
***Драйвер обмена со счетчиком электроэнергии
«ЕвроАльфа»***

Руководство программиста

Содержание

1.	Назначение драйвера	3
2.	Краткое описание счетчика «ЕвроАльфа»	4
3.	Командная строка запуска драйвера	5
4.	Правила конфигурации	9
5.	Структура пакетов обмена между «модулем опроса» и драйвером	13
6.	Протокол обмена счётчика «ЕвроАльфа» с драйвером	14

1 Назначение

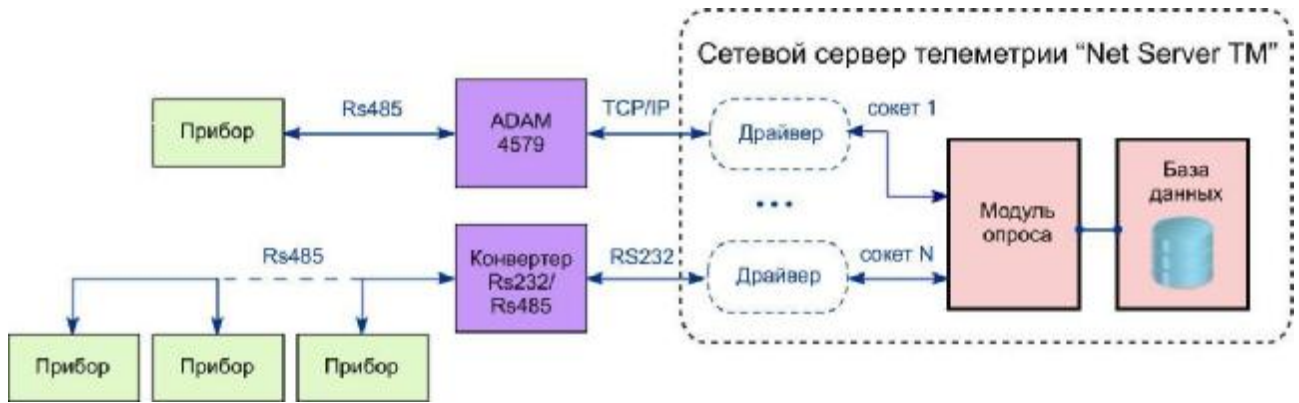


Рис. 1. Структурная схема взаимодействия драйвера с «модулем опроса».

Драйвер служит интерфейсным модулем между счётчиками «ЕвроАльфа» (далее по тексту счётчиками или устройствами) и «модулем опроса», который входит в состав сервера телеметрии «Net server TM». (См. Рис.1.) «Модуль опроса» общается с драйвером с помощью пакетов посылок. Более подробно о структуре пакетов изложено в п.5. Между драйвером и счётчиками обмен информацией осуществляется с использованием протокола, описанного в п.6. Запуск драйвера производится автоматически «модулем опроса» при его (модуля) старте. Командная строка запуска драйвера формируется на основе заданной в конфигурации сервера телеметрии информации. В командной строке указываются параметры инициализации, среди них: системный IP-адрес и порт для соединения с датчиком по протоколу TCP/IP, либо COM-порт и параметры последовательного соединения с устройством, параметры для инициализации диагностики и др. При некорректной инициализации (отсутствие необходимого параметра инициализации, либо присвоение ему некорректного значения) драйвер завершает работу.

После запуска драйвер пытается установить соединение с устройством. В случае неудачного соединения с устройством драйвер повторяет попытки соединения с периодом 20 сек. до нормального завершения. В случае удачного соединения с устройством, драйвер регулярно его опрашивает.

Драйвер постоянно опрашивает устройство. При получении запроса от «модуля опроса» драйвер немедленно формирует ответ.

На основании текущих данных драйвер формирует архивные данные: 3-х минутные, получасовые, часовые и суточные. В каждый момент времени драйвер обрабатывает не более одного заказа. При некорректном заказе драйвер выдает «модулю опроса» признак ошибки обработки заказа. Драйвер завершает свою работу:

- при закрытии «модуля опроса»;
- при отсутствии заказов от «модуля опроса» в течение времени, заданного в конфигурации.

2. Краткое описание счётчиков «ЕвроАльфа».



Многофункциональные, микропроцессорные счетчики электрической энергии серии ЕвроАЛЬФА (A1600) предназначены для учета активной и реактивной энергии и мощности в цепях переменного тока.

Функциональные возможности

- Измерение активных и реактивных энергий и мощностей в двух направлениях.
- Учет электроэнергии по 4 тарифным зонам.
- Фиксация максимальной мощности нагрузки на расчетном интервале времени.
- Запись и хранение данных графика нагрузки в памяти счетчика (до 336 дней по 4 каналам 30 минутные интервалы).
- Передача результатов измерения по цифровым и импульсным каналам связи.
- Измерение (вычисление) и отображение, параметров электроэнергии: напряжение и ток пофазно, частота сети, коэффициент мощности, фазные углы тока и напряжения.
- Автоматический контроль нагрузки с возможностью ее отключения или сигнализации.

Технические характеристики счетчика ЕвроАЛЬФА

Наименование величины	Значение
Класс точности	0,2S и 0,5S
Количество тарифов	4 тарифные зоны в день, 4 типа дней недели, 4 сезона, автоматический переход на летнее и зимнее время
Номинальный ток (максимальный ток)	1-5(10)A
Диапазон рабочих напряжений	3x57-230/100-400, 3x100-400 В
Диапазон частоты сети	47,5 - 52,5 Гц
Рабочий диапазон температур	от -40оС до +60оС
Скорость обмена информацией по оптическому порту (RS-232) по интерфейсу "токовая петля" по интерфейсу RS-485	1200,9600 бод 300, 1200, 2400, 4800, 9600 бод 2400, 4800, 9600, 19200 бод
Передаточное число	от 10 до 50000 имп/кВт.ч в зависимости от модификации счетчика
Сохранность данных в энергонезависимой памяти	20 лет
Регистрация отключений питания	до 255 регистраций
Самодиагностика счетчика	1 раз в сутки
Межповерочный интервал	8 лет
Гарантийный срок эксплуатации	3 года
Срок службы	30 лет

Госреестр №16666-97. Сертифицирован по ГОСТ 30206-94 (МЭК 687), ГОСТ 22261-944
Руководство по эксплуатации ([796 Kb](#))
Паспорт ([178 Kb](#))

3. Командная строка вызова драйвера

3.1. Командная строка вызова.

```
./eaport SERIAL=dev,speed,parity,data_b,stop_b PORT=Nport  
  DEVICES=N1,...,Nn [TKILL=suic_tout] [LOG=log_file]  
  [DEBUG=dbg_val] [CONF=config_file] [BASE=base_file]  
  [BDIR=dir_base] [PARS=par_file] [PAROLE=device_password]
```

или

```
./eaport IP=ip_addr_or_name:ip_port PORT=Nport  
  DEVICES=N1,...,Nn [TKILL=suic_tout] [LOG=log_file]  
  [DEBUG=dbg_val] [CONF=config_file] [BASE=base_file]  
  [BDIR=dir_base] [PARS=par_file] [PAROLE=device_password]
```

где **eaport** – имя запускаемого модуля (имя драйвера);

dev – устройство, обслуживающее COM-порт;

speed – скорость в бодах;

parity – чётность (всегда n, сохранена для «общности» формы);

data_b – количество бит в байте (всегда 8, сохранено для «общности» формы);

stop_b – количество стоповых бит (1 или 2);

ip_addr_or_name:ip_port – ip-адрес и порт (сокет), через который драйвер ведёт диалог с устройствами (драйвер является клиентом);

Nport – сокетный порт верхнего уровня, по которому поступают информационные запросы и отправляются ответы;

N1...Nn – имена устройств, с первой встреченной в имени десятичной цифры начинается адрес устройства;

suic_tout – таймаут в секундах «самоубийства» программы при отсутствии сокетных запросов (умолчание: 0 – никогда);

log_file – файл журнала (умолчание: стандартный вывод – экран);

config_file – файл конфигурации (умолчание: eaport.conf);

dir_base – каталог файлов базы (умолчание: ./ea_b);

base_file – префикс имён файлов базы (умолчание: без префикса);

device_password – 4-ёхбайтовый hex-пароль устройства (умолчание: 0);

par_file – файл списка параметров (умолчание: см. ниже);

dbg_val – битовое поле (hex) разрешения вывода в журнал отладочной информации (умолчание: ffff);:

FD_OK 1 (OK message)

FD_CPACK 2 (device dialog)

FD_MESS 4 (result message)

FD_INPACK 8 (socket dialog - in packets)

FD_OUTPACK 10 (socket dialog - out packets)

FD_ARCH 40 (writing base messages)

FD_POLL 80 (poll dialog in device dialog)

Примеры:

```
earport SERIAL=/dev/ttyS1,9600,n,8,1 PORT=7720 DEVICES=7 TKILL=3600
```

```
earport IP=10.0.1.27:5201 PORT=7720 DEVICES=1,tc16,22 LOG=dep.log DEBUG=18
```

Строка запуска также поясняется при вызове драйвера без аргументов:
./earport

3.2. Список параметров

Имя	Ответ	Тип	Назначение
"s-time"	"time="	c	Опрос времени
"Ua"	"Ua="	c,d,h,mN	Напряжение по фазе А
"Ub"	"Ub="	c,d,h,mN	Напряжение по фазе В
"Uc"	"Uc="	c,d,h,mN	Напряжение по фазе С
"Ia"	"Ia="	c,d,h,mN	Ток по фазе А
"Ib"	"Ib="	c,d,h,mN	Ток по фазе В
"Ic"	"Ic="	c,d,h,mN	Ток по фазе С
"F"	"F="	c,d,h,mN	Частота по фазе А
"cos"	"cos="	c,d,h,mN	Косинус "фи" по сумме фаз
"P"	"P="	c,d,h,mN	Мгновенная активная мощность по сумме фаз
"Q"	"Q="	c,d,h,mN	Мгновенная реактивная мощность по сумме фаз
"So"	"So="	c,d,h,mN	Мгновенная полная мощность по сумме фаз
"Wp_f"	"Wp_f="	c,d,h,mN	Потребленная (прямая) активная энергия
"Wp_b"	"Wp_b="	c,d,h,mN	Выделенная (обратная) активная энергия
"Wp"	"Wp="	c,d,h,mN	Суммарная (разность прямой и обратной) активная энергия
"Wq_f"	"Wq_f="	c,d,h,mN	Потребленная (прямая) реактивная энергия
"Wq_b"	"Wq_b="	c,d,h,mN	Выделенная (обратная) реактивная энергия
"Wq"	"Wq="	c,d,h,mN	Суммарная (разность прямой и обратной) реактивная энергия
"Ws"	"Ws="	d,h,mN	Суммарная (разность прямой и обратной) полная энергия
"Wp_fds"	"Wp_fds="	c,d,h,mN	Потребленная (прямая) активная энергия с начала работы данного

счетчика			
"Wp_bds"	"Wp_bds="	c,d,h,mN	Выделенная (обратная) активная энергия с начала работы данного счетчика
"Wp_ds"	"Wp_ds="	c,d,h,mN	Суммарная (разность прямой и обратной) активная энергия с начала работы данного счетчика
"Wq_fds"	"Wq_fds="	c,d,h,mN	Потребленная (прямая) реактивная энергия с начала работы данного счетчика
"Wq_bds"	"Wq_bds="	c,d,h,mN	Выделенная (обратная) реактивная энергия с начала работы данного счетчика
"Wq_ds"	"Wq_ds="	c,d,h,mN	Суммарная (разность прямой и обратной) реактивная энергия с начала работы данного счетчика
"Wp_fmb"	"Wp_fmb="	c,d,h,mN	Потребленная (прямая) активная энергия с начала месяца
"Wp_bmb"	"Wp_bmb="	c,d,h,mN	Выделенная (обратная) активная энергия с начала месяца
"Wp_mb"	"Wp_mb="	c,d,h,mN	Суммарная (разность прямой и обратной) активная энергия с начала месяца
"Wq_fmb"	"Wq_fmb="	c,d,h,mN	Потребленная (прямая) реактивная энергия с начала месяца
"Wq_bmb"	"Wq_bmb="	c,d,h,mN	Выделенная (обратная) реактивная энергия с начала месяца
"Wq_mb"	"Wq_mb="	c,d,h,mN	Суммарная (разность прямой и обратной) реактивная энергия с начала месяца
"Ws_mb"	"Ws_mb="	c,d,h,mN	Суммарная (разность прямой и обратной) полная энергия с начала месяца
"Two"	"Two="	d,h,mN	Время работы драйвера
"Ton"	"Ton="	d,h,mN	Время потребления энергии

Типы: **c** - немедленное значение;
d - архивное значение за день;
h - архивное значение за час;
mN - архивное значение (или их сумма) за N минут.

При запросе параметр "s-time" для устройства возвращается время драйвера.

3.3. Файл конфигурации

Файл `earport.conf` (имя может быть специфицировано в строке запуска драйвера) регулярно перепрочитывается (с интервалом 10 секунд).

Строка, относящаяся к устройству `dev` имеет вид:

```
dev [debug=ffff] [log=] [live_tout=60] [ok_tout=1] [err_tout=1]
[repeat_tout=1] [ip_init_tout=10] [recs=100000] [parole=0] [day_reps=0]
[hour_reps=0]
```

```
byte *cstr[]= {"quit","debug=","disc","log=","live_tout=",
"ok_tout=","repeat_tout=","err_tout=","ip_init_tout=","recs=","parole=",
"day_reps=","hour_reps="};
```

Указанные значения – умолчания.

Пример:

```
1 debug=18
```

dev – одно из устройств перечисленных в подстроке **DEVICES=**, или `ip_addr:port` из подстроки **IP=** (для всех устройств обслуживаемых драйвером);

debug - режим вывода отладочных сообщений (см. строку запуска);

log - файл журнала;

live_tout – таймаут (все таймауты в секундах) с последнего ответа устройства пока оно считается драйвером “живым”;

ok_tout – таймаут опроса “живого” устройства;

err_tout – таймаут опроса устройства, которое перестало отвечать;

repeat_tout – таймаут опроса “живого” устройства при неполученном последнем (одном) ответе;

ip_init_tout – таймаут повторных попыток переинициировать пропавшее ip-соединение с устройствами;

recs – максимальное количество записей журнала;

parole - 4-ёхбайтовый hex-пароль устройства;

Область действия **debug**, **log**, **recs** - весь драйвер, а не конкретное устройство.

4. Правила конфигурации.

Создание конфигурации осуществляется с помощью программы **STMConf** (C:\Igel\StmConf\StmConf.exe). В данном разделе будут описаны настройки, необходимые для взаимодействия сервера телеметрии и драйвера. Подробное описание работы с конфигуратором **StmConf** и создание конфигураций содержит документ «Конфигуратор сервера телеметрии StmConf. Руководство пользователя». Также дополнительную информацию можно найти в web-справочнике, нажав F1 в IgelView3, раздел Средства конфигурации > Конфигуратор сервера телеметрии StmConf.

4.1. Обработчики событий.

В конфигурации сервера телеметрии должен присутствовать обработчик событий «Модуль опроса». Подключение модуля на рис 2.

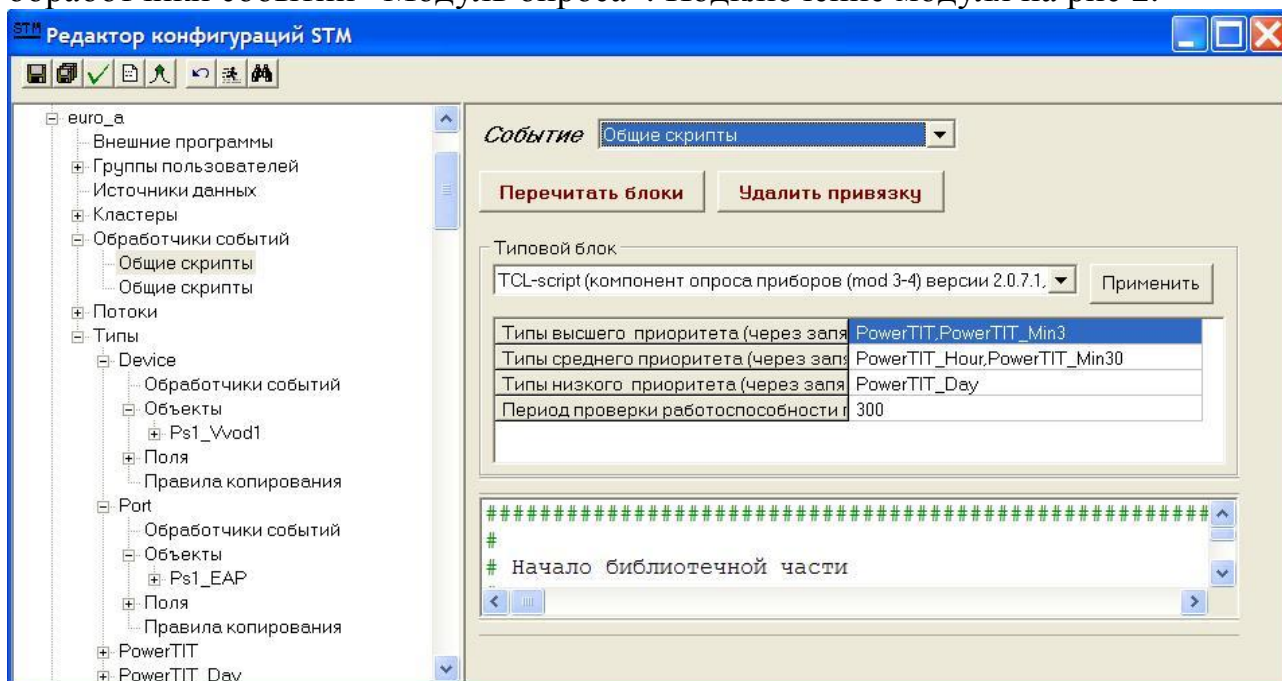


Рис. 2. Подключение «Модуля опроса»

4.2. Типы объектов и устройств.

В конфигурации должны присутствовать следующие типы

Port – Приборный порт, описывает режимы работы с конвертером ADAM 4579;

Device – Описание прибора;

TechTIT – значение измерения – текущие значения.

TechTIT_... – значение измерения – архивные значения.

На основании введенной информации в объектах типа **Port** и **Device** «модуль опроса» сформирует командные строки и запустит копии драйвера.

4.3. Структура типов.

Типы должны иметь следующую структуру полей:

Port

Тип поля	Наим. Поля	Комментарий	Пример значения
String	Driver	Командная строка запуска драйверного модуля	/stm/eaport LOG=130Ps25st1.log DEBUG=FF
Byte	Parallel	Признак параллельности	1
String	PortData	Характеристики порта	10.12.20.16:4001
Long	tOutRequest	таймаут отправляемый в запросе	15000
Long	tOutTest	таймаут тестовых запросов	80000
Byte	Version	Признак для модуля № 4	0

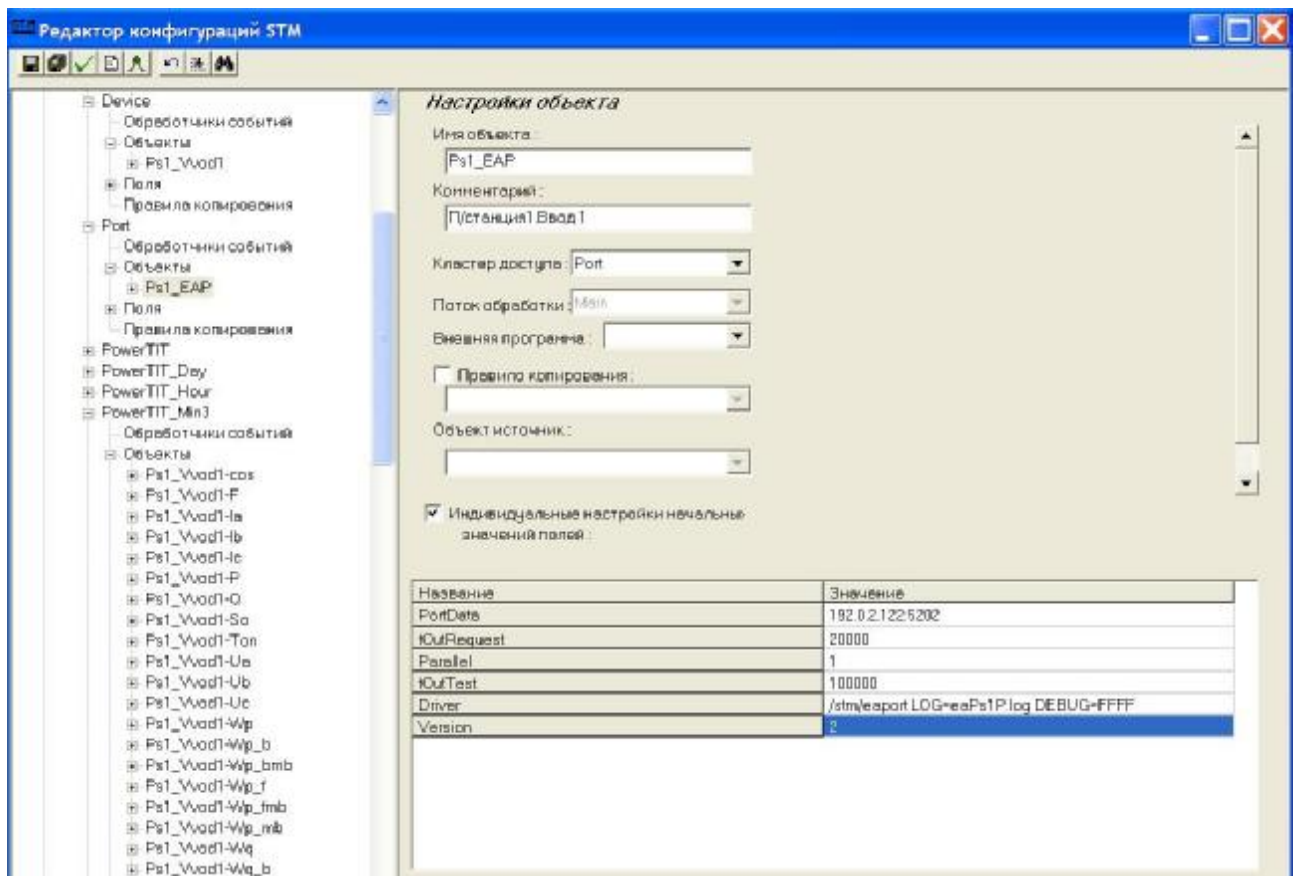


Рис. 3. Пример конфигурации драйвера (объект типа Port)

Device

Тип поля	Наим. Поля	Комментарий	Пример значения
Byte	Activate	1 - прибор в работе, 0 – прибор выведен из работы	1
Int	Address	Адрес устройства	15
Long	DeviceTimeDelt	Опережение времени устройства относительно времени сервера (в секундах)	0
Int	EndPeriod	Запрос по концу/началу	
Int	EndPeriodDB	Запись в базу по концу/началу	
Date	Pereopros	Переопрос устройства с дата:время	
uLong	Period	Период опроса текущих значений в мс	0
String	Port	Имя объекта типа порт, которому соответствует прибор	port1
Time	TimeActivate	время вывода прибора из работы	0

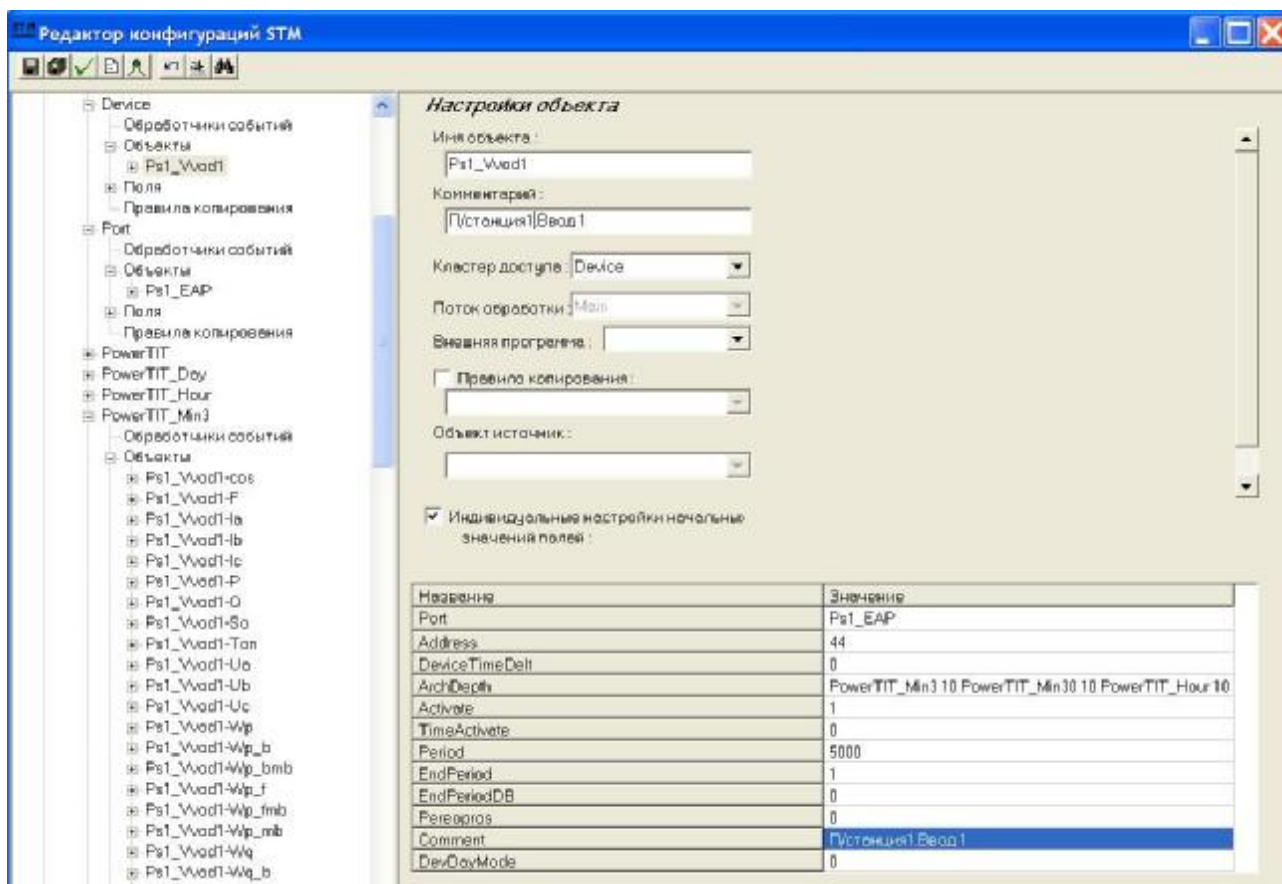


Рис. 4. Пример конфигурации устройства (объект типа Device)

PowerTIT (_Min3, _Min30, _Hour, _Day)

Тип поля	Наим. поля	Коммент.	Пример значения
Double	_avalue	значение телеизмерения от оборудования	
Double	_mvalue	значение телеизмерения от ручного ввода	
Byte	Arc	Номер параметра в приборе	
String	Calc	Формула пересчета	
String	Comment	Комментарий	
String	Device	Имя прибора, к которому привязан тит	
Byte	Man,	признак режима ручного ввода (0/1)	
Double	Value	значение телеизмерения	

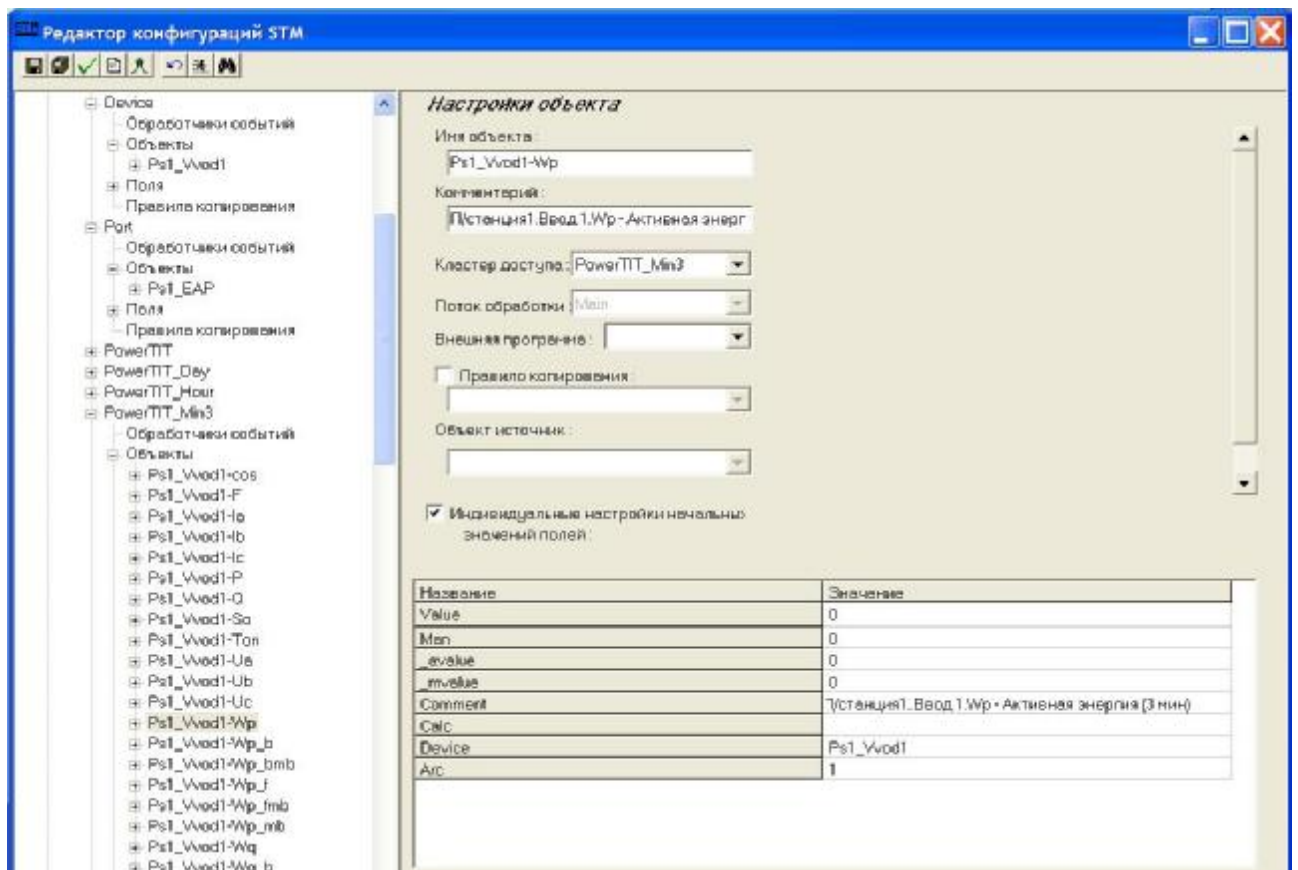


Рис. 5. Пример конфигурации объекта – Активная энергия за 3 минуты

5. Структура пакетов обмена между «Модулем опроса» и драйвером.

Структура пакетов обмена между «модулем опроса» и драйвером подробно описана в документе ««Модуль опроса». Руководство пользователя». Типы запросов, ответов и значения параметров запросов приводятся в таблице 1, 2 и 3.

Таблица 1. Возможные запросы от «модуля опроса» к драйверу

№	Пакет	Описание
0.	{num=N }\n	Пакет контроля работоспособности соединения
1.	{num=N type=c par=L dev=y tout=TO }\n	Запрос текущего уровня
2.	{num=N type=h par=L dev=y tout=TO time= TI }\n	Запрос архивного уровня: type=d – суточного, type=h – часового, type=m30 – получасового, type=m3 – трёхминутного.
3.	{num=N type=c par=s-time dev=y tout=TO }\n	Запрос текущего времени устройства у.

Таблица 2. Возможные ответы на запросы

№ за-прос а	Пакет	Описание
1.	{num=N }\n	Пакет контроля работоспособности соединения
2.	{num=N type=c dev=y sit=s L=значение }\n	Текущий уровень
3.	{num=N type=h dev=y sit=s time=TI L=значение }\n	Архивный уровень

где L может быть любым параметром из списка.

6. Протокол обмена драйвера с счётчиком «ЕвроАльфа».

Связь в Протоколе осуществляется по схеме главный/подчиненный. Под мастером (главным) понимается устройство (контроллер, устройство сбора информации, управляющий компьютер и т.д.), которое инициализирует действие канала связи, посылая сообщение подчиненному устройству. Под подчинённым (ЕвроАльфа) - пассивное устройство, которое только посылает ответ на сообщение от мастера.

Описание протокола открыто, не опубликовано, за более подробными разъяснениями следует обратиться к программисту (koff_i@mail.ru).

Сrc-процедура:

```
word get_crc(byte *p, word len) {
word crc, j;

for (crc=0; len--; ++p) {
((byte *)&crc)[1] ^= *p;
for (j=8; j--;)
if (crc & 0x8000) crc<<= 1, crc ^= 0x1021;
else crc<<= 1;
}
return swapw(crc);
}
```

Шифрование пароля по ключу:

```
dword psw_crc(dword key, dword psw) {
word j, k;
int i;

key+= 0xab41;
for (k=0, i= key + (key>>8) + (key>>16) + (key>>24) & 0xf; i-- >= 0; k=j)
j= key >= 0x80000000, key= (key<<1) + k, psw^= key;
return swapd(psw);
}
```

Диалоги:

1. Открытие сессии:

request: 02 18 06 00 01 address(1 byte) crc(2 bytes)

answer: 02 xx xx xx xx xx xx xx xx key(4 bytes) crc(2 bytes)

request: 02 18 01 00 04 psw_crc(4 byte) crc(2 bytes)

answer: 02 18 00 20 crc(2 bytes)

Примечание: ключ и пароль подаются в процедуру шифрования с байтами в обратном порядке (swarp).

2. Запрос текущих значений:

request: 02 1c 01 00 03 param(1 byte) phase(1 byte) 00 crc(2 bytes)

answer: 02 1c 00 a2 0d data(8 bytes) xx crc(2 bytes)

		ПАРАМЕТРЫ		СЕТИ			
1st byte	2nd byte:	0	2	4	7		
(param)	(phase)	1 ФАЗА	2 ФАЗА	3 ФАЗА	ОБЩЕЕ		
1	Частота	~ 50.00					
6	Напряжн	63.18	1.99	1.95	1st		
b	Углы фаз напряжения	0.00	330.00	330.00	byte		
5	Ток (Амп)	0.00	0.00	0.00			
c	Углы фаз тока	0.00	330.00	330.00			
18	Козф мощ *	1.00	1.00	1.00	11	1.00	
8	Козф мощнос (град)	0.00	0.00	0.00	12	0.00	
9	kW	0.00	0.00	0.00	f	0.00	
7	kvar	0.00	0.00	0.00	e	0.00	
4	kVA	0.00	0.00	0.00	10	0.00	

* "-" опережающий коэф мощ

3. Запрос хранимых данных:

request: 02 05 00 req_len(2 byte) req_offs(2 byte) class(1 byte) crc(2 bytes)

answer: 02 05 00 20 data_len(1 byte) data(data_len bytes) crc(2 bytes)

Примечание: старший бит data_len – признак того, является ли кадр последним для «класса» (80) или может быть запрошено продолжение (00).

4. Продолжение запроса:

request: 02 81 crc(2 bytes)

answer: 02 05 00 20 data_len(1 byte) data(data_len bytes) crc(2 bytes)

5. Закрытие сессии:

request: 02 85 crc(2 bytes)

no answer