Протокол обмена данных SKU-02

Протокол обмена данных со стороны прибора и со стороны персонального компютера (ПК) одинаковый.

Протокол составляет «шапку» и «тело». В «шапке» указываются параметры данных. Ее можно описать следующей структурой.

Примечание : все word, long, float типы байты отображены первым старшем байтом, далее младшим и так далее.

```
typedef struct {
                    byte Start 1;              // первый стартовый байт пакета 0x68
                    word BlokLen;              // полная длина пакета
                    byte Start2;               // второй стартовый байт пакета 0x68
*                   word DeviceName;    //   тип прибора (SKU-02 , 0x02)
*                   float Version;      //   версия прибора
                    byte Status;               //   статус передаваемых данных
*                   word BlokNum;       //  номер блока передаваемых данных
                    unsigned long Ри11Ьеп;//полная длина передаваемых данных
*                   byte Modifikacija;  // модификация прибора
*                   byte DimenE;               // энергия
*                   byte DimenP;               //  мощность
*                   byte FDim;                 //  количество воды
**                  unsigned long DeviceNum; // номер прибора
*           TShortTime FullTime; // реальное время прибора
            byte Command;          // посылаемая команда
            byte Body[n];          // «тело» данных
            } TKepure;
```

следующий 4 байта:

```
byte Suml; byte Sum2; byte Sum3; byte  Stop;
```

BlokLen : длина посылаемого пакета Start 1 ... Stop включительно
Status : A000000B - структура байта, где
A - посылается не последний пакет "1"
B - посылается первый пакет "1"
BlokNum : посылается номер пакета 0 ... m
FullLen : посылается длина "тела"
Command : посланная и принятая команды :

```
#define cm_RealPR          0x20    //  Реальные параметры
# define cm_FullHP         0x21    //полная структура среднечасовой статистики
#define cm_FullDP          0x22    // полная структура среднесуточной статистики
# define cm_FullMP         0x23    //   Среднемесячная статистика
# define cm_AHStopStat     0x24   //  Статистика остановов
#define cm_DeviceSettings  0x25   // Установки прибора
# define cm_UserSettings   0x26   // Установки пользователя
```

```
#defme cm_SetTime          0x27     // Установить время прибора
#define cm_GetTime         0x28   //  Считать время прибора

#define cm_RealStat        0x29   //  Зарезервировано
#define cm_OldRealPR       Ox2A  //  Параметры на отчетный день
#defme cm_PartialHP        0x30  //   Частичная среднечасовая статистика
#defme cm_PartialDP        0x31   //  Частичная среднесуточная статистика

#define cm_NextData        0x70 //   Описание следующего массива данных
#define cm_ErrorData       0x80 //   Ошибочно считаны данные
#define cm_OK              0x90 //   Команда принята
#define cm_NoCommand       OxAO //Нет такой команды
#defmecm NoData            OxBO //  Нет данных
```

Suml : сумма по модулю два Startl ... Body[n] включительно
Sum2 : инвертированная контрольная сумма Startl ... Sum3 включительно
Sum3 : прямая контрольная сумма Startl ... Sum3 включительно

[п] - длина данных в байтах.

При запросе данных из прибора всегда [п] = 0

Где структура данных: TShortTime:

```
typedef struct {
                signed char yy; signed      // год 0..99 /
                char mm;                    /месяц 1..12
                signed char dd;              // день месяца 1..31
                 signed char hh; }          // час 0..23
                TShortTime;
```

При запросе данных из прибора, элементы структуры, обозначенные (*) должны быть заполнены нулями. DeviceNum (**) обозначив нулями, отвечают все приборы, подключенные к линии. Если указан конкретный номер, отвечает только прибор с указанным номером.

Считывание реальных параметров
cm RealPR

Структура данных реальных параметров :

```
typedef struct {
                    long E1;  //Суммарная энергия E1
                    long E2;   // Суммарная энергия E2
                    long VI; //Количество воды V1
                    long V2; // Количество воды V2
                    long   A;   // Время работы прибора (A/3600) h
                    long E;    // Суммарная энергия E
                    long   V;    // Количество воды V
                    float   P;    // Мощность P
                    float   PI; //МощностьP1
                    float   P2; //Мощность P2
                    float   F1; //Расход F1
                    float   F2; //Расход F2
                    float   dTl;
                    float   dT2;
                    float   Tl; // Температура Tl
                    float   T2; // Температура T2
                    float   T3; // Температура T3
                    float   p1; //Давление p1
                    float   p2; //Давлениер2
                    byte    Bus; // Статус прибора
                    byte    Sum;
                    byte    Suml;
                    byte    Sum2;
          // Контрольные суммы рассчитываются, как и в протоколе  связи

                                        } TRealParams;
```

В протоколе связи Body[n] = Body[sizeof(TRealParams)]

Считывание полной среднечасовой, среднесуточной и среднемесячной статистики *cm FullHP. cm FullDP. cm FullMP*

Структура одного блока всех этих статистик :

```
typedef struct {
        byte    Status; // Если - 0, запись правильная,
                        // если не равено 0, запись игнорируется
        long    El; //Суммарная Энергия E1, за час, сутки, месяц
        long    E2; // Суммарная Энергия E2, за час, сутки, месяц
        long    VI; //Количество воды VI, за час, сутки, месяц
        long    V2; // Количество воды V2, за час, сутки, месяц
        long    A;   // время работы, за час, сутки, месяц
        int     T1;  // Средняя T1, за час, сутки, месяц
        int     T2;  // Средняя T2, за час, сутки, месяц
        int     T3;  // Средняя T3, за час, сутки, месяц
        int     T4;  // Средняя T4, за час, сутки, месяц
        int     p1;  // Среднее давление p 1, за час, сутки, месяц
        int     p2;  // Среднее давление p2, за час, сутки, месяц
        byte    Err; // Ошибки работы за час, сутки, месяц
        } TStatParams;
```

Полученные данные располагаются по порядку, считая точку отсчета времени от TShortTime, по порядку убывания

В протоколе связи (sizeof(TKepure)+Body[l]+3+l) < 500 байтов

В протоколе связи (sizeof(TKepure)+Body[2]+3+l) < 500 байтов .

.

В протоколе связи (sizeof(TKepure)+Body[j]+3+l) < 500 байтов

j = sizeof(TStatParams)*k

Считывание реальных параметров на отчетный день

cm_OldRealPR

Структура данных реальных параметров на отчетный день :

```
typedef struct {
        TDateTime  PreCL;
        TRealParams OldRP;
        byte       Sum;
        byte       Suml;
        byte       Sum2;
      } TOldRealParams;
```

Где :

```
typedef struct {
        int       yy;
        signed char mm;
        signed char dd;
        signed char hh;
        signed char nn;
        signed char ss;
        signed char wd;
      } TDateTime;
```

В протоколе связи Body[n] =
Body[sizeof(TOldRealParams)]

Считывание статистики остановок

cm_AHStopStat

Структура данных статистики остановок :

```
typedef struct {
                byte        Status; // Если - 0, запись правильная,
                                    // Если не равен 0, запись игнорируется
                byte        StopType;
                TStopParams Stop;
              } TStop;
```

Где :

StopType :

```
#defme    e Tl    0x02
#define   e T2    0x04
#define   e_T3    0x20
#define   e Fl    0x08
#define   e F2    0x80
#define   e OF1   0x01
#define   e_OF2   0x40
#define   e_Stop  0x10
```

```
typedef struct {
                TShortTimel WT1;
                TShortTime2 WT2;
              } TStopParams;
```

```
typedef struct {
                int         yy;
                signed char mm;
                signed char dd;
                signed char hh;
              } TShortTimel;
```

```
typedef struct {
                signed char mm;
                signed char dd;
                signed char hh;
                signed char nn;
                signed char ss;
              } TshortTime2;
```

Полученные данные располагаются по порядку, считая точку отчета времени от TShortTime, по порядку убывания.
В протоколе связи (sizeof(TKepure)+Body[l]+3+l) < 500 байтов
В протоколе связи (sizeof(TKepure)+Body[2]+3+l) < 500 байтов

В протоколе связи (sizeof(TKepure)+Body[j]+3+l) < 500 байтов
j = sizeof(TStatParams)*k