

**ПРИКЛАДНОЕ ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ  
СИСТЕМЫ СБОРА ТЕХНОЛОГИЧЕСКОЙ ИНФОРМАЦИИ**

**КОМПЛЕКС ПРОГРАММ ЗОНД**  
*(Версия 4.40.0210)*

**Утилиты**

Москва, 2008

**СОДЕРЖАНИЕ**

<b>1. Общие сведения.....</b>	<b>4</b>
<b>2. Утилиты трансляции баз данных.....</b>	<b>5</b>
2.1 Утилита mag3_db.exe.....	5
2.2 Утилита mbm_conf.exe.....	7
2.3 Утилита mbs_conf.exe.....	8
2.4 Утилита mbs2.exe.....	11
2.5 Утилита mag2sbor.exe.....	11
2.6 Утилита db2disk.exe.....	12
2.7 Утилита disk2db.exe.....	13
2.8 Утилита mbf_conf.exe.....	13
2.9 Утилита for_sbor.exe.....	14
2.10 Утилита mls_conf.exe.....	15
2.11 Утилита ftrans.exe.....	15
2.12 Утилита smsserv.exe и smsservw.exe.....	16
<b>3. Утилиты конфигурирования удаленного сервиса.....</b>	<b>17</b>
3.1 Утилита red_num.exe.....	17
3.2 Утилита red_conf.exe.....	17
<b>4. Утилиты формирования библиотек графических фрагментов.....</b>	<b>19</b>
4.1 Утилита getfrag.exe.....	19
4.2 Утилита putfrag.exe.....	19
4.3 Утилита fragsort.exe.....	20
4.4 Утилита delstat.exe.....	20
4.5 Утилита clrfrag.exe.....	20
<b>5. Утилиты формирования графической истории.....</b>	<b>21</b>
5.1 Утилита grhflset.exe.....	21
5.2 Утилита grhmk.exe.....	22
5.3 Утилита grhadd.exe.....	22
5.4 Утилита grhstep.exe.....	23
<b>6. Утилиты трансляции размерности и таблицы цветов.....</b>	<b>24</b>
6.1 Утилита get_rt.exe.....	24
6.2 Утилита put_rt.exe.....	24
<b>7. Утилиты обработки базы данных.....</b>	<b>25</b>
7.1 Утилита gofo_sys.exe.....	25
7.2 Утилита gofo_chk.exe.....	26
7.3 Утилита basecmp.exe и basecmpw.exe.....	26
7.4 Утилита dbf_clear.exe.....	26
7.5 Утилита tula_tab.exe.....	27
7.6 Утилита clrdelta.exe.....	27
7.7 Утилита clr_toti.exe.....	28
7.8 Утилита clr_knum.exe.....	28
7.9 Утилита chgcolor.exe.....	28
<b>8. Экспорт базы данных и конфигурационных файлов.....</b>	<b>30</b>

8.1 Утилита mbs_list.exe.....	30
8.2 Утилита prn_base.exe.....	30
8.3 Утилита dbf_fill.exe.....	31
<b>9. Коммуникационные утилиты.....</b>	<b>32</b>
9.1 Утилита nikm.exe.....	32
9.2 Утилиты set_mba.exe и set_mbaw.exe.....	33
9.3 Утилиты test_id.exe и test_idw.exe.....	33
9.4 Утилита mb_uho.exe.....	34
9.5 Утилита mbtest.exe.....	34
<b>10. Утилиты для слежения за работой УСО .....</b>	<b>36</b>
10.1 Утилита convtrac.exe и convtracw.exe.....	36
<b>11. Список используемых документов.....</b>	<b>37</b>

## 1. Общие сведения

Все документированные здесь утилиты могут быть запущены на компьютерах совместимых с IBM PC AT, и оснащенных процессорами с архитектурой I386, работающих под управлением операционной системы Microsoft DOS/Windows 3.0/3.1/95/98/NT/2000/XP/2003/Vista, кроме коммуникационных утилит, работающих только под управлением операционной системы MS-DOS.

Все утилиты запускаются из корневой директории ПК Зонд (см. Док. 6), кроме коммуникационных утилит, которые могут работать в любой директории.

Все утилиты выводят сообщение:

***Утилита <имя> (Комплекс программ Зонд v4.40.XXXX)  
Copyright SCADA-pro Ltd. Moscow. Russia ,***

где XXXX – номер версии сборки. При внесении изменений в утилиты в файле whatsnew440\_all.htm (whatsnew440\_all.txt) отмечается номер сборки ПК Зонд, в которую изменение вошло впервые. Таким образом, для каждого exe-файла утилиты известна степень реализации. Впервые номер версии сборки введен для версий утилит сборки 13.

## 2. Утилиты трансляции баз данных

Утилиты трансляции баз данных (БД) позволяют осуществлять преобразования, необходимые для согласованного ведения нескольких БД, объединенных в проект. Процесс параметризации проекта описан в Док. 7.

Операция автоматического построения блока параметров одной БД на основе блока параметров другой БД с согласованным изменением конфигураций интерфейсов modbus\_s исходной БД и modbus\_m результирующей БД называется трансляцией блока БД. Таким образом, базы данных узлов проекта могут состоять из областей (блоков параметров) двух типов:

- \* полученных путем трансляции блоков БД нижестоящих уровней (транслированные области);
- \* сформированных в данной БД (первичные области).

### 2.1 Утилита mag3\_db.exe

Утилита mag3\_db.exe применяется для БД ПК Зонд, которые входят в программное обеспечение контролируемых пунктов (КП) системы линейной телемеханики (СЛТМ) «Магистраль-2».

Утилита mag3\_db.exe реализует две операции:

- \* настраивает паспорта параметров заданного первичного блока БД КП;
- \* создает конфигурацию modbus\_m для опроса modbus-линии узла КП.

Синтаксис:

***mag3\_db.exe <flags> <start#> <num#> <dir> <kp> <mba> <idn> <name>***

<b><i>&lt;flags&gt;</i></b>	- битовые флаги, шестнадцатеричное число: bit0=1 - не трогать флаги паспорта (окно, печать, файл, квитиование); bit1=1 - не трогать признак «необходимость формирования графиков»;
<b><i>&lt;start#&gt;, &lt;num#&gt;</i></b>	- начальный адрес и размер блока параметров контроллера в БД;
<b><i>&lt;dir&gt;, &lt;kp&gt;</i></b>	- номер линии (1-16) и контроллера (1-254 dec);
<b><i>&lt;mba&gt;</i></b>	- адрес контроллера в линии (1-ff,hex);
<b><i>&lt;idn&gt;</i></b>	- уникальный номер контроллера (hex);
<b><i>&lt;name&gt;</i></b>	- имя контроллера в modbus_m.cfg (без пробелов).

Утилита mag3\_db.exe настраивает паспорта параметров первичного блока, определяя тип модуля СЛТМ, которому принадлежит блок, по найденной в параметре «имя контроллера» подстроке. Допустимы подстроки: "ДЭ01", "ДЭ02", "ДЭ03", "ДЭ04", "ДЭ05", "СЭ01", "СЭ02", "СЭ03", "СЭ06", "ИЭ01", "ИЭ02", "ИЭ03", "ИЭ06", "НЭ01", "ЭР01", "ЭР02", "ЭР03", "МФ01", "МЭ01", "ТАПАС\_КИК", "ТАПАС\_СКЗ", "ГАЗ-6", "ПУЛЬСАР".

Блок может состоять из параметров УСО «Диагностика» и «modbus\_m». Значения параметров утилиты «номер линии» и «номер контроллера» проставляются в соответствующие поля разделов «подключение».

Внутри блока утилита “ориентируется” по значениям полей разделов “подключение” “тип сигнала” и “адрес сигнала”, которые должны быть проставлены заранее вручную, и таким образом адресует любой из параметров, соответствующих поддерживаемым сигналам модулей СЛТМ (таблицы приведены в Док. 7). Для найденных параметров проставляются значение полей разделов “паспорт” (“нормальное значение”, “номер таблицы цветов”, “уникальный номер”), “статус” (“тип управления”, “обработка” (разрешена), “датчик” (исправен), “аппарат” (в работе), “разрешить запись в окно, на принтер, в журнал” (да), “разрешить звуковой сигнал”, “разрешить квитирование”) и “подключение” (“адрес управления”, “дополнительный фактор”, “тип ссылки”, “адрес ссылки”, “нормальное/достоверное значение”).

Утилита резервирует первые шесть строк таблицы цветов, считая что они заполнены следующим образом:

0	- 3 К Ж К	(сдвоенные поршни)
1	- 3 К К К	(ТС норма - 0)
2	- Ж 3 К С	(краны)
3	- 3 Ж	(ТС норма – 0, активность -1)
4	- К 3	(ТС норма - 1)
5	- К Ж Ж 3	(ТС норма - 1 реле отк/зак)

Утилита mag3\_db.exe при настройке конфигурации modbus\_m выделяет каждому типу сигналов пространство опроса (в таблице – начальный адрес и число адресов через запятую) и управления и проставляет параметры опроса (опрос – “Да”, граница – “0”, пропуск – “16”), а если число управлений не равно 0, то разрешить управление (упр – “Да” при положении маркера на области Coils или Holding). Параметры модулей СЛТМ сведены в Таб. 1 и выбраны на в рамках технологии параметризации, документированной в Док. 7.

Модуль	Тип Контр	Coils (нач.адр. кол-во)	Discret (нач.адр. кол-во)	Input (нач.адр. кол-во)	Holding (нач.адр. кол-во)	Число управ.
ДЭ01	МАГ2	0,0	0,24	0,0	0,0	0
ДЭ02	МАГ2	0,0	0,0	0,0	0,2	2
ДЭ03	МАГ2	0,0	1,24	0,0	0,0	25
ДЭ04	МАГ2	0,0	1,14	0,0	0,0	15
ДЭ05	МАГ2	0,0	0,0	0,0	0,2	2
СЭ01	СЭ01	0,0	0,19	0,0	0,0	1
СЭ02	МАГ2	0,0	1,16	1,5	0,0	12
СЭ03	МАГ2	0,0	1,18	0,0	1,3	6
СЭ06	МАГ2	0,0	1,24	0,0	15,4	19
ИЭ01	МАГ2	0,0	1,8	1,9	0,0	0
ИЭ02	МАГ2	0,0	1,8	1,3	0,0	0
ИЭ03	МАГ2	0,0	1,8	1,9	0,0	0
ИЭ06	ИЭ06	0,0	1,2	3,2	0,0	0
НЭ01	МАГ2	0,0	0,0	0,0	0,4	0
МФ01	МАГ2	0,0	1,8	5,2	0,0	0
ЭР01	ЭР01	0,0	1,8	0,0	1,1	2
ЭР02	ЭР02	0,0	0,0	0,0	3,2	2
ЭР03	МАГ2	0,0	1,8	0,0	0,0	9
МЭ01	МАГ2	0,0	6,1	0,0	0,0	0
ТАПАС-	МАГ2	0,0	0,0	0,2	0,0	0

Модуль	Тип Контр	Coils (нач.адр. кол-во)	Discret (нач.адр. кол-во)	Input (нач.адр. кол-во)	Holding (нач.адр. кол-во)	Число управ.
КИК						
ТАПАС-СКЗ	МАГ2	0,0	0,1	0,2	0,0	0
ГАЗ-6	МАГ2	0,0	1,128	1,8	1,16	0
ПУЛЬСАР	ПУЛЬСАР	0,0	0,0	0,0	448,12	460

Таб. 1. Параметры конфигурации Modbus master для различных типов контроллеров

Параметры линии modbus\_m настраиваются так, как того требует линия КП СЛТМ “Магистраль-2” (Таб. 2).

Наименование параметра	Размерность	Значение
Номер канала связи		Из проекта
Скорость передачи данных	бод	57600
Число стоповых битов		1
Контроль четности		NONE
Пауза опроса линии	с	0
Таймаут опроса	с	1
Тип контроллера:		Из таб .
Таймаут между посылками	N*55 мс	1
Метка времени	с	3
Максимальная длина посылки	байт	240
Такт редкого опроса	с	8

Таб. 2. Параметры линии опроса КП СЛТМ “Магистраль-2”

**Внимание!** Модули СЭ01 и СЭ04 различаются только адресом опроса параметра “контроль реле питания” - 17 в СЭ04 и 18 в СЭ01.

**Внимание!** В настоящий момент развитие утилиты как самостоятельного модуля прекращено, используйте программу “Конфигуратор”.

## 2.2 Утилита mbm\_conf.exe

Утилита mbm\_conf.exe реализует две операции:

- \* создает конфигурацию modbus\_m для опроса modbus-линии узла КП;
- \* настраивает паспорта параметров заданного первичного блока БД КП.

Синтаксис:

**mbm\_conf.exe** <dir> <kp> <mba> <name> <typ> <bound> <antype> <cf> <cn> <df> <dn> <if> <in> <hf> <hn> <cnt> <startsys> <n> <flags>

- <dir>, <kp> - номер линии (1-16) и контроллера (1-254) Modbus master;
- <mba> - адрес контроллера в линии (hex);
- <name> - имя контроллера в modbus\_m (без пробелов);
- <typ> - тип контроллера (ССС,...);
- <bound> - граница опроса достоверности дискретных (0-не опрашивать);

<b>&lt;antype&gt;</b>	- формат опроса аналоговых (ключевые слова Word,Doub,Floa);
<b>&lt;cf&gt;, &lt;cn&gt;</b>	- первый адрес и число адресов опроса Coils (0,..);
<b>&lt;df&gt;, &lt;dn&gt;</b>	- первый адрес и число адресов опроса Discret (0,..);
<b>&lt;if&gt;, &lt;in&gt;</b>	- первый адрес и число адресов опроса Inputs (0,..);
<b>&lt;hf&gt;, &lt;hn&gt;</b>	- первый адрес и число адресов опроса Holding (0,..);
<b>&lt;cnt&gt;</b>	- число управлений и регулирований (0,..);
<b>&lt;startsys&gt; &lt;n&gt;</b>	- начальный адрес и размер блока параметров контроллера в БД;
<b>&lt;flags&gt;</b>	- флаги, шестнадцатеричная константа: 1 - автоматически заполнять подключения (работает для типа контроллера HOST, назначает адреса от 0, четные, тип сигнала HOLDING reg).

Утилита `mbm_conf.exe` настраивает часть полей разделов подключения паспортов параметров первичного блока.

Обработке подвергаются находящиеся в блоке параметры УСО “Диагностика” (относящиеся к УСО “`modbus_m`”) и “`modbus_m`”. Значения параметров утилиты “номер линии” и “номер контроллера” проставляются в соответствующие поля разделов “подключение”.

### 2.3 Утилита `mbs_conf.exe`

Утилита `mbs_conf.exe` предназначена для трансляции БД в рамках проектов на базе ПК Зонд, но может быть использована для автоматической настройки сервера `modbus-slave` ПК Зонд. Утилита создает блок параметров БД УСО `modbus_m` и формирует запись файла конфигурации `modbus_s.cfg`.

Синтаксис:

```
mbs_conf.exe <start#> <end#> <flags> <dst blockname> <s dir> <s kp> <m dir> <m kp> <bound> <name> <mba> <es>
```

<b>&lt;start#&gt;, &lt;end#&gt;</b>	- начальный и конечный системный номер в исходной БД;
<b>&lt;flags&gt;</b>	bit0=1 (hex) - стирать предыдущее содержимое <code>modbus_s.cfg</code> ; bit1=1 - обрабатывать только паспорта у которых задано поле 'код 2' *; bit2=1 - режим минимизации дырок в дискретных **; bit3=1 - ставить 'управление извне' для дискретных двубитных управляемых (для параметров <code>MAGISTRAL_1</code> , <code>MODBUS_M</code> , <code>LOMIKONT</code> ); bit4=1 - ставить запрет квитирования на параметры выходного блока БД; bit5=1 - ставить запрет звука на параметры выходного блока БД; bit6=1 - не образовывать файл блока параметров БД (имя любое); bit7=1 - пропускать <code>mmb</code> -пакеты (нужен, если через контроллер предполагается проложить канал удаленного сервиса); bit8=1 - не транслировать управления (нужен, если не предполагается управление со стороны внешней системы); bit9=1 - отменить сообщения в окно у параметров выходного блока БД; bit10=1 - отменить сообщения в файл у параметров выходного блока БД; bit11=1 - отменить сообщения на принтер у параметров выходного

блока БД;  
 bit15=1 - не устанавливать алгоритм регулирования “полный сеанс” (нужен, если не предполагается управление со стороны внешней системы или алгоритм управления не семиступенчатый; фактически, если внешняя система – не ПК Зонд);

**<dst blockname>** - имя файла выходного блока параметров БД;  
**<s dir>,<s kp>** - номер линии (1-16) и контроллера (1-254) modbus slave;  
**<m dir>,<m kp>** - номер линии (1-16) и контроллера (1-254) modbus master в “верхней” БД;  
**<bound>** - граница области достоверности (dec) \*\*\*;  
**<name>** - имя контроллера в Modbus slave (без пробелов);  
**<mba>** - адрес контроллера в линии (hex,1-FF);  
**<es>** - тип внешней системы, допустимы типы skue - СКУЭ, zond - Зонд, kondr - Кондратки, gofo2 - ГОФО-2.

- \* Поле ‘код 2’ в паспорте параметра используется для занесения имен, которые даются параметрам ПК Зонд внешними системами при реализации программных стыков с ними. Значения поля ‘код 2’ хранятся в файле namesprm.dbf стандартного формата DBF в директории исходной БД. Это позволяет производить заполнение полей ‘код 2’ разработчиками внешних систем. Установка флага приведет тогда к трансляции только тех параметров, которые поддерживаются внешней системой.
- \*\* При опросе между двумя ПК Зонд принято выделять двубитному дискретному параметру два адреса Discret, младший из которых четный. Для выполнения этого правила утилита при распределении адресов иногда пропускает нечетный адрес (так называемая “дырка”). Запросы modbus-мастера ПК Зонд допускают дырки. Если внешняя система не допускает дырки или требует максимально компактных транзакций, оправданно ставить этот флаг. Минимизация достигается за счет работы в 2 прохода: сначала обрабатываются дискретные двубитные параметры, потом – остальные. Это гарантирует отсутствие дырок в адресах, но затрудняет ориентирование, т.к. системные номера параметров modbus\_s не идут в порядке их нахождения в БД.
- \*\*\* Граница области достоверности определяет смещение адресов, на запрос по которым modbus\_s выдаст достоверности сигналов (по распределению адресов внутри блока - симметрично значениям). В качестве значения возможно указать как число, так и ключевое слово “AUTO”. В первом случае утилита занесет значение границы в конфигурацию modbus\_s и в раздел “подключение” паспортов созданного блока БД. В дальнейшем это приведет к тому, что modbus\_m внешнего ПК Зонд будет делать в два раза больше запросов по области Discret: по значениям и по достоверности. Во втором случае утилита автоматически рассчитает значение границы равным первому числу, кратному 8, следующему за числом – размером области данных discret, а также проставит в поле “адрес ссылки” раздела “подключение” паспортов созданного блока БД значение “AUTO”. В дальнейшем это приведет к тому, что modbus\_m внешнего ПК Зонд будет делать один запрос по непрерывной области Discret (значения и достоверность), если не будут играть роль ограничения на длину блока. Это уменьшает суммарное время опроса. Также в поле “нормальное/достоверное значение” раздела “подключение” паспортов созданного блока БД будет проставлено значение, при котором значение параметра будет считаться

достоверным.

Утилита просматривает заданный блок БД и формирует транслированный блок БД в файле и ссылки в конфигурации modbus\_s. При трансляции можно передать параметры только тех типов, которые указаны в . При этом типы преобразуются согласно таблице. Распределение ссылок опроса и управления производится в соответствии с реализацией протокола modbus в ПК Зонд Док. 1 в УСО modbus\_m и сервере modbus\_s. Значения параметров не всех типов могут быть корректно перенесены при опросе по линии modbus, что также отражено в Таб. 3.

Тип параметра в исходной БД	Опрос через	Упр/Рег через	Тип параметра в транслированной БД	АЦП	Корректный опрос в modbus_m
Дискретный	Discret	Coil/Holding	Дискретный	-	-
Восьмипозиционный	Discret	-	Восьмипозиционный	-	-
Аналоговый (УСО modbus_m)	Input	Coil/Holding	Аналоговый	Из подкл	Word**, Floa, Doub
Аналоговый (УСО диагностика)	Input	Coil/Holding	Аналоговый	Нет	Word, Floa, Doub
Аналоговый (другие УСО)	Input	Coil/Holding	Аналоговый	Да	Word**, Floa, Doub
Измерит. линия	Input	-	Аналоговый *	Да	Floa, Doub
Дата-время	Input	-	Дата-время	Нет	Floa, Doub
Счетчик времени	Input	-	Внешний таймер	Нет	Floa, Doub
Внешний таймер	Input	-	Внешний таймер	Нет	Floa, Doub
Счетчик импульсов	Input	-	Внешний счетчик	Нет	Floa, Doub
Внешний счетчик	Input	-	Внешний счетчик	Нет	Floa, Doub
Удаленный Зонд	Input	-	***	***	***

**Таб. 3. Преобразование типов при трансляции утилитой mbs\_conf.exe.**

- \* Параметр типа “измерительная линия” имеет несколько значений (текущее значение – мгновенный расход, также есть расход с начала суток по достоверным и недостоверным данным, за прошлые сутки и др.). Рассматриваемый подход обеспечивает перенос только текущего значения. Для переноса остальных значений можно использовать УСО “Вычислитель”.
- \*\* Перенос данными типа Word в определенных случаях может приводить к значительной потере точности. Например, параметры УСО Superflow опрашиваются ПК Зонд в формате float и могут содержать расходы. При приведении ПК Зонд их значений к формату 12-разрядного АЦП точность составляет 0.024% и может быть меньше исходной. В этом случае рекомендуется пользоваться опросом в формате Floa и Doub.
- \*\*\* Параметры типа “Удаленный Зонд” в разделе “подключение” содержат системный номер параметра в удаленной БД, где он имеет свой тип. Этот тип играет далее исходную роль в преобразовании типов.

При трансляции происходит также преобразование типа управления согласно Таб. 4. Это обеспечивает возможность управления (занесения значений) через несколько ПК Зонд. Для ее реализации типы “Управляемый извне” и “Заносимый извне” должны быть предварительно занесены вручную. Для ускорения этой операции при включении в проект штатных БД с параметрами УСО Магистраль-1, Ломиконт и др. можно однократно

запускать утилиту с установленным параметром bit3, и тогда тип “Управляемый извне” установится автоматически у двубитных дискретных параметров.

Тип управления в исходной БД	Тип управления в результирующей БД
Дискретный	Управление не транслируется
Ручной ввод	Управление не транслируется
Заносимый извне	Управляемый извне
Управляемый извне	Управляемый извне

*Таб. 4. Преобразование типов управления при трансляции утилитой `mbs_conf.exe`.*

## 2.4 Утилита `mbs2.exe`

Утилита предназначена для автоматической генерация ссылок конфигурации Modbus Slave (файл `modbus_s.cfg`) по кодам (`code1,code2`) из паспортов параметров базы данных «Зонд».

Используйте синтаксис:

**`mbs2.exe <start#> <end#> <mdir> <skp> <mbs> <trust bound> <codeN>`**

**`<start#>`, **`<end#>`** - начальный и конечный блока системный номер в БД (dec);  
**`<mdir>`**, **`<skp>`** - номер линии (1-8 dec) и контроллера (1-254 dec) `modbus_s`;  
**`<mbs>`** - Modbus адрес контроллера (hex);  
**`<trust bound>`** - граница области достоверности дискретных (dec) или AUTO;  
**`<codeN>`** - номер используемого кода (1 или 2).**

При трансляции адресов аналоговые параметры относятся к Input reg, дискретные и дискретные многопозиционные к Discret reg. Адреса в полях `code1` или `code2` задаются десятичными числами.

## 2.5 Утилита `mag2sbor.exe`

Утилита `mag2sbor.exe` производит вставку блока паспортов УСО Modbus master в БД ПК Зонд и настройку конфигурации соответствующего контроллера в конфигурации `modbus_m`. При этом параметры указанного контроллера конфигурации `modbus_s` исходной БД автоматически переносятся в указанный контроллер конфигурации `modbus_m` результирующей БД.

Синтаксис:

*mag2sbor.exe* <start#> <\*.psp> <\*.cfg> <m dir> <m kp> <name> <s dir> <s kp>

<start#> - начальный системный номер БД, куда вставляется блок;  
 <flags> - bit0=1 - не вставлять поле 'код 2' в БД;  
 - bit1=1 - отбивать этап управления ФИО;  
 - bit2=1 - отбивать этап управления CONT;  
 - bit4,5 = 00 - опрос input registers как Word;  
           01 - опрос input registers как Double;  
           10 - опрос input registers как Float;  
 - bit7=1 - не переносить уставки и статус (квит, звук и т.д.);  
 <\*.psp> - имя файла блока паспортов УСО Modbus master;  
 <\*.cfg> - имя файла modbus\_s.cfg в ПК Зонд уровнем ниже;  
 <m dir>, <m kp> - номер линии (1-16) и контроллера (1-254) modbus master;  
 <name> - имя контроллера в modbus\_m (без пробелов);  
 <s dir>, <s kp> - номер линии (1-16) и контроллера (1-254) modbus slave в “нижней” БД.

Установка флагов “отбивать этап управления ФИО” и “отбивать этап управления CONT” позволяет не переносить идентификатор диспетчера и не делать сквозные запросы при сеансе телеуправления, задействующем несколько ПК Зонд, и таким образом уменьшить время реакции системы в момент телеуправления. Алгоритмы управления ПК Зонд представлены в \*\*\*.

Установка флага “не переносить уставки и статус” позволяет заблокировать изменение значений флагов статуса и уставок БД при вставке блока. В ПК Зонд, находящемся в эксплуатации, эти значения формируются пользователем.

**Внимание!** При вставке в БД блока параметров не анализируется состав области БД, куда делается вставка. При неудачном позиционировании блока может произойти потеря уникальных данных БД.

**Внимание!** В настоящий момент развитие утилиты как самостоятельного модуля прекращено, используйте программу “Конфигуратор”.

## 2.6 Утилита db2disk.exe

Утилита db2disk.exe сохраняет указанный блок в файл на диск.

Синтаксис:

*db2disk.exe* <start#> <end#> <flags> <dst>

<start#>, <end#> - начальный и конечный системный номер в блока БД;  
 <flags> - bit0=1 - ставить 'управление извне' для дискретных двубитных управляемых (для параметров MAGISTRAL\_1, MODBUS\_M, LOMIKONT);  
 - bit1=1 - отменить квитирование у параметров получаемого блока БД;  
 - bit2=1 - отменить звук у параметров получаемого блока БД;  
 <dst> - имя файла выходного блока БД.

## 2.7 Утилита disk2db.exe

Утилита disk2db.exe вставляет блок с диска в БД.

Синтаксис:

*disk2db.exe* <start#> <flags> <\*.psp>

<start#> - начальный системный номер БД, куда вставится блок;  
 <flags> - bit0=1 - не вставлять поле 'код 2' в БД;  
 <\*.psp> - имя файла блока паспортов.

**Внимание!** При вставке в БД блока параметров не анализируется состав области БД, куда делается вставка. При неудачном позиционировании блока может произойти потеря уникальных данных БД.

**Внимание!** В настоящий момент развитие утилиты как самостоятельного модуля прекращено, используйте программу “Конфигуратор”.

## 2.8 Утилита mbf\_conf.exe

Утилита mbf\_conf.exe предназначена для трансляции БД в рамках проектов на базе ПК Зонд, но может быть использована для автоматической настройки сервера modbus-forcer ПК Зонд. Утилита создает блок параметров БД УСО modbus\_1 и формирует запись файла конфигурации modbus\_f.cfg.

Синтаксис:

*mbf\_conf* <start#> <end#> <flags> <dst blockname> <fdir> <fkp> <ldir> <lkp> <bound>  
 <name> <mba>

<start#> <end#> - начальный и конечный системный номер в исходной БД;  
 <flags> - bit0=1 - стирать предыдущее содержимое modbus\_f.cfg;  
 - bit2=1 - режим минимизации дырок в дискретных (не реализован);  
 - bit4=1 - ставить запрет квитирования на параметры блока БД;  
 - bit5=1 - ставить запрет звукового сигнала на параметры блока БД.  
 <dst blockname> - имя файла выходного блока БД;  
 <fdir>, <fkp> - номер линии (1-8) и контроллера (1-254) modbus\_f;  
 <ldir>, <lkp> - номер линии (1-8) и контроллера (1-254) modbus\_1;  
 <bound> - граница области достоверности (dec);  
 <name> - имя контроллера в modbus\_f (без пробелов);  
 <mba> - адрес контроллера в линии (hex, 1-FF).

Алгоритм работы утилиты такой же, как у утилиты mbs\_conf.exe. Так же выполняется преобразование типов, такой же подход с передачей достоверности. Распределение ссылок опроса и управления производится в соответствии с реализацией протокола modbus в ПК Зонд Док. 1. ] в УСО modbus\_f и сервере modbus\_1. Дискретные параметры переносятся через область Coils, а аналоговые – через область Holding. Управление не поддерживается.

Тип параметра в исходной БД	Опрос через	Тип параметра в транслированной БД	АЦП	Корректный перенос в modbus 1
Дискретный	Coils	Дискретный	-	-
Трехпозиционный	Coils	Трехпозиционный	-	-
Аналоговый (УСО modbus_m)	Holding	Аналоговый	Из подкл	Word**, Floa, Doub
Аналоговый (УСО диагностика)	Holding	Аналоговый	Нет	Word, Floa, Doub
Аналоговый (другие УСО)	Holding	Аналоговый	Да	Word, Floa, Doub**
Измерит. линия	Holding	Аналоговый *	Да	Floa, Doub
Дата-время	Holding	Дата-время	Нет	Floa, Doub
Счетчик времени	Holding	Внешний таймер	Нет	Floa, Doub
Внешний таймер	Holding	Внешний таймер	Нет	Floa, Doub
Счетчик импульсов	Holding	Внешний счетчик	Нет	Floa, Doub
Внешний счетчик	Holding	Внешний счетчик	Нет	Floa, Doub
Удаленный Зонд	Holding	***	***	***

**Таб. 5. Преобразование типов при трансляции утилитой *mbf\_conf.exe*.**

**Внимание!** В настоящий момент развитие утилиты как самостоятельного модуля прекращено, используйте программу “Конфигуратор”.

## 2.9 Утилита *for\_sbor.exe*

Утилита *mag2sbor.exe* производит вставку блока паспортов УСО Modbus loader в БД ПК Зонд и настройку конфигурации соответствующего контроллера в конфигурации *modbus\_1*. При этом параметры указанного контроллера конфигурации *modbus\_f* исходной БД автоматически переносятся в указанный контроллер конфигурации *modbus\_1* результирующей БД.

Синтаксис:

*for\_sbor* <start#> <\*.psp> <\*.cfg> <fdir> <fkp> <ldir> <lkp>

<start#> - начальный системный номер БД, куда вставится блок;  
 <\*.psp> - имя файла блока паспортов УСО *modbus\_1*;  
 <\*.cfg> - имя файла *modbus\_f.cfg* в исходном Зонде;  
 <fdir>, <fkp> - номер линии (1-8) и контроллера (1-254) *modbus\_f*;  
 <ldir>, <lkp> - номер линии (1-8) и контроллера (1-254) *modbus\_1*.

**Внимание!** При вставке в БД блока параметров не анализируется состав области БД, куда делается вставка. При неудачном позиционировании блока может произойти потеря уникальных данных БД.

**Внимание!** В настоящий момент развитие утилиты как самостоятельного модуля прекращено, используйте программу “Конфигуратор”.

## 2.10 Утилита `m1s_conf.exe`

Утилита `m1s_conf.exe` предназначена для трансляции БД в рамках проектов на базе ПК Зонд, но может быть использована для автоматической настройки сервера Telehost («Магистраль-1» slave) ПК Зонд. Утилита создает блок параметров БД УСО МАГИСТРАЛЬ-1 и формирует запись файла конфигурации `telecon.cfg`.

Синтаксис:

`m1s_conf.exe <start#> <end#> <flags> <dst> <mdir> <skp> <blktype> <name>`

`<start#>`, `<end#>` - начальный и конечный системный номер в исходной БД;  
`<flags>(hex):` - битовые флаги, задаются шестнадцатеричным числом:  
`bit0=1` - стирать предыдущее содержимое `telecon.cfg`; `bit1=1` - обрабатывать только `#sys` с полем 'код 2' (GOFO); `bit2=1` - выдавать сообщения о трансляции параметров; `bit3=1` - ставить 'управление извне' для дискретных двубитных управляемых (для параметров `MAGISTRAL_1`, `MODBUS_M`, `LOMIKONT`); `bit4=1` - не образовывать файл блока параметров БД (имя любое); `bit5=1` - отменить квитиование у параметров получаемого блока БД; `bit6=1` - отменить звук у параметров получаемого блока БД; `bit7=1` - не используется; `bit8=1` - не транслировать управления; `bit9=1` - отменить вывод сообщений в окно у параметров получаемого блока БД; `bit10=1` - отменить вывод сообщений в файл у параметров получаемого блока БД; `bit11=1` - отменить вывод сообщений на принтер у параметров получаемого блока БД.  
`<dst>` - имя файла выходного блока БД;  
`<mdir>` - номер линии (0-7) в МАГИСТРАЛЬ-1 мастере;  
`<skp>` - номер КП (1-77) в `telecon`;  
`<blktype>` - тип субблока для трансляции аналоговых параметров (ТТ, ТИД или FL);  
`<name>` - имя контроллера в МАГИСТРАЛЬ-1 мастере (без пробелов).

**Внимание!** В настоящий момент развитие утилиты как самостоятельного модуля прекращено, используйте программу «Конфигуратор».

## 2.11 Утилита `ftrans.exe`

Компилятор формул "ФОР". Производит трансляцию исходного текста формул в бинарный код, и сохраняет его в заданном алгоблоке УСО «Вычислитель».

Запуск:

`ftrans.exe <algnum> <flags> <srcfilename> [trunticks] [savemin] [outnum] [algname]`

`<algnum>` - номер алгоблока, в который сохранить результирующий код (1 ... 16);  
`<flags>` - битовые флаги (hex):  
`бит0: 1=не сохранять результаты (проверка кода); бит1: 1=печатать дамп кода, 0=нет; бит2: 1=печатать имена переменных и функций пользователя, 0=нет;`  
`<srcfilename>` - путь к файлу исходного текста;  
`[trunticks]` - период выполнения алгоблока в тиках по 55 мсек (допустимые значения 0 - выполнять непрерывно ... 65535);

- [savemin]** - период сохранения переменных AOUT & DOUT в минутах (допустимые значения 0 не сохранять ... 255);
- [outnum]** - число переменных AOUT & DOUT в алгоблоке (допустимые значения 0 ... 65535);
- [algname]** - название алгоблока (без пробелов).
- Если установлен бит 0 флагов («не сохранять результаты (проверка кода)»), параметры в скобках [] можно не задавать.

Например:

```
utils\frans.exe 1 5 pu04\form\1.txt
```

Экран после завершения работы программы по ошибке:

```
Утилита FTRANS.EXE (Комплекс Программ Зонд v4.40.0092)Copyright SCADA-Pro ltd.
Moscow. Russia.Алгоблок #: 1Флаги : 05h -только проверка исходного текста;
-вывод имен переменных и функций;Файл исходного текста:
pu04\form\1.txtПеременная: AПеременная: BПеременная: CПеременная: DПеременная:
EПеременная: F Переменная: GСтрока 26: Неверное число аргументовВ файле pu04\form\
1.txt обнаружены ошибки, выполняемый код не сохранен.
```

**Внимание!** В настоящий момент развитие утилиты как самостоятельного модуля прекращено, используйте программу “Конфигуратор”.

## 2.12 Утилита smsserv.exe и smsservw.exe

Утилита smsserv.exe (версия для DOS) и smsservw.exe (версия для DOS и Windows) предназначена для автоматизированного создания конфигурационного файла службы коротких сообщений SMS\_SERV.CNF.

Запуск:

```
smsserv.exe <start#> <end#> <flags> <tty#> <type> <telephone#>
```

- <start#>** - начальный системный номер обрабатываемого блока БД;
- <end#>** - конечный системный номер блока БД;
- <flags>** - флаги, HEX:
- bit0=1 - посылать сообщения при изменении значения;
  - bit1=1 - при изменении достоверности;
  - bit2=1 - при запуске-останове задачи;
  - bit4=1 - использовать флаг БД "вкл.в графики";
  - bit5=1 - использовать Code1;
  - bit6=1 - использовать Code2;
  - bit7=1 - включить все в блоке;
  - bit8=1 - включить все в БД (независимо от границ блока);
- <tty#>** - номер канала ввода-вывода RESIDENT.CNF или WINTTY.CNF;
- <type>** - получатель ЗОНД/ПОЛЬЗОВАТЕЛЬ;
- <telephone#>** - номер телефона получателя SMS.

### 3. Утилиты конфигурирования удаленного сервиса

#### 3.1 Утилита `red_num.exe`

Конфигурация службы удаленного сервиса представляет собой список записей статической коммутации каналов сервиса. Также в ПК Зонд реализован специальный процесс сервера удаленных команд с несколькими каналами поступления команд (modbus message block server, mmb-сервер Док. 3).

Утилита `red_num.exe` задает размер таблицы коммутации каналов удаленного сервиса и число каналов сервера удаленных команд:

Синтаксис:

`red_num <rednum> <mmbnum>`

`<rednum>` - число каналов редиректора;  
`<mmbnum>` - число каналов mmb-сервера.

**Внимание!** В настоящий момент развитие утилиты как самостоятельного модуля прекращено, используйте программу “Конфигуратор”.

#### 3.2 Утилита `red_conf.exe`

Каналы удаленного сервиса скоммутированы сквозь несколько узлов. Вверху каждого канала - точка входа пользователя, внизу - сервер удаленного сервиса.

Утилита `red_conf.exe` конфигурирует одну запись таблицы коммутации каналов удаленного сервиса. Различают три типа записей: запись клиента, запись редиректора и запись сервера. Запись клиента содержит ссылку на контроллер конфигурации `modbus_m` и используется как точка входа пользователя. Запись редиректора содержит ссылки на контроллеры конфигураций `modbus_s` и `modbus_m` и используется для реализации промежуточных точек канала. Запись редиректора содержит ссылку на контроллер конфигурации `modbus_s` и номер канала сервера и используется для реализации конечных точек канала и входа в сервер.

Синтаксис:

`red_conf <redir> <port> <name> <rtype> ...`

`<redir>` - номер канала в таблице коммутации;  
`<port>` - уникальный индекс канала в таблице коммутации (1-254);  
`<name>` - имя канала (без пробелов, до 5 информативных символов);  
`<rtype>` = 'REDIR' (канал-редиректор), в этом случае далее:  
`<mdir>` - номер линии в `modbus_s` (1-8);  
`<skp>` - номер контроллера в `modbus_s` (1-254);  
`<mdir>` - номер линии в `modbus_m` (1-16);  
`<mkp>` - номер контроллера в `modbus_m` (1-254).  
`<rtype>` = 'CLIENT' (канал от клиента(ИЧМ)), в этом случае далее:  
`<mdir>` - номер линии в `modbus_m` (1-16);  
`<mkp>` - номер контроллера в `modbus_m` (1-254).  
`<rtype>` = 'SERVER' (канал к mmb-серверу), в этом случае далее:  
`<mdir>` - номер линии в `modbus_s` (1-8);

**<skp>** - номер контроллера в modbus\_s (1-254);  
**<mmbch>** - номер канала сервера.

Параметр “уникальный индекс канала” должен быть уникален для канала в рамках проекта. Если при вызове red\_conf.exe формируемая запись ссылается на контроллеры конфигураций modbus\_m или modbus\_s, которые не были ранее созданы или были созданы без флага “пропускать mmb-пакеты”, то работа утилиты будет закончена аварийно.

**Внимание!** В настоящий момент развитие утилиты как самостоятельного модуля прекращено, используйте программу “Конфигуратор”.

## 4. Утилиты формирования библиотек графических фрагментов

Утилиты работают с библиотеками фрагментов, которые обеспечивают хранение и отображение фрагментов мнемосхем технологических параметров. В структуре библиотеки можно выделить статическую и динамическую части (массив статик и массив динамик). Структура библиотеки описана в Док. 4.

### 4.1 Утилита `getfrag.exe`

Утилита `getfrag.exe` сохраняет фрагмент из библиотеки фрагментов в файл.

Синтаксис:

***getfrag n[+m] [FileName]***

***n*** - номер фрагмента (1...255);

***+m*** - сдвиг системных номеров на образующимся фрагменте в файле (всегда со знаком);

***FileName*** - необязательный параметр, имя результирующего файла. Если имя не задано явно, фрагмент выделяется в файл с именем `frag.nnn` в директорию БД, где `nnn` - номер фрагмента.

Пример: ***getfrag.exe 5-2 c:\frag\frag.\_5\_***  
***getfrag.exe 12+32***

**Внимание!** В настоящий момент развитие утилиты как самостоятельного модуля прекращено, используйте программу “Конфигуратор”.

### 4.2 Утилита `putfrag.exe`

Утилита `putfrag.exe` вставляет фрагмент из файла в библиотеку фрагментов.

Синтаксис:

***putfrag n[y] [FileName]***

***n[y]*** - номер вставляемого фрагмента (1...255). Если не указан 'y', и в библиотеке уже существует фрагмент с номером `n`, то вставки не произойдет. Если за 'n' указан 'y' - произойдет вставка с перезаписью существующего фрагмента.

***FileName*** - необязательный параметр, имя файла содержащего фрагмент. Если имя не задано явно, фрагмент берется из файла с именем `FRAG.nnn` из директории БД, где `nnn` - номер фрагмента.

Пример: ***putfrag.exe 10y***  
***putfrag.exe 210 base\fragment.10***

Вставляемые файлы фрагментов могут быть образованы как автоматически утилитой `getfrag.exe`, так и программой `viza440.exe`. Их структура приведена в Док. 4. Если в библиотеке уже существует статика, номер которой совпадает с номером статик вставляемого фрагмента, то утилита сравнивает контрольные суммы статик. Если они равны, то статика не вставляется (считается, что она не изменилась). Если они не равны,

то статика вставляется на свободное место, и на нее устанавливается ссылка на вставленном фрагменте.

**Внимание!** В настоящий момент развитие утилиты как самостоятельного модуля прекращено, используйте программу “Конфигуратор”.

### 4.3 Утилита *fragsort.exe*

Утилита *fragsort.exe* сортирует фрагменты библиотеки фрагментов таким образом, чтобы выполнялось правило “номер фрагмента равен номеру статике”.

Синтаксис:

*fragsort.exe*

**Внимание!** В настоящий момент развитие утилиты как самостоятельного модуля прекращено, используйте программу “Конфигуратор”.

### 4.4 Утилита *delstat.exe*

Утилита *delstat.exe* удаляет неиспользуемые статике из библиотеки фрагментов.

Синтаксис:

*delstat.exe*

**Внимание!** В настоящий момент развитие утилиты как самостоятельного модуля прекращено, используйте программу “Конфигуратор”.

### 4.5 Утилита *clrfrag.exe*

Утилита *clrfrag.exe* удаляет с фрагмента ссылки на параметры, отсутствующие в базе данных.

Синтаксис:

*clrfrag.exe [frag\_num]*

*frag\_num* – номер фрагмента (специальное значение ALL – обработать все фрагменты библиотеки).

**Внимание!** В настоящий момент развитие утилиты как самостоятельного модуля прекращено, используйте программу “Конфигуратор”.

## 5. Утилиты формирования графической истории

Подсистема «Графическая история» обеспечивает хранение и отображение в виде графиков историю изменения технологических параметров. Конфигурация и текущие значения хранятся в файле `history.grh` в директории базы данных ПК Зонд. Архивные файлы сохраняются в поддиректории `hyst_arg` директории базы данных. Архивные файлы упакованы встроенным архиватором `asu_pack`.

Исторические значения хранятся в кольцевом буфере. Вновь записываемое значение переписывает самое старое. Файл `history.grh` содержит историю по заданному числу технологических параметров. Для всех графиков одновременно задается число значений, хранящихся в кольцевом буфере и период сохранения данных. Также файл графической истории содержит ссылки на параметры в базе данных, по которым должна собираться история.

Все утилиты должны запускаться из корневого каталога ПК Зонд. При работе они используют данные файла конфигурации `zondviza.cfg`.

### 5.1 Утилита `grhflset.exe`

Утилита `grhflset.exe` устанавливает флаг “требуется вести архивацию текущих значений” в паспортах параметров. В дальнейшем на основании значения этого флага ссылки на параметры заносятся в файл графической истории `history.grh` утилитой `grhadd.exe`.

Синтаксис:

**`grhflset.exe`**

В окне отображается список параметров (в строке системный номер, репер и название параметра) и флаги архивации, которые изменяются по клавише <Enter> (Рис. 1).

00060	ШЛФ КР57 ЦЗП	ЦЕХ3	ПАЛЛАСОВКА	ШЛЕЙФ КРАНА	57	НЕТ
00061	ШЛФ КР72 ЦЗП	ЦЕХ3	ПАЛЛАСОВКА	ШЛЕЙФ КРАНА	72	НЕТ
00062	ШЛФ КР20 ЦЗП	ЦЕХ3	ПАЛЛАСОВКА	ШЛЕЙФ КРАНА	20	НЕТ
00063	ШЛФ КР7 ЦЗП	ЦЕХ3	ПАЛЛАСОВКА	ШЛЕЙФ КРАНА	7	НЕТ
00064	ШЛФ КР8 ЦЗП	ЦЕХ3	ПАЛЛАСОВКА	ШЛЕЙФ КРАНА	8	НЕТ
00065	#000065 РЕЗЕРВ					НЕТ
00066	КП21 БТИ 1ИЭ03	КП21	СВЯЗЬ С БТИ	1ИЭ03		НЕТ
00067	ДАТЧ. Т БТИКП21	КП21	ДАТЧ. ТЕМПЕРАТУРЫ	В БТИ		НЕТ
00068	Т БТИ КП21	КП21	ТЕМПЕРАТУРА	В БТИ		ДА
00069	РВХ ЦЗП	ЦЕХ3	ПАЛЛАСОВКА	ДАВЛЕНИЕ ГАЗА	НА ВХОДЕ	ДА
00070	РВЫХ ЦЗП	ЦЕХ3	ПАЛЛАСОВКА	ДАВЛЕНИЕ ГАЗА	НА ВЫХОДЕ	ДА
00071	ТВХ ЦЗП	ЦЕХ3	ПАЛЛАСОВКА	ТЕМПЕРАТУРА ГАЗА	НА ВХОДЕ	ДА
00072	ТВЫХ ЦЗП	ЦЕХ3	ПАЛЛАСОВКА	ТЕМПЕРАТУРА ГАЗА	НА ВЫХОДЕ	ДА
00073	ТОКР.ВОЗД. ЦЗП	ЦЕХ3	ПАЛЛАСОВКА	ТЕМПЕРАТУРА	ОКРУЖАЮЩЕГО ВОЗДУХА	ДА
00074	#000074 РЕЗЕРВ					НЕТ
00075	ОБ ТНД А1 ЦЗП	ЦЕХ3	ПАЛЛАСОВКА	АГРЕГАТ 1	ОБОРОТЫ ТНД	ДА
00076	ТВЫХЛОП А1 ЦЗП	ЦЕХ3	ПАЛЛАСОВКА	АГРЕГАТ 1	Т ВЫХЛОПНЫХ ГАЗОВ	ДА
00077	КП21 БТИ 2ИЭ03	ЦЕХ3	ПАЛЛАСОВКА	СВЯЗЬ С БТИ	2ИЭ03	НЕТ
00078	ОБ.ТНД А2 ЦЗП	ЦЕХ3	ПАЛЛАСОВКА	АГРЕГАТ 2	ОБОРОТЫ ТНД	ДА
00079	ТВЫХЛОП А2 ЦЗП	ЦЕХ3	ПАЛЛАСОВКА	АГРЕГАТ 2	Т ВЫХЛОПНЫХ ГАЗОВ	ДА
00080	ОБ.ТНД А3 ЦЗП	ЦЕХ3	ПАЛЛАСОВКА	АГРЕГАТ 3	ОБОРОТЫ ТНД	ДА
00081	ТВЫХЛОП А3 ЦЗП	ЦЕХ3	ПАЛЛАСОВКА	АГРЕГАТ 3	Т ВЫХЛОПНЫХ ГАЗОВ	ДА
00082	ОБ.ТНД А4 ЦЗП	ЦЕХ3	ПАЛЛАСОВКА	АГРЕГАТ 4	ОБОРОТЫ ТНД	ДА

Рис. 1. Экран при работе утилиты `grhflset.exe`

**Внимание!** В настоящий момент развитие утилиты как самостоятельного модуля прекращено, используйте программу “Конфигуратор”.

## 5.2 Утилита **grhmk.exe**

Утилита **grhmk.exe** (Graphics History Make) предназначена для создания новых файлов графической истории.

Утилита может быть запущена на выполнение из любого каталога, поскольку не использует файлов конфигурации ПК Зонд.

Синтаксис:

**grhmk.exe** [*y*] [*fGRHPATH*] *<gr\_quant>* *<steptime>* *<point\_quant>* *<pack\_count>*

<b>-y</b>	- на все вопросы отвечать Y (Да);
<b>-fGRHPATH</b>	- полный путь к создаваемому файлу; По умолчанию, если опция -f не задана, создается файл HISTORY.GRH в текущей директории.
<b>gr_quant</b>	- емкость файла (число параметров, по которым будет собираться история);
<b>steptime</b>	- период занесения значений в файл в секундах. В дальнейшем, при помощи утилиты <b>grhstep</b> - можно выбрать размерность единицы измерения секунды или минуты;
<b>point_quant</b>	- число точек в графике (глубина истории);
<b>pack_count</b>	- упаковывать архивный файл при записи каждой кратной pack_count точки.

В квадратных скобках указаны необязательные параметры.

Пример: **grhmk.exe -y -fBASE\history.grh 100 5 320 -1**  
**grhmk.exe 320 60 1000 50**

**Внимание!** В настоящий момент развитие утилиты как самостоятельного модуля прекращено, используйте программу “Конфигуратор”.

## 5.3 Утилита **grhadd.exe**

Утилита **grhadd.exe** (Graphics History Add) предназначена для занесения ссылок на параметры базы данных «Зонд» в файл графической истории.

Синтаксис:

**grhadd.exe** [*fGRHPATH*] *startnum endnum ...*

<b>-fGRHPATH</b>	- полный путь к файлу истории, в который добавляются параметры; Если файл не указан явно, то по умолчанию, используется файл HISTORY.GRH в директории БД "Зонд", указанный в конфигурации (ZONDVIZA.CFG).
<b>startnum</b>	- начальный номер диапазона вставляемых системных номеров;
<b>endnum</b>	- конечный номер диапазона вставляемых системных номеров параметров.

Начальный и конечный номер разделены знаком "минус". Для вставки одного параметра, укажите его системный номер. В квадратных скобках указаны необязательные параметры.

Пример: **grhadd.exe -fBASE\history.grh 10 12 54-110**

### *grhadd.exe 1001-1141 54-110*

**Внимание!** В настоящий момент развитие утилиты как самостоятельного модуля прекращено, используйте программу “Конфигуратор”.

#### **5.4 Утилита *grhstep.exe***

Утилита *grhstep.exe* (Graphics History Step) предназначена для задания интервала занесения значений в графическую историю.

Синтаксис:

***grhstep.exe [NNNsec]***

**NNNsec** - интервал NNN секунд, **NNNmin** - интервал NNN минут;

Если NNN не указано, то задается размерность. Если не указана размерность (min/sec), изменяется только период. Если аргумент не указан, на экране отображаются текущие установки.

Пример:            *grhstep.exe 10sec*  
                      *grhstep.exe 2min*  
                      *grhstep.exe 30*  
                      *grhstep.exe sec*

**Внимание!** В настоящий момент развитие утилиты как самостоятельного модуля прекращено, используйте программу “Конфигуратор”.

## 6. Утилиты трансляции размерности и таблицы цветов

Утилиты должны запускаться из корневого каталога ПК Зонд. При работе они используют данные файла конфигурации `zondviza.cfg`.

### 6.1 Утилита `get_rt.exe`

Утилита `get_rt.exe` сохраняет размерности и таблицу цветов из файла БД ПК Зонд в файл.

Синтаксис:

***get\_rt*** *<path>*

*<path>* - имя файла.

**Внимание!** В настоящий момент развитие утилиты как самостоятельного модуля прекращено, используйте программу “Конфигуратор”.

### 6.2 Утилита `put_rt.exe`

Утилита `put_rt.exe` вставляет размерности и таблицу цветов файла в БД ПК Зонд из файла.

Синтаксис:

***put\_rt*** *<path>*

*<path>* - имя файла.

**Внимание!** В настоящий момент развитие утилиты как самостоятельного модуля прекращено, используйте программу “Конфигуратор”.

## 7. Утилиты обработки базы данных

### 7.1 Утилита gofo\_sys.exe

Утилита gofo\_sys предназначена для формирования конфигурационных файлов системы GOFO для последующего обмена данными по протоколу modbus.

Утилита gofo\_sys должна запускаться из корневой директории ПК Зонд. Она формирует в каталоге БД ПК Зонд следующие текстовые файлы:

- proccomm           - файл конфигурации сигналов управления;
- acqzond            - файл конфигурации сигналов опроса.

Синтаксис:

**gofo\_sys** <sdir>, <skp> <acsysname> [<flags>] <output dir>

- <sdir>, <skp>       - номер линии (1..8) и контроллера (1..254) modbus slave;
- <acsysname>        - имя системы сбора;
- <flags>             - необязательный параметр.  
                   bit0=1 - выходные файлы в формате DOS;  
                   bit0=0 - выходные файлы в формате UNIX;  
                   bit1=1 - имя поля оборудования EQUIP\_NAME;  
                   bit1=0 - имя поля оборудования EQUIP\_SYSTEM\_NAME;  
                   bit2=1 - не ставить CONV\_COEFF = -1 для параметров типа «потенциал трубы».  
                   По умолчанию все биты флагов равны 0.
- <output dir>       - директория, куда записывать выходные файлы (proccomm, acqzond).

Задание бита 1 во флагах позволяет генерировать файлы совместимые с программным обеспечением GOFO проекта «GOFO-2» (bit1=1) и проекта «Ямал-Европа» (bit1=0).

Информация в UNIX-файлах имеет формат, позволяющий напрямую воспринимать ее конфигуратором GOFO. В файлы попадет информация о параметре, который в текущей БД ПК Зонд обладает следующими свойствами:

- \* В паспорте параметра заполнено поле “код2” в соответствии с системой имен GOFO;
- \* В конфигурации указанной линии modbus-slave имеется ссылка на параметр (тип “значение”), то есть параметру выделен modbus-адрес пространстве Discret или Input.  
       Для возможности управления со стороны GOFO - дополнительно нужна ссылка на параметр в пространстве Coils/Holding.

Ссылки формируются вручную в панели инженера modbus-slave или автоматически утилитой mbs\_conf.exe. Для автоматического формирования ссылки для управления в пространстве Coils/Holding Registers в паспорте параметра в поле “тип параметра” должно быть указано “управляемый извне”.

## 7.2 Утилита gofo\_chk.exe

Утилита gofo\_chk.exe проверяет уникальность имен, кодов и подключений параметров.

Синтаксис:

**gofo\_chk.exe [flags]**

**flags**

- bit0=1 - проверять "Код 1";
- bit1=1 - проверять "Код 2";
- bit2=1 - проверять "Полные наименования";
- bit3=1 - проверять реперы параметров;
- bit4=1 - проверять подключения параметров.

По умолчанию flags=0Fh (проверять все имена).

## 7.3 Утилита basecmp.exe и basecmpw.exe

Утилита basecmp.exe для DOS производит сравнение двух БД ПК Зонд. Файл basecmp.exe может выполняться как в DOS так и в Windows. Доступ к обеим БД осуществляется с использованием данных соответствующих файлов zondviza.cfg, которые задаются явно в параметрах командной строки.

Синтаксис:

**basecmp flags zondviza1 zondviza2 path1 path2**

**flags**

- bit0=1 (0x1) - Не сравнивать контрольные суммы паспортов;
- bit0=2 (0x2) - Не сравнивать полные наименования параметров;
- bit0=3 (0x4) - Не сравнивать Код1 параметров";
- bit0=4 (0x8) - Не сравнивать Код2 параметров";

**zondviza1** - путь и имя файла zondviza.cfg первой базы;

**zondviza2** - путь и имя файла zondviza.cfg второй базы.

**path1** - путь к каталогу первой БД, должен заканчиваться '\'

**path2** - путь к каталогу второй БД, должен заканчиваться '\'

Утилита выдает отчет о различиях в паспортах параметров с одинаковыми системными номерами.

Например, имеется две БД расположенные по путям D:\B1\BASE и D:\B2\BASE.  
Командная строка запуска утилиты выглядит следующим образом:

**utils\basecmp.exe 1 d:\b1\zondviza.cfg d:\b2\zondviza.cfg d:\b1\base\ d:\b2\base\**

## 7.4 Утилита dbf\_clear.exe

Утилита dbfclear.exe удаляет из файла namesprm.dbf те записи (поля "имя параметра", "код1", "код2"), которым соответствуют системные номера с отсутствующими системными номерами в файле БД ПК Зонд zond.db.

Синтаксис:

*dbfclear.exe*

### 7.5 Утилита *tula\_tab.exe*

Утилита *tula\_tab.exe* автоматически формирует конфигурацию *modbus\_s* на основе данных текстового файла следующего формата:

Синтаксис:

*tula\_tab* <*discr conf*> <*analog conf*> <*s dir*> <*s kp*> <*kvit di*> <*kvit an*> <*gr di*> <*bound*>

<*discr conf*> - текстовый файл ссылок по дискретным;  
 <*analog conf*> - текстовый файл ссылок по аналоговым;  
 <*sdir*>, <*skp*> - номер линии (1-8) и контроллера (1-254) *modbus\_s*;  
 <*kvit di*> - первый #sys области квитирования дискретных;  
 <*kvit an*> - первый #sys области квитирования аналоговых;  
 <*gr di*> - граница области достоверности в *modbus\_m* (dec).

Текстовые файлы имеют формат:

```
0 185
1 186
2 78
...
```

где первый столбец содержит *modbus*-адреса, а второй – системные номера параметров БД ПК Зонд. Если нужно зарезервировать адрес, но сопоставить ему параметр невозможно, нужно писать “-1” во втором столбце.

Для дискретных параметров автоматически (без специального задания из файла) формируется также область адресов достоверности и квитирования, для аналоговых – квитирования.

### 7.6 Утилита *clrdelta.exe*

Утилита *clrdelta.exe* очищает поля ДельтаТГ (дельта технологической границы) и ДельтаАГ (дельта аварийной границы) паспортов параметров базы данных (см. Док. 5).

Используйте синтаксис:

*clrdelta.exe* <*flags*> <*start#*> <*num#*>

<*flags*> - режим работы утилиты:  
 bit0=1 - не трогать ДельтаТГ;  
 bit1=1 - не трогать ДельтаАГ.  
 <*start#*> - начальный системный номер (0...) очищаемого блока параметров;  
 <*num#*> - размер блока параметров в БД.

## 7.7 Утилита **clr\_toti.exe**

Утилита **clr\_toti.exe** очищает номера телеопераций телеизмерения (ТИ), опрашиваемых при телеуправлении (ТУ) в подключениях параметров БД УСО «Магистраль-1М».

Используйте синтаксис:

**clr\_toti** <start#> <num#>

<start#> - начальный системный номер (0...) очищаемого блока параметров;  
 <num#> - размер блока параметров в БД.

## 7.8 Утилита **clr\_knum.exe**

Утилита **clr\_knum.exe** очищает старшие биты в подключении номера контроллера параметров БД УСО «Modbus master».

В подключении параметров УСО «Modbus master» номер контроллера может быть от 1 до 255. Очистка приведет номера контроллеров в диапазон от 1 до 63.

Утилита используется при конвертировании конфигурационных файлов БД во время смены версии ПО «Зонд».

Синтаксис:

**clr\_knum** <start#> <num#>

<start#> - начальный системный номер (0...) очищаемого блока параметров;  
 <num#> - размер блока параметров в БД.

## 7.9 Утилита **chgcolor.exe**

Утилита **chgcolor.exe** позволяет производить замену номеров таблиц цветов и признаки инверсия и перестановка битов в паспортах дискретных параметров БД "ЗОНД".  
 Параметры запуска:

**chgcolor.exe what new invchg uso** <start#> <end#>

**what** - какую таблицу цветов заменять (номер таблицы 0...);  
**new** - на какую таблицу заменять (номер таблицы 0...);  
**invchg** - в какое состояние устанавливать признак инверсии (Hex);  
 0-ой бит (младший) - значение инверсии (0 - нет, 1 - да); 1-ый бит - перестановка бит (0 - нет, 1 - да);  
 Значение 0 - перестановка НЕТ, инверсия НЕТ;  
           1 - перестановка НЕТ, инверсия ДА;           2 - перестановка ДА, инверсия НЕТ;  
           3 - перестановка ДА, инверсия ДА;  
**uso** - номер УСО, (-1 - для всех);   ORION=0,   MAGISTRAL\_1=1,  
 MAGISTRAL\_UHO=2, IMPULS=3,   FUXON=4,   SF\_2=5,  
 MODBUS\_M=6, SPRINT=7,   MICRODAT=8,   EMICON=9,  
 MARK\_CKD=10, MODBUS\_L=11, EVAL\_USO=12,   REMIKONT=13,  
 LOMIKONT=14, SH\_711=15,   SEVC\_D\_USO=16,   VNS\_USO=17,  
 GF\_3P/PM=18, EK\_88/260=19,   GVC\_2010=20,   IMPULS\_SF=21,  
 OPC\_M=22, IRTM=23,   ABB\_EUROALPHA=24,   METRAN\_120X=25,  
 SET4TM=26, USO\_FILE=27, REMZOND =30, DIAGNOSTIC=31

***start#*** - начальный системный номер (по умолчанию 0);  
***end#*** - конечный системный номер (по умолчанию, последний в БД);

Параметры *start#* и *end#* могут не указываться. По умолчанию, утилита работает по всей БД. Параметр *end#* не может быть указан без *start#*.

Например: ***d:\zond.440>utils\chgcolor.exe 2 3 1 -1d:\zond.440> utils\chgcolor.exe 2 3 3  
1 10 420***

## 8. Экспорт базы данных и конфигурационных файлов

### 8.1 Утилита `mbs_list.exe`

Утилита `mbs_list.exe` позволяет сохранить список параметров Modbus-slave контроллера в CSV файл.

Синтаксис:

***mbs\_list.exe flags dir# kont# <filename.txt>***

***flags*** - флаги, управляющие полями вывода (шестнадцатеричная константа):  
 bit0 : 1-выводить системные номера / 0-нет;  
 bit1 : 1-выводить полное наименование / 0-нет;  
 bit2 : 1-выводить code1 / 0-нет;  
 bit3 : 1-выводить code2 / 0-нет;  
 tbit4 : 1-использовать кодировку Windows 1251 / 0-использовать кодировку CP 866 (DOS).

***dir#*** - номер направления Modbus slave (1...8);  
***kont#*** - номер slave-контроллера (1...254);  
***<filename.txt>*** - имя результирующего файла (необязательный параметр, `sl_list.txt`-по умолчанию).

Например: ***d:\zond.440>utils\mbs\_list.exe 0 1 1 sl\_1\_1.txt***

Фрагмент результирующего файла:

```
33,V,СИРЕНА ББ 520
34,V,ДЕЛЕГ.ПРАВ ПУ
10003,V,ДАТ.ПРИС.ББ606
15003,T,ДАТ.ПРИС.ББ606
10005,V,КРАН 109Б1
15005,T,КРАН 109Б1
```

### 8.2 Утилита `prn_base.exe`

Утилита `prn_base.exe` позволяет сделать распечатку базы данных (Создание перечня входных сигналов по РД 50-34.698-90). Распечатка ведется в текстовый файл формата CSV (текст разделенный запятыми). Символ разделитель указывается.

Синтаксис:

***prn\_base <flags> <param type> <file name> <delimiter>***

***<flags>***

- битовые флаги, задающие режим работы программы:
- bit0=1 печатать системный номер;
- bit1=1 печатать репер;
- bit2=1 печатать имя код1, только с имеющимся код1;
- bit3=1 печатать имя код2, только с имеющимся код2;
- bit4=1 только с имеющимся код1;
- bit5=1 только с имеющимся код2;
- bit6=1 печатать только текстовые имена;

	- bit7=1 печатать в кодировке Win1251;
	Значение <flags> задается шестнадцатеричной константой.
<param type>	- тип параметров БД:- 0 - аналоговые;
	- 1 - дискретные;- 6 - измерительные линии;
<file name>	- имя результирующего файла;
<delimiter>	- символ-разделитель полей.

Задание всех параметров для работы утилиты обязательно.

Пример: *prn\_base.exe 03 0 analog.txt ;'*  
*prn\_base.exe 83 1 d\_win.txt ;'*

### 8.3 Утилита dbf\_fill.exe

Утилита dbf\_fill предназначена для экспорта данных паспортов параметров и значений параметров. Утилита работает в среде DOS и Windows.

Синтаксис:

*dbf\_fill [/Cconf\_file\_name] [/Llog\_file\_name] [/B]<dst>*

*[/Cconf\_file\_name]* - путь и имя файла конфигурации (по умолчанию zondviza.cfg);  
*[/Llog\_file\_name]* - путь и имя LOG файла (по умолчанию dbf\_fill.log);  
*[/B]* - (Batch) пакетный режим (без ожидания нажатия на клавишу);  
*<dst>* - имя dbf-файла.

DBF файл создается по правилам, описанным в документе \*\*\* для команды FILL DBF.

Пример: *dbf\_fill.exe out\values.dbf*  
*dbf\_fill.exe /Cbase\zondviza.cfg repers.dbf*  
*dbf\_fill.exe /Cor\zondviza.or repers.dbf /Lor.log /B or\or.dbf*

## 9. Коммуникационные утилиты

Коммуникационные утилиты построены на базе тех же многозадачного ядра реального времени и драйверов устройств, которые используются в ПК Зонд. Поэтому они поддерживают все типы каналов, реализованные в ПК Зонд. Состав и настройки каналов связи задаются в файле `resident.cnf`, который должен присутствовать в директории запуска. Коммуникационные утилиты работают только в ОС MS\_DOS.

### 9.1 Утилита `nikm.exe`

Утилита `nikm.exe` предназначена для проверки связи с оборудованием, поддерживающим протокол modbus Док. 1. ]. Утилита запускается из корневой директории ПК Зонд или из любого другого места.

Синтаксис:

**`nikm <outport> <inport> [<timeout,tick> [/CRC...] [/biNNN] [/ssMMM]]`**

<b><code>&lt;outport&gt;</code>,</b>	- номера каналов связи соответственно приема и передачи в файле <code>resident.cnf</code> ;
<b><code>&lt;inport&gt;</code></b>	- номера каналов связи соответственно приема и передачи в файле <code>resident.cnf</code> ;
<b><code>&lt;timeout,tick&gt;</code></b>	- необязательный параметр, задающий временной интервал между запросами в линию в тиках (по умолчанию 18 тиков=1с.);
<b><code>[/CRC...]</code></b>	- способ вычисления контрольной суммы: CRCNO - CRC не посылается и не анализируется; CRC16 - используется CRC16 (по умолчанию); CRC32 - используется CRC32; CRCHART - используется сумма по модулю 2.
<b><code>[/biNNN]</code></b>	- если задано, между байтами формируются паузы ~NNN мс.;
<b><code>[/ssMMM]</code></b>	- если задано, до запроса посылаются MMM байт синхросерии (FF).

Пример: **`nikm 0 0 10 /CRCNO`**

- означает прием/передачу с 0 канала и такт запросов примерно 550 мс. При этом первая строка в `resident.cnf` (cm) должна быть

<b><code>device=com1,,100,100</code></b>	- для работы через Com-порты,
<b><code>device=slink485,5,2A0,100,100</code></b>	- для работы через плату RS-485 и т.д.

Пользователю предоставляются два окна-терминала, в которых отражается состав переданных (верхнее окно) и принятых (нижнее окно) посылок и всплывающая командная строка, позволяющая редактировать формат посылки-запроса. Таким образом, работа с программой сводится к набору и послылке запросов. Набор ведется в шестнадцатиричном коде, разделитель – пробел.

После нажатия клавиши **<Enter>** происходит добавление к посылке контрольного CRC-кода и начинается цикл посылки-прослушивания линии с тактом `<time>`, который может быть остановлен нажатием любой клавиши, после чего опять следует переход в режим редактирования.

В режиме посылок возможны следующие операции: установка скорости передачи (**<F2>**), установка типа четности (**<F3>**), установка числа стоп-битов (**<F4>**). Значения трех последних параметров сохраняются при выходе в бинарном файле `nikm.cfg` и загружаются и устанавливаются при следующем запуске. В режиме редактирования возможен выход из программы (**<Esc>**).

## 9.2 Утилиты *set\_mba.exe* и *set\_mbaw.exe*

Утилита *set\_mba.exe* предназначена для задания логических адресов (Modbus адресов) функциональных модулей СЛТМ «Магистраль-2».

Синтаксис:

```
set_mba.exe chl mod_id mba,  
set_mba.exe chl /Fid_table_file
```

*chl* - номер канала в resident.cnf/wintty.cnf (dec) (см. Док. 6);  
*mod\_id* - заводской номер функционального модуля;  
*mba* - задаваемый логический адрес (Modbus адрес 1...FF) (hex);  
*id\_table\_file* - брать данные из файла, полученного экспортом параметров входного порта «Магистраль-2» (программа «Конфигуратор Баз Данных», Док. 7).

Утилита *set\_mbaw.exe* отличается от *set\_mba.exe* только тем, что работает под управлением DOS и Win32 ОС.

Пример: *set\_mba.exe 0 12440053 10*  
*set\_mbaw.exe 0 12440053 12*  
*set\_mbaw.exe 0 /Fkp06.txt*

## 9.3 Утилиты *test\_id.exe* и *test\_idw.exe*

Утилита *test\_id.exe* предназначена для опроса линии modbus с целью получения информации об уникальных адресах подключенных к ней функциональных модулей СЛТМ «Магистраль-2».

Синтаксис:

```
test_id chl file_name </startN> </stopN> </timeoutN>,
```

*chl* - номер канала в resident.cnf/wintty.cnf (dec) (см. Док. 6);  
*file\_name* - имя файла результатов. Если имя результирующего файла задано, в него выводятся сведения об обнаруженных функциональных модулях, если не задано, вывод осуществляется на экран.  
</startN> - начальный Modbus адрес сканирования (N dec). По умолчанию начальный адрес равен 1.  
</stopN> - конечный Modbus адрес сканирования (N dec). По умолчанию конечный адрес равен 255.  
</timeoutN> - время ожидания ответа в тиках таймера (1/18 сек). По умолчанию равно 18 (примерно 1 сек);  
*/csv[FileName]* - создавать результирующий файл в формате CSV (текст с разделителем ‘;’). Если имя файла не задано явно (FileName), по умолчанию m2e.csv.

Утилита *test\_idw.exe* отличается от *test\_id.exe* только тем, что работает под управлением DOS и Win32 ОС.

Пример:

```
test_id 0 kp19.txt /stat1 /stop60 /timeout8
test_idw 0 kp22.txt /stat1 /stop30
test_idw 1 /csvkp22.csv /timeout10
```

#### 9.4 Утилита **mb\_uho.exe**

Утилита **mb\_uho.exe** предназначена для прослушивания линии modbus с целью проверки исправности канала связи.

Синтаксис:

```
mb_uho <dev> <speed> <parity> <stop> <fname> <ticks>,
```

**dev** - номер канала в resident.cnf (dec) (см. Док. 6);  
**speed** - скорость в линии 1200 - 115200;  
**parity** - контроль четности **e**, **o**, **n**;  
 e - even;  
 o - odd;  
 n - none.  
**stop** - число стоп-бит, 1,2;  
**fname** - файл записи трафика;  
**ticks** - время синхронизации прослушивания в тиках (1тик=55мс).

#### 9.5 Утилита **mbtest.exe**

Утилита предназначена для оценки качества канала связи. Программ запускается на разных концах канала и ведет обмен по протоколу Modbus RTU, с подсчетом ошибок различных типов.

Синтаксис:

```
MbTest.exe chl mode [speed] [pause]
```

**chl** - номер канала в resident.cnf (dec) (см. Док. 6);  
**mode** - /M - master (ведущее устройство), /S - slave (ведомое устройство);  
**speed** - sXXX (s50, s100, s150, s200, s300, s600, s1200, s2400, s4800, s9600(\*), s19200, s38400, s57600, s115200, (\*) скорость по умолчанию), s230400, s460800, s921600 - для высокоскоростных плат (COMxH, AST4H, AST5554H, ARNETH);  
**pause** - пауза между запросами, тиках (только для режима работы /M - master).

Например, для запуска теста в режиме master нужно ввести строку:

```
mbtest.exe 0 /M 18
```

Для запуска программы в режиме slave:

```
mbtest.exe 0 /S
```

В обоих примерах для работы программы используется нулевой канал ввода-вывода (см. Док. 6). Пример файла resident.cnf, используемого при тестировании связи, обеспечиваемой радиомодемами Dataradio T-96 приведен ниже.

```
;----- RESIDENT.CNF -----  
; Выбрать нужный порт удалив или поставив комментарий '!'  
  
DEVICE=COM1,,,100,100  
;DEVICE=COM2,,,100,100  
;DEVICE=COM3,,,100,100  
;DEVICE=COM4,,,100,100  
  
; Управление сигналами UART  
CHANNEL_PARAM=0,RTS,CTS,RCD  
  
XTIMER=8  
;----- Конец файла -----
```

## 10. Утилиты для слежения за работой УСО

### 10.1 Утилита convtrac.exe и convtracw.exe

Утилита convtrac.exe предназначена для распечатки бинарного файла обмена с УСО в текстовый файл. Модуль convtrac.exe работает в среде DOS. Модуль convtracw.exe работает в среде DOS и Windows.

Синтаксис:

*convtrac <trace-файл> <txt-файл> [/F<format>]*

*<trace-файл>* - имя (путь) к исходному, бинарному файлу трассировки обменов с УСО;

*<txt-файл>* - имя (путь) к результирующему, текстовому файлу, распечатки обменов;

*[/F<format>]* - формат вывода байта данных в распечатку  
format - nF, где n-число знаков F-формат вывода байта:  
X - шестнадцатеричный (рекомендуем формат "02X");  
o - восьмеричный (используется по умолчанию "-3o");  
d - десятичный.

Пример:

*convtrac.exe modbus\_m.0 modbusm0.txt /F02X*  
*convtrac.exe impuls.0 impuls0.txt /F03o*

## **11.Список используемых документов**

- Док. 1. Комплекс программ «ЗОНД». Реализация протокола Modbus в ПК Зонд. (44000035)
- Док. 2. Комплекс программ «ЗОНД». Система линейной телемеханики "Магистраль-2. Руководство по параметризации. (44000009)
- Док. 3. Комплекс программ «ЗОНД». Реализация служб удаленного сервиса. (44000071)
- Док. 4. Комплекс программ «ЗОНД». Подсистема отображения технологической информации «ВИЗА». (44000008)
- Док. 5. Комплекс программ «ЗОНД». Описание применения. (44000003)
- Док. 6. Комплекс программ «ЗОНД». Установка, конфигурирование и запуск. (44000002)
- Док. 7. Программа «Конфигуратор Баз Данных». (44000010)