

Текст ДОБАВЛЕН

Текст ИЗМЕНЕН

Текст УДАЛЕН

ce308 - трехфазный, 3 измерителя, данные фаз (ф1), (ф2) и (ф3)

ce208_2 - однофазный двухэлементный, 2 измерителя, данные фазы (ф1) и нейтраль (фN)

ce208_1 - однофазный одноэлементный, 1 измеритель, данные фазы (ф1)

Алгоритм входа в сеанс

Запрос: безадресный - `/?!<CR><LF>` или адресный `/?адрес!<CR><LF>`,

адрес - значение параметра IDPAS()

Ответ: без DLMS - `/EMRzCE3081.1<CR><LF>` или с DLMS - `/EMRz2CE2089.1<CR><LF>`,

EMR - идентификатор концерна;

z - символ рабочей скорости обмена, значение параметра SPDzz();

CEv2.v4 - идентификатор прибора, где v2.v4 - значение полей СКОП - параметр SCSD_().

v2 - индекс "платформы" для **ce308** - 3080 / 3081, для **ce208_2** - 2087 / 2089, для **ce208_1** - 2086 / 2088

v4 - номер версии протокола обмена счетчика

Запрос: без перехода в DLMS - `<ACK>0z1<CR><LF>` или с переходом в DLMS - `<ACK>2z2<CR><LF>`

z - символ подтверждения рабочей скорости обмена (при совпадении переход на рабочую скорость, иначе остается на начальной скорости);

Ответ: без перехода в DLMS - `<SOH>P0<STX>(hex32)<ETX><BCC>`

или переход в DLMS - `<ACK>2z2<CR><LF>`

hex32 - входное значение для расчета ХЕШ-кода пароля (случайное число Random32).

Запрос: `<SOH>P1<STX>(psw)<ETX><BCC>` или `<SOH>P2<STX>(hex32)<ETX><BCC>`

psw - символы пароля, значение параметра PSWzz()

hex32 - значение ХЕШ-кода пароля.

Ответ: при совпадении пароля `<ACK>` иначе `<SOH>B0<ETX><BCC>`

Обмен в сеансе

`<SOH>R1<STX>name()<ETX><BCC>`,

`<SOH>W1<STX>name(value)<ETX><BCC>`,

`<SOH>B0<ETX><BCC>`

`<ACK>`,

`<NAK>`,

`<STX>name(value)<CR><LF>name(ERRnn)<CR><LF>...<ETX><BCC>`,

`<STX>(ERRnn)<ETX><BCC>`

`<SOH>B0<ETX><BCC>`

name - имя параметра 5 символов

value - значение параметра, не более 32 символов

ERRnn - сообщение номер nn

Чтение без сеанса (на начальной скорости)

Запрос: `/?!<SOH>R1<STX>name()<ETX><BCC>` или `/?адрес!<SOH>R1<STX>name()<ETX><BCC>`

Ответ: `<STX>name(ERRnn)<CR><LF>name(value)<CR><LF>...<ETX><BCC>`

`<STX>(ERRnn)<ETX><BCC>`

Запись без сеанса (на начальной скорости)

Запрос: ширококвещательный - `/?!<SOH>W1(psw)<STX>name(value)<ETX><BCC>`

адресный - `/?адрес!<SOH>W1(psw)<STX>name()<ETX><BCC>`

Ответ: на ширококвещательный - нет ответа

на адресный - `<ACK>` или `<STX>(ERRnn)<ETX><BCC>`

Для ширококвещательной записи (без ответа), необходимо выдержать паузу не менее 200мс, перед передачей следующего запроса.

Синхронизация часов (ширококвещательная)

Запрос: с обнулением секунд - `/?CTIME!<CR><LF>`

В сторону точного времени - `/?CTIME(чч:мм:сс)!<CR><LF>`

На заданное значение коррекции - `/?CTIME(sXX)!<CR><LF>`

Информация внешнего модуля связи (для вывода на ЖКИ)

Запрос: передача информации модуля связи - `/?MCVAL(id,tim,val)!<CR><LF>`

Все поля параметра передаются в формате HEX.

id - идентификатор модуля связи (0-нет модуля), 16 бит, на ЖКИ выводится 4-я символами HEX

tim - время отображения информации на ЖКИ, 16 бит, сек

val - информация модуля связи, 32 бит, на ЖКИ выводится 8-ю символами HEX

Непечатные символы

<SOH> - 0x01 начало заголовка
<STX> - 0x02 начало текста
<ETX> - 0x03 конец текста
<ACK> - 0x06 подтверждение
<NAK> - 0x15 отрицательное подтверждение
<CR> - 0x0D перевод строки
<LF> - 0x0A возврат каретки
<BCC> - 0x00...0x7F контрольный код

Перечень сообщений (ERRnn)

В случае невозможности передачи запрошенной информации выдается номерное сообщение, где **nn** - идентификационный номер (десятичный).

```
#define UnknownCom 11 // "Команда не поддерживается устройством"
#define UnknownPar 12 // "Неизвестный параметр (имя)"
#define WrongStructPar 13 // "Неправильная структура параметра"
#define CloseAccess 14 // "Ненажата кнопка ДСТП"
#define RejectAccess 15 // "Отказано в доступе к параметру (запись/чтение по списку)"
#define DisAccess 16 // "Запрещено программирование (нет перемычки)"
#define ImpossiblePar 17 // "Недопустимое значение параметра"
#define NonExistDate 18 // "Несуществующая дата/отсутствует фиксация"
#define BusyAccess 19 // "Занят доступ для программирования (запись по другому порту)"
#define ExcSizeBuf 22 // "Превышен допустимый размер ответа" (групповой запрос)
//---
#define ProblemPower 30 // Проблемы с питанием (нет записи в NV-память)
#define ProblemI2C 31 // Проблемы аппаратные i2c
#define ProblemNVmem 32 // Проблемы с NV-памятью (достоверность)
#define ProblemNVsav 33 // Проблемы с NV-памятью (запись)
```

Расчет контрольного кода <BCC>

Контрольный код рассчитывается двумя методами:

- арифметическое суммирование, ADD (проприетарный метод);
- суммирование по модулю 2, "исключающее ИЛИ", XOR (метод ГОСТ Р МЭК 61107-2001).

Расчет выполняется над символами, следующими после первого символа <SOH> или <STX> до символа <ETX> включительно. Размерность контрольного кода <BCC> составляет 7 бит.

Расчет ХЕШ-кода пароля.

Хэш-код пароля для передачи в счетчик – результат вычисления циклической контрольной суммы CRC32, по полиному 0x04C11DB7, сделанного над содержанием строки пароля (символы до ограничителя 0x0).

Начальным входным значением расчета является полученное случайное число при входе в сеанс,

Init = Random32. Выходное значение расчета не инвертируется, XorOut = 0x00000000

// Name : CRC-32 Ethernet

// Polynom : 0x04C11DB7 ($x^{32}+x^{26}+x^{23}+x^{22}+x^{16}+x^{12}+x^{11}+x^{10}+x^8+x^7+x^5+x^4+x^2+x+1$)

// Mirror: 0xEDB88320

// Init : 0xFFFFFFFF (используется Random32)

// Revert: true

// XorOut: 0xFFFFFFFF (используется 0x00000000)

// Check : 0xCBF43926 ("123456789")

Группировка параметров

Для уменьшения количества запросов реализована компоновка (слияние) нескольких запросов в один.

Формат запроса чтения группы - <SOH>R1<STX>GRPNM(NAME1()NAME2()...NAMEN())<ETX><BCC>

Формат запроса записи группы -

<SOH>W1<STX>GRPNM(NAME1(знач)NAME2(знач)NAME2(знач)...NAMEN(знач))<ETX><BCC>

Размер сформированного запроса группы параметров ограничен величиной буфера приемника счетчика, который равен 160 символов (байт). В размере запроса учитываются все байты запроса, включая непечатные символы <SOH>,<STX>,<ETX>...<CR>,<LF>...<BCC>.

Параметры в запрос группы добавляются в формате и структурно соответствующие описанию для обычного одиночного запроса. Разделители между параметрами отсутствуют.

Размер ответа зависит от настройки, параметр SIZzz(), и задается в диапазоне от 80 до 60000 байт. В размере ответа учитываются все байты, включая непечатные символы. При достижении переданного количества байт заданного размера, выдается сообщение (ERR22) и ответ заканчивается. Поэтому при определении размера ответа необходимо учитывать дополнительно размер сообщения - 7 байт. При значении настройки равно нулю - контроль размера ответа не выполняется.

Формат ответа на запрос чтения группы -

<STX>NAME1(знач_1)(знач_2)(знач_K)NAME2(знач_1)...NAMEN(знач_1)(знач_2)<ETX><BCC>

Ответ на запрос чтения выдается в количестве, формате и структурно соответствующие описанию имени параметра. Имена одноименных параметров не выдаются, независимо от настройки условия выдачи одноименных параметров (CDPzz.bit.DisName).

Разделители между значениями одного параметра и между различными параметрами отсутствуют.

При передаче ответа, в канале возможны паузы между значениями параметров до 80 мс.

Формат ответа на запрос записи группы -

<STX>NAME1(OK)NAME2(ERR17)NAME2(OK)...NAMEN(OK)<ETX><BCC>

Ответ на запрос записи выдается на каждый параметр в группе с указанием имени и сообщением результата записи. Результат положительного подтверждения - сообщение (OK), вместо <ACK> для запроса записи одиночного параметра. При отказе записи выдается номерное сообщение (ERRnn).

Разделители между сообщениями параметра отсутствуют.

При запросе записи группы вне сеанса без адреса (широковещательная запись) ответ физически не выдается.

При передаче ответа, в канале возможны паузы между сообщениями до 80 мс.

Примеры построения запросов и получаемых ответов.

1. Чтение группы:

вне сеанса: (настройки - не выводить имена, размер ответа не более 130 байт)

Запрос: `/?Example!<SOH>R1<STX>GRPNM(MODEL()SNUM_()END01(1.0,1,3)LOE028())<ETX><BCC>`

Ответ: `<STX>MODEL(2C05)SNUM_(ERR12)END01(19.05.15 216.59202)(0.0)(210.37)LOE02(31, 18.05.15, 13:30:21, 1)(30, 16.05.15, 19:02:17,0)(ERR22)<ETX><BCC>`

в сеансе: (настройки - выводить имена, размер ответа не ограничен)

Запрос: `<SOH>R1<STX>GRPNM(MODEL()SNUMB()END01(1.0,1,3)LOE02())<ETX><BCC>`

Ответ: `<STX>MODEL(2C05)SNUMB(1234567)END01(19.05.15 216.59202)END01(0.0)END01(210.37)LOE02(31, 18.05.15, 13:30:21, 1)LOE02(30, 16.05.15, 19:02:17,0)LOE02(29, 16.05.15, 18:58:24,1)LOE02(28, 10.05.15, 02:13:45,0)<ETX><BCC>`

2. Запись группы:

вне сеанса:

Запрос:

`/?Example!<SOH>W1(777777)<STX>GRPNM(MODE_(C05)GRF01(7:00,1)GRF01(23:0,2))<ETX><BCC>`

Ответ: `<STX>MODE_(ERR12)GRF01(OK)GRF01(OK)<ETX><BCC>`

в сеансе:

Запрос: `<SOH>W1<STX>GRPNM(MODEL(C05)GRF01(7:00,1)GRF01(23:0,2))<ETX><BCC>`

Ответ: `<STX>MODEL(OK)GRF01(OK)GRF01(OK)<ETX><BCC>`

//----- (9) Параметры сети -----

- VOLTA()** - фазное напряжение, В
се308 - 3 параметра ф1, ф2, ф3; се208_2 - ф1, фN, се208_1 - ф1
- VOLTLO** - линейное напряжение, В
се308 - 3 параметра ф1-2, ф2-3, ф3-1
- CURRE()** - фазный ток, А
се308 - 3 параметра ф1, ф2, ф3; се208_2 - ф1, фN, се208_1 - ф1
- POWER()** - активная мощность, кВт
се308 - 4 параметра ф1, ф2, ф3, сумма; се208_2 - ф1, фN, се208_1 - ф1
- POWEQ()** - реактивная мощность, квар
се308 - 4 параметра ф1, ф2, ф3, сумма; се208_2 - ф1, фN, се208_1 - ф1
- POWES()** - полная мощность, кВА
се308 - 4 параметра ф1, ф2, ф3, сумма; се208_2 - ф1, фN, се208_1 - ф1
- COS_f()** - коэф активной мощности
се308 - 4 параметра ф1, ф2, ф3, сумма; се208_2 - ф1, фN, се208_1 - ф1
- SIN_f()** - коэф реактивной мощности
се308 - 4 параметра ф1, ф2, ф3, сумма; се208_2 - ф1, фN, се208_1 - ф1
- TAN_f()** - коэф реактивной мощности
се308 - 4 параметра ф1, ф2, ф3, сумма; се208_2 - ф1, фN, се208_1 - ф1
- CORIU()** - угол между током и напряжением фазы, град
се308 - 3 параметра ф1, ф2, ф3; се208_2 - ф1, фN, се208_1 - ф1
- CORUU()** - угол между напряжениями фаз, град
се308 - 3 параметра ф1-2, ф2-3, ф3-1
- FREQU()** - частота сети, Гц
- PAVER()** - прогнозируемая мощность (активная потребленная), с начала текущего интервала до текущего времени, кВт
се308 - суммарная; се208_2, се208_1 - учтенная по ф1 или фN
- TERMO()** - температура контроллера, целое десятичное значение со знаком, LSB = 0,01°C
- CORTT()** - значение суточного ухода часов с учетом температуры окружающей среды, целое десятичное значение со знаком (LSB = 0,1сек). Рассчитанный суточный уход (для спешащих +XX сек, для отстающих -XX сек) для температуры 25 °C соответствует параметру CORTI().
Диапазон значений параметра от -11 сек, до +5,5 сек (-110...+55).
- VOLTD()** - отклонение фазного напряжения от номинального/согласованного, на интервале 600 секунд, %
се308 - 6 параметров, отрицательное ф1, ф2, ф3 и положительное ф1, ф2, ф3
се208 - 2 параметра, отрицательное ф1 и положительное ф1
- FREQD()** - отклонение основной частоты напряжения сети от номинальной, на интервале 10 секунд, Гц

//----- (7) Параметры учета -----

1. Список идентификаторов ретроспективы (архивов)

zz = 01 - суточные (до 128)

zz = 02 - месячные или расчетный период (до 40)

zz = 03 - годовые (до 10)

zz = 04 - профиль (до 128 для 60 минут)

zz = 05 - максимумы (до 40)

zz = 06 - события (до 20)

zz = 00 - списки всех архивов по порядку

Если счетчик был выключен в течение календарных суток, данные в архивах суточные и профиль не фиксируются.

Если счетчик был выключен в течение календарного месяца или в течение всего периода между датами расчетного периода, данные в архивах месячные и максимумы не фиксируются.

Если счетчик был выключен в течение календарного года или в течение всего периода между датами декабря расчетного дня, данные в архиве годовые не фиксируются.

Доступные форматы запроса - **LSTzz()**, **LSTzz(k)**, **LSTzz(k,n)**

Запрос - **LSTzz()** - список идентификаторов ретроспективы, начиная с текущего идентификатора, в количестве выполненных фиксаций.

Запрос - **LSTzz(k,n)** - список идентификаторов ретроспективы, начиная с запрошенного номера k (отсчет с текущего = 1), в количестве n штук.

Форматы ответа:

ERR18 - запрошенной фиксации еще не существует

Ответ - LST01(dd.mm.yy) - день, месяц, год

- LST02(bb.mm.yy) - дата расчетного периода (0-конец месяца), месяц, год

- LST03(bb.00.yy) - дата расчетного периода (0-конец месяца), год

- LST04(dd.mm.yy) - день, месяц, год

- LST05(bb.mm.yy) - дата расчетного периода (0-конец месяца), месяц, год

- LST06(dd.mm.yy,чч:мм,ее) - день, месяц, год, час, минута, номер события eEventAction_TypeDef

(1..32)

1а. Глубина хранения ретроспективы (архивов)

Архив представляет собой циклический массив записей. При заполнении архива, новая запись затирует самую старую запись. Параметр показывает глубину архива и текущий номер записи в архиве.

Если текущий номер записи больше глубины архива, значит, архив заполнен полностью.

Доступные форматы запроса - **LSLzz()**

Ответ - LSLzz(hex,hex) - глубина и текущий номер записи архива

Если в запросе указать zz = 00, в ответ придут глубины всех архивов по порядку

2. Накопления Энергий ретроспектив (суточные, расчетные, годовые, событий)

ВНИМАНИЕ: Разрядность счетного механизма накопителей ограничена, при переполнении перенос в старший разряд отбрасывается. Учет энергий осуществляется без учета коэффициентов внешних трансформаторов тока и напряжения.

Для се208/се308 емкость накопителя составляет $1.0e+7$ кВт*ч (10 млн кВт*ч)

zz = 01 - активная потребление A+

zz = 02 - активная генерация A-

zz = 03 - реактивная потребление R+

zz = 04 - реактивная генерация R-

zz = 00 - все энергии по порядку

2а. Запросы энергий по индексу фиксации в архиве и номеру накопителя.

В запросе указывается признак:

a = 0 - фиксация НА (накопители сумма и по тарифам 1..10)

a = 1 - накопление ЗА (накопители сумма и по тарифам 1..10), кроме архива событий

a = 2 - фиксация НА (накопители сумма и по фазам 1..3)

a = 3 - накопление ЗА (накопители сумма и по фазам 1..3), кроме архива событий

для се208_2 - фазные накопление/фиксация энергии определяются как: 1 - учетная по статусу баланса,

2 - учетная по фазе, 3 - учетная по нейтрالي.

для се208_1 - фазные накопление/фиксация энергии определяются как: 1 - учетная по фазе,

2 - учетная по фазе.

Доступные форматы запроса:

- **ENDzz()**, **ENDzz(i.a)**, **ENDzz(i.a,k)**, **ENDzz(i.a,k,n)** - суточный архив
- **ENMzz()**, **ENMzz(i.a)**, **ENMzz(i.a,k)**, **ENMzz(i.a,k,n)** - месячный (расчетный) архив
- **ENYzz()**, **ENYzz(i.a)**, **ENYzz(i.a,k)**, **ENYzz(i.a,k,n)** - годовой архив
- **ENEzz()**, **ENEzz(i.a)**, **ENEzz(i.a,k)**, **ENEzz(i.a,k,n)** - событий архив

Форматы ответа:

ERR18 - запрошенной фиксации еще не существует

В первой строке ответа, передается: идентификатор как в списке, из параметра LSTzz() и один накопитель.

ENDzz(dd.mm.yy, ...) - идентификатор как в списке суточного архива LST01()

ENMzz(bb.mm.yy, ...) - идентификатор как в списке месячного архива LST02()

ENYzz(bb.00.yy, ...) - идентификатор как в списке годового архива LST03()

ENEzz(dd.mm.yy,чч:мм:сс, ...) - идентификатор как в списке архива событий LST06()

Ответ для a=0 или 1 - все накопители по тарифам

ENMzz(bb.mm.yy,сумма) - идентификатор как в списке и сумма энергий по тарифам 1..8 (кВт*ч)

ENMzz(тариф1) - энергия кВт*ч по тарифу 1

.....

ENMzz(тариф10) - энергия кВт*ч по тарифу 10

Ответ для a=2 или 3 - все накопители по фазам

ENMzz(bb.mm.yy,сумма) - идентификатор как в списке и сумма энергий по фазам 1..3 (кВт*ч)

ENMzz(фаза1) - энергия кВт*ч по фазе 1

ENMzz(фаза2) - энергия кВт*ч по фазе 2

ENMzz(фаза3) - энергия кВт*ч по фазе 3

Запрос - **ENDzz()** - выдается суточный архив на всю задействованную глубину, только фиксации НА, сумма и тарифы 1..10. Глубина определяется количеством выполненных фиксаций (максимум - текущие сутки и 127 предыдущих фиксаций)

Запрос - **ENDzz(i.a)** - суточная фиксация i=0 - текущие сутки, 1..127 - предыдущие фиксации, для a = 0, 1 выдается сумма и все тарифы, для a = 2, 3 выдается сумма и все фазы.

Запрос - **ENDzz(i.a,k)** - суточная фиксация i=0 - текущие сутки, 1..127 - предыдущие фиксации, для a = 0, 1, 2, 3 выдается k = какой накопитель {отсчет с 1}.

Запрос - **ENDzz(i.a,k,n)** - суточная фиксация i=0 - текущие сутки, 1..127 - предыдущие фиксации, для a = 0, 1, 2, 3 выдается k = с какого накопителя {отсчет с 1}, n = сколько накопителей.

Запрос - **ENMzz()** - выдается месячный архив на всю задействованную глубину, только для фиксации НА, сумма и тарифы 1..10. Глубина определяется количеством выполненных фиксаций (максимум - текущий месяц и 39 предыдущих фиксаций).

Запрос - **ENMzz(i.a)** - месячная фиксация i=0 - текущий месяц, 1..39 - предыдущие фиксации, для a = 0, 1 выдается сумма и все тарифы, для a = 2, 3 выдается сумма и все фазы.

Запрос - **ENMzz(i.a,k)** - месячная фиксация i=0 - текущий месяц, 1..39 - предыдущие фиксации, для a = 0, 1, 2, 3 выдается k = какой накопитель {отсчет с 1}.

Запрос - **ENMzz(i.a,k,n)** - месячная фиксация i=0 - текущий месяц, 1..39 - предыдущие фиксации, для a = 0, 1, 2, 3 выдается k = с какого накопителя {отсчет с 1}, n = сколько накопителей.

Запрос - **ENYzz()** - выдается годовой архив на всю задействованную глубину, только для фиксации НА, сумма и тарифы 1..10. Глубина определяется количеством выполненных фиксаций (максимум - текущий год и 9 предыдущих фиксаций).

Запрос - **ENYzz(i.a)** - годовая фиксация i=0 - текущий год, 1..9 - предыдущие фиксации, для a = 0, 1 выдается сумма и все тарифы, для a = 2, 3 выдается сумма и все фазы.

Запрос - **ENYzz(i.a,k)** - годовая фиксация i=0 - текущий год, 1..9 - предыдущие фиксации, для a = 0, 1, 2, 3 выдается k = какой накопитель {отсчет с 1}.

Запрос - **ENYzz(i.a,k,n)** - годовая фиксация i=0 - текущий год, 1..9 - предыдущие фиксации, для a = 0, 1, 2, 3 выдается k = с какого накопителя {отсчет с 1}, n = сколько накопителей.

Запрос - **ENEzz()** - выдается архив событий на всю задействованную глубину, только для фиксации НА, сумма и тарифы 1..10. Глубина определяется количеством выполненных фиксаций (максимум - последнее событие и 19 предыдущих событий).

Запрос - **ENEzz(i.a)** - события фиксация i=0 - последнее, 1..19 - предыдущие события, для a = 0 выдается сумма и все тарифы, для a = 2 выдается сумма и все фазы.

Запрос - **ENEzz(i.a,k)** - события фиксация i=0 - последнее, 1..19 - предыдущие события, для a = 0, 2 выдается k = какой накопитель {отсчет с 1}.

Запрос - **ENEzz(i.a,k,n)** - события фиксация i=0 - последнее, 1..19 - предыдущие события, для a = 0, 2 выдается k = с какого накопителя {отсчет с 1}, n = сколько накопителей.

26. Запросы энергий по индексу фиксации в архиве и маске накопителей.

В запросе указывается признак:

a = 0 - фиксация НА (сумма, накопители по тарифам 1..10 и фазам 1..3)

a = 1 - накопление ЗА (сумма, накопители по тарифам 1..10 и фазам 1..3), кроме архива событий

В запросе указывается битовая маска требуемых накопителей, согласно описанию структуры "ENATAR_WIN_BITS" в параметре WTRzz().

Установленный бит (= 1) определяет накопитель, если все биты = 0 - выводится только сумма.

для se208_2 - фазные накопление/фиксация энергии определяются как (fazA = по фазе, fazB = по нейтралю, fazC всегда =0)

для se208_1 - фазные накопление/фиксация энергии не определяются (fazA, fazB, fazC всегда =0)

Доступные форматы запроса:

- **EMDzz(), EMDzz(i.a), EMDzz(i.a,m)** - суточный архив
- **EMMzz(), EMMzz(i.a), EMMzz(i.a,m)** - месячный (расчетный) архив
- **EMYzz(), EMYzz(i.a), EMYzz(i.a,m)** - годовой архив
- **EMEzz(), EMEzz(i.a), EMEz(i.a,m)** - событий архив

Форматы ответа:

ERR18 - запрошенной фиксации еще не существует

В первой строке ответа, передается: идентификатор как в списке, из параметра LSTzz() и сумма энергий по тарифам 1..8 (кВт*ч).

EMDzz(dd.mm.yy,сумма) - идентификатор как в списке суточного архива LST01().

EMMzz(bb.mm.yy,сумма) - идентификатор как в списке месячного архива LST02().

EMYzz(bb.00.yy,сумма) - идентификатор как в списке годового архива LST03().

EMEzz(dd.mm.yy,чч:мм:сс,сумма) - идентификатор как в списке архива событий LST06().

Ответ для a=0 или 1 - все накопители по тарифам 1..10 и фазам 1..3

EMMzz(bb.mm.yy,сумма) - идентификатор как в списке и сумма энергий по тарифам 1..8 (кВт*ч)

EMMzz(тариф1) - энергия кВт*ч по тарифу 1

.....

.....

EMMzz(фаза3) - энергия кВт*ч по фазе 3

Запрос - **EMDzz()** - выдается суточный архив на всю задействованную глубину, начиная с текущих суток, только фиксации НА, сумма, тарифы 1..10, фазы 1..3. Глубина определяется количеством выполненных фиксаций (максимум - текущие сутки и 127 предыдущих фиксаций).

Запрос - **EMDzz(i.a)** - суточная фиксация i = 0 - текущие сутки или 1..127 - предыдущие фиксации, для a = 0 и для a = 1, выдается сумма и все накопители.

Запрос - **EMDzz(i.a,m)** - суточная фиксация i = 0 - текущие сутки или 1..127 - предыдущие фиксации, для a = 0 и для a = 1, выдается сумма и накопители по m = маска накопителей энергий HEX (установленный бит определяет накопитель, для m = 0 - выводится только сумма).

Запрос - **EMMzz()** - выдается месячный архив на всю задействованную глубину, начиная с текущего месяца, только фиксации НА, сумма, тарифы 1..10, фазы 1..3. Глубина определяется количеством выполненных фиксаций (максимум - текущий месяц и 39 предыдущих фиксаций).

Запрос - **EMMzz(i.a)** - месячная фиксация i=0 - текущий месяц, 1..39 - предыдущие месяцы, для a = 0 и для a = 1, выдается сумма и все накопители.

Запрос - **EMMzz(i.a,m)** - месячная фиксация i=0 - текущий месяц, 1..39 - предыдущие месяцы, для a = 0 и для a = 1, выдается сумма и накопители по m= маска накопителей энергий HEX (установленный бит определяет накопитель, для m=0 - выводится только сумма).

Запрос - **EMYzz()** - выдается годовой архив на всю задействованную глубину, начиная с текущего года, только фиксации НА, сумма, тарифы 1..10, фазы 1..3. Глубина определяется количеством выполненных фиксаций (максимум - текущий год и 9 предыдущих фиксаций).

Запрос - **EMYzz(i.a)** - годовая фиксация i=0 - текущий год, 1..9 - предыдущие фиксации, для a = 0 и для a = 1, выдается сумма и все накопители.

Запрос - **EMYzz(i.a,m)** - годовая фиксация i=0 - текущий год, 1..9 - предыдущие фиксации, для a = 0 и для a = 1, выдается сумма и накопители по m= маска накопителей энергий HEX (установленный бит определяет накопитель, для m=0 - выводится только сумма).

Запрос - **EMEzz()** - выдается архив событий на всю задействованную глубину, начиная с последнего события, только фиксации НА, сумма, тарифы 1..10, фазы 1..3. Глубина определяется количеством выполненных фиксаций (максимум - последнее событие и 19 предыдущих событий).

Запрос - **ЕМЕzz(i,a)** - фиксация событий i=0 - последнее событие, 1..19 - предыдущие события, только для a = 0, выдается сумма и все накопители.

Запрос - **ЕМЕzz(i,a,m)** - фиксация событий i=0 - последнее событие, 1..19 - предыдущие события, только для a = 0, выдается сумма и накопители по m= маска накопителей энергий HEX (установленный бит определяет накопитель, для m=0 - выводится только сумма).

2в. Запросы энергий по идентификатору фиксации в архиве и маске накопителей.

Вывод тарифных и фазных накопителей по маске аналогичен выводу по 2а.

В запросе указывается идентификатор, по которому будет выполнен поиск записи в соответствующем архиве:

dd.mm.yy - идентификатор суточного архива (день, месяц, год);

00.mm.yy - идентификатор месячного/расчетного периода архива (месяц, год);

00.00.yy - идентификатор годового архива (год).

Поиск запрошенного идентификатора выполняется, начиная с текущей записи на полное соответствие или до ближайшего идентификатора который меньше запрошенного. Для месячного и годового архива поле день, в котором располагается "дата расчетного периода" в поиске не используется.

Если полученный в ответе идентификатор:

1) соответствует запрошенному идентификатору, значит, получена энергия, зафиксированная на конец запрошенного интервала (сутки, месяц, год) или энергия, накопленная на запрошенном интервале.

2) меньше запрошенного, значит, получена энергия, зафиксированная на конец всех интервалов, в диапазоне от полученного идентификатора до запрошенного. Для накоплений: получена энергия, накопленная на полученном идентификаторе интервала, для остальных интервалов в диапазоне до запрошенного идентификатора, накопленная энергия равна 0.

Если в ответе выдано сообщение ERR18 значит:

1) запрошенный идентификатор больше текущего, поэтому, фиксации еще не существует.

2) в архиве отсутствует идентификатор меньше запрошенного, поэтому, фиксация уже потеряна.

Пример поиска идентификатора в суточном архиве.

Идентификаторы суточного архива LST01:

26.12.16 (текущие), 25.12.16, 24.12.16, 21.12.16, 20.12.16, 15.12.16 (первые сутки).

С 16.12.16 по 19.12.16, 22.12.16 и 23.12.16 счетчик не работал, был выключен.

На запросы EMIzz(14.12.16) или EMIzz(27.12.16) будет получен ответ ERR18

На запрос EMIzz(20.12.16,0,0) - выдать суммарную энергию, зафиксированную НА конец суток 20.12.16г. ответ EMIzz(20.12.16,сумма) - энергия, зафиксированная на конец суток 20.12.16г.

На запрос EMIzz(24.12.16,1,0) - выдать суммарную энергию, накопленную ЗА сутки 24.12.16г. ответ EMIzz(24.12.16, сумма) - энергия, накопленная за сутки 24.12.16г.

На запрос EMIzz(23.12.16,0,0) - выдать суммарную энергию, зафиксированную НА конец суток 23.12.16г. ответ EMIzz(21.12.16, сумма) - энергия, зафиксированная на конец суток 21.12.16г.

Это означает, что счетчик 22.12.16 и 23.12.16 не работал, поэтому энергия на конец этих суток соответствует энергии на конец суток 21.12.16г

На запрос EMIzz(19.12.16,1,0) - выдать суммарную энергию, накопленную ЗА сутки 19.12.16г.

ответ EMIzz(15.12.16, сумма) - энергия, накопленная за сутки 15.12.16г.

Это означает, что счетчик с 16.12.16 по 19.12.16 не работал, поэтому энергия, накопленная за эти сутки равна 0.

В запросе указывается признак:

a = 0 - фиксация НА (сумма, накопители по тарифам 1..10 и фазам 1..3)

a = 1 - накопление ЗА (сумма, накопители по тарифам 1..10 и фазам 1..3)

В запросе указывается битовая маска требуемых накопителей, согласно описанию структуры "ENATAR_WIN_BITS" в параметре WTRzz().

Установленный бит (= 1) определяет накопитель, если все биты = 0 - выводится только сумма.

для се208_2 - фазные накопление/фиксация энергии определяются как (fazA = по фазе, fazB = по нейтрали, fazC всегда =0)

для се208_1 - фазные накопление/фиксация энергии не определяются (fazA, fazB, fazC всегда =0)

Доступные форматы запроса:

- **EMIzz()** - суточный архив полностью, аналог EMDzz()

- **EMIzz(dd.mm.yy)** , **EMDzz(dd.mm.yy,a)** , **EMDzz(dd.mm.yy,a,m)** - суточный архив

- **EMIzz(00.mm.yy)** , **EMDzz(00.mm.yy,a)** , **EMDzz(00.mm.yy,a,m)** - месячный (расчетный) архив

- **EMIzz(00.00.yy)** , **EMDzz(00.00.yy,a)** , **EMDzz(00.00.yy,a,m)** - годового архив

Форматы ответа:

ERR18 - запрошенной фиксации еще/уже не существует

В первой строке ответа, передается: идентификатор как в списке, из параметра LSTzz() и сумма энергий по тарифам 1..8 (кВт*ч).

EMIzz(dd.mm.yy,сумма) - идентификатор как в списке суточного архива LST01().

EMIzz(bb.mm.yy,сумма) - идентификатор как в списке месячного архива LST02().

EMIzz(bb.00.yy,сумма) - идентификатор как в списке годового архива LST03().

Ответ для a=0 или 1 - все накопители по тарифам 1...10 и фазам 1...3

EMIzz(bb.mm.yy,сумма) - идентификатор как в списке и сумма энергий по тарифам 1..8 (кВт*ч)

EMIzz(тариф1) - энергия кВт*ч по тарифу 1

.....

.....

EMIzz(фаза3) - энергия кВт*ч по фазе 3

Запрос - **EMIzz()** - выдается суточный архив на всю задействованную глубину, начиная с текущих суток, только фиксации НА, выдается сумма, тарифы 1..10, фазы 1..3. Глубина определяется количеством выполненных фиксаций (максимум - текущие сутки и 127 предыдущих фиксаций).

Запрос - **EMIzz(dd.mm.yy)** - суточная фиксация с идентификатором dd.mm.yy, только для a=0 фиксация НА, выдается сумма и все накопители.

Запрос - **EMIzz(dd.mm.yy,a)** - суточная фиксация с идентификатором dd.mm.yy, для a=0 и для a=1, выдается сумма и все накопители.

Запрос - **EMIzz(dd.mm.yy,a,m)** - суточная фиксация с идентификатором dd.mm.yy, для a=0 и для a=1, выдается сумма и накопители по m = маска накопителей энергий HEX (установленный бит определяет накопитель, для m = 0 - выводится только сумма).

Запрос - **EMIzz(00.mm.yy)** - месячная фиксация с идентификатором mm.yy, только для a=0 фиксация НА, выдается сумма и все накопители.

Запрос - **EMIzz(00.mm.yy,a)** - месячная фиксация с идентификатором mm.yy, для a=0 и для a=1, выдается сумма и все накопители.

Запрос - **EMIzz(00.mm.yy,a,m)** - месячная фиксация с идентификатором mm.yy, для a=0 и для a=1, выдается сумма и накопители по m = маска накопителей энергий HEX (установленный бит определяет накопитель, для m = 0 - выводится только сумма).

Запрос - **EMIzz(00.00.yy)** - годовая фиксация с идентификатором yy, только для a=0 фиксация НА, выдается сумма и все накопители.

Запрос - **EMIzz(00.00.yy,a)** - годовая фиксация с идентификатором yy, для a=0 и для a=1, выдается сумма и все накопители.

Запрос - **EMIzz(00.00.yy,a,m)** - годовая фиксация с идентификатором yy, для a=0 и для a=1, выдается сумма и накопители по m = маска накопителей энергий HEX (установленный бит определяет накопитель, для m = 0 - выводится только сумма).

3. Максимумы мощности интервала

zz = 01 - активная потребление A+

3а. Запросы максимумов мощности по индексу фиксации в архиве.

Доступные форматы запроса:

- **MAXzz(),MAXzz(i), MAXzz(i,k), MAXzz(i,k,n)**

Форматы ответа:

ERR18 - запрошенной фиксации еще не существует

В первой строке ответа, передается: идентификатор как в списке, из параметра LST05() и одна зона контроля - день, час, минута начала интервала усреднения и максимальная мощность в кВт. Если день равен 0, значит, максимум мощности в этой зоне в этом месяце не контролировался.

Ответ для всех зон:

MAXzz(bb.mm.yy,дд,чч:мм,Зона1) - идентификатор как в списке,

день, час, минута и максимальная мощность 1-й зоны контроля (кВт)

MAXzz(дд,чч:мм,Зона2) - день, час, минута и максимальная мощность 2-й зоны контроля (кВт)

MAXzz(дд,чч:мм,Зона3) - день, час, минута и максимальная мощность 3-й зоны контроля (кВт)

Запрос - **MAXzz()** - выдается архив максимумов мощности на всю задействованную глубину, начиная с текущего месяца, выдаются все зоны. Глубина определяется количеством выполненных фиксаций (максимум - текущий месяц и 39 предыдущих фиксаций).

Запрос - **MAXzz(i)** - фиксация максимумов i=0 - текущий месяц, 1..39 - предыдущие фиксации, все зоны.

Запрос - **MAXzz(i,k,n)** - фиксация максимумов i=0 - текущий месяц, 1..39 - предыдущие фиксации, k= какую зону {отсчет с 1}.

Запрос - **MAXzz(i,k,n)** - фиксация максимумов $i=0$ - текущий месяц, 1..39 - предыдущие фиксации, $k=$ с какой зоны {отсчет с 1}, n = сколько зон.

36. Запросы максимумов мощности по идентификатору фиксации в архиве.

В запросе указывается идентификатор, по которому будет выполнен поиск записи в архиве:

00.mm.yy - идентификатор месячного/расчетного периода архива (месяц, год);

Поиск запрошенного идентификатора выполняется, начиная с текущей записи на полное соответствие или до ближайшего идентификатора который меньше запрошенного. Поле день архива, в котором располагается "дата расчетного периода" в поиске не используется.

Если полученный в ответе идентификатор:

1) соответствует запрошенному идентификатору, значит, получены максимумы мощности, зафиксированные в запрошенном месяце.

2) меньше запрошенного, значит, получены максимумы мощности, зафиксированные в полученном месяце, для остальных месяцев в диапазоне до запрошенного идентификатора, максимумы не определялись.

Если в ответе выдано сообщение ERR18 значит:

1) запрошенный идентификатор больше текущего, поэтому, фиксации еще не существует.

2) в архиве отсутствует идентификатор меньше запрошенного, поэтому, фиксация уже потеряна.

Доступные форматы запроса:

- **MXIzz(), MXIzz(00.mm.yy), MXIzz(00.mm.yy,k), MXIzz(00.mm.yy,k,n)**

Форматы ответа:

ERR18 - запрошенной фиксации еще/уже не существует

В первой строке ответа, передается: идентификатор как в списке, из параметра LST05() и одна зона контроля - день, час, минута начала интервала усреднения и максимальная мощность в кВт.

Если день равен 0, значит, максимум мощности в этой зоне в этом месяце не контролировался.

Ответ для всех зон:

MXIzz(bb.mm.yy,дд,чч:мм,Зона1) - идентификатор как в списке,

день, час, минута и максимальная мощность 1-й зоны контроля (кВт)

MXIzz(дд,чч:мм,Зона2) - день, час, минута и максимальная мощность 2-й зоны контроля (кВт)

MXIzz(дд,чч:мм,Зона3) - день, час, минута и максимальная мощность 3-й зоны контроля (кВт)

Запрос - **MXIzz()** - выдается архив максимумов мощности на всю задействованную глубину, начиная с текущего месяца, выдаются все зоны. Глубина определяется количеством выполненных фиксаций (максимум - текущий месяц и 39 предыдущих фиксаций).

Запрос - **MXIzz(00.mm.yy)** - фиксация максимумов с идентификатором mm.yy, все зоны.

Запрос - **MXIzz(00.mm.yy,k)** - фиксация максимумов с идентификатором mm.yy, k = какую зону {отсчет с 1}.

Запрос - **MXIzz(00.mm.yy,k,n)** - фиксация максимумов с идентификатором mm.yy, k = с какой зоны {отсчет с 1}, n = сколько зон.

4. Профиль

zz = 01 - первый параметр профиля

zz = 02 - второй параметр профиля

zz = 03 - третий параметр профиля

zz = 04 - четвертый параметр профиля

zz = 05 - пятый параметр профиля

zz = 06 - шестой параметр профиля

zz = 00 - все параметры по порядку

Состав параметров профиля и интервал усреднения находятся в параметре PROFI().

Количество интервалов в сутках зависит от интервала усреднения (для 30 мин - 48 интервалов, для 60 мин - 24 интервала).

Глубина архива профиля зависит от интервала усреднения (для 30 мин - 64 суток, для 60 мин - 128 суток) для других интервалов глубина = 3072 / количество интервалов в сутках.

Всего 3072 записи (24 x 128) + 60 записей 25 часа.

Если счетчик не работал в течение календарных суток, эти сутки в архиве профилей отсутствуют.

Содержимое признака интервала профиля:

```
typedef union {
    uint8_t u8;
    struct STAT_REC_ARC {
        uint8_t NoVoid:1; // 0 (NVR) не пустая запись
        uint8_t SumTime:1; // 1 (DST) признак летнего времени
        uint8_t PowDn:1; // 2 (PDN) отключение питания
    };
};
```

```

uint8_t WrRtc:1;          // 3 (CAD) изменение времени (запись в часы)
uint8_t OvrVal:1;         // 4 (ODA) несколько накоплений (два и более проходов)
uint8_t ClrArc:1;         // 5 (CDA) очистка архива
uint8_t FailRtc:1;        // 6 (CIV) сбой часов
uint8_t FailRec:1;        // 7 (ERR) ошибка(испорчены) данные записи
    } bit;
} uStatRec_TypeDef;

```

4а. Запросы данных профиля по индексу фиксации суток в архиве.

Доступные форматы запроса по суткам:

- **VPRzz(), VPRzz(i), VPRzz(i,k), VPRzz(i,k,n)**

Форматы ответа:

ERR18 - запрошенной фиксации еще не существует

В первой строке ответа, передается: идентификатор как в списке, из параметра LST04() и значение параметра на одном интервале суток с признаком интервала uStatRec_TypeDef в формате hex. Остальные интервалы суток в ответе сопровождаются только признаком интервала.

Ответ:

VPRzz(dd.mm.yy,X.X,hex) - идентификатор как в списке,
 значение параметра на интервале суток, признак интервала
 VPRzz(X.X,hex) - значение параметра на следующем интервале суток, признак интервала

 VPRzz(X.X,hex) - значение параметра на следующем интервале суток, признак интервала

Запрос - **VPRzz()** - выдается значение параметра на всех интервалах профиля архива суточных профилей, на всю задействованную глубину, начиная с текущих суток. Глубина определяется количеством выполненных фиксаций (максимум для 60 мин - текущие сутки и 127 предыдущих фиксаций).

Запрос - **VPRzz(i)** - значение параметра на всех интервалах суточного профиля,
 i=0 - текущие сутки, 1..127 - предыдущие фиксации.

Запрос - **VPRzz(i,k)** - значение параметра одного интервала суточного профиля,
 i=0 - текущие сутки, 1..127 - предыдущие фиксации, k = какой интервал {отсчет с 1}

Запрос - **VPRzz(i,k,n)** - значение параметра на интервалах суточного профиля,
 i=0 - текущие сутки, 1..127 - предыдущие фиксации, k= с какого интервала, n = сколько.

. Запросы данных профиля 25 часа сформированного при переходе на зимнее время.

Количество интервалов 25 часа зависит от интервала усреднения (для 30 мин - 2 интервала).

Доступные форматы запроса 25 часа:

- **V25zz(), V25zz(k), V25zz(k,n)**

Форматы ответа:

ERR18 - запрошенной фиксации еще не существует

Ответ:

V25zz(dd.mm.yy,X.X,hex) - идентификатор суток 25 часа,
 значение параметра на интервале 25 часа, признак интервала
 V25zz(X.X,hex) - значение параметра на интервале 25 часа, признак интервала

Запрос - **V25zz()** - значение параметра на всех интервалах профиля 25 часа.

Запрос - **V25zz(k)** - значение параметра одного интервала профиля 25 часа, k= какой интервал {отсчет с 1}.

Запрос - **V25zz(k,n)** - значение параметра на интервалах профиля 25 часа, k= с какого интервала, n= сколько.

Примечание. Для корректного формирования значений интервалов 25 часа профиля:

- Автоматический переход должен быть разрешен ДО наступления даты перехода в счетчике.

- НЕ рекомендуется запись даты/времени в счетчик в день перехода, особенно в интервале перехода.

При "пропуске" полных суток перехода на зимнее время - идентификатор и значения интервалов 25 часа не формируются (остаются "старыми"). Если вообще отсутствовало формирование 25 часа, выдается сообщение (ERR18).

4б. Запросы данных профиля по идентификатору фиксации суток в архиве.

В запросе указывается идентификатор, по которому будет выполнен поиск суточного профиля в архиве:

dd.mm.yy - идентификатор суток профиля в архиве (день, месяц, год);

Поиск запрошенного идентификатора суток выполняется, начиная с текущей записи на полное соответствие или до ближайшего идентификатора который меньше запрошенного.

Если полученный в ответе идентификатор:

1) соответствует запрошенному идентификатору, значит, получены данные интервалов суточного профиля, зафиксированные в запрошенных сутках.

2) меньше запрошенного, значит, получены данные интервалов суточного профиля, зафиксированные в полученных сутках. Для остальных суток в диапазоне до запрошенного идентификатора, все данные интервалов суточного профиля равны 0 с признаком "0x0 - нет накоплений".

Если в ответе выдано сообщение ERR18 значит:

1) запрошенный идентификатор больше текущего, поэтому, фиксации еще не существует.

2) в архиве отсутствует идентификатор меньше запрошенного, поэтому, фиксация уже потеряна.

Доступные форматы запроса по суткам:

- **VPIzz()**, **VPIzz(dd.mm.yy)**, **VPIzz(dd.mm.yy,k)**, **VPIzz(dd.mm.yy,k,n)**

Форматы ответа:

ERR18 - запрошенной фиксации еще/уже не существует

В первой строке ответа, передается: идентификатор как в списке, из параметра LST04() и значение параметра на одном интервале суток с признаком интервала uStatRec_TypeDef в формате hex. Остальные интервалы суток в ответе сопровождаются только признаком интервала.

Ответ:

VPIzz(dd.mm.yy,X.X,hex) - идентификатор как в списке,

значение параметра на интервале суток, признак интервала

VPIzz(X.X,hex) - значение параметра на следующем интервале суток, признак интервала

.....

.....

VPIzz(X.X,hex) - значение параметра на следующем интервале суток, признак интервала

Запрос - **VPIzz()** - выдается значение параметра на всех интервалах профиля архива суточных профилей, на всю задействованную глубину, начиная с текущих суток. Глубина определяется количеством выполненных фиксаций (максимум для 60 мин - текущие сутки и 127 предыдущих фиксаций).

Запрос - **VPIzz(dd.mm.yy)** - значение параметра на всех интервалах суточного профиля с идентификатором dd.mm.yy.

Запрос - **VPIzz(dd.mm.yy,k)** - значение параметра одного интервала суточного профиля с идентификатором dd.mm.yy, k = какой интервал {отсчет с 1}.

Запрос - **VPIzz(dd.mm.yy,k,n)** - значение параметра на интервалах суточного профиля с идентификатором dd.mm.yy, k = с какого интервала {отсчет с 1}, n = сколько интервалов.

//----- (9) Фискальные параметры (только чтение) -----

1. Журналы

Журналы разбиты на 2 группы фиксирующие:

- изменение (программирование) параметров настройки, перечень имен и количество реализованных журналов по списку eNameLogProg_TypeDef;
- контролируемые события, перечень имен и количество реализованных журналов по списку eNameLogEvent_TypeDef.

Глубина и состав данных журнала зависит от типа журнала

Состав журнала: счетчик записей в журнал (32бит) и несколько записей на глубину журнала.

Состав записи:

- штамп времени по часам счетчика 32бит содержит номер секунды от 00:00:00 01.01.2000г.;
- информационное поле 8бит;
- дополнительное поле 32бит, может отсутствовать.

Глубины журналов: 4, 12, 20 записей

(5x12) - журнал без доп поля и глубиной 12 записей

(9x20) - журнал с доп полем и глубиной 20 записей

Структура информационного поля 8 бит при записи параметра

```
struct INFO8_WRITE {
    uint8_t AreaBit:4;      // 3:0 битовое поле
    uint8_t NumPsw:2;       // 5:4 номер пароля "0"-без пароля(JUMP), "1,2,3"-номер
    uint8_t NumPort:2;      // 7:6 номер порта "0"-с кнопок, "1,2,3"- OPTO,com1,com2
} bit;
```

Структура информационного поля 8 бит при контроле события

```
struct INFO8_CHANGE_EVENT {
    uint8_t AreaBit:8;      // 7:0 битовое поле
} bit;
```

typedef enum { //----- Кодировка имени журналов программирования (zz) -----

//----- фиксирует запись параметра в счетчик -----

```
LOG_WriteWatch=1,      // 1 (9x20) Запись ЧПВ (psw,port); (a32 = новое время dd.mm.yy,чч:м:сс, для
u32hex - номер секунды от 00:00:00 01.01.2000г)
LOG_SyncWatch,         // 2 (5x20) Синхронизация времени ЧПВ (от -29 до +29)
LOG_WriteClbrRtc,      // 3 (5x4) Запись поправки хода ЧПВ (psw,port,bit.0)
LOG_WriteTransRtc,     // 4 (5x4) Запись настроек перехода на лето/зиму (psw,port,bit.0)
LOG_WriteMetrol,       // 5 (5x4) Запись метрологии (psw,port,bit.0-сер номер,модель; bit.1-мощн
номин, мощн углов, мощн малая; bit.2-напряж,ток,осн кварц; bit.3-автокалибровка)
LOG_WriteLcd,          // 6 (5x4) Запись настроек индикации и условий CDWIN (psw,port; bit.0-
вывод парам; bit.1-глуб выв арх, время индик; bit.2-выв энергий, выв тарифов, bit.3-условия CDWIN)
LOG_WriteCondUser,     // 7 (5x4) Запись условий учета энергий CDUSE (psw,port,bit.0)
LOG_WriteEnaCache,     // 8 (5x4) Запись разрешения размещения журналов в КЭШе (psw,port,bit.0)
LOG_WritePort,         // 9 (5x12) Запись настроек портов связи (psw,port; bit.0-пароль; bit.1-
адрес,протокол; bit.2-акт,скор,задержка; bit.3-условия,группы сплошн чтен, размер выходного буфера)
LOG_WriteRelAlrm1,     // 10 (5x4) Запись настройки реле сигнализации 1 (psw,port; bit.0-условия;
bit.1-настр контакта; bit.2-время защ интервала, время подтвержд сост; bit.3-время задержки возврата)
LOG_WriteRelAlrm2,     // 11 (5x4) Запись настройки реле сигнализации 2 (psw,port; bit.0-условия;
bit.1-настр контакта; bit.2-время защ интервала, время подтвержд сост; bit.3-время задержки возврата)
LOG_WriteRelAlrm3,     // 12 (5x4) Запись настройки реле сигнализации 3 (psw,port; bit.0-условия;
bit.1-настр контакта; bit.2-время защ интервала, время подтвержд сост; bit.3-время задержки возврата)
LOG_WriteRelLoad,      // 13 (5x4) Запись настроек реле нагрузки (psw,port; bit.0-условия; bit.1-
настр контакта; bit.2-время защ интервала, время подтвержд сост; bit.3-время задержки возврата)
LOG_WriteLevSup,       // 14 (5x12) Запись параметров контроля сети (psw,port; bit.0)
LOG_WritePowMomen,     // 15 (5x12) Запись параметров контроля мгновен мощности (psw,port; bit.0-
лимит мощности; bit.1-время задержки контроля)
LOG_WriteEnLow,        // 16 (5x12) Запись параметров контроля малого потребления (psw,port;
bit.0-лимит энергии; bit.1-время задержки контроля)
LOG_WritePassTar,      // 17 (5x20) Запись пассивной группы тарифных расписаний (psw,port; bit.0-
сут расп; bit.1-сезоны; bit.3- особые даты)
LOG_WriteActTar,       // 18 (5x12) Запись настройки активации тарифных расписаний (psw,port)
LOG_WriteActCfg,       // 19 (5x12) Запись настройки активации расписания зон (psw,port)
LOG_PayMent,           // 20 (9x12) Пополнение счета (psw,port или 1-иниц, 2-очистка счета) (a32
(DEC) = u32hex (HEX) - поправка счета XXX LSB=0,01y.e.)
LOG_21,                // 21 резерв
```

```

LOG_ChangBilling,      // 22 (5x4) Запись даты расчетного периода (psw,port)
LOG_ChangGenTar,      // 23 (5x4) Запись номера "общего" тарифа (psw,port)
LOG_ChangExtTar,      // 24 (5x4) Запись номера тарифа внешнего управления (psw,port)
LOG_ChangEvMatrix,    // 25 (5x12) Запись настройки матрицы событий (psw,port)
LOG_ChangProf,        // 26 (5x12) Запись конфигурации профиля (psw,port)
LOG_ChangModeTar,     // 27 (5x12) Запись режима тарификации (psw,port)
LOG_ChangCntrApos,    // 28 (5x12) Запись режима контроля потребления (psw,port)
LOG_ChangPowIntrv,    // 29 (5x12) Запись параметров контроля мощности интервала (psw,port,
bit.0 - расписание (пассив группа); bit.1 - лимиты мощности (пассив группа); bit.2 - интервал контроля)
LOG_ChangEnLimit,     // 30 (5x12) Запись параметров контроля энергии лимитов (psw,port)
LOG_ChangPrePaid,     // 31 (5x12) Запись параметров контроля энергии предоплаты (psw,port)
LOG_ChangCostEnrg,    // 32 (5x12) Запись стоимости энергии по тарифам (psw,port)
LOG_ChangQuality,     // 33 (5x12) Запись настроек анализа качества (psw,port , bit.0-день недели
начала; bit.1-согласованное напряжение; bit.2-границы контроля; bit.3-максимальная мощность и задержка
для качества ППРФ)
LOG_ChangScTrans,     // 34 (5x4) Запись коэф. внешних трансформаторов (psw,port)
} eNameLogProg_TypeDef; // всего журналов 34

typedef enum { //----- Кодировка имени журналов контролируемых событий (zz) -----
LOG_SelfTest = 1,      // 1 (5x20) Самодиагностика, проблемы бит="1" (bit.0-код, b.1-данные, b.2-
тактирование, b.3-измеритель, b.4,5-рестарт 1-soft,2-wdt,3-pin, b.6-запись, b.7-озу)
LOG_RtcFail,          // 2 (5x12) Состояние ЧРВ (0-норма (при записи в ЧРВ), 1-сбой)
LOG_ClearRetro,       // 3 (5x4) Выполнена очистка ретроспектив
LOG_PowerSupply,      // 4 (5x20) Силовое питание (0=вкл/1=выкл)
LOG_FaultSupply,      // 5 (5x4) Полное пропадание питания
LOG_EmergSupply,      // 6 (5x12) Нештатная ситуация сети (bit.0 - обратн мощность в счетчике на
одно направление, bit.1 - разнонаправленные мощности; bit.2 - неправильное чередование фаз)
LOG_CurrBreakVolt,    // 7 (5x4) Наличие тока при отсутствии напряжения по фазам (bit.0-bit.2 -
фазы A,B,C) - только для 3 фазный
LOG_NoBalancePow=LOG_CurrBreakVolt, // 7 (5x4) НЕ баланс мощности (0=баланс, учет по L,
1=фаза больше, уч.L, 2-нейтраль больше, уч.L, 3-нейтраль больше, учет по N) - только для 1фазный 2эле
LOG_DeviatVolta,      // 8 (5x20) Отклонение частоты и напряжения сети по фазам (по два бита на
частоту b.6-7 и фазу A=b.0-1, B=b.2-3, C=b.4-5 со значениями 0-норма, 1-больше верхн, 2-меньше нижнего,
3-меньше включено=выкл). Для ce208 — частота b.6-7 и фаза L=b.0-1.
LOG_DeviatCurre,      // 9 (5x20) Отклонение тока сети по фазам (по два бита на фазу фазу A=b.0-
1, B=b.2-3, C=b.4-5 со значениями 0-норма, 1-больше верхнего, 2-меньше нижнего, 3-меньше
включено=выкл). Для ce208 L=b.0-1, N=b.2-3.
LOG_QualitySupply,    // 10 (5x20) Оценка качества сети ГОСТ (установленный бит - определяет
нарушение качества, bit.0-частота, bit.1-напряж ф1, bit.2-напряж ф2, bit.3-напряж ф3)
LOG_UnderVoltaQ,      // 11 (9x12) Провал напряжения ГОСТ,
для LOE11,LNE11- (длит, 0.01сек) (a32= глубина,% XX.XX),
для LLE11 - (длит, 0.01сек)(u32hex - глубина, % в формате float32).
LOG_OverVoltaQ,       // 12 (9x12) Перенапряжение ГОСТ,
для LOE12,LNE12- (длит, 0.01сек) (a32= макс значение,B XX.XX),
для LLE12 - (длит, 0.01сек)(u32hex - макс значение,B в формате float32).
LOG_BreakVoltaQ,      // 13 (9x12) Перерыв электроснабжения ГОСТ(сотые секунды длительности
перерыва) (a32 (DEC) = u32hex (HEX) - длительность перерыва, сек XXX)
LOG_AverPowLimit,     // 14 (5x20) Превышение лимита интервала мощности
LOG_ForePowLimit,     // 15 (5x20) Превышение лимита прогнозир мощности (0-нет/1-превыш)
LOG_MomentPowLimit,   // 16 (5x20) Превышение лимита мгновенной мощности (0-нет/1-превыш)
LOG_LowEnergy,        // 17 (5x4) Низкое потребление (0-нет/1-есть низкое потребл)
LOG_EnergyLimit,      // 18 (5x12) Превышение лимитов энергии (bit.0-bit.2 - прев лимит 1,2,3;
bit.4 - есть контроль)
LOG_EvPrePaid,        // 19 (5x12) События предоплатного режима(bit.0-конец предоплаты,bit.1-
конец кредита; bit.4 - есть контроль)
LOG_SyncLimitDay,     // 20 (5x12) Превышен суточный лимит синхронизации (29 секунд)
LOG_SyncLimitMon,     // 21 (5x12) Превышен месячный лимит синхронизации (0-нет/1-превыш)
LOG_StatBattery,      // 22 (5x4) Состояние батарейки (0 - ОК, 1 - плохое, 2 - отсутствует)
LOG_TransSumWin,      // 23 (5x4) Переход на лето/зиму (0- на зиму/1- на лето).
LOG_StatReleAlrm1,    // 24 (5x12) Состояние реле сигнал 1 (код срабатывания: b.7=1 + 0-прямое,
1-32 номер события, ИЛИ код возврата: b.7=0 + 0-авто, 1-по кнопке, 2-по интерф, 3-по кноп и интерф).
LOG_StatReleAlrm2,    // 25 (5x12) Состояние реле сигнал 2 (код срабатывания/возврата).
LOG_StatReleAlrm3,    // 26 (5x12) Состояние реле сигнал 3 (код срабатывания/возврата).

```


LOG_StatReloLoad, // 27 (5x12) Состояние реле нагрузки (код срабатывания/возврата, для реле в подрежиме СПОДЭС номер события и код возврата равен 64).

LOG_ClampsSeal, // 28 (9x4) Пломба зажимов (0-нет вскрытий, 1-обжата, 2-взлом,3-датчик вскрытий/последнее, +4-полное отсутствие питания) (a32 (DEC) = u32hex (HEX) - счетчик вскрытий XXX). Текущее состояние пломбы находится в последней записи.

LOG_CasingSeal, // 29 (9x4) Пломба корпуса (0-нет вскрытий, 1-обжата, 2-взлом,3-датчик вскрытий/последнее, +4-полное отсутствие питания) (a32 (DEC) = u32hex (HEX) - счетчик вскрытий XXX). Текущее состояние пломбы находится в последней записи.

LOG_ExistMagnet, // 30 (5x12) Воздействие магнитом (b.0=1-есть магнит, b.1=1-дефект ДМП)

LOG_SwGrpTar, // 31 (5x12) Переключение активной группы тарифных расписаний

LOG_SwGrpCfg, // 32 (5x12) Переключение активной группы зон мощности

LOG_WrongPassw, // 33 (5x12) Обращение по неверным паролям (port)

LOG_LockPassw, // 34 (5x4) Блокировка по 3 неверным паролям (port)

} eNameLogEvent_TypeDef; // всего журналов 34

zz = 01 ... 34 - номера имени журналов программирования

zz = 01 ... 34 - номера имени журналов контролируемых событий

Если в запросе установить zz = 00, будут выданы все доступные записи журналов с номера 01 по 34

k - смещение запроса начальной записи, т.е. с какой записи от базы начать выдачу (отсчет с 1)

n - запрошенное количество записей на выдачу (отсутствие эквивалентно 1)

rec - номер записи в журнале по порядку в десятичном формате (DEC)

rechex32 - номер записи в журнале по порядку в формате HEX

dd.mm.yy,чч:мм:сс - развернутый штамп времени записи по ЧРВ

stp32hex - штамп времени записи по ЧРВ (формат HEX) номер секунды от 00:00:00 01.01.2000г

hex8 - информационное поле 8 бит (формат HEX)

a32 - доп. информационное поле (может отсутствовать) в индивидуальном формате журнала

u32hex - доп. информационное поле (может отсутствовать) передается в формате uint32 HEX,

содержит информацию в индивидуальном формате журнала (int32/float32)

Информация, заключенная в [квадратные скобки], может отсутствовать.

При отсутствии записей в журнале будет выдана одна запись с номером 0.

При наличии записей в журнале, запись с номером 0 не выдается.

При невозможности получения достоверной записи журнала выдается сообщение в формате ERRnn.

Чтение журналов, начиная с последней записи.

форматы запроса журнала программирования - **LOPzz(), LOPzz(k), LOPzz(k,n)**

форматы запроса журнала событий - **LOEzz(), LOEzz(k), LOEzz(k,n)**

формат ответа развернутый - LOPzz/LOEzz(rec,dd.mm.yy,чч:мм:сс,hex8[a32]).

Запрос: LOPzz(k,n) - выдать записи журнала, начиная с последней (уточняющие индексы k= с какой, n=сколько.) Если запрошено больше доступного количества записей - выдаются только доступные записи.

Запрос: LOPzz() - выдать все доступные записи журнала, начиная с последней

Ответ: LOPzz(rec,dd.mm.yy,чч:мм:сс,hex8[a32]) - все записи журнала.

Чтение журналов, начиная с указанного номера записи.

форматы запроса журнала программирования - **LNPzz(), LNPzz(rec), LNPzz(rec,k), LNPzz(rec,k,n)**

LLPzz(), LLPzz(rec), LLPzz(rec,k), LLPzz(rec,k,n)

форматы запроса журнала событий - **LNEzz(), LNEzz(rec), LNEzz(rec,k), LNEzz(rec,k,n)**

LLEzz(), LLEzz(rec), LLEzz(rec,k), LLEzz(rec,k,n)

формат ответа развернутый - LNPzz/LNEzz(rec,dd.mm.yy,чч:мм:сс,hex8[a32]).

формат ответа свернутый - LLPzz/LLEzz(rechex,stp32hex,hex8[u32hex]).

Запрос: LNPzz(rec,k,n) - выдать записи журнала, начиная с номера записи rec (уточняющие индексы k= с какой, n=сколько.)

если номер записи ЕЩЕ недоступен - выдается номер последней записи журнала,

если номер записи УЖЕ недоступен отсчет ведется с первой (самой старой) доступной записи,

если запрошено больше доступного количества записей - выдаются только доступные записи.

Запрос: LNPzz(nr) - выдать записи журнала, начиная с номера записи rec

если номер записи ЕЩЕ недоступен - выдается номер последней записи журнала,

если номер записи УЖЕ недоступен отсчет ведется с первой (самой старой) доступной записи,

выдаются все доступные записи.

Запрос: LNPzz() - выдать записи журнала начиная с первой (самой старой) записи

выдаются все доступные записи

Ответ: LNPzz(rec,dd.mm.yy,чч:мм:сс,hex8[a32]) - все записи журнала.

LLPzz(rechex,stp32hex,hex8[u32hex]) - первая запись журнала.

[LLPzz](stp32hex,hex8[u32hex]) - последующие записи журнала (номер записи + 1).

Чтение журналов, начиная с указанного номера метки КЭШа.

Дополнительно каждая запись всех журналов помечается сквозным (общим) номером записи.

Эта метка позволяет выполнять чтение записей журналов в порядке их фиксации.

Чтение производится, начиная с указанного общего номера записи, в размере не более затребованного количества записей.

Если запись журнала с общим номером утеряна, в результате замещения свежими записями, эта запись журнала не выдается и не учитывается в затребованном количестве.

Признаком окончания записей журналов, является выдача только общего номера записи (будущего), без информации журнала.

Форматы запроса общего журнала - **GLLUP()**, **GLLUP(num)**, **GLLUP(num,n)**

num - начальный общий номер записи в **десятичном** формате (запрос), диапазон 1...2 147 483 647;

n - количество записей в **десятичном** формате (запрос), диапазон 1...65535;

Формат ответа свернутый - GLLUP(numhex~loghex=rechex,stp32hex,hex8[,u32hex]).

numhex - общий номер записи в **шестнадцатеричном** формате HEX (ответ),

loghex - имя журнала в формате HEX,

для журналов контролируемых событий имя = 0 + код журнала (eNameLogEvent_TypeDef)

для журналов программирования имя = 100 + код журнала (eNameLogProg_TypeDef)

rechex,stp32hex,hex8[,u32hex] - информация записи журнала.

Запрос: GLLUP() - выдать все записи журналов начиная с первой (самой старой) до текущей (самой свежей).

Запрос: GLLUP(num) - выдать все записи журналов, начиная с общего номера записи **num** до текущей (самой свежей).

Запрос: GLLUP(num,n) - выдать записи журналов, начиная с общего номера записи **num** в количестве не более **n** штук

1) признак окончания выдачи записей журналов - выдается только будущий общий номер (текущий общий номер записи + 1)

2) если запрошенный общий номер записи ЕЩЕ недоступен - выдается только будущий общий номер (текущий общий номер записи + 1);

3) если запрошенный общий номер записи УЖЕ недоступен (запись с таким общим номером утеряна, в результате замещения свежими записями) отсчет ведется с ближайшего доступного общего номера;

4) если запрошено больше доступного количества записей - выдаются только записи журналов существующих общих номеров.

Ответ: когда запрошенный номер общей записи ЕЩЕ недоступен

GLLUP(numhex) - номер будущей общий номер записи.

Ответ: когда имеется запрошенный или ближайший номер общей записи

[GLLUP](numhex~loghex=rechex,stp32hex,hex8[,u32hex])

1а. Журналы оценки качества электроэнергии

Чтение журнала начинается с последней записи. Выдаются только существующие записи.

k - смещение запроса начальной записи, т.е. с какой записи от последней начать выдачу (отсчет с 1)

n - запрошенное количество записей на выдачу (отсутствие эквивалентно 1)

rechex32 - номер записи в журнале по порядку в формате HEX

stp32hex - штамп времени записи по ЧРВ (формат HEX) номер секунды от 00:00:00 01.01.2000г

Формат запроса журнала - **LLQzz()**, **LLQzz(k)**, **LLQzz(k,n)**

Если в запросе установить zz = 00, будут выданы все доступные записи журналов с номера 01 по 09

Если смещение запроса больше реальной глубины журнала, выдается одна самая старая запись.

Если смещение запроса попадает на несуществующую запись или записи в журнале отсутствуют, выдается только rechex32=0

Ответ: LLQzz(0).

zz = 01 - журнал отклонения напряжения однофазного счетчика,

zz = 01,02,03 - журналы отклонения фазного напряжения трехфазного счетчика, фазы А, В, С

zz = 04,05,06 - журналы отклонения линейного напряжения трехфазного счетчика, АВ, ВС, СА

Ответ: LLQzz(rechex32, stp32hex,decDeviat,hexM,hexAp, hexAn, hexRp, hexRn, hexPmax, hexSCso).

decDeviat - отклонение напряжения со знаком, LSB=0.01%

hexM - маркер интервала b.0-прерывание напряжения, b.1-провал, b.2-перенапряжение, формат HEX

hexAp - энергия активная потребления накопленная на интервале, LSB=0.001кВт*ч, формат HEX

hexAn - энергия активная отпущенная накопленная на интервале, LSB=0.001кВт*ч, формат HEX
hexRp - энергия реактивная потребления накопленная на интервале, LSB=0.001квар*ч, формат HEX
hexRn - энергия реактивная отпущенная накопленная на интервале, LSB=0.001квар*ч, формат HEX
hexPmax - максимальное значение активной мощности на интервале, LSB=0.001кВт, формат HEX
hexSCso - максимальное значение коэффициента несимметрии по нулевой последовательности на интервале, LSB=0.01%, формат HEX

zz = 07 - журнал превышения напряжения,

Ответ: LLQ08(rechex32,str32hex,end32hex,hexAp,hexAn,hexRp,hexRn,hexUmax,str8hex,end8hex).
hexUmax - максимальное значение напряжения на интервале, LSB=0.01В, формат HEX

zz = 08 - журнал прерывания напряжения,

Ответ: LLQ07(rechex32,str32hex,end32hex,str8hex,end8hex).

str32hex - штамп времени начала события, номер секунды от 00:00:00 01.01.2000г, формат HEX

end32hex - штамп времени конца события, номер секунды от 00:00:00 01.01.2000г, формат HEX

str8hex - штамп времени начала события, сотые доли секунды, формат HEX

end8hex - штамп времени конца события, сотые доли секунды, формат HEX

zz = 09 - журнал оценки качества расчетного периода (месяца).

Ответ: LLQ09(rechex32, str32hex,hexTim,hexCnt,bb.mm.yy).

hexTim - длительность нарушений за месяц- отклонение напряжения, в секундах, формат HEX

hexCnt - количество нарушений за месяц- перенапряжение, формат HEX

bb.mm.yy - идентификатор месяца, как в списке месячного архива LST02() - расчетный день, месяц,

год.

2. Состояние дискретных входов и выходов

zz=01 – дискретные входы

zz=02 – дискретные выходы

Чтение - **STDzz()**

Ответ - STDzz(hex) – значение бита =1 вход/выход замкнут, бит =0 вход/выход разомкнут.

bit.0 - первый вход/выход, bit.7 – восьмой вход/выход

3. Счетчики времени / событий по 8 в группе

zz=01 - группа времени

zz=02 - группа событий

доступные форматы запроса:

Чтение - **CNTzz(), CNTzz(k), CNTzz(k,n)** k= с какого счетчика (начиная с 1), n=сколько

Ответ - CNTzz(n dd.mm.yy,чч:мм:сс) - счетчик и штамп времени очистки счетчика

Запись - CNTzz(hex) установленный бит активирует очистку выбранного счетчика

Биты (0...7) соответствуют номеру счетчика (1...8) в группе

typedef enum { // для zz=01 - группа времени

tiBATT, // таймер работы под батареейкой, с

tiMAGNET, // таймер действия магнита, с

tiDNSUP, // таймер пониженного напряжения, с

tiUPSUP, // таймер повышенного напряжения, с

tiFSUP, // таймер отклонения частоты, с

tiRSVD_6,

tiDEVIQUAL, // таймер отклонений напряжения качества в текущем расчетном периоде (месяц)

tiRSVD_8

} eNameTimeCnt_TypeDef;

typedef enum { // для zz=02 - группа событий

evSUCCESES_OP=1, // успешный прием OPTO

evPROBL_OP, // проблемный прием OPTO

evSUCCESES_C1, // успешный прием com1

evPROBL_C1, // проблемный прием com1

evSUCCESES_C2, // успешный прием com2

evPROBL_C2, // проблемный прием com2

evOVERQUAL, // кол-во перенапряжений качества в текущем расчетном периоде (месяц)

evPROBL_I2C // аппаратные проблемы обмена по i2c

} eNameEventCnt_TypeDef;

4. Накопители синхронизации времени по расчетным периодам (месяц)

всего 12 месяцев (только чтение)

доступные форматы запроса:

Чтение - **SYMON()**, **SYMON(k)**, **SYMON(k,n)** k= с какого месяца (начиная с 1 - текущий), n=сколько

Ответ - **SYMON(-XX,XX)** - количество учтенных секунд отрицательной и положительной синхронизации времени в месяце

5. 6. Идентификаторы

FWDAT(), **FWDAT(ver X.X Mmm dd yyyu=дд.мм.гг,чч:мм:сс= дд.мм.гг,чч:мм:сс)** - версия и дата компоновки ВПО, дата и время "рождения" счетчика, дата и время прошивки ВПО (только чтение).

ID_FW(), **ID_FW(3080v1,hex)** - описание, версия и CRC32 метрологически значимой части ВПО (только чтение)

SCSD_(), **SCSD_(v0,v1,v2,v3,v4,v5)** - значения полей "Система Комплексного Описания версии встроенного Программного обеспечения счетчиков электроэнергии (СКОП)" (только чтение)

7. Статус устройства

STAT_(), **STAT_(hex)** - статусное слово счетчика 32 бит (только чтение)

```
struct STAT_DEV_BITS {
    uint16_t ActTarif:4;      // 3:0 номер активного тарифа T1...T8, 0-нет тарификации
    uint16_t Tarif_9:1;      // 4 накопление в T9
    uint16_t Tarif_10:1;     // 5 накопление в T10
    uint16_t StatBalance:2;   // 7:6 статус баланса (0=баланс, учет по L, 1=фаза больше, уч. по L,
                             // 2=нейтраль больше, уч. по L, 3=нейтраль больше, учет по N) - только для се208_2
    uint16_t ProblemRtc:1;   // 8 проблемы с ЧРВ
    uint16_t SummerRtc:1;    // 9 летнее время ЧРВ (иначе зимнее или выключен автопереход)
    uint16_t EnaCorRtc:1;    // 10 доступна коррекция времени (синхронизация)
    uint16_t LimitSynch:1;   // 11 превышен месячный лимит синхронизации ЧРВ
    uint16_t PositRele:4;    // 15:12 положение 4 реле (0-норма/возврат, 1-сработало) b.12-PC1, b.13-
                             // PC2, b.14-PC3, b.15-PUH
    uint16_t BattLow:1;      // 16 пониженное напряжение батарейки
    uint16_t CrackSeal:1;    // 17 взлом любой пломбы (индикация замка)
    uint16_t FixMagnet:1;    // 18 фиксация воздействия магнитом (индикация подковы)
    uint16_t EmergSupply:1;  // 19 нештатная ситуация сети
    uint16_t RangeVolta:1;   // 20 напряжение любой фазы вне допуска или отсутствует
    uint16_t OverPower:1;    // 21 превышен лимит мощности (прогнозируем или "мгновенная")
    uint16_t OverEnergy:1;   // 22 превышен лимит энергии или предоплаты
    uint16_t forbid23:1;     // 23 занято (используется в технологическом режиме)
    uint16_t LockPassw:1;    // 24 превышен лимит ввода неправильных паролей;
    uint16_t FailSelfTest:1; // 25 проблемы самодиагностики
    uint16_t rsvd26_30:5;    // 30:26 резерв
    uint16_t Technology:1;   // 31 технологический режим
} bit;
```

8. Предварительный статус анализа качества сети

QUALS(), **QUALS(hex)** - значение соответствует hex8 журнала LOG_QualitySupply - HE качество сети ГОСТ (установленный bit.0-частота, bit.1-напр ф1, bit.2-напр ф2, bit.3-напр ф3). Для се208 только bit.0, bit.1.

//----- ТАРИФНЫЕ ГРУППЫ (Sched, Season, ExdayFix) -----

Идентификатор пассивной группы "0" - ноль, активной группы "!0" - не ноль (любое число)

i=0 - пассивная группа, i=!0 - активная группа

Читается - пассивная (без признака в запросе или с признаком "0") и активная (с признаком "!0") группа

Пишется - только пассивная группа

Для активации пассивной группы необходимо записать дату активации группы с признаком активации

Запись расписаний и особых дней выполняется последовательной записью одноименных параметров.

Запись первого параметра очищает список.

Примечание:

Отображение лишних задействованных тарифов 1...8 выключается командой **CMDCT(bit.ClrEnaTarWin)**.

1. Суточные расписания переключения тарифов

zz = 01 ... 32 - номер суточного расписания

максимум 16 точек переключения тарифа

записываются непрерывным массивом, при получении 1 параметра список очищается

доступные форматы запроса - **GRFzz()**, **GRFzz(i)**, **GRFzz(i,k)**, **GRFzz(i,k,n)**, **GRFzz(чч:мм,т)**,

i=0 - пассивная группа, i!=0 - активная (k= с какой точки {отсчет с 1}, n=сколько точек)

Запрос чтения - **GRFzz()**, **GRFzz(0)**, **GRFzz(0,k)**, **GRFzz(0,k,n)** - пассивная группа,

- **GRFzz(1)**, **GRFzz(1,k)**, **GRFzz(1,k,n)** - активная группа

Ответ, запрос записи - **GRFzz(чч:мм,т)** - час, минута переключения и номер тарифа (0,1..8)

По умолчанию:

GRF01(07:00,1)(23:00,2)(00:00,0)...(00:00,0)

GRF02...GRF32(00:00,0)...(00:00,0)

2. Сезонные расписания по дням недели

максимум 12 сезонов

записываются непрерывным массивом, при получении 1 параметра список очищается

доступные форматы запроса - **SESON()**, **SESON(i)**, **SESON(i,k)**, **SESON(i,k,n)**,

SESON(dd.mm,вс-пн-вт-ср-чт-пт-сб),

i=0 - пассивная группа, i!=0 - активная (k= с какого сезона {отсчет с 1}, n=сколько сезонов)

Запрос чтения - **SESON()**, **SESON(0)**, **SESON(0,k)**, **SESON(0,k,n)** - пассивная группа,

- **SESON(1)**, **SESON(1,k)**, **SESON(1,k,n)** - активная группа

Ответ, запрос записи - **SESON(dd.mm,вс-пн-вт-ср-чт-пт-сб)** - день, месяц начала сезона, 7 номеров суточных расписаний по дням недели (0,1..32). dd.mm = 00.00 - сезон выключен.

По умолчанию: **SESON(01.01,1,1,1,1,1,1)(00.00,0,0,0,0,0,0)...**(00.00,0,0,0,0,0,0)

4. Особые даты

максимум 48 дат, одна группа

записываются непрерывным массивом, при получении 1 параметра список очищается

доступные форматы запроса - **EXFIX()**, **EXFIX(i)**, **EXFIX(i,k)**, **EXFIX(i,k,n)**, **EXFIX(dd.mm.yy,p)**,

i=0 - активная группа, i!=0 - активная (k= с какой даты {отсчет с 1}, n=сколько дат)

Запрос чтения - **EXFIX()**, **EXFIX(0)**, **EXFIX(0,k)**, **EXFIX(0,k,n)** - активная группа,

- **EXFIX(1)**, **EXFIX(1,k)**, **EXFIX(1,k,n)** - активная группа

Ответ, запрос записи - **EXFIX(dd.mm.yy,p)** - день, месяц, год, номер суточного расписания (0,1..32)

dd = 0...31 и 255, при значении 255 - все дни заданного месяца и года

mm = 0...12 и 255, при значении 255 - все месяцы с заданным днем и годом

yy = 0...67 и 255, при значении 255 - заданный день и месяц любого года (аналог Особые даты циклические)

dd.mm.yy = 00.00.00 - день выключен и/или p = 0 - день выключен.

По умолчанию: **EXFIX(00.00,0)...**(00.00,0)

5. Дата активации

доступные форматы запроса - **DATPR()**, **DATPR(dd.mm.yy,n)**

Запрос чтения - **DATPR()**

Ответ, запрос записи

- **DATPR(dd.mm.yy,n)** - день, месяц, год и признак запроса активации 0 - пассивный, !0 - запрос активации.

Активация пассивной группы выполняется при установленном признаке активации и даты меньше/равной по часам счетчика. При активации пассивной группы запрос активации снимается, а в параметре находится реальная дата активации по часам счетчика.

Для срочной активации достаточно записать **DATPR(01.01.00,1)**

После активации значения всех параметров в группах идентичны.

По умолчанию: **DATPR(00.00.00,0)**

6. Контрольные коды

1) Контрольный код CRC32 прошивки ВПО (только чтение).

CRCFW(), **CRCFW(XXXXXXXXXX)** – десятичное значение контрольного кода ВПО.

Контрольный код CRC16 рассчитывается по полиному 0x8005 с начальной инициализацией 0xFFFF.

2) Контрольные коды **CRC16** групп тарификации (только чтение).

CRCPR(), **CRCPR(hex,hex)** - Первое значение код пассивной группы, второе значение код активной группы. Расчет контрольного кода выполняется над массивом контрольных кодов частей тарификации (34 кодов частей или 68 байт).

3) Контрольные коды **CRC16** частей тарификации по группам (только чтение)

zz = 01 ... 32 - код суточного расписания 1...32

zz = 33 - код расписания сезонов

zz = 34 - код особых дат

доступные форматы запроса - **CHSzz(),CHSzz(i)**

i=0 - пассивная группа, i!=0 - активная группа

Запрос чтения - CHSzz(),CHSzz(0) - пассивная группа, CHS(1) - активная группа

Ответ: CHSzz(hex)

- Для суточного расписания расчет контрольного кода выполняется над массивом байт 16-и точек переключения тарифов. Каждая точка переключения состоит из 3-х байт (чч,мм,тариф). Общая размерность массива 36 байт.

Для расписания 7:00т1, 23:00т2, остальные байты равны 0 - CRC16 = 0x28AD.

Для расписания все байты равны 0 - CRC16 = 0xFF55.

- Для сезонного расписания расчет контрольного кода выполняется над массивом байт 12-и сезонов. Каждый сезон состоит из 9-и байт (дд,мм,вс,пн,вт,ср,чт,пт,сб). Общая размерность массива 108 байт.

Для расписания 1.1,1,1,1,1,1,1,1,1, остальные байты равны 0 - CRC16 = 0x1F71.

- Для особых дней расчет контрольного кода выполняется над массивом байт 48-и особых дат. Каждая особая дата состоит из 4-х байт (дд.мм.гг.р). Общая размерность массива 192 байт.

Для особых дат все байты равны 0 - CRC16 = 0x1E84.

//----- (7) ПРЯМАЯ запись (время, команды) -----

WATCH(), WATCH(ЧЧ:ММ:СС,дн.ДД.ММ.ГГ) - показания часов

WATCL(), WATCL(hex) - показания часов (номер секунды от 00:00:00 01.01.2000г)

По умолчанию: *WATCH(12:00:00,0.28.06.17)*

Значение даты для записи ограничено диапазоном от 28.06.17г до 31.12.67г

DATE_(), DATE_(дн.ДД.ММ.ГГ) - дата часов (0-вск,1-пнд...6-сбт)

По умолчанию: *DATE_(0.28.06.17)*

Значение даты для записи ограничено диапазоном от 28.06.17г до 31.12.67г

TIME_(), TIME_(ЧЧ:ММ:СС) - время часов

По умолчанию: *TIME_(12:00:00)*

TSUWI(), TSUWI(s,hex,hex) - автоперевод ЧРВ (s,зима,лето), s = 0-выкл/1-вкл

TSUWI(0,740,460) - выкл и настроить зима - октябрь 3 часа, лето - март 2 часа

TSUWI(1,0,0) - включить перевод без изменения настроек

```
struct MARK_TRANSF_PART { // чч+мм+дд
    uint16_t Date:5;        // 4:0 дата 0-посл вск,1..31
    uint16_t Month:4;       // 8:5 месяц 1..12
    uint16_t Hour:5;        // 13:9 час перевода
    uint16_t forb:2;        // 15:13 всегда "0"
```

```
    } part;
```

По умолчанию: *TSUWI(1,0740,0460)*

CORTI(), CORTI(sXX) - поправка хода часов, содержит значение суточного ухода часов, целое десятичное значение со знаком (LSB = 0,1сек). Рассчитанный суточный уход (для спешащих +XX сек, для отстающих -XX сек), при определении по сигналам точного времени за несколько суток, необходимо добавить со знаком к предыдущему значению параметра. Рассчитанный суточный уход, при определении по частотомеру, непосредственно заносится в параметр.

Диапазон значений параметра от -11 сек, до +5,5 сек (-110...+55).

По умолчанию: *CORTI(0)*

/?CTIME!\r\n, /?CTIME(sXX)!\r\n, /?CTIME(ЧЧ:ММ:СС)!\r\n - широковещательная синхронизация часов

CTIME(), CTIME(sXX), CTIME(ЧЧ:ММ:СС) - сеансовая синхронизация часов до +/-29сек в сутки

Синхронизация часов выполняется на время не менее 2-х секунд.

Параметр без значения - корректировка ЧРВ к началу минуты (если время в часах равно 0..29 сек - устанавливается 0 секунд, если время в часах 30..59 сек - устанавливается 59 секунд).

Параметр с заданным значением синхронизации (целое десятичное со знаком) - корректировка ЧРВ на заданную величину, в ближайшие 29 секунд.

Параметр с эталонным временем - корректировка ЧРВ на разницу времени, в ближайшие 29 секунд.

CMDCT(hex) - команды системные, бит=1 - активная, бит=0 - не актуален, формат HEX

```
struct SYS_CMD_BITS {
    uint8_t SetSeal:1;        // 0 Установить электронные пломбы (снять событие взлома,
очистить признак взлома на ЖКИ и в статусе)
    uint8_t KeySeal:1;        // 1 Установить отложенное пломбирование по кнопке (снять
событие взлома, очистить признак взлома на ЖКИ и в статусе)
    uint8_t ClrMagnet:1;      // 2 Очистить признак фиксации магнита на ЖКИ
    uint8_t LockClrEn:1;      // 3 Закрыть возможность очистки энергий с кнопок
    uint8_t ClrEnaTarWin:1;   // 4 Очистка разрешений индикации задействованных тарифов 1..8
    uint8_t ClrPurse:1;       // 5 Очистка кошелька предоплатного режима
    uint8_t rsvd6:1;          // 6 резерв
    uint8_t rsvd7:1;          // 7 резерв
} bit;
```

CMDRL(hex) - команда разрешения возврата реле, бит=1 - активная, бит=0 - не актуален, формат HEX

```
struct POSIT_RELE_BITS {
    uint8_t Alrm_1:1;         // 0 реле ALRM_1 / оптика ТМ_1
    uint8_t Alrm_2:1;         // 1 реле ALRM_2 / оптика ТМ_2
    uint8_t Alrm_3:1;         // 2 реле ALRM_3
    uint8_t Load:1;          // 3 реле нагрузки
    uint8_t rsvd4_7:4;        // 7:4 резерв
} bit;
```

По умолчанию: CMDRL(0)

PRPAY(xxx) - пополнение основного счета, только запись. Десятичное целое со знаком вес LSB=0,01у.е., диапазон значений от -1 000 000 000 до +1 000 000 000 (от -10 000 000,00у.е. до +10 000 000,00у.е.)

PURzz(zz.xx) - состояние счета у.е., только чтение (zz=01-основной счет, zz=02-счет суточного кредита). Диапазон значений от -1E10 до +1E10 (от -10 000 000 000,00у.е. до +10 000 000 000,00у.е.)

По умолчанию: PUR01(0), PUR02(0).

SUBSC(), SUBSC(hex) - абонентский номер, 0..99 999 999 (0..5F5E0FF)

По умолчанию: SUBSC(0)

//----- ПРЯМАЯ запись (ParUser) -----

1. Конфигурация ПОРТОВ СВЯЗИ

NPORT(), NPORT(hex) - номер порта (1=OPTO, 2=com1, 3=com2), по которому выполняется соединение, только чтение.

IDPAS(), IDPAS(str) - адрес-идентификатор протокола МЭК, до 17 символов

По умолчанию: IDPAS()

CONPS(), CONPS(XXXXXX) - пароль модуля связи BLE (Bluetooth_Low_Energy), значение содержит 6 десятичных знаков от 000000 до 999999.

По умолчанию: CONPS(случайное число)

PSWzz(str) - пароли для записи параметров, до 9 символов (пароль zz=01-Администратор, 02-Пользователь)

Пароль администратора не может быть "пустой".

Пароли можно только записывать, чтение паролей запрещено.

По умолчанию: PSW01(777777), PSW02()

По паролю "Администратор" доступны для записи ВСЕ параметры.

Список параметров доступных для записи по паролю "Пользователь" :

PSW02, CORTI, CDWIN, WINzz, WTlzz, WENzz, WTRzz, DEPzz, TIMBP, LSUzz, LIMPP, TIMMP, PERCP, PERCE, DATCF, TAVER, DATZN, DZNzz, LZNzz.

Номер порта zz=01 - оптика, zz=02 - com1, zz=03 - com2, **zz=04 - com3**

CDPzz(), CDPzz(hex) - условия обмена порта интерфейса

struct CONDI_COM_BITS {

```
uint8_t UnLock:1;      // 0 "1" - программирование без кнопки "ДСТП" (МЭК-1107)
uint8_t FixSpeed:1;    // 1 "1" - работать только на начальной скорости (МЭК-1107)
uint8_t DisName:1;     // 2 "1" - запрет вывода имени одноимённых параметров (МЭК-1107)
uint8_t EnaFullRd:1;   // 3 "1" - разрешение вывода профилей в сплошном чтении (МЭК-1107)
uint8_t rsvd4_6:3;     // 6:4 reserved
uint8_t EnaProAdam:1; // 7 работать по проприетарному протоколу ADAM
```

} bit;

Для счетчиков с моделью MODEL.bit.Split = 1, значение bit.0(UnLock) всегда "1".

По умолчанию: CDPzz(4)

PRTzz(), PRTzz(hex) - базовый протокол порта 0=Авто, 1=МЭК-add, 2=МЭК-xor, 3=DLMS, 5=IEC104

(только для zz=03), 6=ADAM (только для zz=02)

По умолчанию: PRTzz(0)

ACTzz(), ACTzz(hex) - время активности интерфейса МЭК, 2..120 сек

По умолчанию: ACTzz(4)

SPPzz(), SPPzz(hex) - скорость обмена интерфейса и начальная скорость протокола МЭК, 0..6 (для com2 - 0..9)

По умолчанию: SPPzz(5)

SPDzz(), SPDzz(hex) - рабочая скорость обмена интерфейса протокола МЭК, 0..6 (для com2 - 0..9)

По умолчанию: SPDzz(5)

DLYzz(), DLYzz(hex) - время задержки перед ответом интерфейса МЭК, 0..200мс

По умолчанию: DLYzz(14)

SIZzz(), SIZzz(hex) - размер выходного буфера только для группового запроса МЭК (максимально допустимая длина ответа), 0-без контроля, 80..60000 байт. Размер ответа обычного запроса не контролируется.

По умолчанию: SIZzz(0)

ADLzz(), ADLzz(hex) - адрес протокола DLMS, 16...16381 (0x0010...0x3FFD)

По умолчанию: ADLzz(10)

EVEzz(), EVEzz(hex,hex) - при чтении выдаются сообщения о **появлении** событий и **пропадании** событий отдельно для каждого порта связи.. Установленные биты указывают на наличие событий разрешенных для сообщения по интерфейсу. Биты (0...31) соответствуют номеру события (1...32) списка eEventAction_TypeDef. При записи выполняется очистка сообщений для порта связи.

2. Конфигурация учета энергий

CDUSE(), CDUSE(hex) - условия учета энергий

```
struct CONDI_USER {
    uint8_t EnaClrEn:1;    // 0 разрешение очистки энергий с кнопок (авто сброс разрешения при
                           // очистке энергий или по новым календарным суткам)
    uint8_t LevBalance:1;  // 1 диапазон контроля баланса мощности (0=5%,1=10%), только se208_2
    uint8_t EnaNeutral:1;  // 2 разрешение (=1) учета по нейтралю при небалансе, только se208_2
    uint8_t DisBalance:1;  // 3 запрет (=1) контроля баланса мощности, только se208_2
    uint8_t rsvd4_5:2;     // 5:4
    uint8_t ThreeWire:1;   // 6 3ф 4пр (=0), 3ф 3 проводное (=1) подключение прибора, только se308
    uint8_t TwoDir:1;      // 7 кол-во направлений учета акт. энергии "0" - одно, "1" - два
} bit;
По умолчанию: CDUSE(6)
```

3. Конфигурация отображения на ЖКИ

CDWIN(), CDWIN(hex) - условия индикации, синхронизации ЧРВ и гудка

```
struct CONDI_WIN {
    uint8_t ModeScr:2;     // 1:0 индикация по неактивн кнопок (0-ничего, 1-начало, 2-авто)
    uint8_t EnaAllScr:1;   // 2 разрешение отображения всех параметров по кнопкам
    uint8_t SumSynchRtc:1; // 3 способ контроля месячной синхронизации (0-мод, 1-знак)
    uint8_t DisSynchRtcKey:1; // 4 запрет синхронизации ЧРВ с кнопок (0-разр, 1-запр)
    uint8_t EnaSetParGrp:1; // 5 разрешение установки параметров в группы индикации (0-все
                           // параметры всегда в заводских группах, 1-разрешен перенос параметров в другие группы)
    uint8_t DisClrBeepKey:1; // 6 запрет очистки реакции гудка кнопкой (0-разр, 1-запр)
    uint8_t EnaQualV:1;    // 7 разрешение отображения признака нарушения КЭ (0-запр, 1-разр)
} bit;
По умолчанию: CDWIN(0)
```

CDAUX(), CDAUX(hex) - дополнительные условия пользователя

```
struct ADD_CONDI_BITS {
    uint8_t EnaEngLocale:1; // 0 Включение Англ. локализации (0 – рус. локаль, 1 – англ. локаль)
    uint8_t rsvd1_7:7;      // 7:1 reserved
} bit;
По умолчанию: CDAUX(0)
```

WINzz(), WINzz(hex) - вывод 48 параметров на индикацию, (параметр zz=01..48)

По умолчанию включены: WIN01(e3), WIN10(d3), WIN12(12), WIN14(12), WIN15(12), WIN16(12), WIN17(12), WIN18(12), WIN19(12), WIN20(12), WIN21(14), WIN22(14), WIN23(14), WIN48(93). Остальные выключены.

При снятом бите 5 параметра CDWIN (EnaSetParGrp) значения битов 0..3 (NumGrp) фиксированы и соответствуют заводским настройкам (не изменяются)

В группу -1 (батарейка) допускается индикация только параметров с номерами winENRG_CURR_TAR, winRTC, winPURSE, winSCOP.

Параметр winENRG_CURR_TAR всегда индицируется в группах -1 (батарейка), 0 (авто), 1 (главная).

```
struct GROUP_WIN_BITS {
    uint8_t NumGrp:4;      // 3:0 вывод в группу 2..15 (0,1-нет вывода)
    uint8_t EnaGrpNum:1;   // 4 разрешение в группу N (NumGrp)
    uint8_t EnaGrpMain:1;  // 5 разрешение в группу 1 (главная)
    uint8_t EnaGrpAuto:1;  // 6 разрешение в группу 0 (авто)
    uint8_t EnaGrpBatt:1;  // 7 разрешение в группу -1 (батарейка)
} bit;
```

```
typedef enum { // список параметров (zz) - первая цифра комментария настройка заводской группы
    winENRG_CURR_TAR=1,    // 1 1 Энергии (сум+тар+фаз) текущих показаний
    winENRG_FIX_DATE,      // 5 2 Энергии (сум+тар+фаз) НА конец СУТОК
    winENRG_ACC_DATE,      // 5 3 Энергии (сум+тар+фаз) ЗА СУТКИ
    winENRG_FIX_MONTH,     // 6 4 Энергии (сум+тар+фаз) НА конец МЕСЯЦА (расч. периода)
```

```

winENRG_ACC_MONTH, // 6 5 Энергии (сум+тар+фаз) ЗА МЕСЯЦ (расч. период)
winENRG_FIX_YEAR, // 7 6 Энергии (сум+тар+фаз) НА конец ГОДА (расч. периода)
winENRG_ACC_YEAR, // 7 7 Энергии (сум+тар+фаз) ЗА ГОД (расч. период)
winENRG_FIX_EVENT, // 8 8 Энергии (сум+тар+фаз) фиксированные по событию
winMAXPOW_MONTH, // 9 9 Максимумы месяца (расч период) по зонам
winRTC, // 3 10 Показания встроенных ЧРВ
winCONFIG_RTC, // 3 11 Конфигурация параметров ЧРВ (зима/лето, калибр, синхр)
winVOLTAGE, // 2 12 Действующее напряжение по фазам
winVOLTLINE, // 2 13 Действующее напряжение межфазное (линейное), для се308
winCURRENT, // 2 14 Действующий ток по фазам
winACTIV_POW, // 2 15 Активная мощность по фазам и сумма
winREACT_POW, // 2 16 Реактивная мощность по фазам и сумма
winAPPAR_POW, // 2 17 Полная мощность по фазам и сумма
winCOS_POW, // 2 18 Коэффициент активной мощности по фазам и сумма
winDEG_IU, // 2 19 Угол между током и напряжением по фазам, дополнительно
для се308 - углы между напряжениями фаз
winFREQU, // 2 20 Частота напряжения питающей сети
winRELE, // 4 21 Статус и настройки реле
winSEAL, // 4 22 Электронные пломбы
winCOM_SETUP, // 4 23 Настройки интерфейсов
winSCALE_TRANSF, // 4 24 Коэффициенты внешних трансформаторов, для се308
winTARIF, //11 25 Состояние и конфигурация тарификации, дата расчета
winSCHED_ACT, //11 26 Тарифное расписание текущих суток
winSCHED_DAY, //11 27 Суточное тарифное расписание (32 расписания)
winSCHED_SEASON, //11 28 Сезонное тарифное расписание (12 сезонов)
winEXDAY_CYC, //11 29 Особые даты циклические (16 дат)
winEXDAY_FIX, //11 30 Особые даты фиксированные (64 даты)
winINTRV_POWER, //12 31 Прогноз мощность, Интервал контроля мощности
winZONE_POWER, //12 32 Расписание зон контроля мощности с лимитами (12 сезон)
winSTATE_QUAL, //13 33 ГОСТ Состояние и настройки контроля качества сети
winSURGE_VOLTA, //13 34 ГОСТ Последн провал и перенапряж (знач и длит, штамп)
winBREAK_VOLTA, //13 35 ГОСТ Перерыв энергоснабжения (длительность, штамп)
winDEVIAT_VOLFR, //13 36 ГОСТ Отклонение напряжен (значения по фазам) и частоты
winBORDER_SUPPLY, //13 37 Границы контроля силовой сети (Udn,Uup,F,Idn,Iup,Ion)
winPURSE, //10 38 Счета предоплаты энергии (кошелек)
winCONFIG_PAY, //10 39 Конфигурация предоплаты энергии
winLIMIT_ENRG, //10 40 Контроль лимитов энергии расчетного периода
winLOW_ENRG, //10 41 Контроль малого потребления
winINOUTPUT, // 4 42 Состояние дискретных входов / выходов
winRSVD_43, // 43
-----
winRSVD_47, // 47
winSCOP, // 3 48 Система комплексного описания ПО (СКОП) и информация
метрологически значимой части ВПО (МЗЧ)
} eNamePar_TypeDef;

```

DEPzz(), DEPzz(hex) - глубина отображения архивов на ЖКИ
Отображается текущее значение + глубина архивов (кроме событий)

```

zz = 01 - суточные, значение 0..99(0x63)
zz = 02 - месячные, значение 0..39(0x27)
zz = 03 - годовые, значение 0..9(0x9)
zz = 04 - профиль (не выводится), значение 0..255(0xFF)
zz = 05 - максимумы, значение 0..12(0x0C)
zz = 06 - события, значение 0..19(0x13), отображается 1 событие при 0 и 20 событий при 19.

```

По умолчанию: DEP01(8), DEP02(6), DEP03(1), DEP04(0), DEP05(6), DEP06(5).

WTIzz(), WTIzz(hex) - время индикации, 5..30 сек, zz=01-автопрокрутка, zz=02-батарейка
По умолчанию: WTI01(5), WTI02(A)

WENzz(), WENzz(hex) - отображение типов энергий, zz=01-автопрокрутка, zz=02-ручной
struct TYPEN_WIN_BITS {
uint8_t Apos:1; // 0 активная потребление (всегда вкл=1)
uint8_t Aneg:1; // 1 активная генерация

```

uint8_t Rpos:1; // 2 реактивная потребление
uint8_t Rneg:1; // 3 реактивная генерация
uint8_t forb4_7:4; // 7:4 не используется
} bit;

```

По умолчанию: WEN01(1), WEN02(F)

WTRzz(), WTRzz(hex) - принудительное отображение тарифов и фаз, zz=01-автопрокрутка, zz=02-ручной

```

struct ENATAR_WIN_BITS {
    uint16_t tar1:1; // 0 тариф 1
    uint16_t tar2:1; // 1 тариф 2
    uint16_t tar3:1; // 2 тариф 3
    uint16_t tar4:1; // 3 тариф 4
    uint16_t tar5:1; // 4 тариф 5
    uint16_t tar6:1; // 5 тариф 6
    uint16_t tar7:1; // 6 тариф 7
    uint16_t tar8:1; // 7 тариф 8
    uint16_t tar9:1; // 8 тариф 9 (парал)
    uint16_t tar10:1; // 9 тариф 10 (парал)
    uint16_t forb10_12:3; // 12:10 нет (всегда =0)
    uint16_t fazA:1; // 13 ce308 - ф1, ce208_2 - фL, ce208_1 - нет (всегда =0)
    uint16_t fazB:1; // 14 ce308 - ф2, ce208_2 - фN, ce208_1 - нет (всегда =0)
    uint16_t fazC:1; // 15 ce308 - ф3, ce208_2 - нет, ce208_1 - нет (всегда =0)
} bit;

```

По умолчанию: WTR01(0), WTR02(3FF)

Примечание:

Отображение лишних задействованных тарифов 1...8 выключается командой **CMDCT(bit.ClrEnaTarWin)**.

TIMBP(), TIMBP(hex) - настройка длительности гудка по событиям, 0x00 - без гудка, 0xFF - без ограничения времени гудка.

```

struct TIME_BEEP_BITS {
    uint8_t Minute:6; // 5:0 длительность гудка, минуты (0..63)
    uint8_t EnaClrDate:1; // 6 длительность гудка до новых суток
    uint8_t EnaClrMonth:1; // 7 длительность гудка до нового месяца
} bit;

```

По умолчанию: TIMBP(0)

SPLIT(XX), SPLIT(buf) - параметр внешнего модуля индикации, только чтение.

При запросе указывается десятичное значение байта списка нажатых кнопок.

```

struct KEY_PRESS_BITS {
    uint8_t Short_Look:1; // 0 короткое нажатие кнопки "ПРСМ"
    uint8_t Long_Look:1; // 1 длинное нажатие кнопки "ПРСМ"
    uint8_t Short_Kadr:1; // 2 короткое нажатие кнопки "КАДР"
    uint8_t Long_Kadr:1; // 3 длинное нажатие кнопки "КАДР"
    uint8_t Short_Acces:1; // 4 короткое нажатие кнопки "ДСТП"
    uint8_t Long_Acces:1; // 5 длинное нажатие кнопки "ДСТП"
} bit;

```

В ответе приходит буфер 24 байта в формате HEX, побайтно двумя символами без разделителей.

Первые 22 это дампы памяти передающегося в драйвер HT16C22G ЖКИ типа Z1003A5HSN8G-B0, последние 2 отведены под статус счетчика и равны 0.

4. Конфигурация РЕЛЕ

zz = 01 - реле сигнализации РС 1/ТМ 1

zz = 02 - реле сигнализации РС 2/ТМ 2

zz = 03 - реле сигнализации РС 3

zz = 04 - реле управления нагрузкой РУН

RCOzz(), RCOzz(hex) - условия реле

```

struct CONDI_RELE_BITS {
    uint8_t TypeTM:3; // 2:0 Вывод по типу телеметрии
    uint8_t Mode:2; // 4:3 Режим реле (0-реле выключено, 1-ТМ, 2-событ, 3-прямое)
    uint8_t AckRetKey:1; // 5 требуется подтверждение возврата кнопкой
    uint8_t AckRetCmd:1; // 6 требуется подтверждение возврата командой
    uint8_t Spodes:1; // 7 Подрежим РУН для системы СПОДЭС (работает при Mode = 2 или 3).
} bit;

```

```

    } bit;
typedef enum { // TypeTM - Тип телеметрии
    TM_RtcBQ=0, // 0 частота часового кварца 512Гц
    TM_Apos, // 1 активная потребление Apos
    TM_Asub, // 2 активная разность |Apos-Aneg|
    TM_Aall, // 3 активная всего (Apos+Aneg)
    TM_Rpos, // 4 реактивная потребление Rpos
    TM_Rneg, // 5 реактивная генерация Rneg
    TM_Rall, // 6 реактивная всего (Rpos+Rneg)
    TM_MainBQ, // 7 частота основного кварца 1Гц
} eEventTM_TypeDef;
typedef enum { // Mode - Режим реле
    mREL_OFF=0, // 0 реле отсутствует
    mREL_TM, // 1 телеметрия по типу
    mREL_EV, // 2 управление по событиям
    mREL_DIR // 3 прямое управление
} eModeRele_TypeDef;

```

По умолчанию: RCO01(B), RCO02(E)или(0), RCO03(18)или(0), RCO04(18)или(0)

При физическом отсутствии реле, в описании параметра MODEL не установлен соответствующий реле бит, режим реле всегда равен mREL_OFF.

RCTzz(), RCTzz(hex) - настройки реле

```

struct CONT_RELE_BITS {
    uint8_t rsvd0_1:2; // 1:0 резерв
    uint8_t NormClose:1; // 2 контакт нормально разомкнут=0/замкнут=1
    uint8_t rsvd3_7:5; // 7:3 резерв
} bit;

```

По умолчанию: RCT01(0), RCT02(0), RCT03(0), RCO04(4)

Для РУН настройка контакта всегда НЗ (нормально замкнут).

RDLzz(), RDLzz(hex) - уровень прямого управления, 0-возврат, 1-сработал

По умолчанию: RDLzz(0)

RTPzz(), RTPzz(hex) - старшая тетрада: время защитного интервала переключения, 0..15сек и младшая тетрада: ширина импульса контакта реле при срабатывании, 0..15сек. Если время защитного интервала меньше аппаратного - выдерживается аппаратный защитный интервал. Если ширина импульса контакта равна 0 - импульс не формируется.

Время защитного интервала переключения - время неготовности реле к переключению контакта.

Для РУН импульс контакта запрещен, ширина импульса всегда равна 0.

По умолчанию: RTPzz(0)

RTCzz(), RTCzz(hex) - время подтверждения состояния, 0..255мин

По умолчанию: RTCzz(1E)

RTDzz(), RTDzz(hex) - время задержки возврата в нормальное состояние, 0..255мин

По умолчанию: RTDzz(0)

EVRzz(), EVRzz(hex) - номер события срабатывания (удержания) реле, 1...32, при 0 - нет события для срабатывания реле, только чтение. Параметр актуален для реле в режиме управления по событиям, для других режимов не имеет смысла.

5. Контрольные уровни

LSUzz(), LSUzz(hex) - уровни контроля сети, 0 - нет контроля

zz = 01 - нижняя граница напряжения, % от Уном (0...10000)

zz = 02 - верхняя граница напряжения, % от Уном (0...10000)

zz = 03 - отклонение частоты от номинальной, LSB=0,01Гц (0...10000)

zz = 04 - нижняя граница тока, % от Iбаз (0...10000)

zz = 05 - верхняя граница тока, % от Iбаз (0...10000)

zz = 06 - граница появления тока, % от Iбаз (0...10000)

По умолчанию: LSU01(5A), LSU02(6E), LSU03(32), LSU04(0), LSU05(0), LSU06(A)

TIMLE(), TIMLE(hex) - время определения низкого потребления, суток (0-нет контроля, 1..45)

По умолчанию: *TIMLE(0)*

TIMMP(), TIMMP(hex) - время задержки контроля мгновенной мощности, сек 0..255

По умолчанию: *TIMMP(0)*

PERCP(), PERCP(hex) - процент контроля мощности интервала, % 0..255 (0 - нет контроля)

По умолчанию: *PERCP(0)*

PERCE(), PERCE(hex) - процент контроля энергии (от первого лимита), % 0..255 (0 - нет контроля)

По умолчанию: *PERCE(0)*

LIMLE(), LIMLE(xx.xxx) - порог энергии малого потребления, кВт*ч (0.0 - нет контроля)

По умолчанию: *LIMLE(0.0)*

LIMMP(), LIMMP(xx.xxx) - порог мгновенной мощности, кВт (0.0 - нет контроля)

По умолчанию: *LIMMP(0.0)*

LIMSY(), LIMSY(hex) - порог месячной синхронизации ЧРВ, сек (0 - нет контроля, 1..65535)

По умолчанию: *LIMSY(0)*

//---- БУФЕРНАЯ запись расписания зон контроля мощности (PowSched) ----

Идентификатор пассивной группы "0" - ноль, активной группы "!0" - не ноль (любое число)

i=0 - пассивная группа, i=!0 - активная группа

Читается - пассивная (без признака в запросе или с признаком "0") и активная (с признаком "!0") группа

Пишется - только пассивная группа

Для активации пассивной группы необходимо записать дату активации группы с признаком активации

Дата активации

доступные форматы запроса - **DATZN(), DATZN(dd.mm.yy, n)**

Запрос чтения - *DATZN()*

Ответ, запрос записи

- *DATZN(dd.mm.yy, n)* - день, месяц, год и признак запроса активации 0 - пассивный, !0- запрос активации.

Активация пассивной группы выполняется при установленном признаке активации и даты меньше/равной по часам счетчика. При активации пассивной группы запрос активации снимается, а в параметре находится реальная дата активации по часам счетчика.

Для срочной активации достаточно записать *DATZN (01.01.00,1)*

После активации значения всех параметров в группах идентичны.

По умолчанию: *DATZN(00.00.00,0)*

DZNzz(), DZNzz(0), DZNzz(1), DZNzz(дд.мм,н1-к1,н2-к2,н3-к3) - 12 расписаний зон контроля мощности интервала (zz=01...12).

Запрос чтения - *DZNzz()*, *DZNzz(0)* - пассивная группа, *DZNzz(1)* - активная группа

Ответ, запрос записи - *DZNzz(дд.мм,н1-к1,н2-к2,н3-к3)*

дд.мм - день и месяц начала действия расписания (00.00 - расписание выключено)

н1 и к1 - начало и конец действия зоны 1 (номер минуты с начала суток 0...1439)

н2 и к2 - начало и конец действия зоны 1 (номер минуты с начала суток 0...1439)

н3 и к3 - начало и конец действия зоны 1 (номер минуты с начала суток 0...1439)

Примечания:

Если минута начала зоны равна минуте окончания зоны - зона действует круглосуточно

Если минута начала зоны равна 0 и минута окончания зоны равна 0 - зона выключена (не активна)

Если минута начала зоны больше минуты окончания зоны - зона действует с переходом через начало суток

По умолчанию: *DZNzz(00.00,0-0,0-0,0-0)*

LZNzz(), LZNzz(0), LZNzz(1), LZNzz(XX.X,XX.X,XX.X) - лимиты мощности 3 зон контроля мощности интервала для 12 расписаний (zz=01...12). Значения в кВт, при значении равном 0.0 контроль лимита в зоне выключен

Запрос чтения - *LZNzz()*, *LZNzz(0)* - пассивная группа, *LZNzz(1)* - активная группа

Ответ, запрос записи - *LZNzz(XX.X,XX.X,XX.X)*

По умолчанию: *LZNzz(0.0,0.0,0.0)*

CRCZN(), CRCZN (hex, hex) - два значения контрольных кодов групп расписания зон контроля мощности интервала (только чтение). Первое для пассивной группы, второе для активной группы.

Контрольные коды частей расписания зон контроля мощности интервала по группам (только чтение)
 zz = 01 ... 12 - код расписания зон контроля мощности интервала 1...12
 доступные форматы запроса - **CHZzz(),CHZzz(i)**
 i=0 - пассивная группа, i!=0 - активная группа
 Запрос чтения - CHZzz(),CHZzz(0) - пассивная группа, CHZ(1) - активная группа
 Ответ: CHZzz(hex)

//---- ПРЯМАЯ запись конфигурации (ParDev) ----

BILNG(), BILNG(hex) - дата расчетного периода (0,1-конец месяца, 2..28).
 Новое значение даты расчетного периода актуализируется при наступлении новых суток.
 По умолчанию: *BILNG(0)*

GETAR(), GETAR(hex) - "общий" тариф безтарифного учета (1..8)
 По умолчанию: *GETAR(5)*

EXTAR(), EXTAR(hex) - тариф внешней тарификации (0-нет, 1..8)
 По умолчанию: *EXTAR(0)*

MDTAR(), MDTAR(hex) - разрешенные режимы тарификации, приоритет режимов по старшинству Event > Extrn > Sched.

```
struct MODE_TARIF_BIT {
    uint8_t Sched:1;          // 0 тарификация по расписанию
    uint8_t Extrn:1;          // 1 тарификация внешняя
    uint8_t Event:1;          // 2 тарификация по событиям
    uint8_t rsvd3_7:5;        // 7:3
} bit;
```

По умолчанию: *MDTAR(1)*

TAVER(), TAVER(hex) - интервал усреднения контроля мощности потребления в минутах (выбирается из ряда 1,2,3,4,5,6,10,12,15,20,30,60)
 По умолчанию: *TAVER(1E)*

PROFI(), PROFI(ht:h1,h2,h3,h4,h5,h6) - ht (hex) - интервал усреднения профиля в минутах (выбирается из ряда 1,2,3,4,5,6,10,12,15,20,30,60), h1..h6 (hex) - идентификаторы 6 данных профиля

```
struct IDENT_PROF_PART { //-- идентификатор данных профиля
    uint8_t Name:4;          // 3:0 имя параметра по списку eNameProf_TypeDef
    uint8_t Faza:2;          // 5:4 фаза параметра по списку eIdFaz_TypeDef
    uint8_t Cond:2;          // 7:6 условие параметра по списку eCondProf_TypeDef
} part;
```

```
typedef enum { //-- Name - имя (0..15) параметра профиля
    eName_Void=0, // 0 пустышка (нет данных)
    eName_Apos,   // 1 активная потребление, кВт*ч
    eName_Aneg,   // 2 активная генерация, кВт*ч
    eName_Rpos,   // 3 реактивная потребление, кВт*ч
    eName_Rneg,   // 4 реактивная генерация, кВт*ч
    eName_rsvd5,  // 5
    eName_rsvd6,  // 6
    eName_rsvd7,  // 7
    eName_rsvd8,  // 8
    eName_F=9,    // 9 частота, Гц
    eName_COS,    // 10 фактор активной мощности
    eName_S,      // 11 полная мощность, кВА
    eName_Q,      // 12 реактивная мощность, квар
    eName_P,      // 13 активная мощность, кВт
    eName_I,      // 14 ток, А
    eName_U,      // 15 напряжение, В
} eNameProf_TypeDef;
```

```
typedef enum { //-- Faza - фаза (0..3) параметра
    p_A=0, // 0 ce308 - ф1, ce208_2 - фаза(Name 9..15)/фаза(Name 1..4), ce208_1 - фаза
    p_B,   // 1 ce308 - ф2, ce208_2 - нейтр(Name 9..15)/нейтр(Name 1..4), ce208_1 - фаза
    p_C,   // 2 ce308 - ф3, ce208_2 - фаза(Name 9..15)/баланс(Name 1..4), ce208_1 - фаза
    p_S    // 3 ce308 - сумма, ce208_2 - нейтр(Name 9..15)/баланс(Name 1..4), ce208_1 - фаза
```

```

} eIdFaz_TypeDef;
typedef enum { //-- Cond - условие накопления (0..3) параметра
    // для Name 1..4
    eCond_Pow=0, // 0 мощность на интервале
    eCond_Enrg, // 1,2,3 энергия на интервале
    // для Name 9..15
    eCond_Aver=0, // 0 среднее арифметическое
    eCond_Min, // 1 минимальное на интервале
    eCond_Max, // 2 максимальное на интервале
    eCond_Rms, // 3 среднее квадратичное
} eCondProf_TypeDef;

```

По умолчанию: ce308 - PROF1(1E:71,72,73,74,0,0), ce208_2 - PROF1(1E:61,62,63,64,0,0) ce208_1 - PROF1(1E:51,52,53,54,0,0)

EVMzz(), EVMzz(hex) - настройка события на действие

```

struct ACTION_RESP_BITS { // действие по событию для
    uint16_t ReleLoad:1; // 0 реле нагрузки РН
    uint16_t ReleAlarm1:1; // 1 реле сигнализации PC 1/TM 1
    uint16_t ReleAlarm2:1; // 2 реле сигнализации PC 2/TM 2
    uint16_t ReleAlarm3:1; // 3 реле сигнализации PC 3
    uint16_t Beep:1; // 4 гудок
    uint16_t FixEnergy:1; // 5 фиксация показаний
    uint16_t AccumTar9:1; // 6 накопление в доп. тариф 9
    uint16_t AccumTar10:1; // 7 накопление в доп. тариф 10
    uint16_t MessUart:1; // 8 сообщение по интерфейсу
    uint16_t SwActTarif:4; // 12:9 переключение активного тарифа на (1..8), 0-выкл
    uint16_t Priority:3; // 15:13 приоритет события переключения тарифа (0..7) 0-высший, при
    // равенстве по очередности в списке событий
} bit;
typedef enum { // перечень (список) событий действия zz=01..zz=32
    EVA_AverPowLimit=1, // 1 превышение мощности интервала лимита любой зоны
    EVA_AverPowPercLimit, // 2 превышение мощности интервала % лимита любой зоны
    EVA_ForePowLimit, // 3 превышение прогноз мощности лимита любой зоны
    EVA_ForePowPercLimit, // 4 превышение прогноз мощности % лимита любой зоны
    EVA_ActZonePower, // 5 любая зона контроля мощности
    EVA_MomentPowLimit, // 6 превышение мгновенной мощности
    EVA_OverVolta, // 7 напряжение любой фазы выше верхней границы (перенапряж)
    EVA_UnderVolta, // 8 напряжение любой фазы ниже нижней границы (провал)
    EVA_BreakVolta, // 9 обрыв любой фазы / нет напряжения (только для ce308)
    EVA_OverCurrent, // 10 ток любой фазы выше верхней границы (перегрузка)
    EVA_UnderCurrent, // 11 ток любой фазы ниже нижней границы (недогрузка)
    EVA_RangeFreq, // 12 частота сети вне диапазона
    EVA_EmergSupply, // 13 нештатная ситуация сети
    EVA_EnergyLimit_3, // 14 превышение лимита энергии 3
    EVA_EnergyLimit_2, // 15 превышение лимита энергии 2
    EVA_EnergyLimit_1, // 16 превышение лимита энергии 1
    EVA_EnergyPercLimit, // 17 превышение процента лимита энергии 1
    EVA_EndPrePaid, // 18 конец предоплаты
    EVA_EndCredit, // 19 конец кредита
    EVA_EndDayCredit, // 20 конец суточного кредита
    EVA_LowEnergy, // 21 низкое потребление
    EVA_ClampsSeal, // 22 взлом пломбы зажимов
    EVA_CasingSeal, // 23 взлом пломбы корпуса
    EVA_ExistMagnet, // 24 наличие магнита
    EVA_ExtInputMain:1; // 25 внешний дискретный вход главный
    EVA_ExtInputNext:1; // 26 внешние дискретные входа остальные
    EVA_rsvd_27, // 27
    EVA_rsvd_28, // 28
    EVA_NoBalancePow, // 29 не баланс мощности (только для ce208_2)
    EVA_rsvd_30, // 30
    EVA_rsvd_31, // 31
    EVA_ClearRetro=32, // 32 очистка накоплений энергий и ретроспективы - действие по событию
    // не настраивается, только фиксация в архиве событий.
}

```


} eEventAction_TypeDef;
По умолчанию: EVMzz(0)

MDENR(), MDENR(hex) - разрешенные режимы контроля потребления

```
struct ENA_CNTR_APOS_BIT {  
    uint16_t AverPower:1;    // 0 контроль интервала мощности  
    uint16_t ForePower:1;    // 1 контроль прогнозируемой мощности  
    uint16_t MomePower:1;    // 2 контроль мгновенной мощности  
    uint16_t LowEnergy:1;    // 3 контроль малого потребления  
    uint16_t Energy:1;       // 4 контроль энергии  
    uint16_t ModeEnergy:1;   // 5 способ контроля энергии (0-по лимитам,1-по предоплате)  
    uint16_t HeapEnergy:4;   // 9:6 контрольная куча контроля энергии по лимитам(0-  
    сумма,1..10-тарифы)  
    uint16_t MaxPowSpod:1;   // 10 максимум мощности на интервале для первой зоны: 1 - всегда  
    круглосуточно (СПОДЭС), 0 - по расписанию зон контроля мощности  
    uint16_t rsvd11_15:5;    // 15:11  
} bit;
```

По умолчанию: MDENR(1F)

LQBzz(), LQBzz(hex) - уровни контроля качества напряжения сети, 0 - нет контроля
zz = 01 - пороговое значение контроля провала напряжения, % от Uопор (0..255)
zz = 02 - пороговое значение контроля перенапряжения, % от Uопор (0..255)
zz = 03 - пороговое значение контроля перерыва питания, % от Uопор (0..255)

По умолчанию: LQB01(5A), LQB02(78), LQB03(50)

QUALD(), QUALD(hex) - день недели начала контроля качества сети (0 - вск, 1 - пнд, 2 - втр, ..., 6 - сбт)
По умолчанию: QUALD(1)

QUALU(), QUALU(xx.xx) - опорное напряжение, по первичной стороне, контроля качества сети, Вольт,
0.0 - нет контроля
По умолчанию: QUALU(230.0)

QMAXP(), QMAXP(xx.xx) - граница максимальной мощности контроля качества сети, кВт, 0.0 - нет
контроля границы
По умолчанию: QMAXP(0.0)

QDLYP(), QDLYP(hex) - значение задержки времени фиксации максимальной мощности, сек (от 0 до 255).
По умолчанию: QDLYP(5)

CTEzz(), CTEzz(yyy) - стоимость 1 кВт*ч энергии по тарифам (zz=01..08). Десятичное целое LSB=0,001у.е.,
диапазон значений от 0 до +1 000 000 000 (от 0,000у.е. до +1 000 000,000у.е.)
По умолчанию: CTEzz(0)

CENzz(), CENzz(yyy) - значения кредита (zz=01) и суточного сверхкредита (zz=02). Десятичное целое
LSB=0,01у.е., диапазон значений от 0 до +1 000 000 000 (от 0,00у.е. до +10 000 000,00у.е.)
По умолчанию: CENzz(0)

LENzz(), LENzz(xx.xx) - значения лимитов энергии (zz=01..03), кВт*ч (0.0 - нет контроля по лимиту)
По умолчанию: LENzz(0.0)

FSCzz(), FSCzz(X.XXX) - коэффициенты внешних трансформаторов (zz=01 - тока, zz=02 - напряжения).
Диапазон значений от 1 до 10000 (для се208 и се308 прямого включения всегда 1).
По умолчанию: FSCzz(1)

//---- СЕРВИСНЫЕ параметры ----

CMDFD(hex) - флаги проблем (чтение) / команда устранения проблем (запись), блоков данных, бит=1 -
активная, бит=0 - не актуален, формат HEX.

Вызывает инициализацию данных при установленном соответствующем флаге проблемы контрольного кода
блока. Код проблемных блоков также отображается на индикаторе при выводе сообщения "Егг 2". При
устранении проблемы параметров метрологии, требуется установка технологической перемычки.

```
struct PROBLEM_STORAGE_BITS {
```

```

// продуктивные параметры
uint16_t SysStat:1;      // 0 текущий статус прибора (SysStat)
uint16_t SysClock:1;    // 1 системное время (SysClock)
uint16_t EnCurr:1;      // 2 энергия нарастающим итогом (CurrentEnergy)
uint16_t EnIntrv:1;     // 3 энергии интервалов (IntervEnergy)
uint16_t PrfCurr:1;     // 4 интервал профиля (CurrentProf)
uint16_t MaxPow:1;      // 5 максимумы мощности (CurrentMaxPow)
uint16_t IndRetro:1;    // 6 индексы архивные (RetroIndex)
uint16_t PayMent:1;     // 7 счета предоплаты (PrePayment)
uint16_t Rsvd8:1;       // 8
uint16_t Rsvd9:1;       // 9
// конфигурационные параметры
uint16_t CntrTarif:1;   // 10 контроль тарификации (ContrTarif)
uint16_t ParUser:1;     // 11 параметры пользователя (интерф, индик, реле..) (ParUser)
uint16_t ParDev:1;      // 12 параметры конфигурации (проф, лим энерг, стоим..) (ParDev)
uint16_t ActZone:1;     // 13 текущие зоны и лимиты мощности (ActZone)
uint16_t ActSched:1;    // 14 текущее суточное расписание (ActSched)
uint16_t ParMetr:1;     // 15 параметры метрологии (Technology)
} bit;

```

CMDDI(hex) - команда активации прямого соединения между портами связи (только запись).

Выполняется соединение портов с настройками связи "8N1" на начальной скорости порта Master (по которому получена команда) сразу после окончания сеанса (переход в начало протокола МЭК).

Соединение разрывается, если по порту Master время между принятыми байтами, "тишина", превысит значение тайм-аута или при выключении счетчика или при передаче последовательности 4 байт с кодом символа '!'.
Прямое соединение: полученный байт, из приемника одного порта, переносится в передатчик другого порта, без контроля на ошибки, содержимое и т.д. При установленном соединении в окна индикации параметров портов отображается обратный отсчет времени до разрыва соединения.

Недопустимые значения команды: номер порта Slave отсутствует в устройстве или равен Master, время тайм-аута менее 5 секунд.

```

struct ARB_DIRECT_BITS {
    uint16_t TimeDir_ta:8;    // 7:0 время тайм-аута прямого канала, 5..255 сек
    uint16_t PortSlave:2;     // 9:8 Slave порт прямого канала (0-OPTO,1-com1,2-com2)
    uint16_t rsvd10_15:6;     // 15:10 резерв
} bit;

```

!!!! - команда разрыва прямого соединения между портами связи.

Выполняется разрыв установленного прямого соединения при передаче непрерывной последовательности (4 символа '!' код 0x21) в порт Master.

//---- ТЕХНОЛОГИЧЕСКИЕ параметры (только чтение) ----

SNUMB(), SNUMB(str) - серийный номер изделия, до 16 символов.

По умолчанию: SNUMB()

MODEL(), MODEL(hex) - модель счетчика, 32 бит

```

typedef union {
    uint32_t u32;
    struct MODEL_DEV_BITS {
        uint16_t Type:4;      // 3:0 тип счетчика по подключению (напряжение и ток)
        uint16_t rsvd4:1;     // 4
        uint16_t Split:1;     // 5 исполнение СПЛИТ = 1 (без ЖКИ и кнопок)
        uint16_t LockClrEn:1;  // 6 полный запрет очистки накопленных энергий (1 - запрет)
        uint16_t TwoSens:1;    // 7 двухэлементный однофазный = 1
    }
    //--
    uint16_t SetConn:4;       // 8:11 битовые флаги установленных модулей связи (b.8 - OPTO,
    b.9 - COM1, b.10 - COM2, b.11 - COM3), OPTO всегда 1
    uint16_t SetRele:4;       // 12:15 битовые флаги установленных реле (b.12 - PC1, b.13 - PC2,
    b.14 - PC3, b.15 - РУН или PC4); PC1 всегда 1
    //--
    uint16_t rsvd16_20:5;     // 20:16 резерв
    uint16_t Aron:1;          // 21 исполнение 3ф 2эл по схеме Арона = 1
    uint16_t LockBoot:1;      // 22 полная блокировка обновления ВПО = 1
    uint16_t Freq60Hz:1;      // 23 частота сети 1=60Гц, 0=50Гц
}

```

```

//--
uint16_t rsvd24_29:6; // 29:24 резерв
uint16_t ExtEprom:1; // 30 установлен дополнительный чип Eprom (+128кБ)
uint16_t Setup:1; // 31 признак активного SETUP (всегда 0)
} bit;
} uModel_TypeDef;

typedef enum { // перечень типов счетчика по подключению (напряжение и ток)
    mod_57_5_10=0, // 0 57,7V 5-10A (се308 - 8000имп/кВт*ч)
    mod_57_5_10_2, // 1 57,7V 5-10A кл.0.2 (се308 - 8000имп/кВт*ч)
    mod_230_5_10, // 2 230V 5-10A (се308 - 4000имп/кВт*ч)
    mod_230_5_60_35, // 3 230V 5-60A 350мкОм (се308 - 800имп/кВт*ч, се208 - 2000имп/кВт*ч)
    mod_230_5_80_16, // 4 230V 5-80A 160мкОм (се308 - 450имп/кВт*ч, се208 - 2000имп/кВт*ч)
    mod_230_5_100_10, // 5 230V 5-100A 100мкОм (се308 - 450имп/кВт*ч, се208 - 2000имп/кВт*ч)
    mod_RSVD_6, // 6
    mod_RSVD_7, // 7
    mod_230_5_100_16, // 8 230V 5-100A 160мкОм (се308 - 450имп/кВт*ч, се208 - 2000имп/кВт*ч)
    ----
    mod_RSVD_15, // 15
} eTypeModel_TypeDef;

typedef enum { // перечень типов счетчика по подключению (напряжение и ток) по схеме АРОНа
    mod_100_5_10=0, // 0 100V 5-10A (се308 - 8000имп/кВт*ч)
    mod_100_5_10_2, // 1 100V 5-10A кл.0.2 (се308 - 8000имп/кВт*ч)
    mod_220_5_10, // 2 220V 5-10A (се308 - 4000имп/кВт*ч)
    mod_220_5_60_35, // 3 220V 5-60A 350мкОм (се308 - 800имп/кВт*ч, се208 - 2000имп/кВт*ч)
    mod_220_5_80_16, // 4 220V 5-80A 160мкОм (се308 - 450имп/кВт*ч, се208 - 2000имп/кВт*ч)
    mod_220_5_100_10, // 5 220V 5-100A 100мкОм (се308 - 450имп/кВт*ч, се208 - 2000имп/кВт*ч)
    mod_RSVD_6, // 6
    mod_RSVD_7, // 7
    mod_220_5_100_16, // 8 220V 5-100A 160мкОм (се308 - 450имп/кВт*ч, се208 - 2000имп/кВт*ч)
    ----
    mod_RSVD_15, //15
} eTypeModel_TypeDef;

```

По умолчанию: *MODEL(B728)* - РУН+РС2+РС1 + COM2+COM1+ОПТО + СПЛИТ + 230V 5-100A 160мкОм

CONST(), CONST(XXX) - постоянная счетчика, имп/кВт*ч.

QUART(), QUART(XX.X) - калибровка основного кварца - погрешность (в процентах) измерения периода тестовой частоты 1 секунда.

По умолчанию: *QUART(0.0)*

ВНИМАНИЕ !!!

При чтении фазных калибровочных коэффициентов:

для се208_2 третье значение всегда 0.0, - **(X.XXX,X.XXX,0.0)**;

для се208_1 второе и третье всегда 0.0 - **(X.XXX,0.0,0.0)**.

LGAIN(), LGAIN(X.XXX,X.XXX,X.XXX) - калибровка активной энергии - погрешности (в процентах) измерения активной мощности по фазам А, В и С при базовом (номинальном) токе и угле равно 0 градусов (COS = 1,0). Погрешность определяется как (Ризм - Рэт) / Рэт.

По умолчанию: *LGAIN(0.0,0.0,0.0)*

UGAIN(), UGAIN(X.XXX,X.XXX,X.XXX) - калибровка напряжения - погрешности (в процентах) измерения напряжения по фазам А, В и С при базовом (номинальном) токе и угле равно 0 градусов (COS = 1,0). Погрешность определяется как (Uизм - Uэт) / Uэт.

По умолчанию: *UGAIN(0.0,0.0,0.0)*

IGAIN(), IGAIN(X.XXX,X.XXX,X.XXX) - калибровка тока - погрешности (в процентах) измерения тока по фазам А, В и С при базовом (номинальном) токе и угле равном 0 градусов ($\cos = 1,0$). Погрешность определяется как $(I_{изм} - I_{эт}) / I_{эт}$.
По умолчанию: $IGAIN(0.0,0.0,0.0)$

LPHI_(), LPHI_(X.XXX,X.XXX,X.XXX) - калибровка угла - погрешности (в процентах) измерения активной мощности по фазам А, В и С при базовом (номинальном) токе и угле равном 60 градусов ($\cos = 0,5$). Итоговая погрешность всегда положительная. Погрешность определяется как $(P_{изм} - P_{эт}) / P_{эт}$.
По умолчанию: $LPHI_(0.0,0.0,0.0)$

LOFFS(), LOFFS(X.XXX,X.XXX,X.XXX) - калибровка малой активной мощности - погрешности (в процентах) измерения активной мощности по фазам А, В и С при токе равном 5% от базового (счетчики прямого включения) или 1% от номинального (счетчики косвенного включения) и угле равном 0 градусов ($\cos = 1,0$). Погрешность определяется как $(P_{изм} - P_{эт}) / P_{эт}$.
По умолчанию: $LOFFS(0.0,0.0,0.0)$

ROFFS(), ROFFS(X.XXX,X.XXX,X.XXX) - калибровка малой реактивной мощности - погрешности (в процентах) измерения реактивной мощности по фазам А, В и С при токе равном 5% от базового (счетчики прямого включения) или 1% от номинального (счетчики косвенного включения) и угле равном 90 градусов ($\sin = 1,0$). Погрешность определяется как $(Q_{изм} - Q_{эт}) / Q_{эт}$.
По умолчанию: $ROFFS(0.0,0.0,0.0)$

//---- РАЗМЕРНОСТЬ значений ВСЕХ параметров ----

Максимальное количество символов (байт) значения параметра между скобками (), при ответе чтения

№	ИМЯ	CE308 и CE208	Содержание
НЕПРОГРАММИРУЕМЫЕ			
1.	STAT_	8	Статус устройства
2.	FWDAT	20	Дата компиляции ВПО
3.	ID_FW	15	Описание, версия и CRC метролог значимой части ВПО
4.	SCSD_	19	СКОП
5.	EVENT	17	Сообщения (set,rst) о событиях
Параметры сети			
6.	VOLTA	6	Фазное напряжение, В
7.	VOLTL	7	Линейное напряжение, В
8.	CURRE	7	Фазный ток, А
9.	POWER	8	Активная мощность, кВт
10.	POWEQ	8	Реактивная мощность, квар
11.	POWES	8	Полная мощность, кВА
12.	COS_f	6	Фактор активной мощности
13.	CORIU	6	Угол между током и напряжением фазы, град
14.	FREQU	5	Частота сети, Гц
15.	PAVER	7	Прогнозир активная потребленная мощность интервала, кВт
Параметры учета			
16.	LSTzz	8 (17)	Списки дат 01-сут, 02-мес, 03-год, 04-проф, 05-макс (06-соб)
17.	ENDzz	13 (22)	Энергии суточные, (для 1-го значения массива)
17a.	EMDzz	13 (22)	Энергии суточные, (для 1-го значения массива)
18.	ENMzz	13 (22)	Энергии месячные, (для 1-го значения массива)
18a.	EMMzz	13 (22)	Энергии месячные, (для 1-го значения массива)
19.	ENYzz	13 (22)	Энергии годовые, (для 1-го значения массива)
19a.	EMYzz	13 (22)	Энергии годовые, (для 1-го значения массива)
20.	ENEzz	13 (31)	Энергии события, (для 1-го значения массива)
20a.	EMEzz	13 (31)	Энергии события, (для 1-го значения массива)
21.	VPRzz	10 (19)	Данные профиля, (для 1-го значения массива)
22.	V25zz	10 (19)	Данные 25 часа профиля, (для 1-го значения массива)
23.	MAXzz	16 (25)	Максимумы мощностей, (для 1-го значения массива)
Журналы, Счетчики			
24.	LOPzz LNPzz	27 (45), (37)	Журналы, стандарт (01),(18)
24a.	LOEzz LNE	27 (33), (37), (31)	Журналы, стандарт (11и12), (13), (28и29)
25.	LLPzz LLEzz	11/15 (20/25)	Журналы, стандарт (LLP01и18, LLE11и12и13и28и29) / с номером записи
25a.	GELzz	19 (28)	Журналы, стандарт (01,18,46,44,45,61,62)
26.	CNTzz	26	Счетчики времени, событий
27.	SYMOM	8	Счетчики месячной синхронизации ЧРВ
28.	QUALS	2	Статус анализа качества сети
29.	CRCCF	9	Контрольные коды пассив и актив конфигураций
30.	CRCPR	9	Контрольные коды пассив и актив тарификации
30a.	CRCZN	9	Контрольные коды пассив и актив зон контроля мощности
31.	CHSzz	4	Контрольный код части тарификации
31a.	CHZzz	4	Контрольный код части зон контроля мощности
ПРОГРАММИРУЕМЫЕ по паролю			
32.	WATCH	19	Значение ЧРВ
32a.	WATCL	8	Значение ЧРВ (номер секунды от 00:00:00 01.01.2000г)
33.	DATE_	10	Дата ЧРВ
34.	TIME_	8	Время ЧРВ
35.	CORTI	4	Калибровка хода часов
36.	CTIME	8	Коррекция (синхронизация) времени
37.	TSUWI	11	Автопереход ЧРВ зима/лето
38.	CMDCT	2	Системные команды
39.	CMDRL	2	Команда разрешения возврата реле
40.	PRPAY	10	Пополнение счета (только запись)

41.	PURzz	15	Состояние счета
42.	PSWzz	8	Пароль для записи параметров (только запись)
43.	IDPAS	17	Адрес - идентификатор счётчика в интерфейсе
44.	PRTzz	1	Базовый протокол обмена порта
45.	ACTzz	2	Время активности порта, сек
46.	SPDzz	1	Номер скорости обмена порта
47.	DLYzz	2	Время задержки перед ответом порта, мс
48.	HIDzz	4	Группы параметров сплошного чтения порта
49.	SIZzz	4	Размер выходного буфера для группового запроса
50.	CDPzz	2	Условия обмена порта
51.	SUBSC	8	Абонентский номер
52.	CDUSE	2	Условия пользователя
53.	CDWIN	2	Условия индикации
54.	WINzz	2	Настройка вывода параметров на индикацию
55.	WTIzz	2	Время индикации Auto(1)/Batt(2)
56.	WENzz	1	Отображение типов энергий Auto(1)/Manu(2)
57.	WTRzz	4	Отображение тарифов энергий Auto(1)/Manu(2)
58.	DEPzz	2	Глубина отображения архивов
59.	TIMBP	2	Длительность реакции гудка
60.	LSUzz	4	Уровни контроля сети
61.	TIMMP	2	Время задержки контроля мгновенной мощности, с
62.	TIMLE	2	Время определения малого потребления, сутки
62.	LIMMP	6	"Разумный" порог мгновенной мощности, кВт
63.	LIMLE	8	"Разумный" порог энергии малого потребления, кВт*ч
64.	PERCP	2	Процент контроля мощности интервала
65.	PERCE	2	Процент контроля энергии (от первого лимита)
66.	LIMSY	4	Порог месячной синхронизации ЧРВ, сек
67.	RCO	2	Условия реле
68.	RCT	2	Настройки реле
69.	RDL	1	Уровень прямого управления
70.	RTP	2	Защитный интервал переключен и ширина импульса контакта
71.	RTC	2	Время подтверждения состояния, мин
72.	RTD	2	Время задержки возврата в нормальное состояние, мин
73.	BILNG	2	Дата расчетного периода
74.	GETAR	1	Номер тарифа для "общего" БЕЗтарифного учета
75.	EXTAR	1	Номер тарифа внешнего управления (0..nTAR_EN)
76.	MDTAR	2	Режимы тарификации бит (Sched,Extern,Event)
77.	TAVER	2	Интервал усреднения мощности в минутах
78.	PROFI	20	Конфигурация профиля
79.	EVMzz	4	Настройка действий по событиям (матрица событий)
80.	MDENR	4	Режимы контроля потребления
81.	LQBzz	2	Верхняя/нижняя граница контроля качества
82.	QUALD	1	День недели начала контроля качества
83.	QUALU	6	Опорное напряжение контроля качества
84.	LENzz	8	"Разумные" значения лимитов 1..3 контроля энергий, кВт*ч
85.	CENzz	10	Кредит, суточный сверхкредит
86.	CTEzz	10	Стоимость энергии по тарифам 1..8
87.	FSCzz	8	Коэф внешних трансформаторов тока и напряжения
88.	DATZN	10	Дата начала действия группы расписаний зон мощности
89.	DZNzz	23	Расписание зон контроля мощности
90.	LZNzz	6	"Разумные" лимиты мощности по зонам, кВт
91.	DATPR	10	Дата начала действия группы расписаний тарифов
92.	GRFzz	7	Суточные расписания переключения тарифов
93.	SESON	26	Сезонные расписания
94.	EXDAY	8	Циклические особые дни
95.	EXFIX	11	Фиксированные особые дни

		<u>ПРОГРАММИРУЕМЫЕ по паролю и джамперу</u>	
96.	SNUMB	15	Серийный номер счётчика
97.	MODEL	4	Модель счетчика энергии
98.	LGAIN	23	Погрешности номинальной активной мощности
99.	LPHI	17	Угловые погрешности
100.	UGAIN	23	Погрешности напряжения
101.	IGAIN	23	Погрешности тока
102.	LOFFS	20	Погрешности малой активной мощности
103.	ROFFS	20	Погрешности малой реактивной мощности
104.	QUART	6	Погрешности частоты основного кварца
105.	PINIT	4	Очистка калибровки (только запись)
106.	BCCLB	2	Пуск накоплений авто калибровки (только запись)
107.	BCVzz	23	Эталонные значения калибруемого параметра (только запись)