

**ОПИСАНИЕ ПРОТОКОЛА ОБМЕНА
КОРРЕКТОРА ГАЗА В25**

версия 1.2

Физический протокол

Связь на физическом уровне выполняется с помощью переходника RS232 либо инфракрасного порта

формат передаваемого байта:

Data bits 8

Parity none

Stop bits 1

Bit per second 4800 / 9600

для инфракрасного порта - 115200

Формат протокола (запрос данных)

№ в пакете	Обозначение	Описание	Байт
0	AddrSlave	Адрес устройства (B25)	1 (unsigned char)
1	Len (0x08)	Длина пакета = 0x08	1 (unsigned char)
2	Code Net (0x8E)	Код запроса	1 (unsigned char)
3	Code Struct	Номер структуры	1 (unsigned char)
4	NMOD	Номер модуля (0,1)	1 (unsigned char)
5	CountPack	Кол. запросов	1 (unsigned char)
6	IDPack[0]	Идентификатор пакета	2 (unsigned short)
7	IDPack[1]		
8	NCount[0]	Номер ячейки архива	4 (unsigned long)
9	NCount[1]		
10	NCount[2]		
11	NCount[3]		
12	AddrMaster	Адрес Мастера (компьютер)	1 (unsigned char)
13	CRC[0]	Контрольная сумма (считается суммированием всех байтов пакета исключая CRC)	2 (unsigned short)
14	CRC[1]		

Формат протокола (ответ на запрос)

№ в пакете	Обозначение	Описание	Байт
0	AddrMaster	Адрес Мастера (компьютер)	1 (unsigned char)
1	Len	Длина пакета	1 (unsigned char)
2	IDPack	Идентификатор пакета	2 (unsigned short)
3	data[]	данные	
4	AddrSlave	Адрес устройства (B25)	1 (unsigned char)
5	CRC[0]	Контрольная сумма (считается суммированием всех байтов пакета включая CRC)	2 (unsigned short)
6	CRC[1]		

Номер структуры:

Команда 0 [0x00] - чтение ID

Команда 1 [0x01] - чтение маркеров

Команда 2 [0x02] - чтение текущих значений

Команда 3 [0x03] - чтение конфигурации

Команда 4 [0x04] - чтение названия предприятия

Команда 5 [0x05] - чтение адреса установки

Команда 6 [0x06] - чтение имя счетчика

Команда 7 [0x07] - чтение имя пользователя

Команда 8 [0x08] - чтение имя поставщика

Команда 9 [0x09] - состав AGA

Команда 10 [0x0A] - читаем структуры последних редакций

Команда 11 [0x0B] - читаем 1 запись часового архива

Команда 12 [0x0C] - читаем 1 запись суточного архива

Команда 13 [0x0D] - читаем 1 запись доп. информации. суточного архива

Команда 14 [0x0E] - читаем 1 запись аварийного архива

Команда 15 [0x0F] - читаем 1 запись архива редакций

Команда 16 [0x10] - читаем 1 месячного архива

Описание структур:

Код структуры [0x00]

Описание подключенного устройства

typedef struct

```
{  
    unsigned short DeviceType; - индикация типа устройства  
    unsigned short Version; - версия программного обеспечения устройства  
    unsigned long ID; - локальный идентификационный номер  
    unsigned long Serial; - серийный номер устройства  
    unsigned short Condition; - состояние устройства  
}
```

SIDDeviceB25;

Код структуры [0x01]

Описание архивов устройства

```
typedef struct {  
    unsigned long DATECreateHourArhive; - дата создания часового архива  
    unsigned long DaySPFlash; - маркер на новую запись суточного архива  
    unsigned long MonthSPFlash; - маркер на новую запись месячного архива  
    unsigned long HourSPFlash; - маркер на новую запись часового архива  
    unsigned long EditSPFlash[2]; - маркер на новую запись архива коррекций  
    unsigned long ErrorSPFlash[2]; - маркер на новую запись архива аварий  
    unsigned long UserSPFlash; - маркер на новую запись дополнительного архива  
  
    unsigned long PointDayFlash; - адрес начала суточного архива  
    unsigned long PointMonthFlash; - адрес начала месячного архива  
  
    unsigned long PointHourFlash; - адрес начала часового архива  
    unsigned long PointEditFlash[2]; - адрес начала коррекций архива  
    unsigned long PointErrorFlash[2]; - адрес начала аварийного архива  
    unsigned long PointUserFlash; - адрес начала дополнительного архива  
  
    unsigned long MaxDayFlash; - максимальное кол. записей суточного  
    unsigned long MaxMonthFlash; - максимальное кол. записей часового  
    unsigned long MaxHourFlash; - максимальное кол. записей часового  
    unsigned long MaxUserFlash; - максимальное кол. записей пользовательского  
    unsigned long MaxEditFlash; - максимальное кол. записей коррекций  
    unsigned long MaxErrorFlash; - максимальное кол. записей аварийного архива  
} SLOG_B25;
```

Команда 2 [0x02]

Описание текущих значений

```
typedef struct { // Описание измерительного модуля.  
    unsigned short VerBus; - версия прокола  
    unsigned long SerialDevice; - серийный номер устройства  
    unsigned long SerialSensorPressure; - серийный номер д. давления  
    unsigned short DN;  
    unsigned short AntiContact; - антидребезг контакта  
    float Cimp; - Коэффициент преобразования ( кол. импульсов на мЗ)  
    unsigned long Qmin; - минимальный расход  
    unsigned long Qmax; - максимальный расход  
    unsigned long TimeQStop; - время до остановки счетчика  
  
    float Pmin; - минимальное давление, МПа  
    float Pmax; - максимальное давление, МПа  
    float Tmin; - минимальная температура, К  
    float Tmax; - максимальная температура, К  
    unsigned char MeasuringCycle; - цикл измерения, сек  
    unsigned char Contract_Time; - контрактный час
```

```

float KT;           - коэффициент датчика Термометра
float KP;           - коэффициент датчика Давления
unsigned long TimePlusHour; - дата перехода +1 час
unsigned long TimeMinusHour; - дата перехода -1 час
unsigned char MaxDevSys; - количество модулей установленное в устройстве
unsigned char WaitError; - время пропуска при аварии сек
unsigned short Setting; - флаги настроек
} STMPNetSettingNow;

```

Вывод данных :

```

unsigned short VerBus printf("%5u ", VerBus)
unsigned long SerialDevice printf("%5u ", SerialDevice)
unsigned long SerialSensorPressure printf("%5u ", SerialSensorPressure)
unsigned short DN; printf("%4u ", DN)

```

```

unsigned short AntiContact printf("%4u ", AntiContact)
float Cimp printf("%10.3f ", Cimp)
unsigned long Qmin fl = Qmin/1000.0 printf("%10.3f ", fl)
unsigned long Qmax fl = Qmax /1000.0 printf("%10.3f ", fl)
unsigned long TimeQStop printf("%4u ", TimeQStop)

```

```

float Pmin printf("%10.3f ", Pmin)
float Pmax printf("%10.3f ", Pmax)
float Tmin printf("%10.1f ", Tmin)
float Tmax printf("%10.1f ", Tmax)
unsigned char MeasuringCycle printf("%2d ", MeasuringCycle)
unsigned char Contract_Time printf("%2d ", Contract_Time)
float KT printf("%10.6f ", KT)
float KP printf("%10.6f ", KP)
unsigned long TimePlusHour TimeNormal(TimePlusHour)
unsigned long TimeMinusHour TimeNormal(TimeMinusHour)
unsigned char MaxDevSys printf("%2d ", MaxDevSys)
unsigned char WaitError printf("%2d ", WaitError)
unsigned short Setting printf("%2d ", Setting)

```

Код структуры [0x03]

Текущие значения одного модуля

```

typedef struct {
    unsigned short VerBus; - версия структуры
    unsigned long CurrentDateTime; - текущее время устройства
    unsigned long WorkTime; - рабочее время устройства

    union { double Vk; unsigned char dataVk[8]; } - объем по счетчику, м3
    union { double Vw; unsigned char dataVw[8]; } - объем с учётом работы при Qmin, м3
    union { double Vc; unsigned char dataVc[8]; } - объем приведенный к нормальным условиям, м3
    union { double Aw; unsigned char dataAw[8]; } - объем рабочий при авариях, м3
    union { double Ac; unsigned char dataAc[8]; } - объем прив. к норм. условиям при аварии, м3

    unsigned long Qmoment; - текущий рабочий расход м3/ч
    float Tw; - рабочая температура, К
    float Pw; - рабочее давление, МПа

    float Pnow; - измеренное давление, МПа

```

float Tnow; - измеренная температура, К

float Cf; - коэффициент коррекции
float K; - коэффициент сжимаемости
float Zc; - Фактор сжимаемости при ст. условиях

float Xa; - Состав газа, Xa %
float Xy; - Состав газа, Xy %
float Ro; - Состав газа, Ro кг/м3
float Pa; - Константа по давлению, МПа
float Ta; - Константа по температуре, К

unsigned short ErrorSemaphore; - флаги состояния устройства
unsigned short Operator; - количество изменений категории Оператор
unsigned short Admin; - количество изменений категории Администратор
unsigned short BitSetting; - Системные настройки модуля
unsigned short Vbat; - Напряжение питания батареи

} STEMPNetInfoNow;

Вывод данных :

```
unsigned short VerBus printf("%5u ", VerBus)
unsigned long CurrentDateTime TimeNormal(CurrentDateTime)
unsigned long WorkTime TimeSec(WorkTime)

double Vk; printf("%10.1f ", Vk)
double Vw; printf("%10.1f ", Vw)
double Vc; printf("%10.1f ", Vc)
double Aw; printf("%10.1f ", Aw)
double Ac; printf("%10.1f ", Ac)

unsigned long Qmoment; fl = Qmoment /1000.0 printf("%10.3f ", fl)
float Tw; printf("%10.2f ", Tw)
float Pw; printf("%10.5f ", Pw)

float Pnow; printf("%10.5f ", Pnow)
float Tnow; printf("%10.5f ", Pnow)

float Cf; printf("%10.5f ", Cf)
float K; printf("%10.5f ", K)
float Zc; printf("%10.5f ", Zc)

float Xa; printf("%10.4f ", Xa)
float Xy; printf("%10.4f ", Xy)
float Ro; printf("%10.4f ", Ro)
float Pa; printf("%10.4f ", Pa)
float Ta; printf("%10.2f ", Ta)

unsigned short ErrorSemaphore;
unsigned short Operator; printf("%5u ", Operator)
unsigned short Admin; printf("%5u ", Admin)
unsigned short BitSetting; printf("%5u ", BitSetting)
unsigned short Vbat; fl = Vbat/100.0 printf("%10.2f ", fl)
```

Код структуры [0x04]

Название предприятия

Возвращаемые данные - char str[80]

Код структуры [0x05]
Адрес установки
Возвращаемые данные - char str[80]

Код структуры [0x06]
Текстовое описание счетчика
Возвращаемые данные - char str[40]

Код структуры [0x07]
Текстовое описание Пользователя
Возвращаемые данные - char str[20]

Код структуры [0x08]
Текстовое описание Поставщика
Возвращаемые данные - char str[20]

Команда 9 [0x09]
Состав AGA92-DC :

```
typedef struct {  
    float Xn[18];  
} SFormatXn;  
  
Xn[0]    "Метан"           CH4"    printf("%10.5f ", Xn[0])  
Xn[1]    "Этан"           C2H6"   printf("%10.5f ", Xn[1])  
Xn[2]    "Пропан"        C3H8"   printf("%10.5f ", Xn[2])  
Xn[3]    "н - Бутан"     н-C4H10" printf("%10.5f ", Xn[3])  
Xn[4]    "и - Бутан"     и-C4H10" printf("%10.5f ", Xn[4])  
Xn[5]    "Азот"          N2"     printf("%10.5f ", Xn[5])  
Xn[6]    "Диоксид углерода CO2" printf("%10.5f ", Xn[6])  
Xn[7]    "Сероводород"   H2S"    printf("%10.5f ", Xn[7])  
Xn[8]    "н - Пентан"    н-C5H12" printf("%10.5f ", Xn[8])  
Xn[9]    "и - Пентан"    и-C5H12" printf("%10.5f ", Xn[9])  
Xn[10]   "н - Гексан"    н-C6H14" printf("%10.5f ", Xn[10])  
Xn[11]   "н - Гептан"    н-C7H16" printf("%10.5f ", Xn[11])  
Xn[12]   "н - Октан"     н-C8H18" printf("%10.5f ", Xn[12])  
Xn[13]   "Гелий"         He"     printf("%10.5f ", Xn[13])  
Xn[14]   "Моноксид углерода CO" printf("%10.5f ", Xn[14])  
Xn[15]   "Кислород"     O2"     printf("%10.5f ", Xn[15])  
Xn[16]   "Аргон"        Ar"     printf("%10.5f ", Xn[16])  
Xn[17]   "Водяной пар"   H2O"    printf("%10.5f ", Xn[17])
```

Команда 10 [0x0A]
Структуры последних редакций

```
typedef struct {  
    SFormatEdit Ro; - последняя редакция плотности  
    SFormatEdit Xa; - последняя редакция Xa  
    SFormatEdit Xu; - последняя редакция Xu  
    SFormatEdit LastEditAdmin; - последняя редакция уровня Администратора  
    SFormatEdit LastEditOperator; - последняя редакция уровня Оператора  
} SFormatLastEdit;
```

Вывод данных :
Обрабатывать как архив редакций параметров

Команда 11 [0x0B]

1 запись часового архива

```
typedef struct {
    unsigned long Vk;
    unsigned long Vw;
    unsigned long Vc;
    unsigned long Aw;
    unsigned long Ac;
    unsigned long P;
    float K;
    unsigned short T;
} SFormatHour – описание часового архива одного модуля
```

```
typedef struct {
    unsigned long DateTime;
    SFormatHour FormatHour[2]; - данные двух модулей
    unsigned short AddFlags[2]; - дополнительная информация
} STEMPHour – описание общей структуры часового архива
```

Вывод данных :

```
unsigned long DateTime TimeNormal (DateTime)
unsigned long Vk; double dl = Vk/10.0; printf("%12.1f ",dl)
unsigned long Vw; double dl = Vw/10.0; printf("%12.1f ",dl)
unsigned long Vc; double dl = Vc/10.0; printf("%12.1f ",dl)
unsigned long Aw; double dl = Aw/10.0; printf("%12.1f ",dl)
unsigned long Ac; double dl = Ac/10.0; printf("%12.1f ",dl)
unsigned long P; float fl = P/10000.0; printf("%7.4f ",fl)
float K; printf("%7.5f ",K)
unsigned short T; float fl = T/100.0 - 273.15; printf("%7.2f ",fl)
```

Команда 12 [0x0C]

1 запись суточного архива

```
typedef struct {
    unsigned long Vk; – объем по счетчику
    unsigned long Vw; – объем с учётом работы при Qmin
    unsigned long Vc; – объем приведенный к нормальным условиям
    unsigned long Aw; – объем рабочий при авариях
    unsigned long Ac; – объем приведенный к норм. условиям при аварии
    unsigned long P; - среднее давление
    unsigned long Perror; - время аварийной работы по давлению
    unsigned long Terror; - время аварийной работы по температуре
    unsigned long Qerror; - время аварийной работы при Q > Qmax
    unsigned long Qmin; - время работы при расходе меньшем Qmin
    unsigned char Operator; - количество редакций уровня Оператор
    unsigned char Admin; - кол редакций уровня Администратор
    unsigned short T; - средняя температура
} SFormatDay;
```

```
typedef struct {
    unsigned long DateTime; - дата записи данных архива
```



```
SFormatDay FormatDay[2]; - данные двух модулей
unsigned short AddFlags[2]; - дополнительная информация
} STEMPDay;
```

Вывод данных :

```
unsigned long DateTime TimeNormal (DateTime)
unsigned long Vk; double dl = Vk/10.0; printf("%12.1f ",dl)
unsigned long Vw; double dl = Vw/10.0; printf("%12.1f ",dl)
unsigned long Vc; double dl = Vc/10.0; printf("%12.1f ",dl)
unsigned long Aw; double dl = Aw/10.0; printf("%12.1f ",dl)
unsigned long Ac; double dl = Ac/10.0; printf("%12.1f ",dl)
unsigned long P; float fl = P/10000.0; printf("%7.4f ",fl)
unsigned short T; float fl = T/100.0 - 273.15; printf("%7.2f ",fl)
unsigned long Perror; TimeSec(Perror)
unsigned long Terror; TimeSec(Terror)
unsigned long Qerror; TimeSec(Qerror)
unsigned long Qmin; TimeSec(Qmin)
unsigned char Operator; printf("%5u ", Operator)
unsigned char Admin; printf("%5u ", Admin)
```

Команда 13 [0x0D]

1 запись доп. информ. суточного архива

```
typedef struct { unsigned char byte[3]; } UnitData3Byte;
```

```
typedef struct {
    unsigned long VolQmax; - объем при активной аварии Qmax за сутки
    unsigned long VolPmax; - объем при активной аварии Pmax за сутки
    unsigned long VolTmin; - объем при активной аварии Tmin за сутки
    unsigned long VolTmax; - объем при активной аварии Tmax за сутки
    unsigned long VolPmin; - объем при активной аварии Pmin за сутки
    UnitData3Byte TimeTmax; - время активной аварии Tmax
    UnitData3Byte TimeTmin; - время активной аварии Tmin
    UnitData3Byte TimePmax; - время активной аварии Pmax
    UnitData3Byte TimePmin; - время активной аварии Pmin
} SFormatDayAdd;
```

```
typedef struct {
    unsigned long DateTime; - дата записи данных
    SFormatDayAdd FormatUser[2]; данные двух модулей
} STEMPDayAdd;
```

Вывод данных :

```
unsigned long DateTime TimeNormal (DateTime)
unsigned long VolQmax; double dl = VolQmax /10.0; printf("%12.1f ",dl)
unsigned long VolPmax; double dl = VolPmax /10.0; printf("%12.1f ",dl)
unsigned long VolTmin; double dl = VolTmin /10.0; printf("%12.1f ",dl)
unsigned long VolTmax; double dl = VolTmax /10.0; printf("%12.1f ",dl)
unsigned long VolPmin; double dl = VolPmin /10.0; printf("%12.1f ",dl)
UnitData3Byte TimeTmax; TimeSec(TimeTmax)
UnitData3Byte TimeTmin; TimeSec(TimeTmin)
UnitData3Byte TimePmax; TimeSec(TimePmax)
UnitData3Byte TimePmin; TimeSec(TimePmin)
```

Команда 14 [0x0E]

1 запись аварийного архива

```
typedef struct {
    unsigned long DateTime; - дата записи
    unsigned long DataError; - значение
    unsigned char CodeError; - код аварии
    unsigned char FreeByte; - Резервировано
    unsigned long Vk; - объем по счетчику на момент записи аварии
    unsigned long TimeError; - время действия аварии при выходе или дополнительная ин-
формация при записи сообщения о аварии
} SFormatError;
```

Вывод данных :

```
unsigned long DateTime TimeNormal (DateTime)
unsigned long Vk; double dl = Vk /10.0; printf("%12.1f ",dl)
```

CodeError:

```
нет кода:" Неизвестное сообщение Код = "
код 0: " Горячая перезагрузка "
код 1: " Авария : датчика давления,"
код 2: " Авария : датчика температуры,"
код 3: " [ P < P min ] Авария : Давление меньше Pmin,"
код 4: " [ P < P min ] Авария : Давление меньше Pmin метода расчета"
код 5: " [ T < T min ] Авария : Температура меньше Tmin,"
код 6: " [ T < T min ] Авария : Температура меньше Tmin метода расчета"
код 7: " [ P > P max ] Авария : Давление больше Pmax,"
код 8: " [ P > P max ] Авария : Давление больше Pmax,"
код 9: " [ T > T max ] Авария : Температура больше Tmax,"
код 10: " [ T > T max ] Авария : Температура больше Tmax метода расчета"
код 11: " [ Q > Q max ] Авария : Рабочий расход больше Qmax"
код 12: " [ Q < Q min ] Внимание : Рабочий расход меньше Qmin"
код 14: " Внимание : Низкое напряжение батареи"
код 15: " [ P < P min ] Авария : Давление меньше Pmin. "
код 16: " [ P > P max ] Авария : Давление больше Pmax. "
код 17: " [ T < T min ] Авария : Температура меньше Tmin. "
код 18: " [ T > T max ] Авария : Температура больше Tmax. "
код 50: " Холодная ( Системная ) перезагрузка "
код 51: " Новая батарея установлена "
код 52: " Авария расчета К сжимаемости "
код 55: " Авария! Nmax вне диапазона "
код 101: " Восстановлена работа датчика давления."
код 102: " Восстановлена работа датчика температуры."
код 103: " [ P > P min ] Давление больше Pmin."
код 104: " [ P > P min ] Давление больше Pmin метода расчета."
код 105: " [ T > T min ] Температура больше Tmin."
код 106: " [ T > T min ] Температура больше Tmin метода расчета."
код 107: " [ P < P max ] Давление меньше Pmax."
код 108: " [ P < P max ] Давление меньше Pmax метода расчета."
код 109: " [ T < T max ] Температура меньше Tmax."
код 110: " [ T < T max ] Температура меньше Tmax метода расчета."
код 111: " [ Q < Q max ] Рабочий расход меньше Qmax."
код 112: " [ Q > Q min ] Рабочий расход больше Qmin."
код 115: " [ P > P min ] Давление больше Pmin."
код 116: " [ P < P max ] Давление меньше Pmax."
код 117: " [ T > T min ] Температура больше Tmin."
код 118: " [ T < T max ] Температура меньше Tmax."
код 152: " расчет К сжимаемости восстановлен "
код 153: " Установлена функция датчика давления "
```

код 154: " Установлена функция термометра "
код 155: " Нmax в диапазоне м. расчета "

DataError:

```
нет кода: "-----"  
код 0: "-----"  
код 1: "-----"  
код 2: "-----"  
код 3: fl = DATA/10000.0 printf("%10.4f ", fl)  
код 4: fl = DATA/10000.0 printf("%10.4f ", fl)  
код 5: fl = DATA/100.0 - 273.15 printf("%10.2f ", fl)  
код 6: fl = DATA/100.0 - 273.15 printf("%10.2f ", fl)  
код 7: fl = DATA/10000.0 printf("%10.4f ", fl)  
код 8: fl = DATA/10000.0 printf("%10.4f ", fl)  
код 9: fl = DATA/100.0 - 273.15 printf("%10.2f ", fl)  
код 10: fl = DATA/100.0 - 273.15 printf("%10.2f ", fl)  
код 11: fl = DATA/1000.0 printf("%10.3f ", fl)  
код 12: fl = DATA/1000.0 printf("%10.3f ", fl)  
код 15: fl = DATA/10000.0 printf("%10.4f ", fl)  
код 16: fl = DATA/10000.0 printf("%10.4f ", fl)  
код 17: fl = DATA/100.0 - 273.15 printf("%10.2f ", fl)  
код 18: fl = DATA/100.0 - 273.15 printf("%10.2f ", fl)  
код 50: ""  
код 52: fl = DATA/10000.0 printf("%10.4f ", fl)  
код 55: fl = DATA/10.0 printf("%10.1f ", fl)  
код 101: fl = DATA/10000.0 printf("%10.4f ", fl)  
код 102: fl = DATA/100.0 - 273.15 printf("%10.2f ", fl)  
код 103: fl = DATA/10000.0 printf("%10.4f ", fl)  
код 104: fl = DATA/10000.0 printf("%10.4f ", fl)  
код 105: fl = DATA/100.0 - 273.15 printf("%10.2f ", fl)  
код 106: fl = DATA/100.0 - 273.15 printf("%10.2f ", fl)  
код 107: fl = DATA/10000.0 printf("%10.4f ", fl)  
код 108: fl = DATA/10000.0 printf("%10.4f ", fl)  
код 109: fl = DATA/100.0 - 273.15 printf("%10.2f ", fl)  
код 110: fl = DATA/100.0 - 273.15 printf("%10.2f ", fl)  
код 111: fl = DATA/1000.0 printf("%10.3f ", fl)  
код 112: fl = DATA/1000.0 printf("%10.3f ", fl)  
код 115: fl = DATA/10000.0 printf("%10.4f ", fl)  
код 116: fl = DATA/10000.0 printf("%10.4f ", fl)  
код 117: fl = DATA/100.0 - 273.15 printf("%10.2f ", fl)  
код 118: fl = DATA/100.0 - 273.15 printf("%10.2f ", fl)  
код 153: "----"  
код 154: "----"  
код 152: fl = DATA/10000.0 printf("%10.4f ", fl)  
код 155: fl = DATA/10.0 printf("%10.1f ", fl)
```

Команда 15 [0x0F]

1 запись архива редакций

```
typedef struct {  
    unsigned long DATE; - дата редакции  
    unsigned long DataNew; - новое значение  
    unsigned long DataOld; - предыдущее значение  
    unsigned char CodeEdit; - код редакции  
    unsigned char Reserv; - зарезервировано  
    unsigned short CRC; - контрольная сумма  
} SFormatEdit;
```

Вывод данных :

```
unsigned long DATE TimeNormal (DATE)
```

CodeEdit:

нет кода: " Неизвестное сообщение Код = "
код 2: " Установлен КТ "
код 3: " Установлен КР "
код 5: " Заводская очистка данных "
код 15: " Установлен коэффициент преобразования, имп./м3 "
код 16: " Установлен Тип контакта "
код 17: " Установлен Qmin, м3/ч "
код 18: " Установлен Qmax, м3/ч "
код 19: " Установлен Pmin, МПа "
код 20: " Установлен Pmax, МПа "
код 21: " Установлен Tmin, `С "
код 22: " Установлен Tmax, `С "
код 23: " Установлен Цикл измерения, сек. "
код 24: " Установлен Метод расчета "
код 25: " Установлен Контрактный час "
код 26: " Установлен Vw, м3 "
код 27: " Установлен Xa, % "
код 28: " Установлен Xu, % "
код 29: " Установлен Ro, кг/м3 "
код 33: " Установлен Pa, МПа "
код 34: " Установлен Ta, `С "
код 35: " Установлен пароль Оператора "
код 36: " Установлен пароль Администратора "
код 37: " Установлен интервал пользовательского архива, мин. "
код 38: " Установлена дата предварительного ввода состава газа "
код 39: " Установлен Xa (предварительный ввод), % "
код 40: " Установлен Xu (предварительный ввод), % "
код 41: " Установлена плотность (отложенный ввод), кг/м3 "
код 42: " Установлена дата перехода на летний период "
код 43: " Установлена дата перехода на зимний период "
код 44: " Установлен Vc, м3 "
код 45: " Установлен Vsum (Vw + V(Qmin)), м3 "
код 48: " Системная конфигурация "
код 49: " переход на летний период период "
код 50: " переход на зимний период -1 час "
код 51: " Коррекция времени "
код 52: " Установлена блокировка временем "
код 53: " переход на летний период период "
код 54: " переход на зимний период "
код 55: " Установлен DN "
код 59: " Установлен состав газа AGA8-92DC Zc "
код 61: " Установлен тип модуля "
код 62: " Установлен количество модулей "
код 63: " Установлено событие Qmin в архиве "
код 64: " Установлена функция датчика давления "
код 65: " Установлено время ожидания импульса, сек. "
код 71: " Установлена скорость обмена сети "
код 72: " Установлено время простоя при аварии, сек. "
код 73: " Установлена защита от дребезга контакта, Гц "
код 78: " Установлен C imp, м3/имп. "
код 79: " Контроллер B25 очищен для записи ПО, CRC "
код 81: " Установлена батарея "
код 82: " расчет K остановлен "
код 83: " расчет K восстановлен, CRC "
код 84: " Установлен Pmax Sensor, МПа "
код 254: " Выполнен переход в режим замены батареи "

DataNew, DataOld (DATA)

```
fl = *(float *)&DATA;  
us = *(unsigned short *)&DATA;
```

```

ss = *(short *)&DATA;
uc = DATA;

код 2: printf("%10.6f ",fl)
код 3: printf("%10.6f ",fl)
код 4: printf("%10.6f ",fl)
код 10: printf("%10.6f ",fl)
код 11: printf("%10.6f ",fl)
код 12: printf("%10.6f ",fl)
код 13: printf("%10.6f ",fl)
код 14: printf("%10.6f ",fl)
код 15: printf("%10.3f ",fl)
код 16: if ( uc ) " норм. разомкнут " else " норм. замкнут "
код 17: fl = DATA/1000.0 printf("%10.3f ",fl)
код 19: printf("%10.3f ",fl)
код 20 printf("%10.3f ",fl)
код 21: fl - 273.15 printf("%10.2f ",fl)
код 22: fl - 273.15 printf("%10.2f ",fl)
код 23: printf("%3d",uc)
код 24: if ( uc == 0 ) "NX 19 mod"
        if ( uc == 1 ) "GERG 91 mod"
        if ( uc == 2 ) "AGA8-92DC"
код 25: printf("%3d",uc)
код 26: printf("%5u ",DATA)
код 27: printf("%10.4f ",fl)
код 28: printf("%10.4f ",fl)
код 29: printf("%10.5f ",fl)
код 33: printf("%10.4f ",fl)
код 34: fl - 273.15 printf("%10.2f ",fl)
код 35: "-----"
код 36: "-----"
код 37: printf("%3d",uc)
код 38: TimeNormal (DATA)
код 42: TimeNormal (DATA)
код 43: TimeNormal (DATA)
код 44: printf("%5d",DATA)
код 45: printf("%5d",DATA)
код 48:
        if ( DATA&0x02 ) AS += "A+" else AS += "A-";
        if ( DATA&0x04 ) AS += "U+" else AS += "U-";
        if ( DATA&0x08 ) AS += "K+" else AS += "K-";
        if ( DATA&0x10 ) AS += "D+" else AS += "D-";
        if ( DATA&0x40 ) AS += "PTLi+"; else AS += "PTLi-";
        if ( DATA&0x80 ) AS += "PaTa+"; else AS += "PaTa-";
        if ( DATA&0x100 ) AS += "TLi+"; else AS += "TLi-";
        if ( DATA&0x200 ) AS += "45L+"; else AS += "45L-";
        if ( DATA&0x400 ) AS += "PLi+"; else AS += "PLi-";
        if ( DATA&0x800 ) AS += "LP+"; else AS += "LP-";

код 49: TimeNormal (DATA)
код 50: TimeNormal (DATA)
код 51: TimeNormal (DATA)
код 52: TimeNormal (DATA)
код 53: TimeNormal (DATA)
код 54: TimeNormal (DATA)
код 55: printf("%5d ",DATA)
код 59: printf("%10.6f ",fl)
код 61: printf("%3d",uc)
код 62: printf("%3d",uc)
код 63: if ( uc == 0 ) " фиксируется " else " не фиксируется "
код 65: printf("%5u ",DATA);
код 71: if ( uc ) "4800" else "9600"

```

```
код 72: printf("%3d",uc)
код 73: if ( us ) 4096/us
код 78: fl = DATA/1000.0 printf("%10.3f ",fl)
код 79: printf("%5u ",DATA)
код 80: fl = (ss/4096.0)*1000.0 printf("%10.2f ",fl)
```

```
case 83: printf("%8u ",DATA)
case 84: printf("%10.1f ",fl)
```

TimeNormal - функция пересчета времени (секунды) в календарную дату, значение 0 соответствует «01/01/2000 00:00»

TimeSec - функция пересчета времени (секунды) в формат чччч:мм:сс

Команда 16 [0x10]

1 месячного архива

Соответствует формату суточного архива

Формат протокола (Ввод данных)

№ в пакете	Обозначение	Описание	Байт
0	AddrSlave	Адрес устройства (В25)	1 (unsigned char)
1	Len	Длина пакета = 8 + Len(данных)	1 (unsigned char)
2	Code Net (0x8D)	Код запроса	1 (unsigned char)
3	NMOD	Номер модуля (0,1)	1 (unsigned char)
4	Reserve	Зарезервировано	1 (unsigned char)
5	CodeSet	Код записываемых данных	1 (unsigned char)
6	Passwd [0]	Пароль	4 (unsigned long)
7	Passwd [1]		
8	Passwd [2]		
9	Passwd [3]		
10	char* data	Данные	
11+ Len	AddrMaster	Адрес Мастера (компьютер)	1 (unsigned char)
12+ Len	CRC[0]	Контрольная сумма (считается суммированием всех байтов пакета, включая CRC)	2 (unsigned short)
13+ Len	CRC[1]		

Ответ устройства 0x00 – в вводе отказано, 0xFF – новые данные приняты

Код записываемых данных

Код	Описание	Тип	Доступ	Разрядность
27	Xa	%	Опер.	46 float
28	Xy	%	Опер.	46 float
29	Ro	кг/м3	Опер.	46 float
33	Pa (константа)	МПа	Опер.	46 float
34	Ta (константа)	К	Опер.	46 float