждений.

- Область **messages1** содержит следующие коды (двоичный вид):

Код	Наименование
0000000000000000	Нет сообщений
messages	
1000000000000000	Установлен режим учёта активной энергии по модулю
0100000000000000	Установлен режим учёта активной импортируемой энергии
0010000000000000	Установлен режим учёта активной экспортируемой энергии
0001000000000000	Установлен режим двунаправленный учёт активной энергии
0000100000000000	Установлен режим учёта реактивной энергии по модулю
0000010000000000	Установлен режим учёта реактивной импортируемой энергии
0000001000000000	Установлен режим учёта реактивной экспортируемой энергии
0000000100000000	Установлен режим двунаправленный учёт реактивной энергии
0000000010000000	На импульсный выход активной энергии подана активная импортируемая энергия
0000000001000000	На импульсный выход активной энергии подана активная экспортируемая энергия
0000000000100000	На импульсный выход активной энергии подана реактивная импортируемая энергия
0000000000010000	На импульсный выход активной энергии подана реактивная экспортируемая энергия
0000000000001000	На импульсный выход реактивной энергии подана активная импортируемая энергия
0000000000000100	На импульсный выход реактивной энергии подана активная экспортируемая энергия
0000000000000010	На импульсный выход реактивной энергии подана реактивная импортируемая энергия
0000000000000001	На импульсный выход реактивной энергии подана реактивная экспортируемая энергия

- Область **warnings1** содержит следующие коды (двоичный вид):

Код	Наименование
0000000000000000	Нет предупреждений
warnings	
1000000000000000	Возврат в пределы порога по напряжению по фазе А
0100000000000000	Выход за пределы порога по напряжению по фазе В
0010000000000000	Возврат в пределы порога по напряжению по фазе В
0001000000000000	Выход за пределы порога по напряжению по фазе С
0000100000000000	Возврат в пределы порога по напряжению по фазе С
0000010000000000	Возврат в пределы порога по мощности по фазе А
0000001000000000	Выход за пределы порога по мощности по фазе В
0000000100000000	Возврат в пределы порога по мощности по фазе В
0000000010000000	Выход за пределы порога по мощности по фазе С
0000000001000000	Возврат в пределы порога по мощности по фазе С
0000000000100000	Закрытие клеммной крышки
0000000000010000	
0000000000001000	
0000000000000100	
0000000000000010	

000000000000000001	

- Область **warning2** содержит следующие коды (двоичный вид):

Код	Наименование
000000000000000000	Нет предупреждений
warnings	
100000000000000000	Отклонение частоты за 100% недельного интервала
010000000000000000	Отклонение частоты за 95% недельного интервала
001000000000000000	Отклонение напряжения по фазе А за 100% недельного интервала
000100000000000000	Отклонение напряжения по фазе В за 100% недельного интервала
000010000000000000	Отклонение напряжения по фазе С за 100% недельного интервала
000001000000000000	
000000100000000000	
000000010000000000	
000000001000000000	
000000000100000000	
000000000010000000	
000000000001000000	
000000000000100000	
000000000000010000	
000000000000001000	
000000000000000100	
000000000000000010	
000000000000000001	

4.1.14. Интерфейсный объект «Список событий (ошибки)», «Список событий (сообщения)», «Список событий (предупреждения)»

Список событий (ошибки, сообщения, предупреждения) представляет собой кольцевой список, заполняемый по изменению соответствующей области буфера ошибок. Дискрет опроса буфера событий 1 секунда.

Элемент списка событий представляет собой следующую структуру:

```
typedef struct {
    BYTE seconds;    // Секунды (0..59)
    BYTE minutes;   // Минуты (0..59)
    BYTE hours;     // Часы (0...12)
    BYTE day;       // День месяца (1...31)
    BYTE month;     // Месяц (1...12)
    BYTE year;      // Год (00...99)
    BYTE events[2]; // События (2 байта)
} EVTRecord_t;
```

Элемент структуры «events» представляет собой копию соответствующей области буфера событий.

Управление функциональностью объектов «Список событий» осуществляется посредством команд **LISTINIT**, **GETLISTNE**, **GET_EVTLIST**.

Для работы со списками, необходимо сначала выставить время в счетчике, а затем проинициализировать их командой **LISTINIT**.

После успешной инициализации список событий будет пустой. Записи в списках будут отсутствовать.

Командами **GET_EVTLIST** с указанием объекта и параметром порядкового номера записи можно получить её содержимое одной записи списка в формате **EVTRecord_t**.

Максимальное количество записей в списке сообщений и предупреждений – 400. Номер запрашиваемой записи в команде **GET_EVTLIST** должен быть в диапазоне от 0 до 399, в противном случае будет возвращен код ошибки.

Максимальное количество записей в списке ошибок – 200. Номер запрашиваемой записи в команде **GET_EVTLIST** должен быть в диапазоне от 0 до 199, в противном случае будет возвращен код ошибки.

Командой **GETLISTNE** запрашивается общее количество имеющихся в списке записей. Когда список заполнен, новые записи добавляются с начала списка, образуя тем самым кольцо. Количество записей для заполненного списка всегда будет возвращаться всегда одно и то же – 400 (или 200).

Командой **GETCURINDEX** запрашивается индекс последнего заполненного элемента списка событий.

При установке времени счетчика, переводе часов, потере времени из-за разряженной батареи резервного питания в профиле мощности могут появляться пропуски или появляться дублирующие записи с одинаковым временем.

В выключенном состоянии новые записи в списке событий не формируются.

Начиная с версии ПО 1.09 включительно, фиксирование события «Коррекция времени» производится по следующему алгоритму: При подаче команды **SETRTC** («Установить часы реального времени») счетчик в списке **messages** сформирует две записи:

1. Запись «Нет сообщений» с текущей датой/временем счетчика до коррекции.
2. Запись «Коррекция времени» с новой датой/временем после коррекции.

На основе этих двух записей, в ПО верхнего уровня может быть вычислена разница времени между временем счетчика до коррекции и после коррекции.

В более ранних версиях ПО (до версии 1.09) при подаче команды **SETRTC** формируется только запись «Коррекция времени» с новой датой/временем после коррекции.

4.1.15. Интерфейсный объект «Калибровка часов реального времени»

Интерфейсный объект «Калибровка часов реального времени» предназначен для коррекции хода часов. Длина объекта – 2 байта. Калибровочный коэффициент должен вычисляться по формуле:

$$k = 1048576 * \left(1 - \frac{4095,5}{f_{изм}} \right), \text{ где}$$

$f_{изм}$ – частота часового кварцевого резонатора, измеренная частотомером на импульсном выходе счётчика. Для перевода импульсного выхода счетчика в режим формирования часовых импульсов, нужно изменить режим импульсного выхода. См. п.п.4.1.23.

Доступ к интерфейсному объекту «Калибровка часов реального времени» возможен только с установленным джампером на печатной плате счётчика. В случае отсутствия джампера, счётчик вернёт ошибку **ACCESS_DENIED**.

Если джампер установлен на выключенном счётчике, то при включении счётчика будет только разрешён доступ к объекту. Если джампер установлен на включённом счётчике, будет однократно выполнена функция «Заводские установки» и разрешён доступ к объекту.

4.1.16. Интерфейсный объект «Время интегрирования профиля мощности»

Объект «Время интегрирования профиля мощности» задаёт время в минутах, через которое формируются записи в списке профиля мощности. Предусмотрены следующие значения времени интегрирования: 5 мин, 10 мин, 15 мин, 20 мин, 30 мин. По умолчанию «Заводская установка» время интегрирования 30 мин. При попытке установить в объект значения, отличающиеся от перечисленных выше, счётчик сформирует исключение **ILLEGAL_DATA_VALUE**.

При изменении значения объекта, счётчик автоматически пересчитывает значение порога автоматического отключения нагрузки в импульсах. Для более подробной информации о пересчёте порога отключения нагрузки см п.п.4.1.25.

4.1.17. Интерфейсный объект «Параметры калибровки»

Объект «Параметры калибровки» содержит следующие калибровочные коэффициенты:

Индекс	Обозначение	Наименование	Диапазон калибровки
0	AVRMSGAIN	Коэффициент передачи по напряжению. Фаза А	
1	BVRMSGAIN	Коэффициент передачи по напряжению. Фаза В	
2	CVRMSGAIN	Коэффициент передачи по напряжению. Фаза С	
3	AIRMSGAIN	Коэффициент передачи по току. Фаза А	
4	BIRMSGAIN	Коэффициент передачи по току. Фаза В	
5	CIRMSGAIN	Коэффициент передачи по току. Фаза С	
6	AWATTGAIN	Коэффициент передачи по активной мощности. Фаза А.	
7	BWATTGAIN	Коэффициент передачи по активной мощности. Фаза В.	
8	CWATTGAIN	Коэффициент передачи по активной мощности. Фаза С.	
9	AVARGAIN	Коэффициент передачи по реактивной мощности. Фаза А.	
10	BVARGAIN	Коэффициент передачи по реактивной мощности. Фаза В.	
11	CVARGAIN	Коэффициент передачи по реактивной мощности. Фаза С.	
12	AVAGAIN	Коэффициент передачи по полной мощности. Фаза А.	
13	BVAGAIN	Коэффициент передачи по полной мощности. Фаза В.	
14	CVAGAIN	Коэффициент передачи по полной мощности. Фаза С.	
15	APHCAL	Коэффициент сдвига фазы. Фаза А.	
16	BPHCAL	Коэффициент сдвига фазы. Фаза В.	
17	CPHCAL	Коэффициент сдвига фазы. Фаза С.	
18	AVRMSOS	Коэффициент смещения по напряжению. Фаза А.	
19	BVRMSOS	Коэффициент смещения по напряжению. Фаза В.	
20	CVRMSOS	Коэффициент смещения по напряжению. Фаза С.	
21	AIRMSOS	Коэффициент смещения по току. Фаза А.	
22	BIRMSOS	Коэффициент смещения по току. Фаза В.	
23	CIRMSOS	Коэффициент смещения по току. Фаза С.	
24	AWATTOS	Коэффициент смещения по активной мощности. Фаза А.	
25	BWATTOS	Коэффициент смещения по активной мощности. Фаза В.	
26	CWATTOS	Коэффициент смещения по активной мощности. Фаза С.	
27	AVAROS	Коэффициент смещения по реактивной мощности. Фаза А.	
28	BVAROS	Коэффициент смещения по реактивной мощности. Фаза В.	
29	CVAROS	Коэффициент смещения по реактивной мощности. Фаза С.	
30	AVAOS	Коэффициент смещения по полной мощности. Фаза А.	
31	BVAOS	Коэффициент смещения по полной мощности. Фаза В.	
32	CVAOS	Коэффициент смещения по полной мощности. Фаза С.	
33	STUPC	Ограничение стартового тока	

Каждый калибровочный коэффициент имеет длину 4 байта. После записи калибровочного коэффициента в энергонезависимую память, автоматически вызывается процедура загрузки калибровочного коэффициента в измерительный модуль.

Калибровочные коэффициенты должны рассчитываться следующим образом:

1. Калибровка коэффициента передачи по напряжению.

Калибровка коэффициента передачи по напряжению должна производиться при номинальном напряжении $U_{ном} = 230В$ для счетчиков Милур 305.32, Милур 305.12 и $U_{ном} = 57,7В$ для счетчика Милур 305.11.

Коэффициент калибровки коэффициента передачи по напряжению для фазы n вычисляется по формуле:

$$nVRMSGAIN = \frac{U_{NOM_n} - U_{TEST_n}}{U_{TEST_n}} \cdot 1562, \text{ где}$$

$nVRMSGAIN$ – Калибровочный коэффициент передачи по напряжению по фазе n.

- U_{NOM_n} – номинальное напряжение по фазе n, установленное на калибровочной установке;
 U_{TEST_n} – измеренное напряжение по фазе n на калибруемом счётчике до калибровки;
n – индекс фазы, принимает значения А, В, С.

2. Калибровка коэффициента передачи по току.

Калибровка коэффициента передачи по току должна производиться при номинальном напряжении $U_{НОМ} = 230$ В и токе $I = 40$ А для счетчика Милур 305.32, при номинальном напряжении $U_{НОМ} = 230$ В и токе $I = 10$ А для счетчика Милур 305.12, при номинальном напряжении $U_{НОМ} = 57,7$ В и токе $I = 10$ А для счетчика Милур 305.11.

Коэффициент калибровки коэффициента передачи по току для фазы n вычисляется по формуле:

$$nIRMSGAIN = \frac{(I_{NOM_n} - I_{TEST_n})}{I_{TEST_n}} \cdot 195, \text{ где}$$

$nIRMSGAIN$ – Калибровочный коэффициент передачи по току по фазе n.

- I_{NOM_n} – ток по фазе n, установленный на калибровочной установке;
 I_{TEST_n} – измеренный ток по фазе n на калибруемом счётчике до калибровки;
n – индекс фазы, принимает значения А, В, С.

3. Калибровка коэффициента передачи по активной мощности.

Калибровка коэффициента передачи по активной мощности должна производиться при номинальном напряжении $U_{НОМ} = 230$ В и токе $I = 40$ А для счетчика Милур 305.32, при номинальном напряжении $U_{НОМ} = 230$ В и токе $I = 10$ А для счетчика Милур 305.12, при номинальном напряжении $U_{НОМ} = 57,7$ В и токе $I = 10$ А для счетчика Милур 305.11. Фактор мощности должен быть равен 1 ($\cos \varphi = 1$, $\varphi = 0^\circ$). Коэффициент смещения активной мощности ($nWATTOS$) должен быть равен нулю.

Коэффициент калибровки коэффициента передачи по активной мощности для фазы n вычисляется по формуле:

$$nWATTGAIN = \frac{(P_{NOM_n} - P_{TEST_n})}{P_{TEST_n}} \cdot 5000, \text{ где}$$

$nWATTGAIN$ – Калибровочный коэффициент передачи по активной мощности по фазе n.

- P_{NOM_n} – активная мощность по фазе n, установленная на калибровочной установке;
 P_{TEST_n} – активная мощность по фазе n на калибруемом счётчике до калибровки;
n – индекс фазы, принимает значения А, В, С.

4. Калибровка коэффициента передачи по полной мощности.

Калибровка коэффициента передачи по полной мощности должна производиться при номинальном напряжении $U_{НОМ} = 230$ В и токе $I = 40$ А для счетчика Милур 305.32, при номинальном напряжении $U_{НОМ} = 230$ В и токе $I = 10$ А для счетчика Милур 305.12, при номинальном напряжении $U_{НОМ} = 57,7$ В и токе $I = 10$ А для счетчика Милур 305.11.

Коэффициент калибровки коэффициента передачи по полной мощности для фазы n вычисляется по той же формуле, что и коэффициент передачи для активной мощности.

5. Калибровка смещения по напряжению.

Калибровка коэффициента передачи по напряжению должна производиться при номинальном напряжении $U_{НОМ} = 160$ В для счетчиков Милур 305.32, Милур 305.12 и $U_{НОМ} = 45$ В для счетчика Милур 305.11.

Коэффициент калибровки коэффициента передачи по напряжению для фазы n вычисляется по формуле:

$$nVRMSOS = (U_{NOM_n} - U_{TEST_n}), \text{ где}$$

$nVRMSOS$ – Калибровочный коэффициент смещения по напряжению по фазе n .

U_{NOM_n} – номинальное напряжение по фазе n , установленное на калибровочной установке, (мВ);

U_{TEST_n} – измеренное напряжение по фазе n на калибруемом счётчике до калибровки, (мВ);

n – индекс фазы, принимает значения А, В, С.

6. Калибровка смещения по току.

Калибровка коэффициента смещения по току должна производиться при номинальном напряжении $U_{НОМ} = 230$ В и токе $I = 0,5$ А для счетчика Милур 305.32, при номинальном напряжении $U_{НОМ} = 230$ В и токе $I = 0,1$ А для счетчика Милур 305.12, при номинальном напряжении $U_{НОМ} = 57,7$ В и токе $I = 0,1$ А для счетчика Милур 305.11.

Коэффициент калибровки коэффициента смещения по току для фазы n вычисляется по формуле:

$$nIRMSOS = (I_{NOM_n} - I_{TEST_n}), \text{ где}$$

$nIRMSOS$ – Калибровочный коэффициент смещения по току по фазе n .

I_{NOM_n} – ток по фазе n , установленный на калибровочной установке, (мА);

I_{TEST_n} – ток по фазе n на калибруемом счётчике до калибровки, (мА);

n – индекс фазы, принимает значения А, В, С.

7. Калибровка смещения по активной мощности.

Калибровка коэффициента смещения по активной мощности должна производиться при номинальном напряжении $U_{НОМ} = 230$ В и токе $I = 0,5$ А для счетчика Милур 305.32, при номинальном напряжении $U_{НОМ} = 230$ В и токе $I = 0,1$ А для счетчика Милур 305.12, при номинальном напряжении $U_{НОМ} = 57,7$ В и токе $I = 0,1$ А для счетчика Милур 305.11. Фактор мощности должен быть равен 1 ($\cos \varphi = 1$, $\varphi = 0^\circ$).

Коэффициент калибровки коэффициента смещения по активной мощности для фазы n вычисляется по формуле:

$$nWATTOS = (P_{NOM_n} - P_{TEST_n}) \cdot 50, \text{ где}$$

$nWATTOS$ – Калибровочный коэффициент смещения по активной мощности по фазе n .

$nWATTGAIN$ – Калибровочный коэффициент передачи по активной мощности по фазе n .

P_{NOM_n} – активная мощность по фазе n , установленная на калибровочной установке, (мВт);

P_{TEST_n} – активная мощность по фазе n на калибруемом счётчике до калибровки, (мВт);

n – индекс фазы, принимает значения А, В, С.

8. Калибровка смещения по полной мощности.

Калибровка коэффициента смещения по полной мощности должна производиться при номинальном напряжении $U_{НОМ} = 230$ В и токе $I = 0,5$ А для счетчика Милур 305.32, при номинальном напряжении $U_{НОМ} = 230$ В и токе $I = 0,1$ А для счетчика Милур 305.12, при номинальном напряжении $U_{НОМ} = 57,7$ В и токе $I = 0,1$ А для счетчика Милур 305.11.

Коэффициент калибровки коэффициента смещения по полной мощности для фазы n вычисляется по той же формуле, что и коэффициент смещения для активной мощности.

9. Калибровка сдвига фазы.

Калибровка коэффициента сдвига фазы между током и напряжением должна производиться при номинальном напряжении $U_{НОМ} = 230$ В и токе $I = 40$ А для счетчика Милур 305.32, при номинальном напряжении $U_{НОМ} = 230$ В и токе $I = 10$ А для счетчика Милур 305.12, при

номинальном напряжении $U_{\text{НОМ}} = 57,7 \text{ В}$ и токе $I = 10 \text{ А}$ для счетчика Милур 305.11. Фактор мощности должен быть равен 0,5 ($\cos \varphi = 0,5$, $\varphi = 60^\circ$).

Коэффициент калибровки коэффициента сдвига фазы для фазы n вычисляется по формуле:

$$nPHCAL = \frac{(P_{NOM_n} - P_{TEST_n})}{0.00054 P_{TEST_n}} + 128$$

$nPHCAL$ – Калибровочный коэффициент сдвига фазы n .

P_{NOM_n} – активная мощность по фазе n , установленная на калибровочной установке, (мВт);

P_{TEST_n} – активная мощность по фазе n на калибруемом счётчике до калибровки, (мВт);

n – индекс фазы, принимает значения А, В, С.

10. Калибровка коэффициента передачи по реактивной мощности.

Калибровка коэффициента передачи по реактивной мощности должна производиться при номинальном напряжении $U_{\text{НОМ}} = 230 \text{ В}$ и токе $I = 40 \text{ А}$ для счетчика Милур 305.32, при номинальном напряжении $U_{\text{НОМ}} = 230 \text{ В}$ и токе $I = 10 \text{ А}$ для счетчика Милур 305.12, при номинальном напряжении $U_{\text{НОМ}} = 57,7 \text{ В}$ и токе $I = 10 \text{ А}$ для счетчика Милур 305.11. Фактор мощности должен быть равен 0 ($\cos \varphi = 0$, $\varphi = 90^\circ$). Коэффициент смещения реактивной мощности ($nVAROS$) должен быть равен нулю.

Коэффициент калибровки коэффициента передачи по реактивной мощности для фазы n вычисляется по той же формуле, что и коэффициент передачи для активной мощности.

11. Калибровка смещения по реактивной мощности.

Калибровка коэффициента смещения по реактивной мощности должна производиться при номинальном напряжении $U_{\text{НОМ}} = 230 \text{ В}$ и токе $I = 0,5 \text{ А}$ для счетчика Милур 305.32, при номинальном напряжении $U_{\text{НОМ}} = 230 \text{ В}$ и токе $I = 0,1 \text{ А}$ для счетчика Милур 305.12, при номинальном напряжении $U_{\text{НОМ}} = 57,7 \text{ В}$ и токе $I = 0,1 \text{ А}$ для счетчика Милур 305.11. Фактор мощности должен быть равен 0 ($\cos \varphi = 0$, $\varphi = 90^\circ$).

Коэффициент калибровки коэффициента смещения по реактивной мощности для фазы n вычисляется по той же формуле, что и коэффициент смещения для активной мощности.

12. Калибровка «Стартовый ток».

Коэффициент калибровки «Стартовый ток» предназначен для предотвращения самохода счётчика. Физически, этот коэффициент калибровки представляет собой пороговое значение мощности (в единицах измерителя), ниже которого счётчик не должен учитывать энергию. Значение коэффициента калибровки «Стартовый ток» должно быть:

- 20 мА для счетчиков Милур 305.32
- 10 мА для счетчиков Милур 305.12.
- 5 мА для счетчика Милур 305.11.

Начальное значение калибровочного коэффициента «Стартовый ток» перед калибровкой должно быть установлено 0x00000000.

Доступ к интерфейсному объекту «Параметры калибровки» возможен только с установленным джампером на печатной плате счётчика. В случае отсутствия джампера, счётчик вернёт ошибку ACCESS_DENIED.

Если джампер установлен на выключенном счётчике, то при включении счётчика будет только разрешён доступ к объекту. Если джампер установлен на включённом счётчике, будет однократно выполнена функция «Заводские установки» и разрешён доступ к объекту.

4.1.18. Интерфейсный объект «Управление автоматическим переходом на летнее/зимнее время»

Объект «Управление автоматическим переходом на летнее/зимнее время» содержит один байт. Для включения автоматического перехода необходимо записать значение value=1. Для отключения автоматического перехода записать value=0. Данные объекта хранятся в энергонезависимой памяти.

В счётчике введено «Поясное время» и «Сезонное время». Поясное время – это время, принятое в данном часовом поясе. Сезонное (летнее/зимнее) время может отличаться от поясного времени, если установлен флаг автоматического перехода на летнее/зимнее время. Зимнее время в счётчике совпадает с поясным временем. Летнее время на один час опережает поясное время.

Основные часы счётчика всегда соответствуют поясному времени. Дата и время в записях срезов мощности соответствуют поясному времени. Это позволяет избежать пропусков и образования дублирующихся записей в срезах мощности при переходе на летнее/зимнее время.

Время, показываемое на ЖК индикаторе и по интерфейсу RS-485, соответствует сезонному времени. Дата и время в записях списков событий соответствуют сезонному времени. При отключении автоматического перехода, сезонное время полностью совпадает с поясным временем.

4.1.19. Исключён. См. журнал замечаний п.п.6.6.

4.1.20. Интерфейсный объект «Напряжение батареи резервного питания»

Интерфейсный объект «Напряжение батареи резервного питания» содержит два байта. Измерение напряжения производится однократно при включении счётчика, а затем один раз в сутки. Значение напряжения батареи истинно через три секунды после начала измерения. Значение напряжения батареи имеет оценочный характер. Значение 0x0E81 будет соответствовать напряжению 3,713 Вольта. На ЖК индикаторе счётчика будет отображаться значение 3,71 Вольта.

4.1.21. Интерфейсный объект «Максимальное число тарифов»

Интерфейсный объект «Максимальное число тарифов» содержит один байт. Значение объекта [0..7] определяет максимальное число тарифов, с которым работает счётчик. Значение «0» соответствует одностарифному варианту, значение «1»- соответствует двухтарифному счетчику. Значение «7»- соответствует 8-тарифному счетчику. При попытке записать в объект число большее, чем «7», счетчик вернет ILLEGAL_DATA_VALUE. С этим параметром должно быть согласовано тарифное расписание, записываемое в счетчик. Если в объекте «Максимальное число тарифов» задано «1», а в тарифном расписании указано переключение на тариф «3», данное переключение производится не будет. Счетчик останется на прежнем тарифе. По умолчанию, при сбросе на заводские установки, в интерфейсный объект записывается число «7», что соответствует 8-тарифному учёту энергии.

4.1.22. Интерфейсный объект «Цифровой идентификатор ПО»

Интерфейсный объект «Цифровой идентификатор ПО» содержит два байта. В интерфейсном объекте содержится Цифровой идентификатор программного обеспечения (контрольная сумма исполняемого кода) в соответствии с требованиями МИ 3286-2010.

4.1.23. Интерфейсный объект «Режим импульсного выхода счётчика»

Интерфейсный объект «Режим импульсного выхода счётчика» содержит один байт. Значение объекта определяет текущий режим импульсного выхода счётчика. Режим импульсного выхода сохраняется в энергонезависимой памяти. При включении счётчика режим импульсного выхода восстанавливается из энергонезависимой памяти. Предусмотрены следующие режимы импульсного выхода:

Значение объекта	Наименование режима
0	Импульсный выход для импортируемой энергии.
1	Импульсный выход. Режим проверки для импортируемой энергии
2	Управление нагрузкой. Нагрузка постоянно включена. При записи этого

	значения, нагрузка включается. Проверка превышения порога автоматического управления нагрузкой не производится .
3	Управление нагрузкой. Нагрузка постоянно выключена. При записи этого значения, нагрузка отключается. Проверка превышения порога автоматического управления нагрузкой не производится .
4	Управление нагрузкой. Автоматическое управление. При записи этого значения, нагрузка включается и один раз в минуту производится проверка превышения порога автоматического управления нагрузкой.
5	Импульсы кварцевого резонатора (32768 / 8) Гц
6	Импульсы кварцевого резонатора 1 Гц

4.1.24. Интерфейсный объект «Тип выхода управления нагрузкой»

Интерфейсный объект «Тип выхода управления нагрузкой» содержит один байт. Значение объекта определяет тип управления нагрузкой. По умолчанию (заводская установка) при включённой нагрузке цепь управления нагрузкой нормально разомкнута. Если логика управления нагрузкой должна быть инверсная, нужно переключить объект в другой режим, где при включённой нагрузке цепь управления нагрузкой нормально замкнута.

Значение объекта	Наименование режима
0	Нагрузка включена. Фототранзистор оптрона закрыт. Цепь управления нагрузкой разомкнута. Светодиод счётчика светится.
1	Нагрузка включена. Фототранзистор оптрона открыт. Цепь управления нагрузкой замкнута. Светодиод счётчика погашен.

4.1.25. Интерфейсный объект «Порог автоматического отключения нагрузки»

Интерфейсный объект «Порог автоматического отключения нагрузки» содержит два байта. Значение объекта определяет пороговое значение мощности, усреднённой за время интегрирования профиля мощности. Значение объекта должно задаваться в Ваттах. Если нужно задать среднюю мощность 21600 Вт, в объект должно быть записано число 0x5460. При изменении значения задаваемого порога или при изменении времени интегрирования профиля мощности, счётчик автоматически пересчитывает порог автоматического отключения нагрузки в импульсах. При превышении порогового значения числа импульсов, нагрузка будет отключена в текущем интервале профиля мощности. При автоматическом отключении нагрузки, повторное включение нагрузки будет возможно только по окончании текущего интервала профиля мощности (в следующей получасовке). Проверка превышения порога мощности производится один раз в минуту. Включение нагрузки должно производиться по интерфейсу, путём записи нужного значения в объект «Режим импульсного выхода счётчика».

Для модели счётчика «Милур 305.32» пороговое значение отключения нагрузки может превышать 2 байта, поэтому для данной модели в счетчик должно записываться значение порога, уменьшенное в 10 раз. Например, если задан порог отключения нагрузки 70 000 Вт (0x11170), в данный объект счётчика модели «Милур 305.32» должно быть записано значение 6 000 (0x1770). Для остальных моделей счётчика «Милур 305», в объект должно быть записано значение, соответствующее значению в Ваттах.

4.1.26. Интерфейсный объект «Энергия в суточных интервалах», «Энергия в месячных интервалах»

Интерфейсные объекты «Энергия в суточных интервалах», «Энергия в месячных интервалах» представляют собой кольцевой список, заполняемый по событиям «Начало новых суток», «Начало нового месяца».

Для версий ПО счетчика с 0100 по 0199 включительно, Элемент списка «Энергия в интервалах» представляет собой следующую структуру:

```
typedef struct{
    BYTE seconds;    // Секунды (0..59)
    BYTE minutes;   // Минуты (0..59)
    BYTE hours;     // Часы (0...12)
    BYTE day;       // День месяца (1...31)
    BYTE month;    // Месяц (1...12)
    BYTE year;     // Год (00...99)
    BYTE energy[72]; // Энергия (72 байта)
} ENTARRecord_t;
```

Для версий ПО счетчика с 0200 по 0299 включительно, Элемент списка «Энергия в интервалах» представляет собой следующую структуру:

```
typedef struct{
    BYTE seconds;    // Секунды (0..59)
    BYTE minutes;   // Минуты (0..59)
    BYTE hours;     // Часы (0...12)
    BYTE day;       // День месяца (1...31)
    BYTE month;    // Месяц (1...12)
    BYTE year;     // Год (00...99)
    BYTE energy[144]; // Энергия (144 байта)
} ENTARRecord_t;
```

Для версий ПО счетчика с 0100 по 0199 включительно, элемент структуры «energy» содержит 72 байта, в которые записываются: значение суммарной активной энергии по всем тарифам, активной энергии по тарифу 1, активной энергии по тарифу 2, активной энергии по тарифу 3, активной энергии по тарифу 4, активной энергии по тарифу 5, активной энергии по тарифу 6, активной энергии по тарифу 7, активной энергии по тарифу 8, суммарной реактивной энергии по всем тарифам, реактивной энергии по тарифу 1, реактивной энергии по тарифу 2, реактивной энергии по тарифу 3, реактивной энергии по тарифу 4, реактивной энергии по тарифу 5, реактивной энергии по тарифу 6, реактивной энергии по тарифу 7, реактивной энергии по тарифу 8.

Для версий ПО счетчика с 0200 по 0299 включительно, элемент структуры «energy» содержит 144 байта, в которые записываются значения: суммарной активной импортируемой и экспортируемой энергии по всем тарифам, активной импортируемой и экспортируемой энергии по тарифу 1, активной импортируемой и экспортируемой энергии по тарифу 2, активной импортируемой и экспортируемой энергии по тарифу 3, активной импортируемой и экспортируемой энергии по тарифу 4, активной импортируемой и экспортируемой энергии по тарифу 5, активной импортируемой и экспортируемой энергии по тарифу 6, активной импортируемой и экспортируемой энергии по тарифу 7, активной импортируемой и экспортируемой энергии по тарифу 8, суммарной реактивной импортируемой и экспортируемой энергии по всем тарифам, реактивной импортируемой и экспортируемой энергии по тарифу 1, реактивной импортируемой и экспортируемой энергии по тарифу 2, реактивной импортируемой и экспортируемой энергии по тарифу 3, реактивной импортируемой и экспортируемой энергии по тарифу 4, реактивной импортируемой и экспортируемой энергии по тарифу 5, реактивной импортируемой и экспортируемой энергии по тарифу 6, реактивной импортируемой и экспортируемой энергии по тарифу 7, реактивной импортируемой и экспортируемой энергии по тарифу 8.

Значение энергий выводится в упакованном BCD-формате, младший байт вперед, младшая тетрада вперед. Каждая значащая цифра энергии располагается в своей тетраде.

Максимальное число записей в списке «Энергия в суточных интервалах» соответствует 35 записей для версии ПО ниже, чем 106. Начиная с версии 106 включительно, максимальное число записей в списке «Энергия в суточных интервалах» соответствует 123 записям.

Размерность значения энергии зависит от модели счётчика. Модель счётчика указывается в объекте «Информация об устройстве» (ID 32).

Для модификаций счётчиков «Милур 305.11», «Милур 305.12» размерность значений энергии составляет 0,001 кВт*ч (квар*ч). Например, значение 85 10 00 00 означает значение энергии 0,158 кВт*ч (квар*ч).

Для модификаций счётчиков «Милур 305.32» размерность значений энергии составляет 0,01 кВт*ч (квар*ч). Например, значение 35 10 00 00 означает значение энергии 1,53 кВт*ч (квар*ч).

Для версий ПО счетчика с 0100 по 0199 включительно, структура элемента «energy» следующая:

Смещение, байт	Наименование параметра	Длина параметра, байт
0	Суммарная активная энергия	4
4	Активная энергия по тарифу 1	4
8	Активная энергия по тарифу 2	4
12	Активная энергия по тарифу 3	4
16	Активная энергия по тарифу 4	4
20	Активная энергия по тарифу 5	4
24	Активная энергия по тарифу 6	4
28	Активная энергия по тарифу 7	4
32	Активная энергия по тарифу 8	4
36	Суммарная реактивная энергия	4
40	Реактивная энергия по тарифу 1	4
44	Реактивная энергия по тарифу 2	4
48	Реактивная энергия по тарифу 3	4
52	Реактивная энергия по тарифу 4	4
56	Реактивная энергия по тарифу 5	4
60	Реактивная энергия по тарифу 6	4
64	Реактивная энергия по тарифу 7	4
68	Реактивная энергия по тарифу 8	4

Для версий ПО счетчика с 0200 по 0299 включительно, структура элемента «energy» следующая:

Смещение, байт	Наименование параметра	Длина параметра, байт
0	Суммарная активная импортируемая энергия	4
4	Активная импортируемая энергия по тарифу 1	4
8	Активная импортируемая энергия по тарифу 2	4
12	Активная импортируемая энергия по тарифу 3	4
16	Активная импортируемая энергия по тарифу 4	4
20	Активная импортируемая энергия по тарифу 5	4
24	Активная импортируемая энергия по тарифу 6	4
28	Активная импортируемая энергия по тарифу 7	4
32	Активная импортируемая энергия по тарифу 8	4
36	Суммарная активная экспортируемая энергия	4
40	Активная экспортируемая энергия по тарифу 1	4
44	Активная экспортируемая энергия по тарифу 2	4
48	Активная экспортируемая энергия по тарифу 3	4
52	Активная экспортируемая энергия по тарифу 4	4
56	Активная экспортируемая энергия по тарифу 5	4

60	Активная экспортируемая энергия по тарифу 6	4
64	Активная экспортируемая энергия по тарифу 7	4
68	Активная экспортируемая энергия по тарифу 8	4
72	Суммарная импортируемая реактивная энергия	4
76	Реактивная импортируемая энергия по тарифу 1	4
80	Реактивная импортируемая энергия по тарифу 2	4
84	Реактивная импортируемая энергия по тарифу 3	4
88	Реактивная импортируемая энергия по тарифу 4	4
92	Реактивная импортируемая энергия по тарифу 5	4
96	Реактивная импортируемая энергия по тарифу 6	4
100	Реактивная импортируемая энергия по тарифу 7	4
104	Реактивная импортируемая энергия по тарифу 8	4
108	Суммарная экспортируемая реактивная энергия	4
112	Реактивная экспортируемая энергия по тарифу 1	4
116	Реактивная экспортируемая энергия по тарифу 2	4
120	Реактивная экспортируемая энергия по тарифу 3	4
124	Реактивная экспортируемая энергия по тарифу 4	4
128	Реактивная экспортируемая энергия по тарифу 5	4
132	Реактивная экспортируемая энергия по тарифу 6	4
136	Реактивная экспортируемая энергия по тарифу 7	4
140	Реактивная экспортируемая энергия по тарифу 8	4

Управление функциональностью объектов «Энергия в интервалах» осуществляется посредством команд **LISTINIT**, **GETLISTNE**, **GET_ENTALIST**.

Для работы со списками, необходимо сначала выставить время в счетчике, а затем проинициализировать их командой **LISTINIT**.

После успешной инициализации список будет содержать одну запись, в которой будет содержаться значение энергий на момент инициализации списка.

Командами **GET_ENTALIST** с указанием объекта и параметром порядкового номера записи можно получить содержимое одной записи списка в формате **ENTARcord_t**.

Максимальное количество записей в списке «Энергия в суточных интервалах» для версии ПО от 0100 до 0105 включительно – 35. Максимальное количество записей в списке «Энергия в суточных интервалах» для версии ПО, начиная с версии 0106 – 123. Максимальное количество записей в списке «Энергия в месячных интервалах» – 12. Номер запрашиваемой записи в команде **GET_ENTALIST** должен быть в диапазоне от 0 до 34(122) и от 0 до 11 соответственно. В противном случае будет возвращен код ошибки.

Командой **GETLISTNE** запрашивается общее количество имеющихся в списке записей. Когда список заполнен, новые записи добавляются с начала списка, образуя тем самым кольцо. Количество записей для заполненного списка всегда будет возвращаться всегда одно и то же – 34 и 12 соответственно.

Командой **GETCURINDEX** запрашивается индекс последнего заполненного элемента списка «Энергия в интервалах».

При установке времени счетчика, переводе часов, потере времени из-за разряженной батареи резервного питания в списке могут появляться пропуски или появляться дублирующие записи с одинаковым временем.

В выключенном состоянии новые записи в списке не формируются.

При включении счетчика, если день или месяц последней записи, не совпадают с текущими, в списке формируется новая запись с датой, временем, энергиями на момент включения счетчика.

4.1.27. Интерфейсный объект «Серийный номер счетчика»

Интерфейсный объект представляет собой строку из 15 байт. Каждый байт соответствует символу в соответствии со стандартной Win-кодировкой. Строка серийного номера "131040000025896" будет представлена в виде последовательности: 31 33 31 30 34 30 30 30 30 32 35 38 39 36 в шестнадцатеричном коде.

Начиная с версии 2.00 включительно, серийный номер счетчика может быть использован для работы со счётчиком по интерфейсу RS-485 или через оптопорт. Для обращения у счётчику по серийному номеру, ПО верхнего уровня должно из серийного номера сформировать 4-байтовый адрес по следующему алгоритму:

Байты	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
пример	1	3	1	0	4	0	0	0	1	2	3	4	5	6	7
формат	y	y	m	m	m	n	n	n	n	n	n	n	n	n	n

y - год, когда был произведён счетчик (2 байта);
m - модель счетчика (3 байта);
n - номер, используемый для вычисления 4-байтового адреса (10 байт).
Для вычисления 4-байтового адреса используется 10-байтовый номер,

Пример вычисления 4-байтового адреса:

Дан серийный номер: «131040001234567». При считывании из счетчика серийного номера будет получена следующая последовательность: 31 33 31 30 34 30 30 30 31 32 33 34 35 36 37. Для формирования адреса необходимо перевести часть строки «30 30 30 31 32 33 34 35 36 37» в число. Получим десятичное число 1234567. 4-байтовый адрес при обращении к счетчику будет следующий: 0x0012D687. Структура фрейма при 4-байтовой адресации описана в п.п.3.6 настоящего протокола.

4.1.28. Интерфейсный объект «Серийный номер печатного узла»

Интерфейсный объект представляет собой строку из 15 байт. Каждый байт соответствует символу в соответствии со стандартной ASCII-кодировкой. Строка серийного номера "131040000025896" будет представлена в виде последовательности: 31 33 31 30 34 30 30 30 30 30 32 35 38 39 36 в шестнадцатеричном коде. Объект используется для версий ПО счётчика, начиная с 3.00 включительно.

4.1.29. Интерфейсный объект «Таймаут ответа счетчика»

Интерфейсный объект содержит 1 байт, задающий время задержки ответа счетчика в секундах. Обработка входного фрейма и выполнение присланной команды производится счётчиком сразу. Посылка ответного фрейма производится с задержкой на заданное число секунд. Если интерфейсный объект содержит значение 0x00, ответный фрейм отсылается без задержки. Значение 0x01- соответствует задержке в 1 сек, значение 0x02- 2 сек и т.д.

4.1.30. Интерфейсный метод «Поиск записи в срезах мощности»

Интерфейсный метод «Поиск записи в срезах мощности», представляет собой метод предназначенный для выполнения операции поиска записи в кольцевом списке «Срезы мощности» по заданным дате и времени. Дата и время должны быть заданы в качестве аргументов метода:

PWILIST_SEARCH (BYTE ObjUID, BYTE minutes, BYTE hour, BYTE mday, BYTE month, BYTE year);

Вызов метода запускает процесс поиска. Поиск записи выполняется в фоновом режиме и может занимать продолжительное время. Во время выполнения операции поиска счетчик продолжает полнофункциональную работу. Также во время поиска работает интерфейс. После завершения операции поиска, результаты поиска доступны для считывания. Считывание результатов поиска производится при помощи повторного вызова того же метода с теми же аргументами. Для версий ПО счетчика с 0100 по 0199 включительно, метод возвращает следующую структуру данных:

```
typedef struct
{
    pwisearch_t search_result; // Код результата поиска;
    unsigned char index[2]; // Индекс найденной записи в списке;
```

```

unsigned char pwr[4];           // Суммарная средняя активная мощность;
unsigned char pwrQ[4];         // Суммарная средняя реактивная мощность;
unsigned char incomplate;      // Признак неполного интервала найденной записи;
} pwisearch_result_t;

```

Для версий ПО счетчика с 0200 по 0299 включительно, метод возвращает следующую структуру данных:

```

typedef struct
{
    pwisearch_t search_result; // Код результата поиска;
    unsigned char index[2];    // Индекс найденной записи в списке;
    unsigned char pwrPI[4];    // Суммарная средняя активная импортируемая мощность;
    unsigned char pwrPE[4];    // Суммарная средняя активная экспортируемая мощность;
    unsigned char pwrQI[4];    // Суммарная средняя реактивная импортируемая мощность;
    unsigned char pwrQE[4];    // Суммарная средняя реактивная экспортируемая мощность;
    unsigned char incomplate;  // Признак неполного интервала найденной записи;
} pwisearch_result_t;

```

Код результата поиска может принимать следующие значения:

```

typedef enum
{
    PWISEARCH_OK = 0,           // Запись найдена;
    PWISEARCH_ACK,             // Запрос на поиск принят. Начат поиск;
    PWISEARCH_IN_PROGRESS,    // Процесс поиска записи активен;
    PWISEARCH_NOT_FOUND,      // Запись не найдена;
    PWISEARCH_LIST_EMPTY,     // Список срезов мощности пустой;
    PWISEARCH_ERROR,          // Ошибка при работе со списком срезов мощности;
} pwisearch_t;

```

При вызове метода поиска счетчик вернет PWISEARCH_ACK. Это означает, что запрос на поиск принят и начат процесс поиска записи. Если счетчик вернул PWISEARCH_LIST_EMPTY, это означает, что список пустой и поиск невозможен. При этом остальные данные структуры будут равны нулю.

При повторном вызове метода счетчик может вернуть:

- PWISEARCH_OK, если поиск завершен и запись найдена;
- PWISEARCH_IN_PROGRESS, если поиск активен и ещё не завершен;
- PWISEARCH_NOT_FOUND, если поиск завершен и запись не найдена;
- PWISEARCH_ERROR, если во время поиска обнаружена ошибка при работе со списком срезов мощности.

Если счетчик вернул PWISEARCH_OK, в остальных членах структуры pwisearch_result_t находятся искомые данные.

Если счетчик вернул любое другое значение, кроме PWISEARCH_OK, остальные члены структуры pwisearch_result_t будут нулевыми.

Если счетчик вернул PWISEARCH_IN_PROGRESS, это означает, что поиск не закончен и необходимо повторить вызов метода через какое-то время. Время поиска зависит от глубины списка и может занимать продолжительное время. Поиск записи производится от записи с индексом 0 до записи с индексом n-1 (от старых к новым), где n- глубина списка срезов мощности.

Для подробной информации о срезах мощности см п.п.4.1.12.

4.1.31. Интерфейсный объект «Версия метрологически значимой части ПО»

Интерфейсный объект содержит 4 байта. Структура объекта идентична объекту «Версия встроенного ПО» с идентификатором ID = 33. См. п.п.4.1.11. Данный интерфейсный объект предназначен для идентификации метрологически значимой части программного обеспечения счётчика.

4.1.32. Интерфейсный объект «Тип адресации»

Интерфейсный объект содержит 1 байт. Значение объекта «0x00» соответствует 1-байтовой адресации. Значение объекта «0x01» соответствует 4-байтовой адресации. Данный интерфейсный объект предназначен для выбора в счетчике типа адресации. При выборе 1-байтовой адресации, обмен с 4-байтовой адресацией будет невозможен и наоборот. По умолчанию при выпуске с завода-изготовителя в счетчике установлена 1-байтовая адресация. См также п.п. 4.1.

4.1.33. Интерфейсный объект «Ключ сети ZigBee»

Интерфейсный объект содержит 2 байта данных типа ushort. Данный интерфейсный объект предназначен для настройки ключа сети ZigBee. По умолчанию при выпуске с завода-изготовителя в счетчике установлено значение «0x0000». Объект используется в ПО счетчика, начиная с версии 4.00 включительно. Доступ к объекту разрешён только для моделей счётчика со встроенным модулем ZigBee. См также п.п. 4.1.

4.1.34. Интерфейсный объект «Управление встроенным реле отключения нагрузки»

Интерфейсный объект содержит 1 байт. Значение объекта хранится в энергонезависимой памяти счётчика и определяет текущий режим реле отключения нагрузки. При включении анализируется модель счётчика и в зависимости от модели определяется доступность данного объекта.

Предусмотрены следующие значения объекта:

Значение объекта	Наименование режима
0	Нагрузка постоянно включена
1	Нагрузка постоянно выключена
2	Автоматическое управление нагрузкой
3	Полуавтоматическое управление нагрузкой

Режим «Нагрузка постоянно включена» является режимом по умолчанию. Данный режим обеспечивает постоянное включение нагрузки и для изменения этого состояния необходимо изменить значение данного объекта.

Режим «Нагрузка постоянно выключена» означает, что нагрузка отключена и для изменения этого состояния необходимо изменить значение данного объекта.

Режим «Автоматическое управление нагрузкой». При включении данного режима счётчик включит нагрузку и начнёт отслеживание средней мощности за период интегрирования профиля мощности. Если среднее значение мощности превысит заданный порог, нагрузка будет автоматически отключена. При наступлении нового интервала профиля мощности (следующей получасовки), нагрузка будет автоматически включена.

Режим «Полуавтоматическое управление нагрузкой» отличается от автоматического режима отсутствием автоматического включения нагрузки при наступлении нового интервала профиля мощности (новой получасовки).

Данный объект используется в ПО счетчика, начиная с модели 306. Доступ к объекту разрешён ТОЛЬКО для модели счетчика со встроенным реле отключения нагрузки. Для других моделей счетчиков при попытке доступа к данному объекту, счётчик вернёт исключение «Некорректный идентификатор объекта». Импульсный выход счётчика также может работать в режиме управления нагрузкой. См п.п. 4.1.23, 4.1.24. Режим управления нагрузкой импульсного выхода счётчика работает независимо от данного объекта. См также описание объекта «Порог автоматического отключения нагрузки» п.п. 4.1.25.

4.1.35. Интерфейсный метод «Поиск записей в списках счетчика»

Интерфейсный метод предназначен для поиска записей в списках счетчика, удовлетворяющих заданным условиям поиска. Данный метод доступен в счетчике с версией ПО 1.06 и выше. Метод может быть вызван только для объектов со следующими идентификаторами: ID= 16, 18, 59, 60, 66, 67. В случае вызова данного метода для других объектов, счётчик вернёт исключение ILLEGAL_DATA_ADDRESS (Некорректный идентификатор объекта). В качестве

условий поиска записей задаётся диапазон дат и времени, которым должны соответствовать найденные записи. Метод возвращает следующую структуру:

```
typedef struct{
    BYTE search_result;      // Код состояния поиска;
    BYTE start_index[2];    // Начальный индекс диапазона найденных записей, 2 байта;
    BYTE end_index[2];      // Конечный индекс диапазона найденных записей, 2 байта.
} ListSearch_t;
```

Код состояния поиска может принимать следующие значения:

```
typedef enum
{
    LIST_SEARCH_OK = 0,      // Запись найдена;
    LIST_SEARCH_ACK,        // Запрос на поиск принят. Начат поиск;
    LIST_SEARCH_IN_PROGRESS, // Процесс поиска записи активен;
    LIST_SEARCH_BEFORE_RANGE, // Найдены более ранние записи;
    LIST_SEARCH_AFTER_RANGE, // Найдены более поздние записи;
    LIST_SEARCH_OUT_RANGE,  // Найдены записи вне диапазона поиска;
    LIST_SEARCH_LIST_EMPTY, // Список пустой;
    LIST_SEARCH_ERROR,      // Ошибка при работе со списком;
} state_search_t;
```

При вызове метода поиска счетчик вернет LIST_SEARCH_ACK. Это означает, что запрос на поиск принят и начат процесс поиска записи. Если счетчик вернул LIST_SEARCH_LIST_EMPTY, это означает, что список пустой и поиск невозможен. При этом остальные данные структуры будут равны нулю.

При повторном вызове метода счетчик может вернуть:

- LIST_SEARCH_OK, если поиск завершен и запись найдена;
- LIST_SEARCH_IN_PROGRESS, если поиск активен и ещё не завершен;
- LIST_SEARCH_BEFORE_RANGE, если найдены более ранние записи;
- LIST_SEARCH_AFTER_RANGE, если найдены более поздние записи;
- LIST_SEARCH_OUT_RANGE, если найдены записи вне диапазона поиска;
- LIST_SEARCH_ERROR, если во время поиска обнаружена ошибка при работе со списком.

Если счетчик вернул LIST_SEARCH_OK, в остальных членах структуры ListSearch_t находятся искомые данные.

Если счетчик вернул любое другое значение, кроме LIST_SEARCH_OK, остальные члены структуры ListSearch_t будут нулевыми.

Если счетчик вернул LIST_SEARCH_IN_PROGRESS, это означает, что поиск не закончен и необходимо повторить вызов метода через какое-то время. Время поиска зависит от глубины списка и может занимать продолжительное время.

4.1.36. Интерфейсный метод «Получение коллекции объектов»

Интерфейсный метод предназначен для оптимизации опроса счётчика. Данный метод позволяет уменьшить число запросов и ускорить получение данных. Данный метод доступен в счетчике с версией ПО 1.06 и выше. В качестве входных параметров указывается идентификатор коллекции (CollectionUID) получаемых данных и индекс (Index) пакета данных в коллекции.

Collect ion UID	Index	Возвращаемые данные	Наименование коллекции
1	0	Для версий ПО счетчика с 0106 по 0199 включительно, возвращаемые данные имеют структуру: typedef struct{	Коллекция энергий от сброса счетчика. Для версий ПО счетчика с 0106 по 0199, длина

Collect ion UID	Index	Возвращаемые данные	Наименование коллекции
		<pre> BYTE active_energy_sum[4]; BYTE active_energy_T1[4]; BYTE active_energy_T2[4]; BYTE active_energy_T3[4]; BYTE active_energy_T4[4]; BYTE active_energy_T5[4]; BYTE active_energy_T6[4]; BYTE active_energy_T7[4]; BYTE active_energy_T8[4]; BYTE reactive_energy_sum[4]; BYTE reactive_energy_T1[4]; BYTE reactive_energy_T2[4]; BYTE reactive_energy_T3[4]; BYTE reactive_energy_T4[4]; BYTE reactive_energy_T5[4]; BYTE reactive_energy_T6[4]; BYTE reactive_energy_T7[4]; BYTE reactive_energy_T8[4]; BYTE RTC[7]; } collection_energy_t; Для версий ПО счетчика с 0200 по 0299 включительно, возвращаемые данные имеют структуру: typedef struct{ BYTE act_imp_energy_sum[4]; BYTE act_imp_energy_T1[4]; BYTE act_imp_energy_T2[4]; BYTE act_imp_energy_T3[4]; BYTE act_imp_energy_T4[4]; BYTE act_imp_energy_T5[4]; BYTE act_imp_energy_T6[4]; BYTE act_imp_energy_T7[4]; BYTE act_imp_energy_T8[4]; BYTE react_imp_energy_sum[4]; BYTE react_imp_energy_T1[4]; BYTE react_imp_energy_T2[4]; BYTE react_imp_energy_T3[4]; BYTE react_imp_energy_T4[4]; BYTE react_imp_energy_T5[4]; BYTE react_imp_energy_T6[4]; BYTE react_imp_energy_T7[4]; BYTE react_imp_energy_T8[4]; BYTE act_exp_energy_sum[4]; BYTE act_exp_energy_T1[4]; BYTE act_exp_energy_T2[4]; BYTE act_exp_energy_T3[4]; BYTE act_exp_energy_T4[4]; BYTE act_exp_energy_T5[4]; BYTE act_exp_energy_T6[4]; BYTE act_exp_energy_T7[4]; BYTE act_exp_energy_T8[4]; </pre>	<p>коллекции 25 байт. Для версий ПО счетчика с 0200 по 0299 включительно, длина коллекции 43 байта.</p>

Collect ion UID	Index	Возвращаемые данные	Наименование коллекции
		<pre> BYTE react_exp_energy_sum[4]; BYTE react_exp_energy_T1[4]; BYTE react_exp_energy_T2[4]; BYTE react_exp_energy_T3[4]; BYTE react_exp_energy_T4[4]; BYTE react_exp_energy_T5[4]; BYTE react_exp_energy_T6[4]; BYTE react_exp_energy_T7[4]; BYTE react_exp_energy_T8[4]; BYTE RTC[7]; } collection_energy_t; </pre>	
2	0	<pre> typedef struct{ BYTE active_power_A[4]; BYTE active_power_B[4]; BYTE active_power_C[4]; BYTE active_power_sum[4]; BYTE reactive_power_A[4]; BYTE reactive_power_B[4]; BYTE reactive_power_C[4]; BYTE reactive_power_sum[4]; BYTE total_power_A[4]; BYTE total_power_B[4]; BYTE total_power_C[4]; BYTE total_power_sum[4]; BYTE voltage_A[3]; BYTE voltage_B[3]; BYTE voltage_C[3]; BYTE current_A[3]; BYTE current_B[3]; BYTE current_C[3]; BYTE frequency[2]; BYTE current_tariff; BYTE RTC[7]; } collection_power_t; </pre>	Коллекция мгновенных значений параметров сети, измеряемых счетчиком. Длина коллекции 76 байт.
3	I (Параметр I может принимать значения от 0 до (n-1) n- глубина списка	<pre> typedef struct{ PWIRecordw_t rec[I + 0]; PWIRecordw_t rec[I + 1]; PWIRecordw_t rec[I + 2]; PWIRecordw_t rec[I + 3]; PWIRecordw_t rec[I + 4]; PWIRecordw_t rec[I + 5]; PWIRecordw_t rec[I + 6]; PWIRecordw_t rec[I + 7]; } collection_pwi_t; </pre>	Коллекция из 8 записей списка срезов мощности. В случае формирования неполной коллекции, оставшаяся часть данных заполняется нулями.
4	I (Параметр I может принимать значения от 0 до (n-1) n- глубина	<p>Для версий ПО счетчика с 0106 по 0199 включительно, возвращаемые данные имеют структуру:</p> <pre> typedef struct{ ENTAQRecord_t rec[I + 0]; ENTAQRecord_t rec[I + 1]; } collection_enta_t; </pre>	Коллекция из 2 записей энергии в суточных интервалах. В случае формирования неполной коллекции, оставшаяся часть данных заполняется нулями.

Collect ion UID	Index	Возвращаемые данные	Наименование коллекции
	списка	Для других версий ПО данная коллекция не используется. На запрос данной коллекции счетчик должен вернуть исключение SLAVE_DEVICE_FAILURE «Невозможно выполнить команду». Для получения записи использовать метод GET_ENTALIST .	
5	I (Параметр I может принимать значения от 0 до (n-1) n- глубина списка	Для версий ПО счетчика с 0106 по 0199 включительно, возвращаемые данные имеют структуру: typedef struct{ ENTAQRecord_t rec[I + 0]; ENTAQRecord_t rec[I + 1]; } collection_enta_t; Для других версий ПО данная коллекция не используется. На запрос данной коллекции счетчик должен вернуть исключение SLAVE_DEVICE_FAILURE «Невозможно выполнить команду». Для получения записи использовать метод GET_ENTALIST .	Коллекция из 2 записей энергии в месячных интервалах. В случае формирования неполной коллекции, оставшаяся часть данных заполняется нулями.
6	I (Параметр I может принимать значения от 0 до (n-1) n- глубина списка	typedef struct{ EVTRecord_t rec[I + 0]; EVTRecord_t rec[I + 1]; EVTRecord_t rec[I + 2]; EVTRecord_t rec[I + 3]; EVTRecord_t rec[I + 4]; EVTRecord_t rec[I + 5]; EVTRecord_t rec[I + 6]; EVTRecord_t rec[I + 7]; EVTRecord_t rec[I + 8]; EVTRecord_t rec[I + 9]; } collection_event_t;	Коллекция из 10 записей списка ошибок errors . В случае формирования неполной коллекции, оставшаяся часть данных заполняется нулями.
7	I (Параметр I может принимать значения от 0 до (n-1) n- глубина списка	typedef struct{ EVTRecord_t rec[I + 0]; EVTRecord_t rec[I + 1]; EVTRecord_t rec[I + 2]; EVTRecord_t rec[I + 3]; EVTRecord_t rec[I + 4]; EVTRecord_t rec[I + 5]; EVTRecord_t rec[I + 6]; EVTRecord_t rec[I + 7]; EVTRecord_t rec[I + 8]; EVTRecord_t rec[I + 9]; } collection_event_t;	Коллекция из 10 записей списка сообщений messages . В случае формирования неполной коллекции, оставшаяся часть данных заполняется нулями. Длина коллекции 80 байт.
8	I (Параметр I может принимать	typedef struct{ EVTRecord_t rec[I + 0]; EVTRecord_t rec[I + 1]; EVTRecord_t rec[I + 2];	Коллекция из 10 записей списка предупреждений warnings . В случае формирования неполной

Collect ion UID	Index	Возвращаемые данные	Наименование коллекции
	значения от 0 до (n-1) n- глубина списка	<pre> EVTRecord_t rec[I + 3]; EVTRecord_t rec[I + 4]; EVTRecord_t rec[I + 5]; EVTRecord_t rec[I + 6]; EVTRecord_t rec[I + 7]; EVTRecord_t rec[I + 8]; EVTRecord_t rec[I + 9]; } collection_event_t; </pre>	коллекции, оставшаяся часть данных заполняется нулями. Длина коллекции 80 байт.
9	0	<pre> typedef struct{ BYTE active_energy_sum[4]; BYTE active_energy_T1[4]; BYTE active_energy_T2[4]; BYTE active_energy_T3[4]; BYTE active_energy_T4[4]; BYTE active_energy_T5[4]; BYTE active_energy_T6[4]; BYTE active_energy_T7[4]; BYTE active_energy_T8[4]; BYTE reactive_energy_sum[4]; BYTE reactive_energy_T1[4]; BYTE reactive_energy_T2[4]; BYTE reactive_energy_T3[4]; BYTE reactive_energy_T4[4]; BYTE reactive_energy_T5[4]; BYTE reactive_energy_T6[4]; BYTE reactive_energy_T7[4]; BYTE reactive_energy_T8[4]; BYTE voltage_A[3]; BYTE voltage_B[3]; BYTE voltage_C[3]; BYTE current_A[3]; BYTE current_B[3]; BYTE current_C[3]; BYTE active_power_A[4]; BYTE active_power_B[4]; BYTE active_power_C[4]; BYTE reactive_power_A[4]; BYTE reactive_power_B[4]; BYTE reactive_power_C[4]; BYTE total_power_A[4]; BYTE total_power_B[4]; BYTE total_power_C[4]; BYTE frequency[2]; BYTE current_tariff; BYTE RTC[7]; BYTE device_model[16]; BYTE firmware_vers[4]; BYTE battery_voltage[2]; BYTE metrology_crc[2]; BYTE load_on_flag; BYTE cover_flag; </pre>	<p>Коллекция для сплит модели. Длина коллекции 188 байт. Коллекция доступна для всех моделей, начиная с версии 0200 включительно.</p> <p>load_on_flag = 0 – нагрузка включена. load_on_flag = 1 – нагрузка выключена</p> <p>cover_flag = 0 – клеммная крышка закрыта. cover_flag = 1 – клеммная крышка открыта</p> <p>phase_item – битовое поле, где: бит 0 – наличие фазы А бит 1 – наличие фазы В бит 2 – наличие фазы С бит 3 – признак неправильного включения фаз.</p>

Collect ion UID	Index	Возвращаемые данные	Наименование коллекции
		BYTE power_factor_A [2]; BYTE power_factor_B [2]; BYTE power_factor_C [2]; BYTE power_factor_S [2]; BYTE power_angle_A [2]; BYTE power_angle_B [2]; BYTE power_angle_C [2]; BYTE power_angle_S [2]; BYTE phase_angle_AB [2]; BYTE phase_angle_AC [2]; BYTE phase_angle_BC [2]; BYTE lcd_parameters [3]; BYTE phase_item ; } collection_split_t ; 	
10	I (Параметр I может принимать значения от 0 до (n-1) n- глубина списка)	typedef struct{ EVTRecord_t rec [I + 0]; EVTRecord_t rec [I + 1]; EVTRecord_t rec [I + 2]; EVTRecord_t rec [I + 3]; EVTRecord_t rec [I + 4]; EVTRecord_t rec [I + 5]; EVTRecord_t rec [I + 6]; EVTRecord_t rec [I + 7]; EVTRecord_t rec [I + 8]; EVTRecord_t rec [I + 9]; } collection_event_t ; 	Коллекция из 10 записей списка сообщений messages1 . В случае формирования неполной коллекции, оставшаяся часть данных заполняется нулями. Длина коллекции 80 байт. Коллекция доступна, начиная с версии 0200 включительно.
11	I (Параметр I может принимать значения от 0 до (n-1) n- глубина списка)	typedef struct{ EVTRecord_t rec [I + 0]; EVTRecord_t rec [I + 1]; EVTRecord_t rec [I + 2]; EVTRecord_t rec [I + 3]; EVTRecord_t rec [I + 4]; EVTRecord_t rec [I + 5]; EVTRecord_t rec [I + 6]; EVTRecord_t rec [I + 7]; EVTRecord_t rec [I + 8]; EVTRecord_t rec [I + 9]; } collection_event_t ; 	Коллекция из 10 записей списка предупреждений warnings1 . В случае формирования неполной коллекции, оставшаяся часть данных заполняется нулями. Длина коллекции 80 байт. Коллекция доступна, начиная с версии 0200 включительно.
12	I (Параметр I может принимать значения от 0 до (n-1) n- глубина списка)	typedef struct{ EVTRecord_t rec [I + 0]; EVTRecord_t rec [I + 1]; EVTRecord_t rec [I + 2]; EVTRecord_t rec [I + 3]; EVTRecord_t rec [I + 4]; EVTRecord_t rec [I + 5]; EVTRecord_t rec [I + 6]; EVTRecord_t rec [I + 7]; EVTRecord_t rec [I + 8]; EVTRecord_t rec [I + 9]; } collection_event_t ; 	Коллекция из 10 записей списка предупреждений warnings2 . В случае формирования неполной коллекции, оставшаяся часть данных заполняется нулями. Длина коллекции 80 байт. Коллекция доступна, начиная с версии 0200 включительно.

4.1.37. Интерфейсный объект «Конфигурация учета энергии и импульсных выходов»

Интерфейсный объект «Конфигурация учета энергии» предназначен для настройки счетчика в части учета активной и реактивной энергии и импульсных выходов счетчика. Данный интерфейс доступен для счетчиков Милур 307, начиная с версии ПО 0200 включительно и только для моделей с двунаправленным учётом (наличие буквы W в наименовании модели).

Длина объекта составляет 1 байт. Младшая тетрада объекта предназначена для конфигурирования учета активной энергии и импульсного выхода активной энергии. Старшая тетрада объекта предназначена для конфигурирования учета реактивной энергии и импульсного выхода реактивной энергии. Младшие два бита тетрад задают вид энергии, подаваемый на импульсный выход. Старшие два бита тетрад определяют конфигурацию учёта энергии счетчиком. При заводской инициализации устанавливаются значения по умолчанию:

- Учет активной энергии по модулю;
- Учет реактивной энергии по модулю;
- Имп. выход активной энергии настроен на активную импортируемую энергию;
- Имп. выход реактивной энергии настроен на реактивную импортируемую энергию;

Конфигурирование учёта активной энергии и импульсного выхода активной энергии:

Двоичный код	Учет активной энергии	Учет реактивной энергии	Импульсный выход активной энергии	Импульсный выход реактивной энергии
0xXXXX0000	По модулю	X	Активная импортируемая	X
0xXXXX0001	По модулю	X	Активная экспортируемая	X
0xXXXX0010	По модулю	X	Реактивная импортируемая	X
0xXXXX0011	По модулю	X	Реактивная экспортируемая	X
0xXXXX0100	Импортируемая	X	Активная импортируемая	X
0xXXXX0101	Импортируемая	X	Активная экспортируемая	X
0xXXXX0110	Импортируемая	X	Реактивная импортируемая	X
0xXXXX0111	Импортируемая	X	Реактивная экспортируемая	X
0xXXXX1000	Экспортируемая	X	Активная импортируемая	X
0xXXXX1001	Экспортируемая	X	Активная экспортируемая	X
0xXXXX1010	Экспортируемая	X	Реактивная импортируемая	X
0xXXXX1011	Экспортируемая	X	Реактивная экспортируемая	X
0xXXXX1100	Двунаправленный учет	X	Активная импортируемая	X
0xXXXX1101	Двунаправленный учет	X	Активная экспортируемая	X
0xXXXX1110	Двунаправленный учет	X	Реактивная импортируемая	X

0xXXXX1111	Двунаправленный учет	X	Реактивная экспортируемая	X

Конфигурирование учёта реактивной энергии и импульсного выхода реактивной энергии:

Двоичный код	Учет активной энергии	Учет реактивной энергии	Импульсный выход активной энергии	Импульсный выход реактивной энергии
0x0000XXXX	X	По модулю	X	Активная импортируемая
0x0001XXXX	X	По модулю	X	Активная экспортируемая
0x0010XXXX	X	По модулю	X	Реактивная импортируемая
0x0011XXXX	X	По модулю	X	Реактивная экспортируемая
0x0100XXXX	X	Импортируемая	X	Активная импортируемая
0x0101XXXX	X	Импортируемая	X	Активная экспортируемая
0x0110XXXX	X	Импортируемая	X	Реактивная импортируемая
0x0111XXXX	X	Импортируемая	X	Реактивная экспортируемая
0x1000XXXX	X	Экспортируемая	X	Активная импортируемая
0x1001XXXX	X	Экспортируемая	X	Активная экспортируемая
0x1010XXXX	X	Экспортируемая	X	Реактивная импортируемая
0x1011XXXX	X	Экспортируемая	X	Реактивная экспортируемая
0x1100XXXX	X	Двунаправленный учет	X	Активная импортируемая
0x1101XXXX	X	Двунаправленный учет	X	Активная экспортируемая
0x1110XXXX	X	Двунаправленный учет	X	Реактивная импортируемая
0x1111XXXX	X	Двунаправленный учет	X	Реактивная экспортируемая

4.1.38. Интерфейсный объект «ПКЭ»

Интерфейсный объект «ПКЭ» предназначен для настройки параметров показателей качества электроэнергии, регистрируемых счетчиком. В интерфейсный объект записываются пределы изменений параметров сети в миллиВольтах, в миллиГерцах. Счетчик вычисляет минимальное и максимальное значения отклонений с учётом номинального значения параметра и заданного значения предела изменений.

Данный интерфейсный объект доступен для счетчиков Милур 307, начиная с версии ПО 0200 включительно и только для моделей с двунаправленным учётом (наличие буквы W в наименовании модели).

По умолчанию в счетчике установлены параметры ПКЭ для синхронизированных систем электроснабжения. Номинальное значение частоты по умолчанию установлено 50000 мГц. Номинальное значение напряжения по умолчанию установлено 220000 мВ для моделей счетчика 307.22 и 307.12, либо 57700 мВ для модели счетчика 307.11.

Для номинального значения напряжения 220000 мВ отклонение напряжения (10%) будет соответствовать 22000 мВ. Отклонение частоты задаётся в мГц. Для синхронизированных систем электроснабжения отклонение частоты задаётся 200 мГц в течение 95 % недельного интервала времени и 400 мГц в течение 100 % недельного интервала времени. Для изолированных систем электроснабжения отклонение частоты задаётся 1000 мГц в течение 95 % недельного интервала времени и 5000 мГц в течение 100 % недельного интервала времени. Максимальные и минимальные значения фазных напряжений и частоты при записи объекта должны быть инициализированы номинальными значениями. Даты начала и окончания интервала контроля при записи объекта должны быть инициализированы текущим значением даты.

Длина объекта составляет 44 байта. Объект представляет собой следующую структуру:

```
typedef struct{
    BYTE voltage_nom[3]; // Номинальное значение напряжения, мВ
    BYTE frequency_nom[2]; // Номинальное значение частоты, мГц
    BYTE voltage_100[3]; // Отклонение напряжения в течение 100% недели, мВ
    BYTE frequency_95[2]; // Отклонение частоты в течение 95% недели, мГц
    BYTE frequency_100[2]; // Отклонение частоты в течение 100% недели, мГц
    BYTE v_min_phA[3]; // Минимальное значение напряжения на фазе А, мВ
    BYTE v_max_phA[3]; // Максимальное значение напряжения на фазе А, мВ
    BYTE v_min_phB[3]; // Минимальное значение напряжения на фазе В, мВ
    BYTE v_max_phB[3]; // Максимальное значение напряжения на фазе В, мВ
    BYTE v_min_phC[3]; // Минимальное значение напряжения на фазе С, мВ
    BYTE v_max_phC[3]; // Максимальное значение напряжения на фазе С, мВ
    BYTE f_min_95[2]; // Минимальное значение частоты в течение 95% недели, мГц
    BYTE f_max_95[2]; // Максимальное значение частоты в течение 95% недели, мГц
    BYTE f_min_100[2]; // Минимальное значение частоты в течение 100% недели, мГц
    BYTE f_max_100[2]; // Максимальное значение частоты в течение 100% недели, мГц
    BYTE start_year; // Год начала интервала контроля ПКЭ
    BYTE start_month; // Месяц начала интервала контроля ПКЭ
    BYTE start_day; // День начала интервала контроля ПКЭ
    BYTE stop_year; // Год окончания интервала контроля ПКЭ
    BYTE stop_month; // Месяц окончания интервала контроля ПКЭ
    BYTE stop_day; // День окончания интервала контроля ПКЭ
} pwr_quality_t;
```

Обновление максимальных и минимальных значений фазных напряжений, частоты, даты начала и окончания интервала контроля производится счётчиком еженедельно в понедельник в 0 часов, 0 минут, 0 секунд. Эти данные хранятся в энергонезависимой памяти в течение недели до следующего обновления. Даты начала и окончания интервала контроля соответствуют прошедшему закрытому недельному интервалу.

4.1.39. Интерфейсный объект «Защитное отключение нагрузки»

Интерфейсный объект «Защитное отключение нагрузки» предназначен для настройки защитного отключения потребителя, в случае выхода параметров электрической сети за допустимые пределы. Данный интерфейсный объект доступен для чтения и записи только для моделей счетчиков со встроенным реле отключения нагрузки. Для всех остальных моделей без встроенного реле отключения нагрузки, счетчик вернёт исключение ILLEGAL_DATA_ADDRESS.

Данный интерфейсный объект доступен, начиная с версии 0107 включительно. Длина объекта составляет 25 байт. Объект представляет собой следующую структуру:

```
typedef struct{
    BYTE vltg_min_phA[3]; // Минимальное значение напряжения по фазе А, мВольт
    BYTE vltg_min_phB[3]; // Минимальное значение напряжения по фазе В, мВольт
```

```

BYTE vltg_min_phC[3]; // Минимальное значение напряжения по фазе С, мВольт
BYTE vltg_max_phA[3]; // Максимальное значение напряжения по фазе А, мВольт
BYTE vltg_max_phB[3]; // Максимальное значение напряжения по фазе В, мВольт
BYTE vltg_max_phC[3]; // Максимальное значение напряжения по фазе С, мВольт
BYTE pwr_max_ph[4]; // Максимальное значение фазной активной мощности, мВт
BYTE power_cut_time; // Время контроля, сек
BYTE power_cut_ctrl[2]; // Набор параметров контроля
} emergency_pwrkut_t;

```

Срабатывание защитного отключения нагрузки происходит в случае, если в течение времени контроля **power_cut_time** будет зафиксировано устойчивое состояние выхода за пределы хотя бы одного из параметров контроля, устойчивое состояние срабатывания датчика магнитного поля или нарушение чередования фаз. Функция защитного отключения предоставляет следующий набор параметров контроля:

Данные	Бит	Описание параметра контроля
power_cut_ctrl[0];	0	Снижение напряжения по фазе А ниже vltg_min_phA
	1	Снижение напряжения по фазе В ниже vltg_min_phB
	2	Снижение напряжения по фазе С ниже vltg_min_phC
	3	Повышение напряжения по фазе А выше vltg_max_phA
	4	Повышение напряжения по фазе В выше vltg_max_phB
	5	Повышение напряжения по фазе С выше vltg_max_phC
	6	Повышение фазной активной мощности выше pwr_max_ph
	7	Срабатывание датчика магнитного поля
power_cut_ctrl[1];	0	Нарушение чередования фаз
	1	Активность таймера контроля. Бит всегда установлен в 1.
	2	Автоматическое включение нагрузки при восстановлении напряжения

Для включения параметра контроля, соответствующий бит в **power_cut_ctrl** должен быть установлен в 1. При сбросе бита в 0, соответствующий параметр не контролируется счетчиком. Таймер контроля включен всегда. Изменение бита активности таймера контроля не оказывает влияния на работу таймера.

Устойчивый выход значения фазного напряжения за пределы значений **vltg_min_phA**, **vltg_min_phB**, **vltg_min_phC**, **vltg_max_phA**, **vltg_max_phB**, **vltg_max_phC** в течение времени **power_cut_time**, приведёт к отключению нагрузки. Повторное включение нагрузки возможно подачей команды по интерфейсу или автоматически по истечении 60 секунд после восстановления напряжений по всем фазам, если установлен бит автоматического включения и нет других причин отключения нагрузки.

Значение параметра **pwr_max_ph** контролируется счетчиком независимо по каждой фазе. При устойчивом превышении значения данного параметра в течение времени **power_cut_time**, нагрузка будет отключена. Повторное включение нагрузки возможно только подачей команды по интерфейсу.

В случае устойчивого срабатывания датчика магнитного поля или обнаружения нарушения чередования фаз в течение времени **power_cut_time**, нагрузка будет отключена. Повторное включение нагрузки возможно только подачей команды по интерфейсу.

Параметр «Автоматическое включение нагрузки при восстановлении напряжения» означает, что счетчик автоматически включит нагрузку если:

- отключение нагрузки произошло по причине выхода напряжения за пределы порогов;
- в течение 60 сек после восстановления напряжения, не будет зафиксировано ни одной причины срабатывания защитного отключения нагрузки из установленного набора параметров контроля.

4.1.40. Интерфейсный объект «Причины защитного отключения нагрузки»

Интерфейсный объект «Причины защитного отключения нагрузки» предназначен для считывания из счетчика причин, по которым сработало защитное отключение нагрузки.

Данный интерфейсный объект доступен, начиная с версии 0107 включительно. Длина объекта составляет 2 байта. Объект представляет собой следующую структуру:

```
typedef struct{  
    BYTE power_cut_case[2]; // Причина отключения нагрузки  
} emergency_pwrkut_case_t;
```

Для обновления информации об отключении нужно прочитать из счетчика значение объекта. Предусматривается следующий набор случаев защитного отключения нагрузки:

Данные	Бит	Описание параметра контроля
power_cut_case[0];	0	Снижение напряжения по фазе А ниже vltg_min_phA
	1	Снижение напряжения по фазе В ниже vltg_min_phB
	2	Снижение напряжения по фазе С ниже vltg_min_phC
	3	Повышение напряжения по фазе А выше vltg_max_phA
	4	Повышение напряжения по фазе В выше vltg_max_phB
	5	Повышение напряжения по фазе С выше vltg_max_phC
	6	Повышение фазной активной мощности выше pwr_max_ph
	7	Срабатывание датчика магнитного поля
power_cut_case[1];	0	Нарушение чередования фаз
	1	Бит всегда установлен в 0.
	2	Бит всегда установлен в 0.

При включении нагрузки командой по интерфейсу, все биты сбрасываются в 0. В случае срабатывания защитного отключения нагрузки, соответствующий бит устанавливается в 1. Интерфейсный объект располагается в оперативной памяти. При включении счетчика все причины отключения нагрузки, кроме выхода напряжения за заданные пределы, устанавливаются в интерфейсном объекте. Если при включении счетчика напряжение находится за пределами порогов, соответствующие биты появятся в интерфейсном объекте через **power_cut_time** секунд. Если при включении счетчика напряжение в норме, то нагрузка будет включена через 60 секунд.

5. ПРИЛОЖЕНИЯ

5.1.

6. ЖУРНАЛ ЗАМЕЧАНИЙ

6.1. 27.09.2011. Закрыто изм.7.1. Коточигов. Поправить объекты «Тарифное расписание» и «Часы реального времени».

6.2. 07.09.2012. Закрыто изм.7.2. Коточигов. Добавить режим импульсного выхода «Импульсы кварцевого резонатора 1 Гц».

6.3. 07.09.2012. Закрыто изм.7.3. Коточигов. Ввести идентификатор управляющей процедуры «Заводская инициализация» для быстрой настройки счетчика после программирования.

6.4. 07.09.2012. Закрыто изм.7.4. Коточигов. Исправить размерность записи среза мощности в таблице объектов с 11 на 9.

6.5. 12.04.2013. Закрыто изм.7.5. Коточигов. Внести изменения в части нумерации объектов, ввести объекты «Энергия в суточных интервалах», «Энергия в месячных интервалах», «Серийный номер счетчика», ввести функцию чтения энергий в интервалах.

6.6. 19.09.2013 Закрыто изм.7.6. Коточигов. Убрать описание объектов «Энергия на начало месяца», Поправить описание объекта «Тарифное расписание», Исправить размерность величин мощности, напряжения, токи в таблице объектов и описаниях объектов. Исправить описание объектов «Энергия в интервалах», «Срезы мощности». «Списки событий». Добавить описание сервисов AOPEN и ARELEASE.

6.7. 23.12.2013 Закрыто изм.7.7. Коточигов. Добавить объекты «Серийный номер печатного узла», «Таймаут ответа счетчика». Поправить описание объекта «Серийный номер счетчика». Ввести описание 4-байтовой адресации.

6.8. 20.03.2014 Закрыто изм.7.8. Коточигов. Добавить метод «Поиск записи в срезах мощности» в список методов. Добавить описание метода «Поиск записи в срезах мощности».

6.9. 02.04.2014 Закрыто изм.7.9. Коточигов. Изменить тарифное расписание в части введения 8 тарифов для 16 тарифных зон. Убрать объекты тарифного расписания на 4 тарифа. Добавить объекты тарифного расписания на 8 тарифов. Добавить объекты активной и реактивной энергий по тарифам 5,6,7,8. Изменить структуру объектов «Энергия в суточных интервалах», «Энергия в месячных интервалах».

6.10. 30.04.2014 Закрыто изм.7.10. Коточигов. Ввести зависимость размерности энергии от модели счётчика. В перечень событий добавить событие «Воздействие магнитным полем».

6.11. 30.04.2014 Закрыто изм.7.11. Коточигов. Исправить описание объекта «Параметры калибровки». Ввести объект и описание объекта «Версия метрологически значимой части ПО».

6.12. 11.09.2014 Закрыто изм.7.12. Коточигов. Ввести интерфейсные объекты «Тип адресации», «Ключ сети ZigBee» и описание интерфейсных объектов «Тип адресации», «Ключ сети ZigBee».

6.13. 24.09.2014 Закрыто изм.7.13. Коточигов. Дополнить интерфейсный объект «Буфер событий» сообщением «Коррекция времени».

6.14. 14.10.2014 Закрыто изм.7.14. Коточигов. Ввести описание интерфейсного объекта «Управление встроенным реле отключения нагрузки».

6.15. 19.01.2014 Закрыто изм.7.15. Коточигов. Ввести метод получения индекса последнего заполненного элемента списка.

6.16. 31.08.2015 Закрыто изм.7.16. Коточигов. Добавить метод поиска диапазона записи в списках, метод выборки коллекции данных.

6.17. 27.03.2016 Закрыто изм.7.17. Коточигов. Добавить объекты экспортируемой активной и реактивной энергий, дополнить описание интерфейсного объекта «Срезы мощности», расширить объект «Буфер событий» в части warnings и messages, дополнить описание объектов «Энергия в суточных интервалах», «Энергия в месячных интервалах». Ввести интерфейсный объект «Конфигурация учета энергии и импульсных выходов».

6.18. 15.05.2016 Закрыто изм.7.18. Коточигов. Добавить коллекцию для сплит модели счетчика «Милур 307». Изменить коллекции для модели «Милур 307».

6.19. 28.07.2016 Закрыто изм.7.19. Коточигов. Добавить объект и описание объекта «Защитное отключение нагрузки».

6.20. 31.08.2016 Закрыто изм.7.20. Коточигов. Добавить объект и описание объекта «Причины защитного отключения нагрузки».

6.21. 23.09.2016 Закрыто изм.7.21. Коточигов. Дополнить описание интерфейсного объекта "Список событий (сообщения)" в части события "Коррекция времени".

7. ЖУРНАЛ ИЗМЕНЕНИЙ

7.1. 27.09.2011 по зам.6.1. Коточигов. Поправлена таблица тарифного расписания. п.п.4.1.8. Поправлено описание объекта «Часы реального времени» п.п. 4.1.7.

7.2. 07.09.2012 по зам.6.2. Коточигов. Добавить режим импульсного выхода «Импульсы кварцевого резонатора 1 Гц». См п.п.4.1.23.

7.3. 07.09.2012 по зам.6.3. Коточигов. Добавлена управляющая процедура «Заводская инициализация». См п.п.4.1.4.

7.4. 07.09.2012 по зам.6.4. Коточигов. Исправлена размерность записи среза мощности в таблице объектов с 11 на 9. См п.п.4.1.

7.5. 12.04.2013 по зам.6.5. Коточигов. Введен метод Получить запись списка «Энергия в интервалах», Введены изменения в нумерации объектов. Добавлены объекты «Энергия в суточных интервалах», «Энергия в месячных интервалах», «Серийный номер счетчика». Дополнено описание объекта «Управляющая процедура». См п.п. 3.6.3, 4.1, 4.1.4, 4.1.19, 4.1.23, 4.1.26, 4.1.27.

7.6. 19.09.2013 по зам.6.6. Коточигов. Убрано описание объектов «Энергия на начало месяца», Поправлено описание объекта «Тарифное расписание», Исправлена размерность величин мощности, напряжения, токи в таблице объектов и описаниях объектов. Исправлено описание объектов «Энергия в интервалах», «Срезы мощности». «Списки событий». Добавлено описание сервисов AOPEN и ARELEASE. См п.п.3.6.2, 3.6.9, 3.6.10, 4.1, 4.1.2, 4.1.3, 4.1.8, 4.1.19, 4.1.12, 4.1.14, 4.1.26.

7.7. 23.12.2013 по зам.6.7. Коточигов. Добавлены объекты «Серийный номер печатного узла», «Таймаут ответа счетчика». Поправлено описание объекта «Серийный номер счетчика». Введено описание 4-байтовой адресации. См п.п.4.1.27, 4.1.28, 4.1.29, 4.1.

7.8. 20.03.2014 по зам.6.8. Коточигов. В список методов добавлен метод «Поиск записи в срезах мощности». Добавлено описание метода «Поиск записи в срезах мощности». См п.п. 3.2.4.34.1.30.

7.9. 10.04.2014 по зам.6.9. Коточигов. Изменено тарифное расписание в части введения 8 тарифов для 16 тарифных зон. Убраны объекты тарифного расписания на 4 тарифа. Добавлены объекты тарифного расписания на 8 тарифов. Добавлены объекты активной и реактивной энергий по тарифам 5,6,7,8. Изменена структура объекта «Энергия в суточных интервалах», «Энергия в месячных интервалах». См п.п.4.1, 4.1.1, 4.1.8, 4.1.21, 4.1.26.

7.10. 30.04.2014 по зам.6.10. Коточигов. Введена зависимость размерности энергии от модели счётчика. В перечень событий добавлено событие «Воздействие магнитным полем». См п.п.4.1, 4.1.1, 4.1.12, 4.1.26, 4.1.13.

7.11. 11.07.2014 по зам.6.11. Коточигов. Поправлено описание объекта «Параметры калибровки». Введён объект и описание объекта «Версия метрологически значимой части ПО». См п.п.4.1, 4.1.1, 4.1.12, 4.1.26, 4.1.13.

7.12. 11.09.2014 по зам.6.12. Коточигов. Введены объекты и описание объектов «Тип адресации», «Ключ сети ZigBee». См п.п. 4.1, 4.1.32, 4.1.33.

7.13. 24.09.2014 по зам.6.13. Коточигов. Дополнен интерфейсный объект «Буфер событий» сообщением «Коррекция времени». См п.п.4.1.13.

7.14. 14.10.2014 по зам.6.14. Коточигов. Введено описание интерфейсного объекта «Управление встроенным реле отключения нагрузки». См п.п.4.1, 4.1.34.

7.15. 19.01.2015 по зам.6.15. Коточигов. Введен метод получения индекса последнего заполненного элемента списка. См п.п. 3.2.4.3, 4.1.12, 4.1.14, 4.1.26,.

7.16. 31.08.2015 По зам. 6.16. Коточигов. Добавлены методы LIST_SEARCH (BYTE ObjUID, BYTE start_minutes, BYTE start_hour, BYTE start_mday, BYTE start_month, BYTE start_year, BYTE end_minutes, BYTE end_hour, BYTE end_mday, BYTE end_month, BYTE end_year), GET_COLLECTION (BYTE CollectionUID, BYTE Index) и описание данных методов. См п.п. 3.2.4.3, 4.1.35, 4.1.36. Дополнен пункт 4.1.26 в части максимального числа записей для списка «Энергия в суточных интервалах».

7.17. 27.03.2016 по зам.6.17. Коточигов. Добавлены объекты экспортируемой активной и реактивной энергий Дополнены интерфейсные объекты «Срезы мощности», «Буфер событий» (расширен список событий в части warnings, добавлены объекты warnings1, warnings2, расширен список событий в части messages, добавлен объект messages1), «Энергия в суточных интервалах», «Энергия в месячных интервалах». Введён объект «Конфигурация учета энергии и импульсных выходов». См п.п. 4.1, 4.1.1, 4.1.12, 4.1.13, 4.1.26, 4.1.30, 4.1.36, 4.1.37.

7.18. 15.05.2016 по зам.6.18. Коточигов. Добавлена коллекция для сплит модели счетчика. Изменены коллекция для модели «Милур 307», начиная с версии 0200 включительно. См п.п. 4.1.36.

7.19. 28.07.2016 по зам.6.19. Коточигов. Добавлен объект и описание объекта «Защитное отключение нагрузки». Объект доступен для моделей счетчиков со встроенным реле отключения нагрузки, начиная с версии 0107 включительно. См п.п. 4.1, 4.1.39.

7.20. 31.08.2016 по зам.6.20. Коточигов. Добавлен объект и описание объекта «Причины защитного отключения нагрузки». Объект доступен для моделей счетчиков со встроенным реле отключения нагрузки, начиная с версии 0107 включительно. См п.п. 4.1, 4.1.39, 4.1.40.

7.21. 23.09.2016 по зам.6.21. Коточигов. Дополнено описание интерфейсного объекта "Список событий (сообщения)" в части события "Коррекция времени". Алгоритм формирования события «Коррекция времени» изменен, начиная с версии 1.09 включительно. См п.п. 4.1.14.

Согласовано

" ___ " _____ 20 г.

" ___ " _____ 20 г.

Согласовано

" ___ " _____ 20 г.

" ___ " _____ 20 г.