

# Модуль подсистемы “Транспорты” <Serial>

Модуль:	Serial
Имя:	Последовательный интерфейс
Тип:	Транспорт
Источник:	tr_Serial.so
Версия:	0.7.3
Автор:	Роман Савоченко
Описание:	Предоставляет последовательный интерфейс. Используется для обмена данными через последовательные интерфейсы типа RS232, RS485, GSM и другое.
Лицензия:	GPL

## Оглавление

<a href="#">Модуль подсистемы “Транспорты” &lt;Serial&gt;</a> .....	1
<a href="#">Введение</a> .....	1
<a href="#">1. Входящие транспорты</a> .....	2
<a href="#">2. Исходящие транспорты</a> .....	4
<a href="#">3. Замечания</a> .....	6

## Введение

Модуль транспорта Serial предоставляет в систему поддержку транспортов, основанных на последовательных интерфейсах типа RS232, RS485, GSM и другие. Поддерживаются входящие и исходящие транспорты. Добавить новые входящие и исходящие интерфейсы можно посредством конфигурации транспортной подсистемы в любом конфигураторе системы OpenSCADA.

В режиме модема модулем поддерживается смешанный режим работы. Смешанный режим подразумевает наличие входящего транспорта, который ожидает внешних подключений, а также исходящего транспорта на том-же устройстве. Т.е. входящий транспорт будет игнорировать все запросы при наличии установленного исходящим транспортом соединения, в то-же время исходящий транспорт не будет осуществлять попыток установить соединение при наличии подключения к входящему транспорту или соединения другого исходящего транспорта например, с другим номером телефона.

Внимание! В обычном режиме последовательного интерфейса не допускается многократное использование одного и того-же порта во входящих и исходящих транспортах. Глобального блокирования последовательного устройства не осуществляется в виду неоднозначности этого процесса на системном уровне, а многократное использование может привести к непредсказуемым проблемам. При необходимости организации локального последовательного канала с парой связанных портов рекомендуется использование команды "\$ socat -d -d pty,raw,echo=0,perm=0666 pty,raw,echo=0,perm=0666".

# 1. Входящие транспорты

Сконфигурированный и запущенный входящий транспорт открывает порт последовательного интерфейса для ожидания запросов клиентов. Каждый входящий интерфейс обязательно связывается с одним из доступных транспортных протоколов, к которому передаются входящие сообщения.

Диалог конфигурации входящего последовательного интерфейса изображён на рис.1.

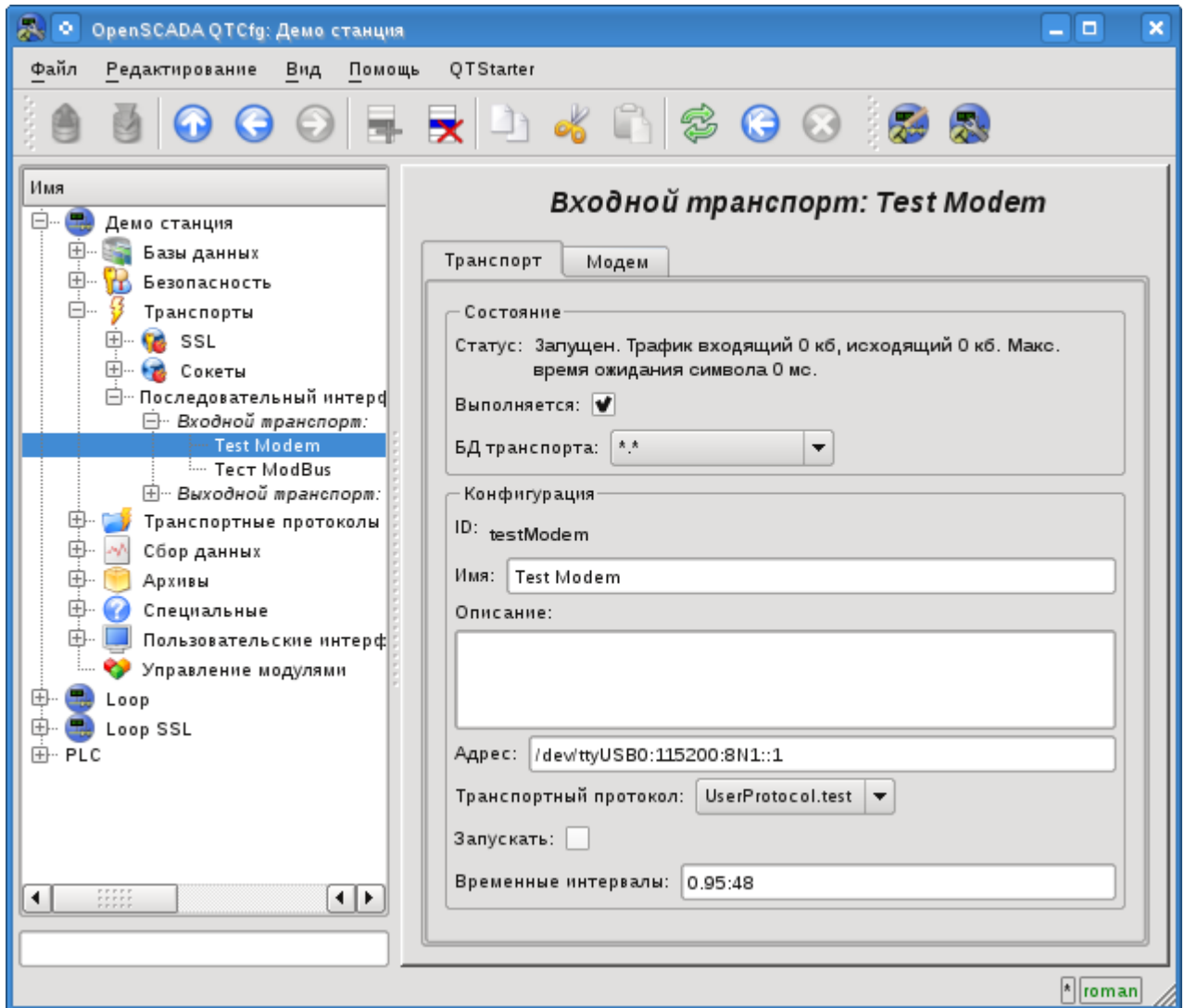


Рис.1. Диалог конфигурации входящего последовательного интерфейса.

С помощью этого диалога можно установить:

- Состояние транспорта, а именно: "Статус", "Выполняется" и имя БД, содержащей конфигурацию.
- Идентификатор, имя и описание транспорта.
- Адрес интерфейса в формате строки: "[dev]:[spd]:[format]:[fc]:[mdm]". Где:
  - *dev* - адрес последовательного устройства (/dev/ttyS0);
  - *spd* - скорость последовательного устройства из ряда: 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200, 230400, 460800, 500000, 576000 или 921600;
  - *format* - формат асинхронных данных "<размер><чётность><стоп>" (8N1, 7E1, 5O2, ...);
  - *fc* - управление потоком: 'h' - аппаратное (CRTSCTS), 's' - программное (IXON|IXOFF);
  - *mdm* - режим модема, ожидание 'RING'.
- Выбор транспортного протокола.
- Состояние, в которое переводить транспорт при загрузке: "Запущен".
- Временные интервалы интерфейса в формате строки: "[symbol]:[frm]". Где:

- *symbol* - время символа в миллисекундах. Используется для контроля факта окончания фрейма;
- *frm* - максимальное время фрейма в миллисекундах. Используется для ограничение максимального размера пакета запроса (фрейма).

Транспорт поддерживает возможность работы в режиме модема. Данный режим включается пятым параметром адреса и подразумевает ожидания звонка от удалённого модема (запрос "RING"), ответа на звонок (команда "ATA") и последующей передачи запросов от удалённой станции протоколу транспорта. Отключение сеанса связи осуществляется инициатором соединения и приводит к переключению модема приёмника на ожидание новых звонков.

Для настройки модема входящего транспорта предусмотрена специальная вкладка "Модем" (рис.2).

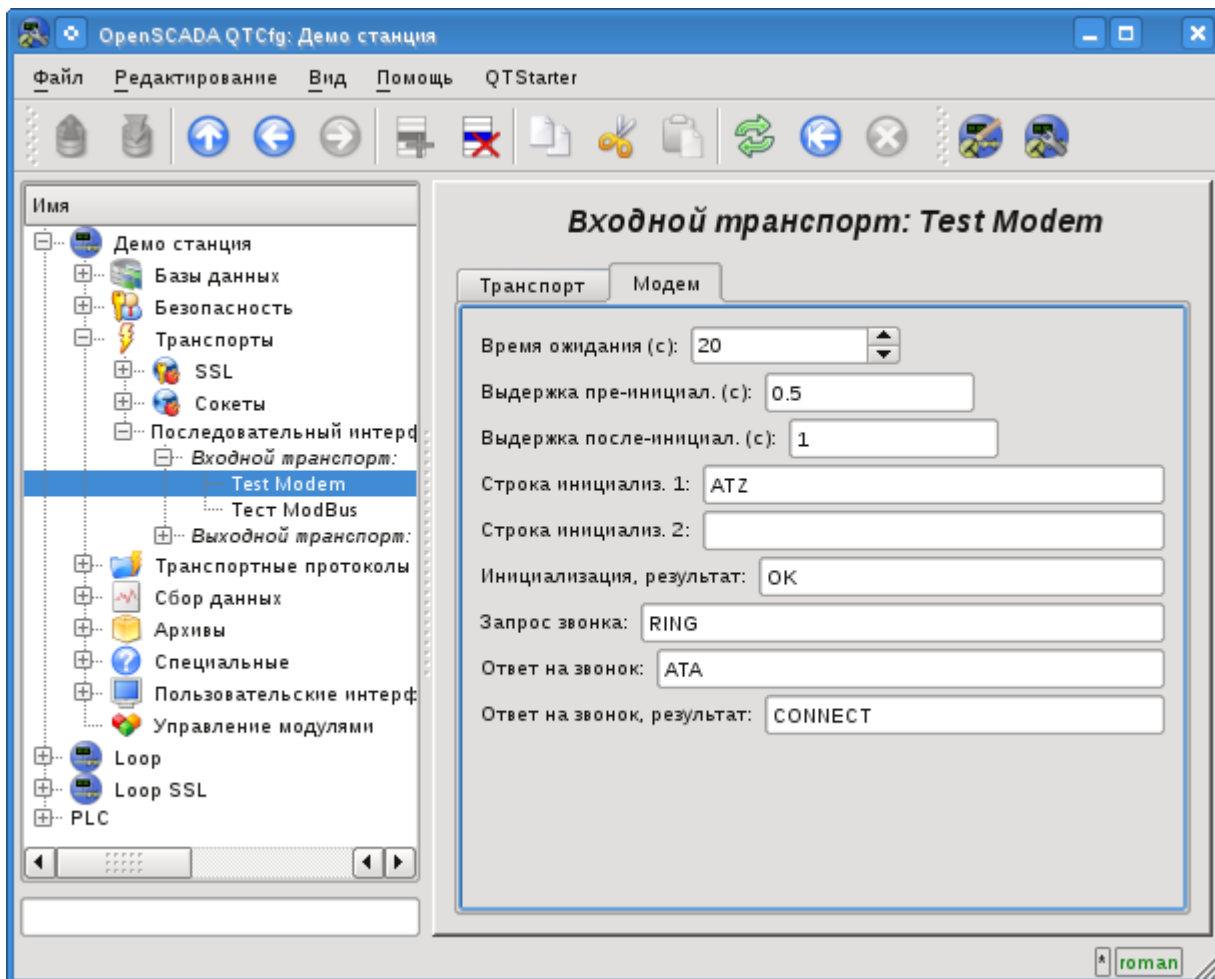


Рис.2. Вкладка "Модем" конфигурации модема входящего последовательного интерфейса.

С помощью этого диалога можно установить следующие свойства работы с модемом:

- Время ожидания, таймаут, модема на запросы, в секундах.
- Выдержка времени перед инициализацией модема, в секундах.
- Выдержка времени после инициализацией модема, в секундах.
- Первая строка инициализации, обычно содержит команду сброса настроек модема "ATZ".
- Вторая строка инициализации.
- Строка результата инициализации модема, обычно "OK", которой отвечает модем на инициализацию и которую нужно ожидать.
- Запрос звонка, обычно "RING", который шлёт модем в случае поступления исходящего вызова.
- Ответ на звонок, обычно "ATA", который отправляется модему для ответа на звонок.
- Строка результата на ответ на звонок, обычно "CONNECT", которой отвечает модем на команду ответа и которую нужно ожидать.

## 2. Исходящие транспорты

Сконфигурированный и запущенный исходящий транспорт открывает порт последовательного интерфейса для отправки запросов через него.

Главная вкладка страницы конфигурации исходящего последовательного интерфейса изображена на рис.3.

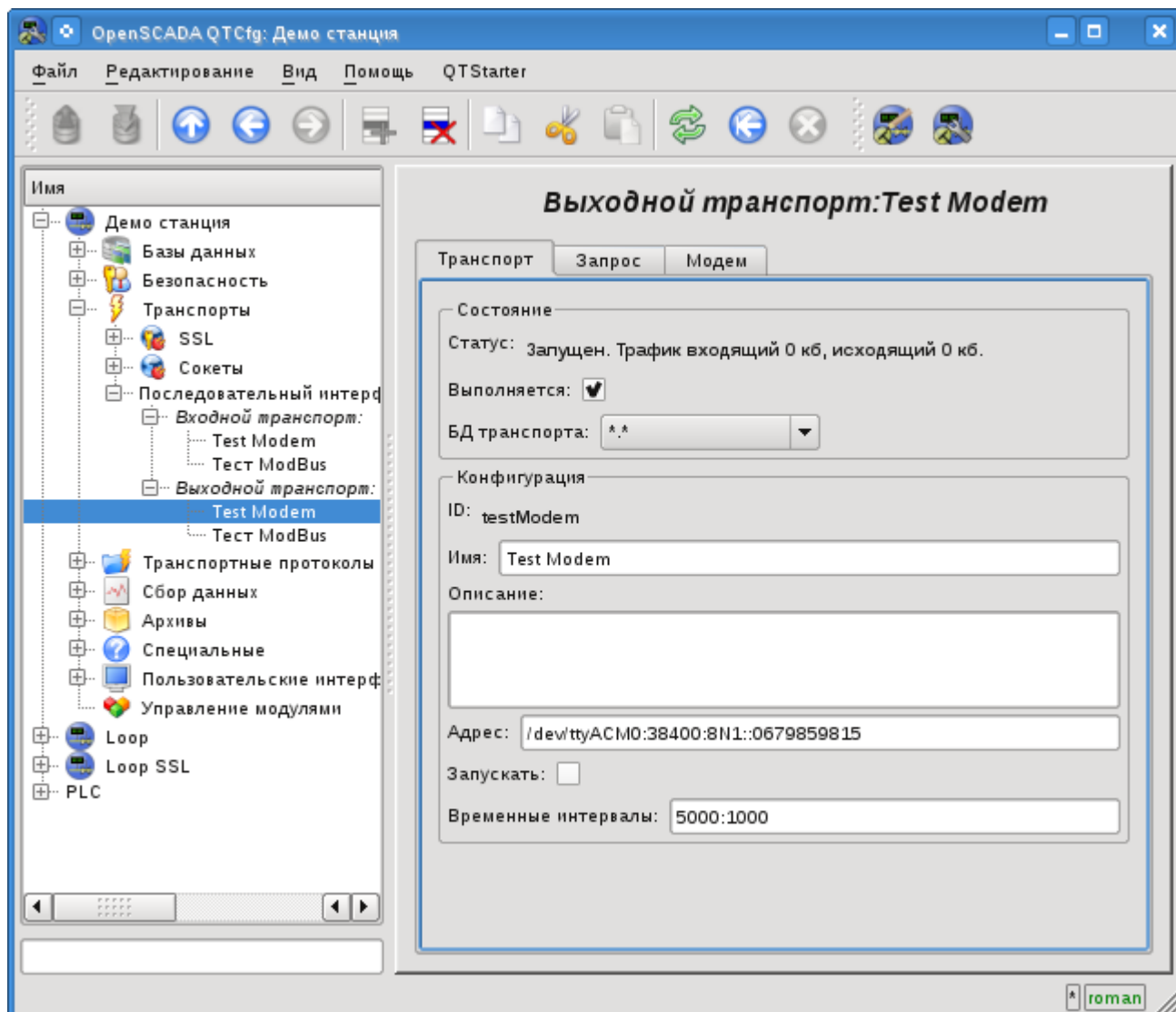


Рис.3. Главная вкладка страницы конфигурации исходящего последовательного интерфейса.

С помощью этого диалога можно установить:

- Состояние транспорта, а именно: "Статус", "Запущен" и имя БД, содержащей конфигурацию.
- Идентификатор, имя и описание транспорта.
- Адрес интерфейса в формате строки: "[dev]:[spd]:[format]:[fc]:[modTel]". Где:
  - *dev* - адрес последовательного устройства (/dev/ttyS0);
  - *spd* - скорость последовательного устройства из ряда: 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200, 230400, 460800, 500000, 576000 или 921600;
  - *format* - формат асинхронных данных "<размер><чётность><стоп>" (8N1, 7E1, 5O2, ...);
  - *fc* - управление потоком: 'h' - аппаратное (CRTSCTS), 's' - программное (IXON|IXOFF);
  - *modTel* - телефон модема, присутствие этого поля переключает транспорт на работу в режиме модема.
- Состояние, в которое переводить транспорт при загрузке: "Запущен".
- Временные интервалы интерфейса в формате строки: "[conn]:[symbol]". Где:

- *conn* - время ожидания соединения т.е. ответа от удалённого устройства.
- *symbol* - время символа в миллисекундах. Используется для контроля факта окончания фрейма.

Транспорт поддерживает возможность работы в режиме модема. Данный режим включается наличием пятого параметра адреса и подразумевает осуществление звонка по телефону, указанному пятым параметром, в момент запуска транспорта. После установки связи с удалённым модемом все запросы передачи направляются станции за удалённым модемом. Отключение сеанса связи, с остановкой транспорта, осуществляется по таймауту активности.

Для настройки модема исходящего транспорта предусмотрена специальная вкладка "Модем" (рис.4).

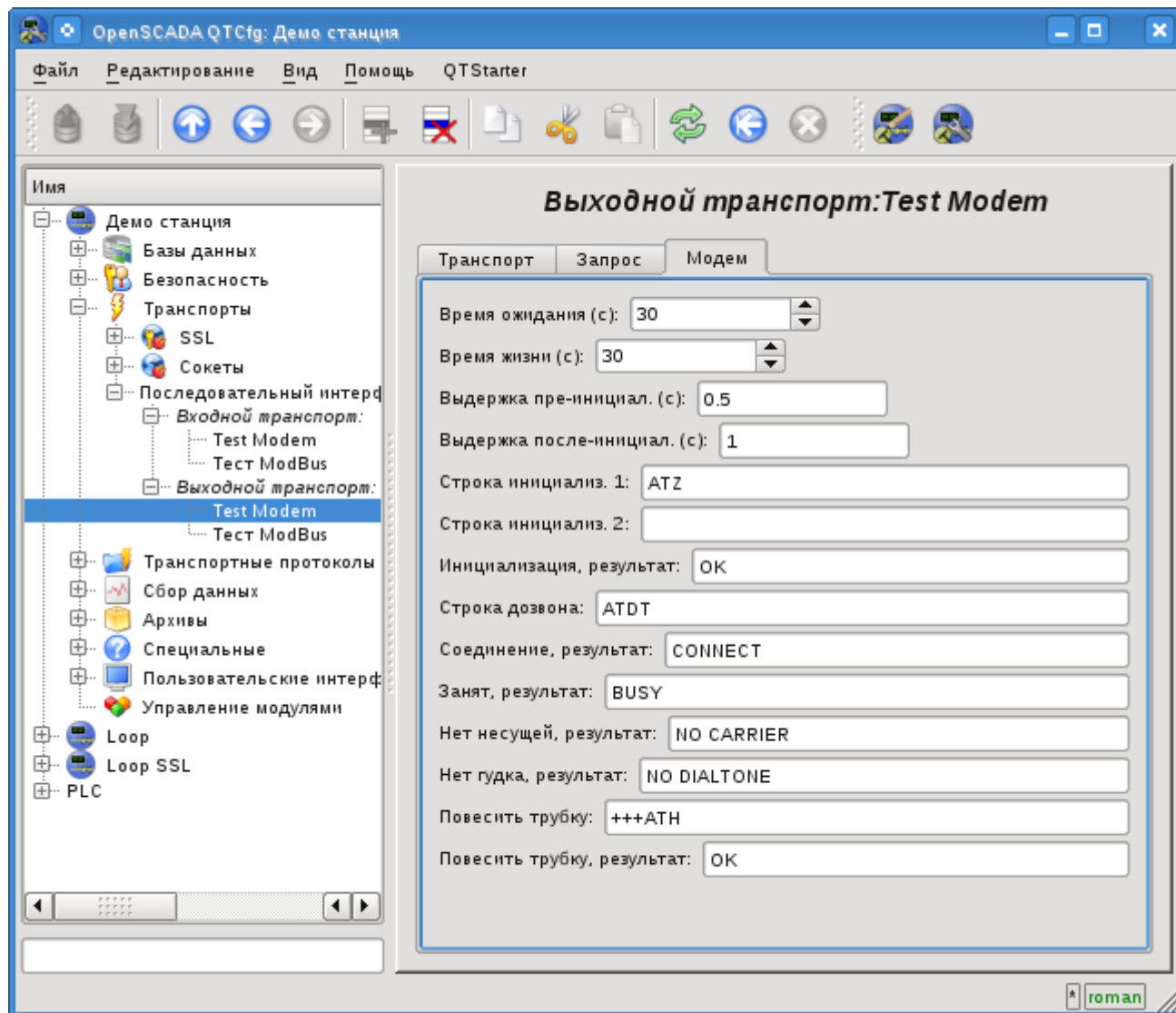


Рис.4. Вкладка "Модем" конфигурации модема исходящего последовательного интерфейса.

С помощью этого диалога можно установить следующие свойства работы с модемом:

- Время ожидания, таймаут, модема на запросы, в секундах.
- Время жизни соединения, в секундах. Если в течении этого времени будет отсутствовать передача данных через транспорт то соединение будет разорвано.
- Выдержка времени перед инициализацией модема, в секундах.
- Выдержка времени после инициализацией модема, в секундах.
- Первая строка инициализации, обычно содержит команду сброса настроек модема "ATZ".
- Вторая строка инициализации.
- Строка результата инициализации модема, обычно "OK", которой отвечает модем на инициализацию и которую нужно ожидать.
- Строка дозвона к удалённому модему, обычно "ATDT". При дозвоне номер телефона добавляется к данному префиксу.

- Строка результата удачного соединения, обычно "CONNECT".
- Строка результата занятости линии, обычно "BUSY".
- Строка результата отсутствия несущей в линии, обычно "NO CARRIER".
- Строка результата отсутствия гудка линии, обычно "NO DIALTONE".
- Команда повесить трубку, обычно "+++ATH". Данная команда вызывается всегда, когда нужно разорвать соединение.
- Строка результата команды повесить трубку, обычно "OK", которой отвечает модем на команду и которую нужно ожидать.

### 3. Замечания

Коммуникации через последовательные интерфейсы имеют ряд особенностей. Наиболее важной особенностью является критерий окончания сообщения и время ожидания этого критерия. В одних протоколах таким критерием является признак окончания или указанный размер сообщения. В других протоколах таким критерием является отсутствие данных во входящем потоке в течение указанного времени, время символа. В обоих случаях время ожидания критерия или символа является ключевым и сильно сказывается на общем времени обмена. Следовательно, чем меньше это время тем лучше. Вот тут и возникает проблема латентности оборудования и его драйверов.

Проверить латентность канала обмена и тем самым оптимально настроить время дождания, символа можно с помощью интерфейса вкладки "Запрос" исходящего транспорта. Для этого необходимо указать образцовый запрос соответствующего протокола, указать "Ожидать таймаут", отослать запрос и проконтролировать его целостность. Для получения более репрезентативного результата необходимо запрос повторить несколько раз. Если наблюдается получение неполных ответов, то время символа необходимо увеличить, иначе можно уменьшить.

На встроенном оборудовании последовательных интерфейсов RS232/422/485 можно добиться низкого уровня латентности, вплоть до единиц миллисекунд. Однако, на высоко-нагруженных системах с множеством задач с приоритетом реального времени латентность может стать недетерминированной в связи с исполнением потока обслуживания событий ядра Linux в низком приоритете. Для решения этой проблемы необходимо установить высокий приоритет этим потокам, что можно сделать с помощью скрипта, поместив его, например, в /etc/rc.local:

```
#!/bin/sh
# High priority set to kernel threads events for serial interfaces reaction rise
events=`ps -Ao pid,comm | sed -n '/[ ]*\([^ ]\)[ ]*events\[0-9\]/s//\1/p'`
for ie in $events; do
    chrt -pr 21 $ie
done
```

На внешнем оборудовании последовательных интерфейсов, например, в переходниках USB->RS232/422/485, может возникнуть проблема высокой латентности связанная с особенностью аппаратной реализации или его драйвера. Решать эту проблему можно путём изучения настроек этого оборудования или установкой большого времени ожидания, символа!