



**Мультипротокольные  
маршрутизаторы  
NSG  
Программное обеспечение NSG Linux**

**Руководство пользователя  
Часть 2  
Физические порты  
и службы канального уровня  
(Ethernet, PPP, Frame Relay)**

Версия программного обеспечения 1.0 build 1

Обновлено 14.09.2007

## АННОТАЦИЯ

Данный документ содержит руководство по настройке и применению мультипротокольных маршрутизаторов NSG, оснащенных программным обеспечением NSG Linux. Руководства по применению других продуктов NSG, а также базового программного обеспечения NSG для серий NPS-7e, NSG-500, NX-300 и NSG-800 содержатся в отдельных документах.

Документ состоит из следующих разделов:

- Часть 1. Общесистемная конфигурация
- Часть 2. Физические порты и службы канального уровня (Ethernet, PPP, Frame Relay)
- Часть 3. Маршрутизация и службы IP
- Часть 4. Туннелирование и виртуальные частные сети (VPN)
- Часть 5. Подсистема X.25

Вторая часть документа посвящена настройке физических портов, а также организации трафика средствами канального уровня: Ethernet bridging, VLAN, коммутации пакетов Frame Relay, доступу через асинхронные порты PPP и по PPP-over-Ethernet.

Общее описание системы, описание общесистемных параметров и командного языка системы приведены в Части 1. Настройка IP-маршрутизации и связанных с ней служб, а также механизмов управления IP-трафиком и обеспечения QoS, описана в Части 3. Часть 4 посвящена построению виртуальных частных сетей (VPN) на базе спецификации IPsec. В Части 5 рассмотрено использование устройств NSG в сетях X.25 и интеграция этих сетей с сетями IP.

**ВНИМАНИЕ** Продукция компании непрерывно совершенствуется, в связи с чем возможны изменения отдельных аппаратных и программных характеристик по сравнению с настоящим описанием. Сведения о последних изменениях приведены в файлах README.TXT, CHANGES, а также в документации на отдельные устройства.

Замечания и комментарии по документации NSG принимаются по адресу: [doc@nsg.net.ru](mailto:doc@nsg.net.ru).

© ООО «Эн-Эс-Джи» 2003–2007

ООО «Эн-Эс-Джи»  
Россия 105187 Москва  
ул. Кирпичная, д.39, офис 1302  
Тел.: (+7-495) 918-32-11  
Факс: (+7-495) 918-27-39

<http://www.nsg.ru/>  
<mailto:info@nsg.net.ru>  
<mailto:sales@nsg.net.ru>  
<mailto:support@nsg.net.ru>

## § СОДЕРЖАНИЕ §

### Часть 2. Физические порты и службы канального уровня (Ethernet, PPP, Frame Relay)

§2.1. Архитектура объектов физического и канального уровней.....	4
§2.2. Настройка физического и канального уровней.....	7
§2.2.1. Общие команды настройки физических портов.....	7
§2.2.2. Настройка синхронных портов — параметры физического уровня.....	8
§2.2.3. Настройка синхронных портов — протокольные параметры.....	10
§2.2.4. Настройка виртуальных каналов Frame Relay.....	11
§2.2.5. Настройка асинхронных портов.....	13
§2.2.6. Особенности использования проводных и сотовых модемных модулей.....	14
§2.2.7. Сценарии соединения для асинхронных портов.....	16
§2.2.8. Особенности использования консольного порта в устройствах NSG-700.....	18
§2.2.9. Особенности работы портов SLIP.....	18
§2.2.10. Настройка портов Ethernet.....	19
§2.2.11. Особенности использования модуля IM-SDSL в устройствах NSG-700.....	20
§2.2.12. Настройка портов USB.....	21
§2.3. Настройка физических интерфейсов E1.....	22
§2.3.1. Принципы настройки интерфейсов E1.....	22
§2.3.2. Терминология.....	22
§2.3.3. Общие параметры и настройка LIU.....	23
§2.3.4. Настройка цикловой структуры потока E1 (framer).....	23
§2.3.5. Формирование групп.....	24
§2.3.6. Коммутация E1.....	25
§2.4. Коммутация пакетов Ethernet.....	26
§2.4.1. Bridge Groups.....	26
§2.4.2. Коммутация VLAN.....	27
§2.4.3. Использование коммутатора Ethernet в устройствах NSG-700.....	28
§2.5. Коммутация пакетов Frame Relay и синхронного трафика.....	29
§2.5.1. Коммутация Frame Relay.....	29
§2.5.2. Коммутация синхронного трафика.....	29
§2.6. Сеансовый доступ по протоколам SLIP, PPP и его производным.....	30
§2.6.1. Шаблон IP-интерфейса.....	30
§2.6.2. Настройка сервера PPPoE.....	33
§2.6.3. Настройка клиента PPPoE.....	33
§2.6.4. Настройка клиента PPTP.....	35
§2.6.5. Настройка локального списка пользователей.....	35
§2.6.6. Настройка клиента RADIUS.....	35
§2.7. Просмотр состояния и статистики портов.....	36
Приложение 2–А. Режимы синхронизации последовательных интерфейсов.....	37
§2–А.1. Синхронные интерфейсы DCE/DTE.....	37
§2–А.2. Режим TTC.....	39
Приложение 2–Б. Примеры настройки PPP и PPPoE.....	41
§2–Б.1. Согласование IP-адресов.....	41
§2–Б.2. Аутентификация.....	43
§2–Б.3. Дополнительные возможности протокола PPP.....	45
§2–Б.4. Подключение клиентов по протоколу PPPoE.....	46
§2–Б.5. Подключение к серверам PPPoE.....	47

## §2.1. Архитектура объектов физического и канального уровней

Инкапсуляция и коммутация пакетов на физическом и канальном уровнях устройств NSG–900 осуществляется при помощи портов. В терминах программного обеспечения NSG Linux, *порт* представляет собой комплексный объект, который может включать драйвер физического интерфейса, обработчик протоколов канального уровня и интерфейс сетевого либо прикладного уровня для подключения к IP-маршрутизатору либо определенному приложению, соответственно. Эти компоненты автоматически создаются, связываются друг с другом и согласованно конфигурируются по мере настройки порта — в тех пределах, пока конфигурация определена однозначно и не требует дополнительных указаний от пользователя.

Порты могут иметь следующие системные имена:

- eth0 Встроенный порт Fast Ethernet.
  - s1, s2, ... Универсальные порты (разъемы расширения). Доступны только в случае, если в системе имеются разъемы расширения, для которых известен тип установленного интерфейсного модуля. Количество таких имен равно количеству сконфигурированных универсальных портов; порты, для которых тип интерфейсного модуля равен empty, недоступны для дальнейшего конфигурирования. Номера портов соответствуют номерам разъемов расширения, указанным на корпусе.
- ПРИМЕЧАНИЕ** Набор доступных команд и параметров конфигурации для портов sN зависит от типа установленного физического интерфейса и/или его настроек.
- a1, a2, ... Встроенные асинхронные порты. Количество таких имен равно количеству обнаруженных в устройстве асинхронных портов. Номера портов соответствуют номерам, указанным на корпусе.
  - x1, x2, ... Виртуальные порты для многоканальных физических интерфейсов E1 (в данной версии не поддерживаются). Каждому порту данного типа соответствует одна группа канальных интервалов (таймслотов) в потоке E1.
  - t1, t2, ... Виртуальные синхронные порты, представляющие собой окончания туннелей HDLC-over-IP (GRE).

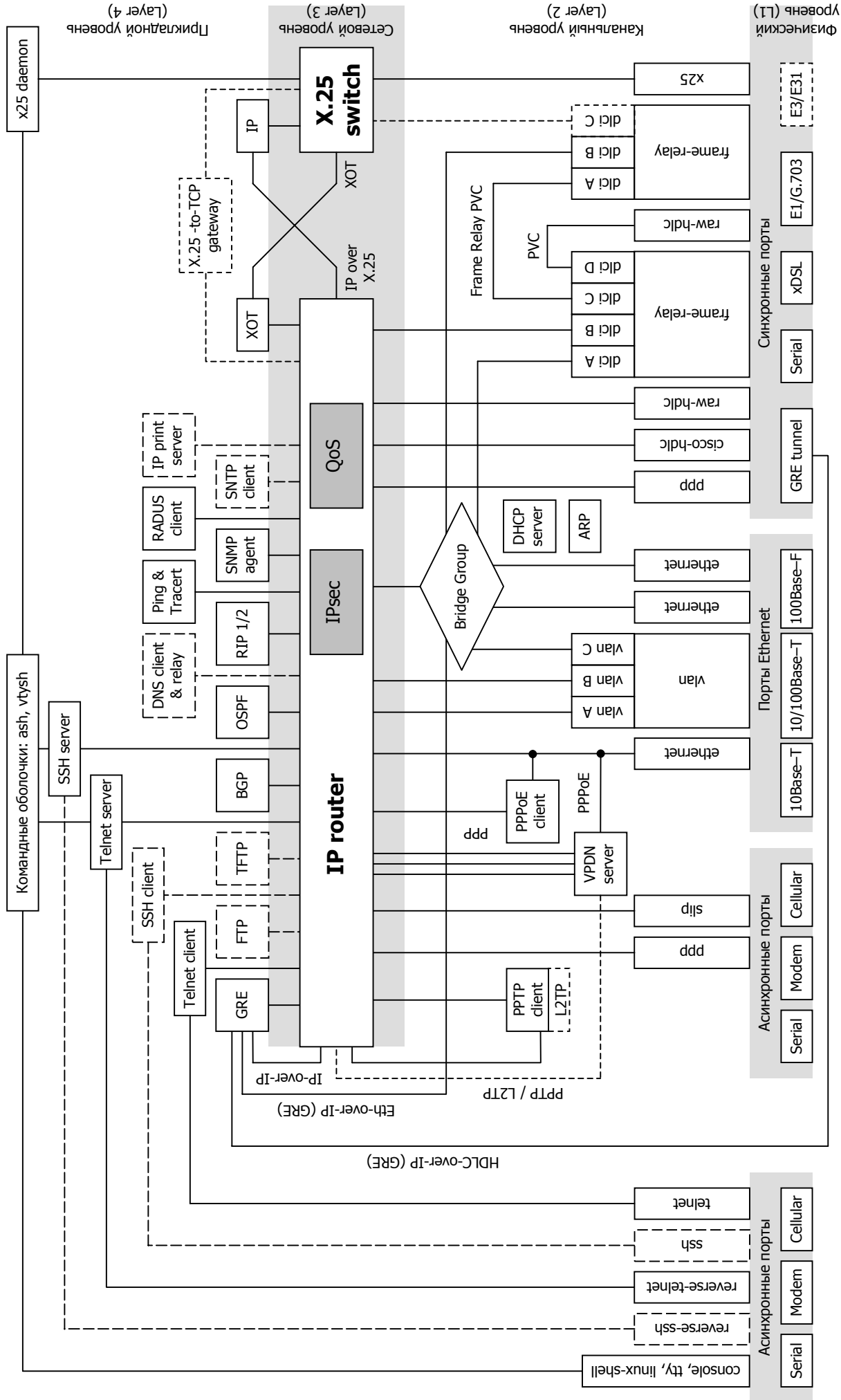
С точки зрения протоколов канального уровня, каждый порт относится к одной из трех категорий:

- Порты Ethernet — для подключения к локальным сетям Ethernet.
- Синхронные порты — для подключения к синхронным каналам WAN "точка-точка".
- Асинхронные порты — для подключения к асинхронным соединениям WAN "точка-точка".

Каждой из этих категорий соответствуют также определенные типы физической среды и режим передачи:

Категория портов	Протоколы канального уровня (encapsulation)	Имена и конфигурация	Физические и виртуальные интерфейсы
Ethernet	ethernet (IEEE 802.3) vlan (IEEE 802.1Q)	eth0	Встроенный Fast Ethernet
		sN при card sN {im-et10   um-et100}	Сменные Ethernet 10Base-T
Асинхронные	ppp slip tty — подключение к основной командной оболочке linux-shell — подключение к командной оболочке Linux console — подключение к одной из двух командных оболочек в зависимости от имени пользователя reverse-telnet — подключение к локальному серверу Telnet telnet — подключение посредством клиента Telnet к удаленному серверу	aN	Встроенные асинхронные
		sN	Сменные RS-485
		sN при card sN { im-v24   im-v35 } port sN physical-layer async либо card sN uim-cdma	Сменные V.24 и V.24/V.35, сотовые модемы, модемы ТФОП
Синхронные	cisco-hdlc frame relay ppp x25 raw-hdlc	xN	Группы на многоканальных интерфейсах E1
		tN	Туннели HDLC-over-IP (GRE)
		sN при card sN { im-e1   im-2e1 }	Группа 1 или режим unframed на одноканальных интерфейсах E1
		sN, кроме вышеприведенных исключений	Сменные V.24, V.24/V.35, RS-530, X.21, xDSL, G.703, G.703.1

Более подробно эти взаимосвязи будут рассмотрены ниже.



Иерархическая архитектура протоколов в NSG Linux (компоненты, показанные длинным пунктиром, доступны только средствами командной оболочки Linux, коротким пунктиром — находятся в разработке)

**ВНИМАНИЕ** Категория порта однозначно связана с его символическим именем только для фиксированных портов (Ethernet или асинхронных). Большинство портов в устройствах NSG являются *универсальными*, т.е. представляют собой разъемы расширения, которые оборудуются сменными интерфейсными модулями. Такие порты могут относиться к любой из трех категорий в зависимости от типа и конфигурации интерфейсного модуля.

На вершине иерархической структуры порта находится системный *интерфейс*, через который непосредственно происходит обмен данными с объектами вышестоящего уровня (IP-маршрутизатором, прикладными программами) или с другими портами. В простейшем случае порт представляет собой полностью *одноканальный* объект, т.е. имеет внутри себя только один канал данных, на всех уровнях сетевой иерархии. Примерами таких портов являются синхронные порты с инкапсуляцией PPP, Cisco-HDLC, с прозрачным протоколом (кроме портов с физическими интерфейсами E1), Ethernet, все типы асинхронных портов.

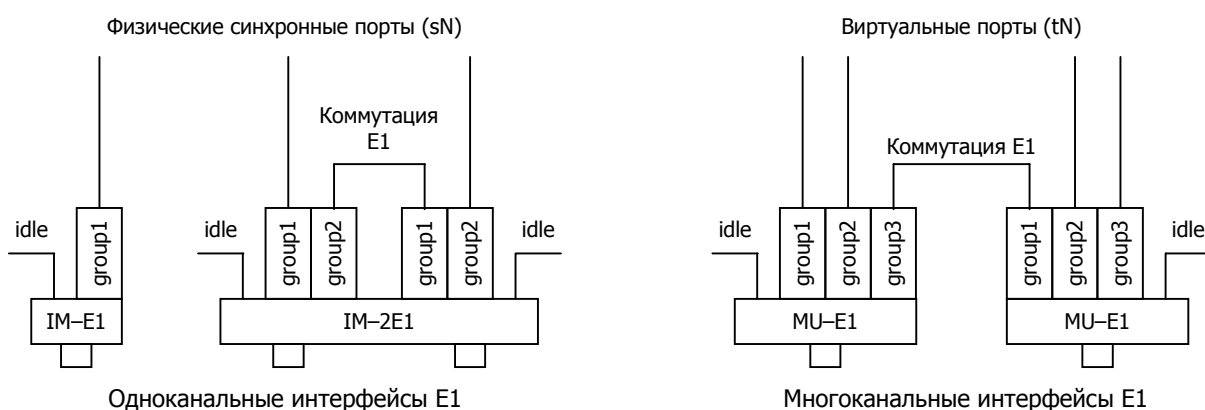
Порты с инкапсуляцией Frame Relay, X.25 и VLAN являются *многоканальными*. Эти протоколы предусматривают эмуляцию нескольких виртуальных сред передачи (каналов или локальных сетей) на канальном уровне. Для каждого из таких портов создаются дочерние объекты — DLCI, логические каналы X.25 или VLAN, соответственно — имеющие уникальные номера в пределах данного порта. Объектам DLCI и VLAN соответствуют свои *суб-интерфейсы*, через которые осуществляется обмен данными; сам порт при этом лишь обрабатывает процедуры протокола.

В частности, для обработки пакетов IP необходимо назначить соответствующему интерфейсу или суб-интерфейсу так называемый *IP-префикс* (совокупность IP-адреса и длины маски, например, 10.0.0.1/8). Интерфейс, которому не присвоен никакой IP-префикс, может:

- участвовать в коммутации на канальном уровне (*Ethernet bridging, Frame Relay switching, raw sync switching*)
- использоваться для мультипротокольной инкапсуляции трафика (например, PPP-over-Ethernet)
- автоматически подключаться к одному из прикладных процессов (командная оболочка, Telnet-сервер или клиент).

**ПРИМЕЧАНИЕ** Фактически при назначении IP-префикса происходит создание IP-интерфейса, который связывает порт (или его дочерний объект) с IP-маршрутизатором. IP-интерфейс относится к третьему (сетевому) уровню протокольной иерархии.

Физические интерфейсы E1, работающие в структурированном (*framed*) режиме G.704, также являются многоканальными, но эта многоканальность реализована на первом уровне сетевой иерархии. Такие интерфейсы выделяют из потока E1 одну или несколько *групп* канальных интервалов (таймслотов) для передачи данных. В зависимости от типа интерфейса E1 (одно- или многоканальный, соответственно), выбранные группы служат в качестве физической среды передачи либо для порта sN, либо для виртуальных портов tN, присоединенных к данному физическому интерфейсу sN. Кроме того, группы могут коммутироваться между собой на физическом уровне, минуя протокольную обработку на канальном и вышестоящих уровнях. Организация групп и их коммутация с портами или друг с другом — дополнительные задачи, которые являются специфическими для настройки интерфейсов E1.



Специфическую задачу представляет мультипротокольная инкапсуляция трафика. В программном обеспечении NSG Linux она реализована следующим образом: порт, обслуживающий "вложенный" протокол, использует в качестве физического уровня порт с "внешним" протоколом. Например, при организации виртуальной сети коммутируемого доступа (Virtual Private Dial-Up Networks, VPDN) на основе протокола PPP-over-Ethernet (PPPoE) в системе автоматически создаются виртуальные порты PPP, транспортом для которых служат не физические интерфейсы, а порт Ethernet.

## §2.2. Настройка физического и канального уровней

### §2.2.1. Общие команды настройки физических портов

Для настройки физического порта необходимо войти в меню (config-nsg)# и ввести следующую команду

```
port { eth0 | s1 | s2 | ... | a1 | a2 | ... }
```

Вход в меню конфигурации порта.

Доступные имена портов зависят от того, какие фиксированные интерфейсы и разъемы расширения были ранее обнаружены в системе или указаны вручную командами chassis, card (см. Часть 1). После ввода вышеуказанной команды пользователь попадает в меню конфигурирования выбранного порта, например:

```
(config-nsg)# port s1
(config-port-s1)#
```

Набор команд и параметров, доступных для каждого порта, зависит от типа порта и его физического интерфейса. Общими для всех типов портов являются следующие команды:

```
adm-state { up | down }
```

Установка административного статуса порта. По умолчанию, все порты находятся в состоянии up (при условии, что для них известен тип физического интерфейса).

```
display { configuration | all | path | version }
```

Просмотр параметров данного объекта меню, пути к нему и версии NSG Linux, соответственно.

```
show
```

Просмотр текущего состояния и статистики работы данного физического или виртуального порта. Подробно о данной команде см. п.2.7.

```
description
```

Административное описание данного порта — текстовая строка длиной до 255 знаков. Если строка содержит пробелы, ее необходимо заключить в кавычки.

Еще несколько команд используются для всех или нескольких типов портов, однако имеют различные параметры для различных типов портов и физических интерфейсов:

```
baudrate
```

Установка скорости для данного порта.

```
encapsulation
```

Установка типа инкапсуляции для данного порта.

```
mode
```

Установка режима работы для данного порта.

```
ip
```

Установка IP-префикса (адреса и маски) для данного интерфейса или субинтерфейса.

Подробно эти команды и команды, специфические для отдельных типов портов, описаны в следующих трех параграфах.

Отдельные типы интерфейсных модулей на шасси NSG-800, NSG-900 могут работать в нескольких существенно различных режимах, а именно:

— IM-V24, IM-V35, IM-V35-2 — в синхронном или в асинхронном режиме

— IM-SDSL *h/w ver.2* — в синхронном режиме или в режиме "удаленного порта" Ethernet-over-HDLC (совместимом с устройствами NSG-50 SDSL bridge на противоположной стороне линии SDSL).

Для этих модулей предусмотрена дополнительная команда:

```
physical-layer { sync | async | ether }
```

Выбор режима для данного порта. (По умолчанию sync). В зависимости от этого выбора изменяются остальные команды меню (config-port-sN)# и их параметры.

**ПРИМЕЧАНИЕ** Интерфейсные модули и карты IM-V35, IM-V35-2, UM-SI, MV-HSI поддерживают два или более протоколов физического уровня (V.24/RS-232, V.35 и др.). Выбор протокола осуществляется аппаратно при помощи соответствующих кабелей NSG.

## §2.2.2. Настройка синхронных портов — параметры физического уровня

В синхронном режиме могут использоваться следующие типы физических интерфейсов:

- Интерфейсы V.24, V.35 с установленным параметром `physical-layer sync`
- Интерфейсы V.36/RS-449, RS-530, RS-530A, X.21 и универсальные последовательные интерфейсы
- Интерфейсы SHDSL-bis, SHDSL, DSL/xxx, IDSL, SRM
- Интерфейсы SDSL с установленным параметром `physical-layer sync`
- Интерфейсы E12 (G.703), E0 (G.703.1)
- Интерфейсы E1, работающие в неструктурированном режиме (`framing none`)
- Сконфигурированные группы таймслотов для передачи данных (`data-group N ...`) на интерфейсах Fractional E1 и Channelized E1, работающих в структурированном режиме.

Для интерфейсных модулей IM-CE1-х каждая группа данных должна быть скоммутирована с некоторым виртуальным портом tN. Для всех остальных типов модулей порты имеют символические имена вида sN. Для настройки синхронных портов WAN используются команды, приведенные ниже.

**ПРИМЕЧАНИЕ** Интерфейсные модули IM-CE1-х в данной версии NSG Linux не поддерживаются. Интерфейсные модули IM-хSHDSL/bis поддерживаются только в неструктурированном режиме.

Параметры настройки порта по смыслу разделяются на две функциональные группы. Первая группа параметров определяет настройки физического уровня:

`baudrate <бит/с>`

Скорость работы порта. Набор допустимых скоростей зависит от типа интерфейсного модуля, назначенного для данного порта. Для просмотра списка скоростей введите "`baudrate ?`".

`mode { external | internal | ttc }` (для последовательных модулей DCE/DTE)  
`mode { coe | cpe }` (для модулей xDSL; значение COE эквивалентно MASTER, CPE — SLAVE)  
`mode { coe | cpe | rcpe | rcpe }` (для модулей SHDSL-bis)  
`mode { local | loop }` (для модулей G.703.х цифровой плезиохронной иерархии)

Режим синхронизации порта. Набор допустимых значений параметра зависит от типа интерфейсного модуля. Для двух физических интерфейсов, непосредственно соединенных друг с другом, всегда должны быть установлены противоположные режимы синхронизации.

Для физических интерфейсов, допускающих работу только в каком-то единственном режиме (UM-HDLC, MU-D128, MU-D64N, MU-IDSL, MU-E0) данный параметр недоступен.

Значение `mode ttc` соответствует работе последовательного интерфейса в режиме TTC в качестве DCE и поддерживается только для интерфейсного модуля IM-V35-2 обр.2003 г. Для двух последовательных интерфейсов DTE/DCE устанавливаются режимы: для одного — DCE (`mode internal` или `mode ttc`), для другого — DTE (`mode external`). Подробно о режимах синхронизации для последовательных интерфейсов DTE/DCE см. Приложение 2-А.

В обоих режимах DCE значение скорости является действующим. В режиме DTE (как обычном, так и TTC) значение скорости является справочным. Значения скоростей выше 2048000 бит/с являются округленными. Устойчивая работа на этих скоростях гарантируется только в режиме TTC (в качестве DTE и/или DCE, в зависимости от того, что из них поддерживается данным модулем).

Для двух интерфейсов xDSL устанавливаются режимы: для одного — `coe`, для другого `cpe`. Источником синхронизации в этой паре является устройство Central Office Equipment (COE, или MASTER). Значение скорости имеет следующий смысл:

- Для модулей IM-DSL/xxx — значение справочное, реальная скорость однозначно определена типом модуля.
- Для модулей IM-SRM, IM-SRM-2, IM-IDSL — значение справочное, реальная скорость устанавливается переключками на модуле. (Для портов IDSL устройства NSG-800/maxU — программированием TDM-коммутатора.)
- Для модулей IM-SDSL, IM-SHDSL, IM-хSHDSL/bis — значение действующее.

Во всех случаях фактическое значение скорости, как бы оно ни устанавливалось, должно быть одинаковым на обеих сторонах.

**ПРИМЕЧАНИЕ** Для интерфейсов SDSL выбор способа интерпретации линейного кода 2B1Q в данной реализации NSG Linux не предусмотрен. Всегда используется нормальная последовательность (сначала знак, затем амплитуда).

Для интерфейсов SHDSL-bis возможны два дополнительных специфических режима:

`rcoe`        Режим *coe with remote startup*  
`rcpe`        Режим *cpe with coe startup*

Данные режимы также используются парой на противоположных концах линии и устанавливаются, что пилотный сигнал для начала установления соединения генерируется устройством COE; устройство CPE находится в пассивном режиме, ожидая этого сигнала. (В обычном режиме COE/CPE соединение инициируется устройством CPE, которое показывает таким сигналом, что оно подключено к линии.)



Для неструктурированных интерфейсов G.703.x скорость однозначно определяется типом модуля. Параметр `mode` имеет следующий смысл:

- Для модуля IM-703/64xxx, работающего как сонаправленный — значение действующее.
- Для модуля IM-703/64CND, сконфигурированного переключками как противонаправленный — значение справочное. Передатчик и приемник модуля синхронизируются по отдельным витым парам от генератора на удаленной стороне.
- Для модулей IM-703 и IM-703-2 — значение справочное. Фактический режим синхронизации устанавливается переключкой на модуле.

При подключении к сети оператора связи источником синхронизации, как правило, является оборудование оператора, поэтому на устройстве NSG следует установить `mode loop`. При непосредственном соединении двух устройств физическим медным кабелем устанавливаются режимы: для одного — `local`, для другого — `loop`.

Для структурированных интерфейсов E1 выбор режима синхронизации производится в меню `controller sn`. Там же производится формирование групп данных, причем скорость для группы определяется числом канальных интервалов в ней. Поэтому в меню порта пункт `mode` отсутствует, а значение скорости является справочным.

`shdsl-annex { a | b }`

Только для модуля IM-SHDSL: спектральное распределение плотности сигнала в линии

- a Спектральная плотность согласно ITU-T G.991.2 Annex A
- b Спектральная плотность согласно ITU-T G.991.2 Annex B (по умолчанию)

`clkmode { 1 | 3a }`

Только для модулей IM-xSHDSL/bis: параметр, описывающий архитектуру линейного окончания SHDSL в плане синхронизации в соответствии с документом ITU-T G.991. По умолчанию установлено значение `clkmode 1`; изменение данного параметра требуется лишь в исключительных случаях.

Параметр должен быть установлен одинаковым на обеих сторонах линии.

`annex { a | b | auto_ab | g | speed-dependent }`

Только для модулей IM-xSHDSL/bis: линейный код и спектральная плотность сигнала SHDSL.

- a TC-PAM16, спектральная плотность согласно G.991 Annex A (США/Канада)
- b TC-PAM16, спектральная плотность согласно G.991 Annex B (Европа)
- auto\_ab TC-PAM16, автоматический выбор между Annex A и Annex B
- g TC-PAM32, спектральная плотность согласно G.991 Annex G (для работы со скоростями выше 2304000 бит/с)

`speed-dependent` Автоматический выбор между режимами `auto_ab` и `g` в зависимости от скорости в линии (для IM-2SHDSL/bis — от скорости в одиночной линии, т.е. с учетом параметра `wiring`).

По умолчанию установлен режим `speed-dependent`.

`wiring { p1 | p2 | p1mp2b | p2mp1b | 2-pair | 4-wire }`

Только для модуля IM-2SHDSL/bis: режим совместной работы портов данного модуля.

Одноканальные режимы (диапазон скоростей 192000...5696000 бит/с):

- p1 Используется только порт P1 модуля.
- p2 Используется только порт P2 модуля.

Режимы работы с горячим резервированием по схеме 1+1:

- p1mp2b Порт P1 является основным (*main*), порт P2 — резервным (*backup*).
- p2mp1b Порт P2 является основным (*main*), порт P1 — резервным (*backup*).

Диапазон скоростей 192000...5696000 бит/с; скорости, `annex` и `clkmode` одинаковы для обоих портов.

При нормальной работе основного порта весь трафик идет через него, независимо от состояния резервного. В случае разрыва соединения весь трафик направляется в резервный порт, если тот активен (т.е. имеет место физическое соединение).

При восстановлении физического соединения на основном порту весь трафик безусловно возвращается в него.

Многоканальные режимы:

- 2-pair Режим работы по двум физическим парам согласно G.shdsl.bis с использованием чередования байтов (*byte interleaving*). Оба порта должны быть активны. При отказе одного из физических соединений передача данных становится невозможна. Диапазон скоростей 384000...11392000 бит/с.

- 4-wire Режим работы по двум физическим парам согласно G.shdsl. Аналогичен 2-pair, но введен для совместимости с оборудованием, не поддерживающим G.shdsl.bis. Диапазон скоростей 384000...4608000 бит/с.

По умолчанию для данного модуля установлен режим `p1`.

**ПРИМЕЧАНИЕ** Совместимость с модулем IM-SHDSL в данной версии программного обеспечения не реализована.

`tx-ring-limit <1...8>`

Для всех типов синхронных модулей: количество буферов на передачу. Данный параметр позволяет улучшить работу механизмов QoS (подробнее см. Часть 3). Значение по умолчанию — 8.

### §2.2.3. Настройка синхронных портов — протокольные параметры

Вторая группа команд в меню портов описывают параметры протоколов второго и третьего уровней. Эти команды относятся к настройке как физических синхронных портов, так и виртуальных портов, представляющих собой окончания туннелей HDLC-over-GRE. (Подробнее о туннелировании см. Часть 4.)

Общей и ключевой для всех типов синхронных портов является следующая команда:

```
encapsulation { cisco-hdlc | frame-relay | ppp | x25 | raw-hdlc }
```

Тип протокола канального уровня.

Порты с инкапсуляцией `raw-hdlc` предназначены для прозрачной передачи синхронного трафика, представленного в виде кадров формата HDLC общего вида (к таковым, в частности, относятся кадры Cisco-HDLC, PPP, Frame Relay, X.25). Этот формат подразумевает:

- флаг — 01111110 в двоичном представлении;
- контрольная последовательность кадра — ITU-T FCS-16;
- порядок приема/передачи байтов данных — младшим битом вперед;
- прием/передача данных — без побитной инверсии (NRZ).

Длина кадра HDLC не должна превышать 1600 байт (это ограничение обусловлено особенностями реализации устройств NSG).

Такие порты могут коммутироваться (с помощью PVC) друг на друга или на виртуальные каналы Frame Relay, а также использоваться для передачи IP-трафика. Остальные типы портов не требуют комментариев.

Если для порта задана инкапсуляция `cisco-hdlc`, `ppp` или `raw-hdlc`, то в системе появляются суб-интерфейсы с именами вида `sN.0`. Примеры: `s1.0`, `s2.0`, и т.д. Если для порта задана инкапсуляция `frame-relay`, то при создании каждого виртуального канала (DLC) в системе появляется суб-интерфейс с именами вида `sN.<dlci>`, где `sN` — имя физического порта, `<dlci>` — номер виртуального канала (DLCI). Примеры: `s2.16`, `s2.100`, и т.д. Именно эти объекты используются для маршрутизации и коммутации пакетов.

Дальнейшая настройка порта зависит от выбранной инкапсуляции.

#### а) Команды для портов с инкапсуляцией Cisco-HDLC, PPP и RAW:

```
ip address <ip-префикс> [peer <ip-адрес>] [broadcast <ip-адрес>] [anycast <ip-адрес>]
```

```
no ip address <ip-префикс>
```

Установка и удаление параметров IP-интерфейса. Обязательным параметром данной команды является IP-префикс (адрес и длина маски, например, 123.134.145.156/24). При необходимости можно отдельно назначить адреса удаленной стороны (`peer`) в соединении "точка-точка", широковещательной (`broadcast`) и групповой (`anycast`) рассылки.

**Пример.** Настроить порт 1 для работы с внешней синхронизацией, интерфейс V.35, протокол Cisco-HDLC.

```
card s1 im-v35
port s1 ip address 192.168.1.1/24
```

При этом следующие значения будут установлены по умолчанию:

```
adm-state up
physical-layer sync
mode external
baudrate 64000
encapsulation cisco-hdlc
cisco-hdlc keepalive 10
cisco-hdlc timeout 25
ip address 192.168.1.1/24 peer 0.0.0.0 broadcast 0.0.0.0 anycast 0.0.0.0
```

**ПРИМЕЧАНИЕ** В данной версии NSG Linux для порта Sync PPP значение IP-адреса является фиктивным и не используется. В ходе процедуры установления PPP-соединения устройство NSG Linux посылает удаленной стороне пакет IPCP со значением адреса 0.0.0.0. Если на удаленной стороне установлено также устройство NSG Linux, оно соглашается на работу с таким адресом, и процедура идет дальше своим чередом. Однако другие устройства (например, под управлением базового ПО NSG) обычно интерпретируют такой пакет как запрос на присвоение динамического IP-адреса. В этом случае на удаленной стороне необходимо сконфигурировать IP-адрес, который она должна назначать устройству NSG Linux.

```
cisco-hdlc { keepalive | timeout } <1...100>
```

Значения параметров протокола Cisco-HDLC. Команда доступна только при инкапсуляции `cisco-hdlc`.

`keepalive` Интервал времени(в секундах), через который будут посылаться пакеты `keepalive`.

`timeout`      Время ожидания (в секундах). Если за этот период не придет пакет *keepalive*, то интерфейс перейдет в состояние DOWN.

`raw-hdlc type { ip | loopback }`

Только для портов с инкапсуляцией *raw-hdlc*: тип пакетов HDLC. Если установлено значение `ip`, данный порт может использоваться в качестве IP-интерфейса (т.е. иметь IP-адрес и другие параметры IP), а также коммутироваться с другим аналогичным портом или с виртуальным каналом Frame Relay (см. п.2.5). Данное значение установлено по умолчанию.

Если для порта установлено значение `loopback`, то он работает в режиме программного шлейфа, т.е. коммутируется сам с собой. В этом режиме он может быть использован для кольцевого теста, инициируемого удаленной стороной.

**ПРИМЕЧАНИЕ** Порт `raw-hdlc type ip` не обязательно должен иметь IP-адрес. При отсутствии адреса он может использоваться для коммутации с другим аналогичным портом или каналом Frame Relay.

#### б) Команды для портов с инкапсуляцией X.25:

Для портов X.25 определяется большое число параметров, относящихся к двум группам: канальному (подменю `lapb`) и сетевому (подменю `x25`) уровням протокола X.25. Подробное описание этих параметров приведено в Части 5 данного руководства.

#### в) Команды для портов с инкапсуляцией Frame Relay:

`frame-relay lmi { none | ansi | ccitt }`

Тип управления в протоколе Frame Relay.

`frame-relay intf-type { dte | dce }`

Логический тип порта Frame Relay:

`dte` — "user" (по умолчанию)

`dce` — "network"

`frame-relay t391 <0...255>`

`frame-relay t392 <0...255>`

`frame-relay n391 <0...255>`

`frame-relay n392 <0...255>`

`frame-relay n393 <0...255>`

Значения параметров протокола Frame Relay.

`frame-relay dlci <16...1022>`

`no frame-relay dlci <16...1022>`

Создание/настройка и удаление виртуального канала (DLC) с указанным идентификатором (DLCI). Первая команда создает канал с указанным номером, если он не существует, и осуществляет вход в меню настройки DLC (см. следующий параграф).

**ПРИМЕЧАНИЕ** Для специалистов, знакомых с базовым программным обеспечением маршрутизаторов NSG, следует заметить, что используемое в нем понятие *станции Frame Relay* аналогично понятию суб-интерфейса Frame Relay.

### §2.2.4. Настройка виртуальных каналов Frame Relay

Для использования виртуального канала Frame Relay необходимо создать его в меню `(config-port-sN)#` и перейти в подменю `(config-dlci-NN)#` для его настройки:

```
(config-port-s1)# frame-relay dlci 17
```

```
(config-dlci-17)#
```

Номер виртуального канала (DLCI) должен быть уникальным в пределах одного порта. Два DLC, принадлежащие к разным портам, могут иметь одинаковые DLCI. В меню `(config-dlci-NN)#` доступны следующие основные команды:

`description`    Административное описание данного DCL — текстовая строка длиной до 255 знаков. Если строка содержит пробелы, ее необходимо заключить в кавычки.

`cir <бит/с>`      Установка значения параметра CIR.

`bc <бит>`        Установка значения параметра BC.

`be <бит>`        Установка значения параметра BE.

**ПРИМЕЧАНИЕ** Параметры CIR, BC, BE относятся только к исходящему трафику, т.е. к пакетам, отправляемым в линию. Входящий трафик принимается весь, независимо от данных параметров.

```
bridge-group <1...255>
```

Присоединить физический интерфейс к мосту Ethernet с указанным номером (см. п.2.4).

```
route port <имя-порта> <dlci>
```

```
route port <имя-порта> no
```

```
route port no no
```

Установить постоянное виртуальное соединение (PVC) для пересылки пакетов из данного DLC в указанный DLC указанного порта Frame Relay, либо в указанный порт с инкапсуляцией RAW (см. п.2.5), либо удалить существующий PVC.

```
ip address <ip-префикс> [peer <ip-адрес>] [broadcast <ip-адрес>] [anycast <ip-адрес>]
```

```
no ip address <ip-префикс>
```

Установка и удаление параметров IP-интерфейса для данного DLC. Обязательным параметром является IP-префикс (адрес и длина маски, например, 123.134.145.156/24). При необходимости можно отдельно указать адреса удаленной стороны (peer) в соединении "точка-точка", широко-вещательной (broadcast) и групповой (anycast) рассылки.

Если DLC предполагается использовать только для прозрачной коммутации пакетов Frame Relay или для работы в составе моста, то параметры IP-интерфейса для него не требуются и, как правило, их следует устанавливать равными нулю (за исключением некоторых экзотических случаев).

**Пример.** Настроить порт 2 для работы с модулем SDSL, скорость 2048000 бит/с, протокол Frame Relay, LMI CCITT, два DLC с номерами 16 и 17.

```
port s2
```

```
physical sdsl
```

```
mode master
```

```
baudrate 2048000
```

```
encapsulation frame-relay
```

```
frame-relay lmi ccitt
```

```
frame-relay dlci 16
```

```
ip address 12.0.0.1/8
```

```
exit
```

```
frame-relay dlci 17
```

```
ip address 12.0.1.1/8
```

```
exit
```

**Особенности реализации протокола Frame Relay в устройствах NSG.** Гарантии качества услуг (QoS) для виртуального соединения Frame Relay определяются в терминах следующих пяти параметров:

CIR	<i>Committed Information Rate</i> — согласованная информационная скорость канала (бит/с).
EIR	<i>Exceeded Information Rate</i> — допустимое превышение информационной скорости канала (бит/с).
T <sub>c</sub>	<i>Committed Rate Measurement Interval</i> — период времени, на котором производится регулировка потока трафика.
BC	<i>Committed Burst Size</i> — количество бит, которое может быть передано по логическому каналу за интервал времени T <sub>c</sub> .
BE	<i>Exceeded Burst Size</i> — количество бит, на которое может быть превышено значение BC за интервал времени T <sub>c</sub> .

Эти параметры связаны друг с другом соотношениями:

$$BC = T_c \times CIR; \quad BE = T_c \times EIR$$

Таким образом, независимыми из них являются любые три, а оставшиеся два в этом случае определяются однозначно. В устройствах NSG пользователем устанавливаются значения CIR, BC и BE. Параметры T<sub>c</sub> (в секундах) и EIR (в битах) вычисляются, если не вдаваться в подробности, по формулам  $T_c = BC / CIR$  и  $EIR = BE / T_c$ .

По данному DLC в каждый интервал T<sub>c</sub> передается BC бит и, если канал загружен не полностью, то еще BE бит сверх того. Например:

```
CIR:256000 BC:128000 BE:128000
```

Тогда  $T_c = BC / CIR = 0,5$  сек, т.е. в каждые полсекунды будет гарантированно передаваться 128000 бит. Если другие DLC неактивны и физическая скорость на канале достаточно велика, то в эти же полсекунды будет передано еще 128000 бит. Максимальная информационная скорость, таким образом, может достигать 512000 бит/с.

Меньшие значения T<sub>c</sub> обеспечивают более равномерный поток трафика, но создают большую нагрузку на процессор.

**ПРИМЕЧАНИЕ** Строго говоря, параметры Frame Relay вычисляются в устройствах NSG несколько более сложным образом из-за того, что системный таймер работает с дискретностью 0,01 с. Таким образом, время T<sub>c</sub>, вычисленное по вышеприведенной формуле, должно быть округлено до десятков миллисекунд; интервалы менее 10 мс не допускаются. Обозначим эту величину

$T_C^*$ . После этого значения BE и BC пересчитываются по следующим формулам:

$$BC^* = T_C^* \times CIR; \quad BE^* = T_C^* \times EIR$$

Именно эти значения используются при формировании трафика Frame Relay. Как видно, они могут немного отличаться от значений, заданных пользователем. Фактически из пяти параметров DLCI неизменными остаются следующие три: CIR, EIR и  $T_C^* = 0,01 \times \text{round}(100 \times BC / CIR)$ , а BC и BE подгоняются под эти значения.

Если CIR:0, то  $T_C$  принимается равным 1 сек., BC независимо от установки обнуляется (т.е. не гарантируется никакая передача), а возможное количество передаваемых бит в секунду (при отсутствии нагрузки по другим DLC) определяется параметром BE.

**ВНИМАНИЕ** Если установить CIR:0 BC:x BE:0, то информация вообще передаваться не будет!

Все вышеописанные параметры относятся только к формированию исходящего трафика. Ограничение входящего трафика (полисинг) в сетях Frame Relay не предусмотрено. Для того, что, чтобы регулировать поток трафика в обратном направлении, необходимо соответствующим образом настроить параметры DLCI на другой стороне соединения. В частности, стандарты Frame Relay допускают создание несимметричных DLCI, т.е. скорости передачи и параметры качества услуги в противоположных направлениях могут быть различными.

### §2.2.5. Настройка асинхронных портов

В асинхронном режиме могут использоваться следующие типы физических интерфейсов:

- Модули IM-V24, IM-V35-x с установленным параметром physical-layer async (символические имена вида sN)
- Модули IM-485-x (символические имена вида sN)
- Модули IM-GPRS, IM-EDGE, UIM-CDMA и UIM-EVDO (символические имена вида sN)
- Модули IM-V34, IM-V92 (символические имена вида sN)
- Встроенные асинхронные порты устройств NSG-900/16A (символические имена вида aN)

Для настройки асинхронных портов используются следующие команды:

`baudrate <50...230400>`

Скорость работы порта. (Для просмотра полного списка поддерживаемых скоростей введите "?") Скорость 230400 бит/с поддерживается модулями IM-V24, IM-V35-x, IM-CDMA и не поддерживается встроенными асинхронными портами устройств NSG-900/16A, NSG-800/16A, модулями IM-485-x и IM-GPRS.

`databits { 5 | 6 | 7 | 8 }`

Число бит данных. Значение по умолчанию – 8.

`stopbits { 1 | 2 }`

Число стоп-бит. Значение по умолчанию – 1.

`parity { even | mark | none | odd | space }`

Четность. Значение по умолчанию – none.

**ПРИМЕЧАНИЕ** Для портов с инкапсуляцией ppp и slip допускается только следующий формат асинхронных данных, известный как 8N1:

```
databits 8
parity none
stopbits 1
```

`flowcontrol { hardware | software | none }`

Управление потоком. Значение по умолчанию – none.

`encapsulation { ppp | slip | tty | linux-shell | reverse-telnet | telnet | unused | console }`

Тип протокола канального уровня:

```
ppp — протокол PPP
slip — протокол SLIP
tty — вход в основную командную оболочку (vtysh)
linux-shell — вход в командную оболочку ОС Linux (ash)
telnet — прозрачное подключение к серверу Telnet
reverse-telnet — прозрачное подключение удаленного клиента Telnet к данному порту
console — специальный тип для разделяемого консольного порта в отдельных типах шасси
unused — формальное значение, используемое по умолчанию.
```

`virtual-template <1...25>`

Только при использовании инкапсуляции ppp или slip: номер шаблона IP-интерфейса. Шаблон содержит совокупность параметров, относящихся к работе протокола IP, аутентификации по PAP/CHAP и т.п. Если номер равен 0, все параметры устанавливаются по умолчанию. Подробно об использовании шаблонов см. п.2.6.1. При изменении шаблона порт рестартует автоматически.

`chat-script <имя>`

Только при использовании инкапсуляции `ppp`: имя сценария, который будет использован для установки физического соединения. Имя может вводиться в кавычках или без них. Пустая строка (" ") означает, что сценарий не используется.

**ПРИМЕЧАНИЕ** Если в шаблоне для данного виртуального интерфейса указан режим установления соединения по требованию (`ppp connection on-demand`), то для нормальной работы интерфейса необходимо указать какой-либо сценарий, хотя бы пустой:

```
chat-script DUMMY " " " "
```

(Первая пара апострофов означает "ничего не ждать", вторая — "ничего не посылать"; все тело сценария взято в двойные кавычки.)

**ПРИМЕЧАНИЕ** В данной версии NSG Linux при изменении сценария соединения необходимо рестартовать порт вручную командами `adm down / adm up`.

`chat-log { previous | current }`

Только при использовании инкапсуляции `ppp`: просмотр журнала выполнения сценария физического соединения. Ключевое слово `previous` выводит журнал последней завершенной попытки, `current` — текущей попытки. Во втором случае, повторяя ввод команды, можно проследить ход работы сценария по мере его выполнения.

`ppp-log { previous | current }`

Только при использовании инкапсуляции `ppp`: просмотр журнала сеанса PPP. Ключевое слово `previous` выводит журнал последней завершенной попытки, `current` — текущей попытки. Во втором случае, повторяя ввод команды, можно проследить ход сеанса по мере его выполнения.

В ходе выполнения сценария физического соединения и PPP-сессии весь вывод направляется также на сервер Syslog, если он включен. Для этого необходимо предварительно перейти в командную оболочку ОС Linux и выполнить команду:

```
syslogd
```

либо включить `syslogd` при помощи сценария, см. Часть 1. Просмотреть файл системного журнала можно также в командной оболочке ОС Linux следующей командой:

```
cat /var/log/messages
```

`tcp-port <номер>`

Только при использовании инкапсуляции `reverse-telnet`: номер порта TCP, по которому осуществляется доступ удаленного клиента Telnet к данному физическому порту. Рекомендуется использовать номера портов TCP старше 3000 во избежание конфликтов с портами стандартных служб и протоколов.

`remote-ip <ip-адрес>`

Только при использовании инкапсуляции `telnet`: IP-адрес удаленного сервера Telnet, к которому автоматически устанавливается соединение. Если адрес не указан (установлен в 0.0.0.0), то при подключении к данному порту пользователь попадает в командный режим клиента Telnet и устанавливает соединение вручную при помощи команды `open`.

`remote-port <tcp-порт>`

Только при использовании инкапсуляции `telnet`: номер порта TCP удаленного сервера Telnet, к которому автоматически устанавливается соединение. По умолчанию устанавливается порт 10023.

`transparent { yes | no }`

Только при использовании инкапсуляции `telnet`: включение/выключение прозрачного режима Telnet. В непрозрачном режиме клиент перехватывает и обрабатывает `escape`-последовательности, в прозрачном — пропускает все символы без изменения. По умолчанию устанавливается непрозрачный режим.

## §2.2.6. Особенности использования проводных и сотовых модемных модулей

Интерфейсные модули IM-GPRS и IM-EDGE предназначены для подключения к сотовым сетям GSM/GPRS/EDGE, при этом они могут работать как в режиме канальной передачи данных (Channel Separated Data, CSD), так и в пакетных режимах GPRS (General Radio Packet Service) и EDGE (Enhanced Data rates for Global Evolution).

Модули IM-CDMA *h/w ver.1* (снят с производства), UIM-CDMA и UIM-EVDO предназначены для подключения к сотовым сетям CDMA 2000 1x/EV-DO (CDMA Evolution, Data Only).

Модули IM-V34 и IM-V92 — аналоговые модемы тональной частоты, предназначенные для передачи данных по коммутируемым телефонным линиям.

Интерфейсные модули CDMA (UIM-CDMA, UIM-EVDO) в устройствах серии NSG-700, за исключением NSG-700/40AR, работают через внутренний интерфейс USB. Для них следует установить тип модуля

```
card s1 uim-cdma
```

Кроме того, при работе в режиме USB **необходимо** снять перемычку аппаратного рестарта (J1) на модуле.

Во всех остальных случаях, в данной версии программного обеспечения, все эти модули рассматриваются не как специфические интерфейсные карты, а как медиа-конвертеры между интерфейсом V.24 (RS-232) и радио- или проводным (FXO) интерфейсом. По этой причине для них следует установить тип модуля

```
card sN im-v24
```

Допустимые инкапсуляции зависят от используемого модуля и режима подключения:

- В канальном режиме (CSD) сети GSM и при проводном соединении устанавливается соединение физического уровня между двумя модемами. Поверх этого соединения может работать любой асинхронный протокол (ppp, slip, reverse-telnet, telnet и др.), однако следует заметить, что из всего этого списка только протокол PPP обеспечивает достаточно развитые механизмы для управления установлением и разрывом соединения.
- В пакетном режиме сетей GSM и CDMA устанавливается соединение на всех уровнях протокольного стека между абонентским терминалом (модемом) и сетью оператора, которая одновременно является и физической средой передачи, и поставщиком сетевых услуг. Соединение осуществляется по тому протоколу, который предлагается сетью. В реальности это исключительно IP-over-PPP (другие протоколы, предусмотренные стандартами, на практике не поддерживаются). Поэтому для пакетных режимов единственная допустимая инкапсуляция — ppp.

Следующая особенность состоит в выборе скорости. Она должна быть установлена одинаковой для порта устройства NSG и для порта интерфейсного модуля. При этом вышеперечисленные модули имеют следующие существенные отличия:

- Модули IM-GPRS, IM-V34 и IM-V92 автоматически настраиваются на скорость в асинхронном порту. (Если иное не задано командой AT+IPR.) Для них можно устанавливать в порту NSG любую из поддерживаемых скоростей.
- Для модулей UIM-CDMA, UIM-EVDO при работе в режиме USB параметр скорости как таковой отсутствует вовсе.
- При работе через внутренний асинхронный интерфейс модули UIM-CDMA, UIM-EVDO не поддерживают автоматическое определение скорости. Для них скорость устанавливается вручную командой вида AT+IPR=nnnn (где nnnn — требуемая скорость порта) и автоматически запоминается в энергонезависимой памяти. После этого необходимо изменить параметр baudrate порта NSG надлежащим образом.
- Модуль IM-EDGE также не поддерживает автоматическое определение скорости, но установленная вручную скорость не запоминается в нем автоматически. Для изменения и немедленного сохранения скорости следует вводить две AT-команды в одной строке:

```
AT+IPR=nnnn&W
```

после чего изменить скорость порта NSG.

Модули IM-GPRS поддерживают скорость до 115200 бит/с включительно, остальные — до 230400 Кбит/с. По умолчанию для всех вышеперечисленных модулей установлена скорость 115200 бит/с.

**ВНИМАНИЕ** Если в порту модуля ошибочно выставлена скорость, не поддерживаемая портом устройства NSG, то связь с модулем будет утеряна!

Дальнейшее управление модулем осуществляется при помощи AT-команд, вводимых при помощи сценариев работы порта PPP, либо напрямую в прозрачном асинхронном режиме. Общим для всех модемов является Hayes-совместимый набор команд (V.25ter). Основные команды для работы в сотовых сетях приведены в документах NSG:

*Управление модулем IM-GPRS с помощью AT-команд*

*Управление модулем IM-EDGE с помощью AT-команд*

*Управление модулями UIM-CDMA и UIM-EVDO с помощью AT-команд.*

**ПРИМЕЧАНИЕ** Если на модеме установлена перемычка аппаратного рестарта по падению сигнала DTR порта (что рекомендуется в большинстве случаев), то сценарий дозвона должен начинаться с искусственной паузы, необходимой для загрузки внутреннего программного обеспечения модема. В течение этого времени модем не может корректно обрабатывать AT-команды. Рекомендуемая задержка для IM-V34, IM-V92 — не менее 3 сек, для сотовых модемов — приведена в вышеупомянутых документах.

**ПРИМЕЧАНИЕ** Доступность ряда услуг (входящие модемные соединения, гарантированное качество услуг GPRS и т.п.) зависит от их поддержки в сети конкретного оператора GSM и от условий конкретного абонентского договора.

### §2.2.7. Сценарии соединения для асинхронных портов

Сценарии соединения предназначены для инициализации оборудования, подключенного к асинхронным портам устройства NSG, и установления соединений на физическом уровне. В данной версии NSG Linux сценарии используются только для асинхронных портов с инкапсуляцией rpp. Для управления сценариями используются следующие команды в меню (config–nsg)#:

```
chat-script <имя> "<строка сценария>"
no chat-script <имя>
```

Создание/изменение и удаление сценария. Имя сценария представляет собой текстовый идентификатор, по которому асинхронный порт может ссылаться на данный сценарий (параметр chat-script в конфигурации порта). Остальная часть команды рассматривается как единая строка, являющаяся собственно сценарием. Максимальная длина сценария — 255 символов.

Просмотреть существующие сценарии можно с помощью команды display all или display config в меню (config–nsg)#.

Сценарий представляет собой последовательность записей "жду" — "посылаю", разделенных пробелами. Нечетные члены последовательности представляют собой сообщения, ожидаемые от модема или удаленной системы, а следующие за ними четные — команды, выдаваемые интерфейсом в линию. Пример ввода сценария:

```
(config–nsg)# chat-script logging_on "ogin: vasya.pupkin assword: qwerty"
```

Поскольку тело сценария содержит пробелы, оно, как строковый параметр, должно быть заключено в кавычки (""). Для ввода кавычек, пробелов и других спецсимволов в текст записей сценария используются особые правила, описанные ниже.

Приведенный выше пример означает, что интерфейс PPP будет ожидать от удаленной системы приглашения, оканчивающегося на ogin: (без пробела). Когда эта последовательность символов будет получена, в линию будет послана строка vasya.pupkin. Затем клиент будет ждать, пока из линии будет получено приглашение assword:, и в ответ пошлет строку qwerty — и так далее до конца сценария. Каждая посылаемая последовательность символов дополняется символом <CR>.

**ВНИМАНИЕ** Пароль для входа в удаленную систему представляет собой, с точки зрения сценария, обычную запись, не выделяющуюся среди остальных, и хранится в конфигурации устройства в открытом виде.

Любая запись сценария (ожидание или посылка) может быть заключена в апострофы, или одинарные прямые кавычки ('). Апостроф необходим, если запись сама по себе должна содержать пробелы, например:

```
(config–nsg)# chat-script logging_on "'ogin: ' vasya.pupkin 'assword: ' qwerty"
```

Здесь, в отличие от предыдущего примера, записи 'ogin: ' и 'assword: ' означают, что устройство будет ждать от удаленной системы подсказок, оканчивающихся пробелами после двоеточия.

Два апострофа означают пустую запись (' ' — не путать с одиночной двойной кавычкой!). Если такая запись стоит в качестве записи ожидания, то клиент PPP ничего не ждет и сразу переходит к посылке следующей команды. В частности, чтобы начать выполнение сценария не с ожидания, а с выдачи команды модему, следует указать пустой первый член последовательности:

```
' ' ATZ OK ATDP1234567 CONNECT ...
```

Если пустая запись стоит на месте записи, посылаемой в модем, то посылается символ <CR>, т.е. пустая строка. В частности, если сценарий должен завершиться получением сообщения CONNECT, то для соблюдения четности записей следует указать:

```
"... CONNECT ' ' "
```

Внутри апострофов не могут использоваться спецсимволы, перечисленные в конце данного параграфа. Однако допускается брать в апострофы не всю запись а только ее фрагмент; в этом случае запись рассматривается как единое целое. Например, следующие две записи эквивалентны:

```
'ABC '123
'ABC 123'
```

а следующая запись может быть сделана только с двумя парами апострофов, чтобы спецсимвол \N оказался вне их:

```
'sending NULL '\N' sent NULL'
```

Если в течение некоторого времени (по умолчанию 45 секунд) ожидаемая последовательность не будет получена, то выполнение сценария заканчивается неудачей и интерфейс PPP переходит в исходное состояние. Продолжительность тайм-аута может быть изменена включением параметра TIMEOUT в сценарий перед строкой ожидания. Например, в сценарии



```
" ' ' ATZ OK ATDT5551212 CONNECT ' ' TIMEOUT 10 'ogin:' sidorov"
```

тайм-аут перед ожиданием строки ogin: будет уменьшен до 10 секунд. Нулевое значение TIMEOUT указывает, что время ожидания не ограничено.

В качестве записи ожидания может быть указана последовательность записей "ожидание"—"посылка"—...—"ожидание", разделенных дефисами. Например, сценарий

```
" ' ' ATD1234 CONNECT-ATD1256-CONNECT-ATD1278-CONNECT ' ' "
```

означает, что если после набора номера 1234 в течение 45 секунд не получен ответ CONNECT, то модем должен набрать номер 1256; если по этому номеру тоже не удастся соединиться (нет ответа, получен BUSY, NO CARRIER, NO DIALTONE и т.п.) — набрать 1278. Как только получен ответ CONNECT, выполнение альтернативной ветви завершается и исполняется следующий шаг основного сценария — в данном случае, посылка пустой строки и нормальное завершение сценария. Если все варианты альтернативных посылок испробованы и ни на одну из них не получен ожидаемый ответ (в общем случае он может быть свой для каждой из посылок), выполнение сценария завершается аварийно.

Аналогичный пример для команд и ответов, содержащих спецсимволы:

```
" ' ' 'AT+CPIN?' ' +CPIN: READY-AT+CPIN=9876-+CPIN: READY' ATD1234 CONNECT ' ' "
```

В этом случае сначала проверяется регистрация модуля IM-GPRS (или внешнего модема) в сети GSM. Если получен ответ +CPIN: READY (с пробелом!), то интерфейс приступает к набору номера; если нет — вводит PIN-код (AT+CPIN=9876) и снова ждет ответа +CPIN: READY.

**ПРИМЕЧАНИЕ** Основные команды для подключения к сотовым сетям см. в документах NSG:

*Управление модулем IM-GPRS с помощью AT-команд.*

*Управление модулем IM-EDGE с помощью AT-команд.*

*Управление модулями UIM-CDMA и UIM-EVDO с помощью AT-команд.*

Специальным образом вводятся символы, перечисленные ниже.

- \? Вопросительный знак. При последовательном нажатии клавиш \ и ? обратный слэш преобразуется в вопросительный знак "на лету"
- \" Двойная кавычка ("), например:  
... OK AT+CGDCONT=1,\"IP\", \"internet.cellprovider.ru\" OK ATD\*99# CONNECT ...  
В этом случае в модуль IM-GPRS будет послан следующий GPRS-контекст:  
+CGDCONT=1, "IP", "internet.cellprovider.ru"
- \\ Обратный слэш (\) — например, для некоторых фирменных команд у отдельных типов модемов.

Три вышеприведенные escape-последовательности являются общими для всех строковых параметров, к которым относится сценарий соединения. Помимо этого, в сценариях предусмотрены дополнительные спецсимволы и инструкции:

- \- Дефис (-) как текстовый символ в записи ожидания — чтобы отличать его от дефиса, разделяющего альтернативные варианты посылок/ответов. Для команд, посылаемых PPP-интерфейсом, допускается вводить дефис обычным образом.
- \b Символ <BS> (0x08)
- \c Подавить символ <CR> в конце строки. Данный спецсимвол может использоваться только в конце строки и только для команд, посылаемых PPP-интерфейсом. По умолчанию, в конце каждой посылаемой строки вставляется символ <CR>.
- \d Задержка на 1 сек (только для команд, посылаемых PPP-интерфейсом)
- \K Послать сигнал BREAK (только для команд, посылаемых PPP-интерфейсом)
- \n Символ <LF> (0x0A)
- \N Послать символ NULL (0x00; только для команд, посылаемых PPP-интерфейсом)
- \p Задержка на 1/10 сек (только для команд, посылаемых PPP-интерфейсом)
- \r Символ <CR> (0x0D)
- \s Пробел (символ 0x20). Это альтернативный способ ввода пробела, не разрывающий целостность отдельной записи. Его можно использовать вместо апострофов, чтобы ввести пробел в тело посылки/ответа, например:  
AT+CGREG? +CGREG:\s1,1
- \t Символ горизонтальной табуляции (0x09)
- ^A ... ^Z Непечатаемые управляющие символы с кодами от 0x01 до 0x1A
- ^[, ^], ^^, ^\_ Непечатаемые управляющие символы с кодами от 0x1B, 0x1D, 0x1E, 0x1F, соответственно
- \^ Символ ^
- \' Апостроф

**Пример.** Для IP-интерфейса типа PPP, работающего в режиме сервера, сценарий обычно содержит команды инициализации модема и перевода его в режим ожидания входящих звонков:

```
chat-script wait4call " ' ' ATZ OK ATSO=1"
```

**ПРИМЕЧАНИЕ** Для модулей сотовых и аналоговых проводных модемов, если установлена переключатель аппаратного рестарта по падению сигнала DTR порта (что рекомендуется в большинстве случаев), необходима принудительная задержка в начале сценария перед вводом AT-команд. Задержка легко реализуется следующим образом:

```
chat-script DIALUP "TIMEOUT n XXX-AT-OK AT TIMEOUT 45 OK ..."
```

При исполнении сценария PPP-интерфейс сначала ждет строку XXX в течение n секунд, затем, не получив ее, посылает AT и получает OK. Следующая пара AT OK восстанавливает таймаут по умолчанию — 45 сек (можно вместо этого вставить TIMEOUT 45 непосредственно перед следующим ожидаемым ответом).

Рекомендуемая задержка для IM-V34, IM-V92 — не менее 3 сек, для сотовых модемов — приведена в дополнительных инструкциях по их настройке.

### §2.2.8. Особенности использования консольного порта в устройствах NSG-700

Консольный порт устройств NSG-700 может также использоваться как для передачи данных наравне с другими асинхронными портами. Для работы в режиме консольного ему назначается специальный тип протокола `encapsulation console`, предусмотренный только для данного порта. Никакие настройки, производимые средствами основного программного обеспечения, в этом режиме не действуют; параметры асинхронного интерфейса устанавливаются в загрузчике U-Boot.

**ПРИМЕЧАНИЕ** В процессе загрузки программного обеспечения консольный порт функционирует как консольный, в частности, выводит сообщения и приглашения U-Boot. Может возникнуть ситуация, когда подключенное оборудование (например, консоль другого сетевого устройства) пытается интерпретировать эти сообщения как команды, отвечает на них (например, ERROR), устройство NSG-700 также воспринимает эти ответы как команды U-Boot, и т.п. В результате устройство бесконечно перезагружается или впадает в иное неработоспособное состояние.

Для устранения подобной ситуации необходимо использовать загрузчик U-Boot версии *NSG build 2* или старше с установленной переменной окружения `silent=yes`.

В заводской конфигурации порт настроен как консольный с параметрами 9600, 8N1. Изменить тип console на любой другой протокол, доступный для асинхронного порта, возможно только при управлении устройством по сети (посредством Telnet, SSH или X.25).

Если порт используется для передачи данных, то установить ему режим console можно в любое время. По этой причине не рекомендуется использовать его для подключения к вышестоящей сети, чтобы избежать ошибок, приводящих к потере удаленного управления устройством.

### §2.2.9. Особенности работы портов SLIP

Для асинхронного порта, работающего в режиме SLIP, при создании IP-интерфейса воспринимаются следующие параметры, описанные в шаблоне `virtual-template` (см. п.2.6.1):

```
ip address
peer ip address
ip mtu
keepalive
```

Остальные параметры, входящие в состав шаблона, относятся только к интерфейсу PPP и в данном случае игнорируются. Особо следует остановиться на параметре `keepalive`, поскольку в самом протоколе SLIP механизм послышки контрольных пакетов не предусмотрен. Вместо него используется так называемый механизм *outfill* (термин из SLIP в Linux). В случае отсутствия входящего трафика через заданный интервал времени посылается пакет-пустышка (байт 0xC0), который воспринимается удаленной стороной как подтверждение работоспособности. Если в течение времени `3×keepalive` не принят ни один пакет, SLIP-соединение разрывается, физические сигналы порта (DTR, RTS) опускаются, и через 1 сек. порт рестартует и пытается установить соединение заново.

### §2.2.10. Настройка портов Ethernet

Для подключения к ЛС Ethernet могут использоваться следующие типы физических интерфейсов:

- Встроенный порт Fast Ethernet (символическое имя eth0)
- Модули IM-ET10, IM-ET10F и UM-ET100 (символические имена вида sN)
- Интерфейсы SDSL, SHDSL.bis с установленным параметром physical-layer ethernet (символические имена вида sN)

Особо следует остановиться на последнем варианте. В программном обеспечении NSG поддерживается инкапсуляция Ethernet-over-HDLC, позволяющая транслировать кадры Ethernet по синхронным каналам WAN. Для этой цели необходимо использовать следующие пары устройств:

- Интерфейсный модуль IM-SDSL *hardware version 2* + устройство NSG-50 SDSL bridge
- Интерфейсный модуль IM-xSHDSL/bis + устройство NSG-50 xSHDSL.BIS bridge
- Интерфейсная карта MV-xSHDSL + устройство NSG-50 xSHDSL.BIS bridge
- Два модуля IM-SDSL *hardware version 2*
- Два модуля IM-xSHDSL/bis и/или карты MV-xSHDSL

Совместно интерфейсный модуль в центральном маршрутизаторе NSG и устройство NSG-50 на удаленной площадке образуют "удаленный" порт Ethernet. На физическом уровне он использует транспорт SDSL, но с точки зрения как программных компонент маршрутизатора 2-3 уровней, так и подключенной локальной сети он выглядит как порт Ethernet.

**ПРИМЕЧАНИЕ** Используемая инкапсуляция Ethernet-over-HDLC является фирменной и в общем случае не совместима с продуктами других производителей. Режим работы Ethernet-over-HDLC поддерживается только аппаратной модификацией IM-SDSL *h/w ver.2* (обр. 2006 г. и позже).

Для настройки физических интерфейсов Ethernet (фиксированных или сменных) используются команды:

`mac-address XX:XX:XX:XX:XX:XX`

Физический адрес порта. Встроенные порты Fast Ethernet по умолчанию имеют MAC-адреса по умолчанию вида 00:09:56:xx:xx:xx.

Сменные модули IM-ET10, IM-ET10F, UM-ET100 не имеют собственных MAC-адресов, и для них устанавливается адрес, равный MAC-адресу встроенного порта. Поскольку два порта Ethernet одного устройства не подключаются к одной локальной сети, это, в общем случае, не вызывает затруднений. Для полностью корректной конфигурации сети можно также назначить модулям уникальные MAC-адреса, но при этом рекомендуется ограничиться изменением только двух младших байт.

`mode { half-duplex | full-duplex }`

Режим работы интерфейсного модуля IM-ET10F:

`half-duplex` — полудуплексный (по умолчанию)

`full-duplex` — полнодуплексный

Автоматический выбор режима для модуля IM-ET10F не предусмотрен.

**ПРИМЕЧАНИЕ** Режим Full Duplex поддерживается только интерфейсными модулями IM-ET10F (выпуск 2005 г. и позже). Установка этого режима для модулей IM-ET10 (выпуск до 2004 г. включительно) приводит только к возникновению большого числа коллизий и снижению пропускной способности порта.

Для устройства NSG-800/4WL рекомендуется использовать в полнодуплексном режиме не более 2 портов Ethernet.

**ПРИМЕЧАНИЕ** Для встроенного порта Fast Ethernet и модуля UM-ET100 выбор скорости (10/100 Мбит/с) и режима передачи (полу/полнодуплексный) в данной версии NSG Linux производится только автоматически.

**ПРИМЕЧАНИЕ** Встроенный порт Fast Ethernet устройств NSG-900 и модуль UM-ET100 автоматически определяют состояние линии, т.е. при отсутствии физического соединения с сетью соответствующий IP-интерфейс ставится в состояние DOWN, а соответствующие ему маршруты удаляются из таблицы маршрутизации.

Для встроенных интерфейсов Fast Ethernet устройства NSG-700, модулей IM-ET10, IM-ET10F, а также интерфейсов xDSL, настроенных в режиме моста, состояние IP-интерфейса не зависит от наличия физического соединения.

Для настройки физических интерфейсов Ethernet-over-HDLC используются следующие команды:

`baudrate <бит/с>`

Скорость работы порта. Для просмотра списка скоростей введите "baudrate ?". Значение действующее и должно быть установлено одинаковым на обеих сторонах.

mode { coe | cpe }

Режим синхронизации порта (значение COE эквивалентно MASTER, CPE — SLAVE). Для двух физических интерфейсов xDSL, непосредственно соединенных друг с другом, всегда должны быть установлены противоположные режимы синхронизации: для одного — coe, для другого cpe. Источником синхронизации в этой паре является устройство Central Office Equipment (COE).

**ПРИМЕЧАНИЕ** Для интерфейсов SDSL выбор способа интерпретации линейного кода 2B1Q в данной реализации NSG Linux не предусмотрен. Всегда используется нормальная последовательность (сначала знак, затем амплитуда).

Для обоих типов портов Ethernet используются также команды:

encapsulation { ethernet | vlan }

Тип пакетов Ethernet:

ethernet — обычная физическая ЛС Ethernet

vlan — VLAN (IEEE 802.1Q)

vlan <2...4094>

no vlan <2...4094>

Создание/настройка и удаление виртуальной локальной сети. Первая команда создает VLAN с указанным номером, если она не существует, и осуществляет вход в меню настройки VLAN.

Внутри меню доступны все те же пункты, что и для физического порта Ethernet, за очевидным исключением adm-state, mac-address и encapsulation.

Команда доступна только при инкапсуляции vlan.

Если для порта задана инкапсуляция vlan, то при создании каждой VLAN в системе появляется суб-интерфейс с именем вида sN.<vlan> или eth0.<vlan>, где sN, eth0 — имя физического порта, <vlan> — номер VLAN. Примеры: s1.2, eth0.6, и т.д. Именно эти объекты используются для маршрутизации и коммутации пакетов.

ip address <ip-префикс> [peer <ip-адрес>] [broadcast <ip-адрес>] [anycast <ip-адрес>]

no ip address <ip-префикс>

Установка и удаление параметров IP-интерфейса. Обязательным параметром является IP-префикс (адрес и длина маски, например, 123.134.145.156/24). При необходимости можно отдельно назначить адреса удаленной стороны (peer) в соединении "точка-точка", широковещательной (broadcast) и групповой (anycast) рассылки. Команда находится в различных узлах меню, в зависимости от выбранной инкапсуляции:

— для инкапсуляции ethernet — в меню настройки порта, например, (config-port-eth0)#

— для инкапсуляции vlan — в меню настройки каждой VLAN, (config-vlan-N)#

bridge-group { <1...255> | no }

Присоединить физический интерфейс или VLAN к мосту с указанным номером (см. п.2.4).

Указанный мост должен уже существовать в системе.

**ПРИМЕЧАНИЕ** При подключении порта Ethernet или VLAN к мосту они безусловно отключаются от IP-интерфейса маршрутизатора. При этом все связанные с ними параметры IP обнуляются и становятся недоступными для конфигурирования.

## §2.2.11. Особенности использования модуля IM–SDSL в устройствах NSG–700

Интерфейсный модуль IM–SDSL *h/w ver.2* может использоваться в устройствах NSG–700/4AU, NSG–700/8A, NSG–700/12A только в фирменном режиме эмуляции порта Ethernet (Ethernet-over-HDLC). На удаленной стороне при этом может использоваться либо такой же модуль во втором маршрутизаторе NSG, либо мост NSG–50 SDSL Bridge.

Модуль устанавливается только в основной разъем расширения 1 и подключается к внутреннему интерфейсу Ethernet как четвертый порт коммутатора (см. след. параграф). После установки

```
card s1 im-sdsl
```

в системе появляется меню phy s1. Данное меню содержит только параметры, относящиеся к физическому уровню SDSL: скорость, режим COE/CPE и т.п. Никакой новый IP-интерфейс при этом не создается. Трафик из линии проходит через коммутатор и обрабатывается интерфейсом eth0, поэтому MAC-адрес и все параметры IP устанавливаются в меню port eth0.

**ПРИМЕЧАНИЕ** Интерфейсный модуль IM–SDSL *h/w ver.1* и другие модули IM–xDSL в устройствах NSG–700 использоваться не могут.

### §2.2.12. Настройка портов USB

Для подключения внешних USB-устройств могут использоваться следующие типы интерфейсов:

- Сменные модули UM-USB на устройствах NSG-700 (символические имена вида sN)
- Фиксированные порты USB на устройстве NSG-1000 (символические имена вида usbN)
- Сменные модули UM-USB в дополнительных разъемах расширения на устройстве NSG-1000 (символические имена вида sN)

Для настройки портов USB используются следующие команды:

`encapsulation { printer | unused }`

Выбор типа внешнего устройства, подключенного к порту USB (или его отмена, соответственно).

Другие типы USB-устройств в данной версии NSG Linux не поддерживаются. После выбора инкапсуляции в меню появляются пункты, специфичные для выбранного класса устройств.

`description <описание>`

Текстовое описание устройства, для удобства администрирования. Максимальная длина описания

— 255 символов. Значение по умолчанию — пустая строка.

`adm-state { up | down }`

Административное состояние устройства. По умолчанию все вновь создаваемые USB-устройства включены.

`tcp-port { 9100 | 9101 | 9102 }`

Только при подключении принтера: номер порта TCP, на котором будет работать данный принтер.

По умолчанию установлен TCP порт 9100. Если к устройству подключено более одного принтера, то для каждого из них необходимо назначить уникальный номер порта.

`bidirectional { yes | no }`

Только при подключении принтера: выбор двух- или одностороннего (от удаленного клиента к принтеру) режима передачи. По умолчанию установлен двусторонний режим. Если драйвер и программное обеспечение принтера не поддерживают его, можно установить односторонний.

## §2.3. Настройка физических интерфейсов E1

### §2.3.1. Принципы настройки интерфейсов E1

Физические интерфейсы E1 представляют собой специфический тип интерфейсов, поскольку формируют структурированный поток байтов на физическом уровне. Организация этой структуры представляет собой дополнительную задачу, которая является специфической для интерфейсов E1.

Для настройки интерфейса необходимо сначала идентифицировать модули E1, имеющиеся в системе, либо автоматически, либо при помощи команд `chassis` и `card` (см. Часть 1). Если в некотором разъеме расширения (например, s3) идентифицирован один из модулей E1, то в меню `(config-nsg)#` появляется следующая команда:

```
controller e1 <разъем>
```

Вход в меню конфигурации физического интерфейса. Параметром команды является символическое имя разъема расширения (s1 ... s4) или фиксированного интерфейса E1, в зависимости от модели устройства.

Дальнейшая настройка производится в меню выбранного физического интерфейса. Формально эта задача решается в терминах *групп канальных интервалов* и разделяется на четыре этапа:

- Настройка общих параметров интерфейса и физического приемопередатчика (LIU — *line interface unit*)
- Настройка цикловой структуры потока E1 (управление фреймером — *framer*)
- Формирование групп из выбранных канальных интервалов (*groups*)
- Коммутация выбранных групп друг с другом или с физическими портами (*cross-connect*)

### §2.3.2. Терминология

Некоторые режимы работы канала E1 (ИКМ–30) иногда обозначаются различными терминами; кроме того, одни и те же термины иногда могут толковаться по-разному различными производителями, специалистами, авторами технической литературы и компьютерной периодики. Определенные разночтения имеются также между различными редакциями стандартов ITU и национальными аналогами соответствующих стандартов. Для единообразного толкования ниже приведены описания режимов интерфейсов E1 и используемые термины.

Режим	Название согласно ITU–Т G.703, G.704 в редакции 2001 г.	Другие названия
Неструктурированный поток 64 Кбит/с	E0	DS0, G703.1
Неструктурированный поток 2048 Кбит/с	E12	G.703.6 (unframed)
		<i>некорректно: G.703, E1</i>
Структурированный поток 32×64 Кбит/с, в котором данный интерфейс организует <i>только одну</i> группу канальных интервалов	E1	Fractional E1 (FE1) sub-E1 ИКМ–30
Структурированный поток 32×64 Кбит/с, в котором данный интерфейс организует <i>более одной</i> группы канальных интервалов	E1	Channelized E1 (CE1) ИКМ–30

Термин "G.703" сам по себе (без указания подпункта) является некорректным, поскольку данный стандарт описывает весь иерархический ряд скоростей PDH. Употреблять его допустимо только в тех случаях, когда из контекста однозначно следует, что речь идет о некоторой скорости — например, 2048 Кбит/с. Это же относится к термину "G.704", которым иногда обозначают структурированный поток E1 (на самом деле он описывает весь ряд цикловых структур).

Термин "E1" является не вполне однозначным, поскольку может означать несколько различных режимов работы интерфейса и, кроме того, иногда ошибочно используется для неструктурированного потока 2048 Кбит/с. По этой причине рекомендуется явно оговаривать, о каком режиме идет речь: структурированном или неструктурированном потоке, с использованием одной или нескольких выделенных групп канальных интервалов. В структурированном режиме канальный интервал 0 всегда является служебным; таким образом, для передачи данных может быть выделено не более 31 канального интервала с суммарной скоростью  $31 \times 64 = 1984$  Кбит/с.

Толкование терминов Fractional E1 и Channelized E1 также может варьироваться в зависимости от контекста (стандарты ITU, телефония, передача данных). В данном Руководстве и в другой документации NSG они применяются в смысле, наиболее близком к сложившемуся применительно к передаче данных. Аналогичное толкование используется другими ведущими производителями сетевого оборудования.

### §2.3.3. Общие параметры и настройка LIU

Настройка общих параметров интерфейса и физического приемопередатчика производится в меню контроллера E1, например, (config-controller-sN)#, при помощи следующих команд:

#### а) общие параметры

adm-state { up | down }

Установка административного статуса интерфейса. По умолчанию, все интерфейсы находятся в состоянии up.

clock-source { line | internal | through }

Выбор источника синхронизации для передатчика интерфейса:

line Передатчик синхронизируется от сигнала, принимаемого из линии. (Этот режим называется также loop.)

internal Передатчик синхронизируется от внутреннего генератора. (Этот режим называется также local.)

through Передатчик синхронизируется от приемника другого интерфейса, работающего в паре с данным в режиме *drop-and-insert*.

По умолчанию используется режим line.

loopback { disable | local | line }

Включение режима кольцевого теста (*loopback*) на интерфейсе:

disable Кольцевой тест выключен, интерфейс может использоваться для передачи данных.

line Режим шлейфа для удаленного теста — все данные, принимаемые интерфейсом из линии, посылаются обратно в линию. Допускается только в режиме синхронизации от линии (clock-source line).

local Локальный кольцевой тест — все данные, передаваемые устройством через данный интерфейс, посылаются обратно в устройство. Допускается только в режиме синхронизации от внутреннего генератора (clock-source local).

По умолчанию кольцевой тест выключен (режим disable).

#### б) параметры LIU E12

linecode { hdb3 | ami }

Выбор линейного кодирования сигнала:

hdb3 Код HDB3 (High-density bipolar 3). Этот код предусмотрен спецификациями ИТУ-Т и используется в большинстве систем связи. По этой причине, он используется по умолчанию.

ami Код AMI (Alternate mark inversion). Используется исключительно для совместимости с некоторыми специфическими системами связи, реализованными с отклонениями от рекомендаций ИТУ-Т.

rcv-gain { max | min }

Выбор чувствительности приемника:

max Максимальная чувствительность. Режим используется по умолчанию.

min Минимальная чувствительность. Режим может быть полезен на коротких линиях.

### §2.3.4. Настройка цикловой структуры потока E1 (framer)

Настройка цикловой структуры потока E1 производится в меню контроллера следующими командами:

framing { no-crc4 | crc4 | none }

Выбор цикловой структуры:

no-crc4 Структурированный (*framed*) поток E1 без использования сверхциклов CRC-4. Поток состоит из 32 канальных интервалов, по 64 Кбит/с каждый.

crc4 Структурированный поток E1 с использованием сверхциклов CRC-4.

none Неструктурированный (*unframed*) поток, т.е. поток E12 (согласно терминологии ИТУ-Т G.703 в редакции 2001 г.) со скоростью 2048 Кбит/с. В этом режиме интерфейс E1 функционирует аналогично модулю IM-703 и может обслуживать только один канал данных.

По умолчанию используется режим no-crc4.

**ВНИМАНИЕ** Если установлен режим framing none, то все нижеперечисленные параметры, а также разбиение потока на группы и коммутация, не имеют смысла.

signaling-mode { disable | cas-mf | ccs-idle }

Использование сигнализации в потоке E1:

- disable Сигнализация не используется. Канальный интервал 16 может быть задействован для передачи данных.
- cas-mf Канальный интервал 16 используется для организации сверхциклов CAS. В битах ABCD для всех каналов передается код-заполнитель, установленный параметром unused-abcd. Все принятые биты ABCD игнорируются.
- ccs-idle В канальном интервале 16 передается октет-заполнитель ccs-idle. По умолчанию сигнализация не используется (режим disable).

alarm-generation { disable | auto-ais | auto-rai }

Выбор режима генерации аварийных сигналов (*alarms*):

- disable Аварийная сигнализация не используется.
  - auto-ais Генерируется сигнал AIS (Alarm Indication Signal).
  - auto-rai Генерируется сигнал RAI (Remote Alarm Indication).
- По умолчанию генерация аварийных сигналов отключена (режим disable).

national-bit { 0 | 1 }

Значение интернационального бита (FAS/NFAS Si bit, канальный интервал 0): 0 или 1. Значение по умолчанию — 1. Параметр не имеет смысла при framing crc4.

additional-bits <0x00...0x1f>

Значение добавочных битов (NFAS Sa bits) в цикловом синхросигнале нечётного цикла (канальный интервал 0). Допустимые значения — 5-битовые маски, задаваемые в виде шестнадцатеричных чисел от 00 до 1F. Младший бит маски (0) соответствует биту Sa4, старший бит (4) — биту Sa8. Значение по умолчанию — 0x1f.

x-bits <0...7> Значение свободных битов в сверхцикловом синхросигнале сверхцикла CAS (цикл 0, канальный интервал 16). Допустимые значения от 0 до 7. Значение по умолчанию — 7. Параметр имеет смысл только при signaling-mode cas-mf.

unused-abcd <0x00...0x0f>

Код-заполнитель для вставки в позиции канальной сигнализации сверхцикла CAS (циклы 1...15, канальный интервал 16). Допустимые значения — шестнадцатеричные числа от 00 до 0F. Младший бит маски (0) соответствует биту A, старший бит (3) — биту D канальной сигнализации. Значение по умолчанию — 0x0b. Параметр имеет смысл только при signaling-mode cas-mf.

ccs-idle <0x00...0xff>

Октет-заполнитель для вставки в канальный интервал 16 при включенной сигнализации CCS. Допустимые значения — шестнадцатеричные числа от 00 до FF. Значение по умолчанию — 0xfc. Параметр имеет смысл только при signaling-mode ccs-idle.

idle-pattern <0x00...0xff>

Октет-заполнитель для вставки в свободные канальные интервалы. Допустимые значения — шестнадцатеричные числа от 00 до FF. Значение по умолчанию — 0x55.

**ПРИМЕЧАНИЕ** Приведенные выше значения параметров по умолчанию выбраны на основании рекомендаций ITU-T G.704 и I.431. Как правило, изменение этих параметров требуется только для совместимости с отдельными специфическими системами.

### §2.3.5. Формирование групп

Каждая группа канальных интервалов в структурированном потоке E1 представляет собой единое целое с точки зрения коммутации и передачи данных. Группа может передавать один и только один канал данных со скоростью  $N \times 64$  Кбит/с (где  $N$  — число канальных интервалов в группе), коммутироваться с другой группой из такого же числа канальных интервалов, или с портом, работающим с такой же скоростью. Формирование групп производится в меню контроллера E1 следующими командами:

data-group <1...31> timeslots { <список> | no }

Создание и удаление группы для передачи данных. Первый параметр — номер группы, который может изменяться от 1 до 31. Каждая группа содержит в себе один или несколько канальных интервалов, используемых для передачи данных, и может быть скомутирована с одним из физических или виртуальных портов устройства. Максимальное количество поддерживаемых групп зависит от типа физического интерфейса.

Список канальных интервалов, включенных в группу, состоит из номеров и диапазонов номеров (через дефис), разделенных запятыми. Пример:

```
data-group 4 timeslots 1,4,7-20,31
```

Другие типы групп в данной версии не поддерживаются.



### §2.3.6. Коммутация E1

Заключительный этап настройки интерфейса E1 состоит в коммутации каждой образованной на нем группы с одним из следующих объектов:

- с синхронным портом устройства (sN, tN) — для обмена данными с канальным и вышестоящими уровнями
- с другой группой канальных интервалов на этом же или другом интерфейсе E1
- с другим физическим интерфейсом

В двух последних случаях данные прозрачно коммутируются из одного интерфейса в другой на физическом уровне, минуя процессор устройства. Настройка коммутации определяется возможностями аппаратной части:

- Для интерфейсных модулей IM-E1-x в структурированном режиме возможна только одна группа для передачи данных (с номером 1), которая всегда жестко скоммутирована с портом sN устройства; номер порта совпадает с номером разъема расширения, в который установлен данный модуль. Ручная настройка коммутации, таким образом, не требуется.

- Для интерфейсных модулей IM-2E1-x в структурированном режиме возможна только одна группа для передачи данных на каждом из двух физических интерфейсов. Эта группа всегда имеет номер 1 и жестко скоммутирована с соответствующим портом sN устройства (модули IM-2E1-x занимают по два разъема расширения).

При работе в режиме *drop-and-insert* (clock-source through) группы данных на двух интерфейсах всегда должны состоять из канальных интервалов с одинаковыми номерами. Остальные интервалы жестко скоммутированы с одного интерфейса на другой.

Если в режиме *drop-and-insert* обмен данными происходит только через один интерфейс, то на другом интерфейсе следует сконфигурировать такую же группу данных и поставить соответствующий порт в `adm-state:down`. Это временное решение, которое будет изменено в ближайших версиях NSG Linux.

Ручная настройка коммутации для данных модулей также не требуется.

Поддержка интерфейсов Channelized E1 и гибкая коммутация между ними будут реализованы в последующих версиях NSG Linux.

**ВНИМАНИЕ** Для работы в режиме *drop-and-insert* два интерфейса должны обязательно иметь одинаковые настройки цикловой структуры (см. п. §2.3.4).

## §2.4. Коммутация пакетов Ethernet

### §2.4.1. Bridge Groups

Устройства под управлением NSG Linux могут работать в режиме прозрачного моста Ethernet между двумя или более физическими локальными сетями Ethernet, виртуальными сетями (VLAN), а также каналами Frame Relay и туннелями GRE, на другой стороне которых также установлены аналогичные мосты. Создание/настройка и удаление мостов производится в меню команд NSG (`config-nsg`)# при помощи команд:

```
bridge <1...255>
no bridge <1...255>
```

При этом первая команда создает мост с указанным номером, если он не существует, и осуществляет вход в меню настройки моста.

Дальнейшая настройка моста производится в меню (`config-bridge-NN`)#, содержащем следующие команды:

**description** Административное описание данного моста — текстовая строка длиной до 255 знаков. Если строка содержит пробелы, ее необходимо заключить в кавычки.

**aging-time <10...1000000>**  
Время жизни MAC-адресов (в секундах) в адресных таблицах моста.

**forward-time <4...200>**  
Задержка перехода, в секундах. (Задержка, которую необходимо выждать перед переходом в новое состояние после изменений сетевой топологии.)

**hello-time <1...10>**  
Интервал между рассылками сообщений Hello (в секундах).

**max-age <6...200>**  
Максимальное время жизни записей в таблицах MAC-адресов.

**priority <0...65535>**  
Приоритет моста в рамках протокола Spanning Tree. Меньшее значение соответствует более высокому приоритету. Мост с наименьшим приоритетом будет выбираться в качестве корневого.

**show-macs** Вывести таблицу MAC-адресов.

**show** Вывести статус и статистику данного моста.

После того, как мост создан, к нему должны быть подключены два или более порта Ethernet (и/или эквивалентных им объектов). Мост будет запоминать MAC-адреса, находящиеся за каждым из этих портов, и перераспределять пакеты Ethernet согласно их MAC-адресам назначения. Подключение и отключение производится следующей командой:

```
bridge-group { <номер> | no }
```

Данная команда используется в меню настройки следующих объектов:

- виртуальных каналов Frame Relay (`config-dlci-N`)#
- физических портов Ethernet (`config-eth0`)#, (`config-sN`)#
- VLAN (`config-vlan-N`)#
- туннелей GRE с инкапсуляцией IP (`config-tunnel-N`)#

При подключении данных объектов к мосту они безусловно отключаются от IP-интерфейса маршрутизатора. При этом все связанные с ними параметры IP обнуляются и становятся недоступными для конфигурирования. Вместо это сам мост, как единое целое, представляет собой IP-интерфейс, подключенный к маршрутизатору, и имеет полный набор параметров IP.

## §2.4.2. Коммутация VLAN

При использовании VLAN коммутация пакетов Ethernet может осуществляться двумя способами:

### а) Коммутация "сеть-на-сеть"

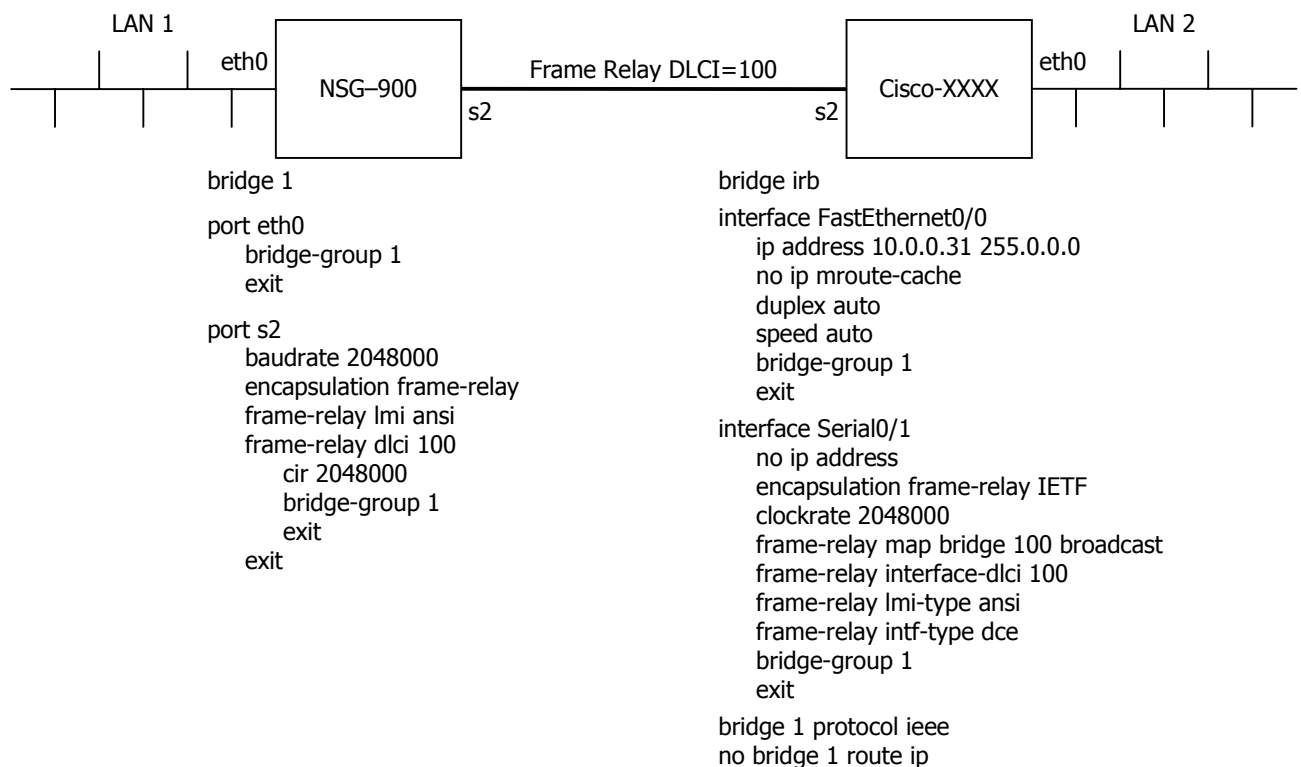
При таком способе коммутации в состав моста могут входить:

- Физические порты Ethernet с инкапсуляцией ethernet
- Индивидуальные VLAN (в т.ч. и расположенные на одном и том же физическом порту)
- Виртуальные каналы Frame Relay и туннели GRE, если их удаленные стороны входят в состав аналогичных мостов (на устройствах NSG, Cisco Systems, или совместимых с ними).

Коммутация осуществляется между индивидуальными физическими или виртуальными сетями Ethernet, непосредственно или через транспорт Frame Relay, GRE. Используемые VLAN могут иметь различные номера, поскольку теги 802.1q через мост не передаются. Во входящем пакете, если он получен из VLAN, тег анализируется (именно по нему определяется принадлежность пакета данному мосту) и затем удаляется. Далее пакеты передаются через мост в обычном формате Ethernet, т.е. без тегов. Если выходная сеть является виртуальной, то при передаче в нее к пакету добавляется тег с номером этой VLAN.

Таким образом, данный способ коммутации обеспечивает обмен данными между любыми двумя хостами, находящимися в физических или виртуальных сетях, подключенных к мосту.

Пример. На одной стороне канала Frame Relay установлено устройство NSG, на другой стороне Cisco. Простая коммутация между двумя физическими сетями Ethernet.



Если на обеих сторонах канала Frame Relay используются устройства NSG, второе устройство настраивается аналогично первому.

### б) Коммутация "порт-на-порт"

При таком способе коммутации в состав моста могут входить только физические порты Ethernet с инкапсуляцией vlan, а также виртуальные каналы Frame Relay и туннели GRE — при условии, что на удаленной стороне канала имеется аналогичный мост. Пакеты передаются через мост в неизменном виде, вместе с тегами 802.1q. Таким образом, адресат может получить пакет только в том случае, если он находится в VLAN с тем же номером, что и отправитель.

Bridge Group в данном случае удобнее рассматривать не один мост, а как совокупность нескольких мостов — по одному на каждый используемый идентификатор VLAN. Пересылка пакетов осуществляется независимо в рамках каждой VLAN, заданной одинаковым номером на нескольких портах Ethernet. Как следствие, необходимо, чтобы VLAN с одним и тем же номером была определена, как минимум, на двух физических портах Ethernet; эти порты могут находиться как на одном устройстве, так и на разных устройствах, соединенных через сеть Frame Relay.

### §2.4.3. Использование коммутатора Ethernet в устройствах NSG–700

Устройства NSG–700 оснащены встроенным коммутатором Fast Ethernet с поддержкой VLAN. Коммутатор подключен одним портом к внутреннему порту `eth0`, находящемуся на процессоре, тремя портами — к физическим интерфейсам `Eth0.1`, `Eth0.2`, `Eth0.3` на корпусе устройства, и одним портом — к разъему расширения 1, если в него установлен модуль IM–SDSL *h/w ver.2*.

Коммутатор и внутренний порт Ethernet в совокупности могут работать в трех конфигурациях:

- Коммутатор и порт в нормальном режиме Ethernet:

```
ethernet-switch normal
port eth0 encapsulation ethernet
```

Это обычный режим работы единой физической сети Ethernet без VLAN.

- Коммутатор в нормальном режиме Ethernet, порт в режиме VLAN:

```
ethernet-switch normal
port eth0 encapsulation vlan
```

Данный режим работы соответствует одной физической сети Ethernet, в которой определено несколько VLAN. Количество и номера VLAN могут быть произвольными.

- Коммутатор и порт в режиме VLAN с номерами 101 ... 104:

```
ethernet-switch vlan
port eth0
  encapsulation vlan
  vlan 101
    ip address ...
  exit
  vlan 102
    ip address ...
  exit
  vlan 103
    ip address ...
  exit
  vlan 104
    ip address ...
  exit
```

В этом случае пакетам Ethernet, поступающим во внешний интерфейс `Eth0.1`, присваивается тег VLAN 101, и в таком виде они поступают во внутренний порт `eth0` (и только в него). Пакеты, исходящие из внутреннего порта `eth0` с тегом VLAN 101, направляются только в интерфейс `Eth0.1`, здесь тег удаляется и пакет передается во внешнюю среду. Аналогичным образом, VLAN 102 и 103 соответствуют интерфейсам `Eth0.2` и `Eth0.3`. VLAN 104 соответствует разъему расширения `s1`, если в него установлен модуль IM–SDSL *h/w ver.2*.

Таким образом, трафик всех этих портов оказывается логически изолирован друг от друга на канальном уровне, и обмен данными между ними возможен только через протокольную обработку в процессоре (с возможным применением туннелирования, VPN и т.п.). При этом каждому интерфейсу VLAN 101 ... 104 необходимо назначить свой IP-адрес.

**ВНИМАНИЕ** Если коммутатор находится в режиме VLAN, то порту `eth0` также необходимо назначить инкапсуляцию VLAN и определить на нем VLAN с указанными выше номерами (в зависимости от используемых внешних портов). Ни при какой другой конфигурации обмен данными не возможен.

## §2.5. Коммутация пакетов Frame Relay и синхронного трафика

### §2.5.1. Коммутация Frame Relay

Для установления постоянного виртуального соединения (PVC) между двумя виртуальными каналами (DLC) сети Frame Relay используется следующая команда в меню настройки DLC:

```
(config-dlci-NN)# route port <имя> <dlci>
```

где NN — номер текущего DLC, <имя> — символическое имя выходного порта (например, s3), <dlci> — номер выходного DLC на этом порту.

Команда является односторонней, т.е. устанавливает только пересылку пакетов из данного DLC в указанный выходной DLC. (Иначе говоря, полученное PVC будет симплексным.) Для того, чтобы установить полноценное двустороннее (дуплексное) PVC, необходимо использовать эту команду два раза — в меню настройки каждого из двух DLC. Пример:

```
(config-nsg)# port s1 frame-relay dlci 17 route port s3 81  
(config-nsg)# port s3 frame-relay dlci 81 route port s1 17
```

В частности, постоянное виртуальное соединение может быть асимметричным, т.е. иметь различную скорость передачи в одну и в другую сторону. Чтобы получить такое соединение, нужно установить для двух DLC различные значения CIR/BC/BE.

Для просмотра имеющихся правил коммутации Frame Relay можно использовать команду `display` в меню портов, параметров Frame Relay или DLC.

Для удаления правил коммутации используется команда

```
(config-dlci-NN)# route no no
```

Разрыв PVC, как и его установление, должен быть выполнен с обеих сторон.

### §2.5.2. Коммутация синхронного трафика

Синхронные порты с инкапсуляцией `raw-hdlc` предназначены для приема/передачи произвольного трафика, представленного пакетами HDLC общего вида (точное определение таких пакетов см. в п.2.2.2). Несколько потоков такого трафика могут быть мультиплексированы в один канал Frame Relay; для этого следует установить PVC между каждым из входных физических портов и соответствующим DLC, например:

```
port s1 encapsulation frame-relay  
port s2 encapsulation raw_hdlc  
port s1 frame-relay dlci 17 route port s2 no  
port s2 route port s1 17
```

Как можно видеть, отличие от коммутации Frame Relay состоит в том, что в команде `route` вместо номера виртуального канала указывается `no`. Аналогичным образом производится удаление PVC:

```
port s1 frame-relay dlci 17 route port no no  
port s2 route port no no
```

Два порта с инкапсуляцией `raw-hdlc` могут быть скомутированы друг на друга:

```
port s1 encapsulation raw_hdlc  
port s2 encapsulation raw_hdlc  
port s1 route port s2 no  
port s2 route port s1 no
```

Устройство с такой конфигурацией де-факто будет работать как преобразователь интерфейсов, синхронный модем, повторитель или устройство доступа к каналу E1 — в зависимости от используемых интерфейсных модулей на обоих портах. Безусловно, для высокопроизводительных устройств NSG-900 и NSG-800 такое употребление является не самым целесообразным, однако этот режим может быть полезен для целей тестирования и диагностики. В частности, синхронный порт может быть замкнут сам на себя, т.е. работать в качестве программного шлейфа:

```
port s1 encapsulation raw_hdlc  
port s1 route port s1 no
```

Все данные, получаемые из линии, отправляются в этом случае обратно в ту же линию.

## §2.6. Сеансовый доступ по протоколам SLIP, PPP и его производным

### §2.6.1. Шаблон IP-интерфейса

Для настройки IP-интерфейсов, которые предполагается использовать с асинхронными портами PPP, SLIP или с протоколами на основе PPP (PPPoE, PPTP), предназначены *шаблоны* (*virtual templates*). Шаблон представляет собой совокупность дополнительных параметров, связанных с установкой физического соединения, аутентификацией, согласованием параметров IP и другими функциями. Эти параметры могут относиться к режиму клиента (т.е. для подключения устройства NSG к удаленной системе), к режиму сервера (т.е. для подключения удаленных пользователей к устройству NSG), или к обоим режимам.

Создание/редактирование и удаление шаблонов производится в меню команд NSG (`config-nsg`)# при помощи команд:

```
virtual-template <1...25>
no virtual-template <1...25>
```

При этом первая команда создает шаблон с указанным номером, если он не существует, и осуществляет вход в меню редактирования шаблона. Ссылка на номер используемого шаблона входит в конфигурацию асинхронного порта с инкапсуляцией PPP или SLIP (см. п.2.2.5), клиента и сервера PPPoE (см. пп. 2.6.3, 2.6.2), туннеля PPTP (см. Часть 4).

В меню настройки шаблона (`config-virtual-template-N`)# (где N — номер шаблона) содержатся следующие основные команды:

`description` <комментарий>

Текстовое описание данного шаблона для удобства администрирования. Если строка содержит пробелы, она должна быть заключена в кавычки. Максимальная длина описания — 255 символов.

`ip address` <ip-адрес>

Установка IP-адреса для создаваемого IP-интерфейса. (Фактический адрес может отличаться от установленного, если использована опция `ppp ipcp accept-address yes`.) Если IP-адрес должен быть получен от удаленной стороны, то значение данного параметра следует установить равным 0.0.0.0.

`ip mtu` <100...1500>

Размер MTU создаваемого интерфейса. По умолчанию для асинхронного порта MTU = 1500. Для интерфейса PPPoE значение данного параметра игнорируется и принудительно устанавливается MTU = 1492, для интерфейса PPTP MTU = 1480.

`keepalive` { no | <0...100> [retry {<1...100> | no } ] }

Проверка целостности соединения. Для порта с инкапсуляцией PPP, а также для протокольных объектов на основе PPP (PPPoE, PPTP и т.п.) используется механизм LCP Echo. Для порта с инкапсуляцией SLIP посылаются пустые пакеты; подробнее об этом механизме см. п.2.2.9. Первый параметр определяет интервал (в секундах) между посылкой контрольных пакетов; если значение параметра равно нулю или no, запросы не посылаются. При этом ответы на входящие запросы отсылаются в любом случае (в т.ч. и при `keepalive 0`).

Второй параметр устанавливает максимальное количество запросов. Если на указанное число запросов подряд не получено ни одного ответа, соединение разрывается. Суммарное время, по истечении которого интерфейс сочтет соединение неработающим и рестартует, равно произведению этих двух параметров. Значение `retry no` показывает, что разрыв соединения не производится, независимо от отсутствия ответов на запросы; такая установка целесообразна, например, если пакеты *keepalive* посылаются с единственной целью предотвратить разрыв соединения на физическом уровне из-за отсутствия трафика (переход сотовых модемов в "спящий" режим и т.п.)

При изменении параметра `keepalive` параметр `retry` автоматически принимает значение no. Таким образом, чтобы использовать механизм зондирования и разрыва соединения, данную команду необходимо вводить полностью.

По умолчанию установлены следующие значения параметров: `keepalive 10 retry no`.

`peer ip address` <ip-адрес>

IP-адрес удаленной стороны. При установлении соединения автоматически создается соответствующая запись в маршрутной таблице.

Этот же адрес используется в случае, если удаленный клиент PPP требует назначить ему динамический IP-адрес, а требуемое значение не определено никакими другими средствами (например, сервером RADIUS в результате аутентификации). Если устройство NSG не должно назначать IP-адрес удаленной стороне, то данный параметр следует установить равным 0.0.0.0.

Если шаблон используется для создания интерфейсов PPPoE сервера, то удаленным клиентам назначаются последовательные IP-адреса, начиная с заданного; максимальное число используемых адресов определяется параметром `pppoe limit` в настройках VPDN-группы (см. п.2.6.2).

Следующая группа команд этого же подменю (`config-virtual-template-N`)# определяет режимы, специфичные для интерфейса PPP.

`ppp connection { permanent | on-demand | passive }`

Режим работы PPP-интерфейса:

- `permanent` Интерфейс стремится поддерживать сеанс PPP постоянно. При разрыве соединения — рестартует и пытается установить его заново. Этот режим установлен по умолчанию.
- `on-demand` Интерфейс устанавливает физическое соединение и сеанс PPP только при наличии пакетов в выходной очереди данного интерфейса.
- `passive` Интерфейс не пытается инициировать сеанс PPP. После рестарта интерфейс обрабатывает сценарий физического соединения (он должен быть задан обязательно) и затем не посылает LCP-запросы, а только ждет запросы от удаленной стороны. Этот режим типичен для сервера PPP-доступа.

`ppp idle-time { <30...86400> | no }`

Тайм-аут неактивности, в секундах. Если в течение указанного времени через интерфейс не передаются данные, соединение разрывается. Опция доступна только при работе в режиме сервера (`connection passive`) или в режиме соединения по требованию (`connection on-demand`). При значении `no` (установлено по умолчанию) разрыв соединения по отсутствию активности не производится.

`ppp session-time { <60...86400> | no }`

Максимальная продолжительность сеанса PPP, в секундах. По истечении указанного времени сеанс принудительно разрывается. Опция доступна только при работе в режиме сервера (`connection passive`) или в режиме соединения по требованию (`connection on-demand`). При значении `no` (установлено по умолчанию) время соединения не ограничено.

`ppp authentication { pap | chap | noauth } { local | radius }`

При работе в режиме сервера: режим аутентификации удаленного клиента при открытии сеанса. Первый параметр определяет протокол аутентификации или отсутствие таковой. Второй параметр определяет способ аутентификации:

- `local` локальный список пользователей (см. п.2.6.5)
- `radius` удаленный сервер RADIUS

`ppp refuse-auth { [pap] [chap] [ms-chap] [ms-chap-v2] | no }`

При работе в режиме клиента: отказываться от выполнения аутентификации по специфическим протоколам, если она запрашивается удаленной стороной. По умолчанию параметр имеет значение `no`, т.е. устройство NSG согласно аутентифицировать себя по любому из перечисленных протоколов. Чтобы сузить этот список, следует запретить конкретные протоколы; в этом случае аутентификация будет производиться только по оставшимся.

`ppp sent-username <имя_пользователя>`

При работе в режиме клиента: в ответ на запрос аутентификации отсылать указанное имя и соответствующий ему пароль из локального списка пользователей (см. п.2.6.5). Данные имя и пароль используются при аутентификации по всем протоколам, не запрещенным командой `ppp refuse-auth`. Если строки с указанным именем пользователя не оказывается в локальном списке пользователей, то запрос на аутентификацию отвергается.

`ppp ipcp accept-address { yes | no }`

При работе в режиме клиента: разрешить использование адреса, присылаемого удаленной стороной, в качестве своего локального адреса. По умолчанию установлено значение `no`.

`ppp ipcp accept-dns { yes | no }`

При работе в режиме клиента: запросить адреса серверов DNS у удаленной стороны. В ответ удаленная сторона может прислать адреса одного или двух серверов. Просмотреть назначенные адреса можно в командной оболочке ОС Linux командой:

```
cat /etc/ppp/resolv.conf
```

и использовать в клиенте/ретрансляторе DNS (см. Часть 3). По умолчанию установлено значение `yes`.

**ПРИМЕЧАНИЕ** На клиентских ПК, работающих через устройство NSG, адреса DNS следует установить вручную. Данное решение является временным. Штатное решение с помощью встроенного клиента и ретранслятора DNS планируется в последующих версиях NSG Linux.

`ppp ipcp dns <ip-адрес>`

При работе в режиме сервера: передавать удаленной стороне указанный адрес DNS. Если значение параметра равно `0.0.0.0`, адрес DNS не передается.

`ppp set-default-route { yes | no } [ metric <1...255>]`

Установка данного интерфейса в качестве маршрута по умолчанию. По умолчанию установлено значение `no`.

Опциональный параметр `metric` позволяет принудительно установить метрику маршрута, определяющую его порядок в списке других маршрутов по умолчанию. По умолчанию маршруты создаются с метрикой 1.

**ВНИМАНИЕ** Указание метрики необходимо, если в системе имеются другие маршруты по умолчанию (как созданные на других PPP-интерфейсах командой `set-default-route`, так и созданные статически командой `ip route ...`). В частности, если интерфейс планируется использовать в качестве резервного, его метрика должна быть больше, чем метрика основного маршрута. Если это правило не соблюдено, то вероятен конфликт, в результате которого не будет создан ни один маршрут, или они будут созданы в неправильном порядке, или все они окажутся неработоспособными.

**ВНИМАНИЕ** Если PPP-интерфейс используется в режиме *dial-up* клиента со следующими настройками:

- Соединение устанавливается по требованию (`connection on-demand`)
- Адреса назначаются удаленной стороной (`ipcp accept-address yes`)

то для него в таблице маршрутизации создается запись с некоторыми фиктивными IP-адресами, необходимая для того, чтобы направить пакеты на интерфейс в то время, когда соединение отсутствует. После установления PPP-соединения она заменяется записью с фактическими адресами.

Если при этом в таблице маршрутизации были явно заданы другие маршруты на такой интерфейс (командой `ip route ...`), то после установления соединения они уничтожаются вместе с фиктивной записью, но никакие новые записи вместо них не создаются. По этой причине использовать в данной ситуации команду `ip route` не следует.

`ppp debug { on | off }`

Включение расширенного режима отладки PPP. По умолчанию установлено значение `off`, т.е. вывод краткого набора сообщений об основных событиях PPP-сеанса.

**ВНИМАНИЕ** На устройствах серии NSG-700 не следует сохранять установку `ppp debug on` в энергонезависимой памяти. В данной версии NSG Linux она может приводить к зависанию командной оболочки на этапе загрузки конфигурации при старте устройства.

`ppp options "<строка>"`

Ввод дополнительных опций PPP, для установки которых в данной версии NSG Linux не предусмотрены отдельные команды. В строку могут быть включены (через пробел) любые опции, предусмотренные для службы PPP (`pppd`) в Linux.

Подробно обо всех возможных опциях см. *man-pages* по `pppd`.

**ПРИМЕЧАНИЕ** Параметры PPP, для которых предусмотрены отдельные команды в настройках шаблона, рекомендуется вводить именно с помощью этих команд.

Созданные шаблоны интерфейсов PPP используются в конфигурации физических асинхронных портов (см. п.2.2.5) и служб PPPoE, PPTP. Для выбора шаблона используются соответствующие команды в меню настройки портов и VPDN-групп, например:

```
(config-nsg)# virtual-template 1
.....
(config-nsg)# port a1
(config-a1)# encapsulation ppp
(config-a1)# virtual-template 1
.....
(config-nsg)# port s2
(config-s2)# physical-layer async
(config-s2)# encapsulation slip
(config-s2)# virtual-template 1
.....
(config-nsg)# vpdn-group 1
(config-vpdn-group)# virtual-template 1
```

Если шаблон интерфейса не задан (значение 0), или указанный шаблон не существует, то PPP-соединение будет устанавливаться, по умолчанию, следующим образом:

- локальный и удаленный адреса назначаются удаленной стороной
- аутентификация не производится



### §2.6.2. Настройка сервера PPPoE

Для работы устройства NSG в качестве сервера PPPoE (PPP-over-Ethernet) необходимо определить виртуальную частную сеть доступа (Virtual Private Dialup Network, VPDN). В данной версии NSG Linux такие сети создаются и удаляются в меню NSG следующим образом:

```
(config-nsg)# vpdn-group <1...25>
(config-nsg)# no vpdn-group <номер>
```

Первая команда создает шаблон с указанным номером, если он не существует, и осуществляет вход в меню редактирования шаблона. Одновременно в устройстве могут работать несколько серверов PPPoE, если каждый из них относится к своему порту Ethernet (или эквивалентному ему объекту).

Настройка сервера VPDN производится в меню (config-vpdn-group)#.

protocol pppoe

Установить протокол туннелирования PPPoE. Другие названия протоколов, которые могут упоминаться в подсказке, зарезервированы для дальнейшего использования.

pppoe name <имя>

Установить имя сервера PPPoE. По умолчанию данный параметр имеет пустое значение; в этом случае созданный PPPoE сервер получает имя, указанное в параметре hostname устройства.

pppoe limit <число>

Установить максимальное число одновременных сеансов работы пользователей.

virtual-template <номер>

Использовать для данной VPDN шаблон интерфейса (virtual-template) с указанным номером.

Запуск и останов сервера PPPoE производится в меню порта, имеющего инкапсуляцию Ethernet (или эквивалентного ему объекта):

```
vpdn-group { <номер> | no }
```

Запустить/остановить сервер PPPoE с указанным номером на данном порту.

**ПРИМЕЧАНИЕ** После изменения какого-либо из параметров PPPoE (в подменю vpdn-group, virtual-template) следует рестартовать сервер PPPoE:

```
pppoe vpdn-group no
pppoe vpdn-group <номер>
```

В следующих версиях это будет выполняться автоматически. При перезапуске сервера PPPoE все пользовательские сеансы, установленные на данный момент, разрываются.

### §2.6.3. Настройка клиента PPPoE

Клиент PPPoE может работать поверх как физической, так и виртуальной сети Ethernet. Меню портов Ethernet с инкапсуляцией ethernet и меню суб-интерфейсов VLAN на портах Ethernet с инкапсуляцией vlan содержит команды:

no pppoe Полное отключение клиента PPPoE. При этом разрывается соединение, если оно существует в данный момент, и все параметры клиента принимают значения по умолчанию (в т.ч. client disable).

pppoe Переход в подменю настройки клиента PPPoE.

Дальнейшая настройка производится в меню (config-pppoe)#, которое содержит следующие пункты:

```
client { enable | disable }
```

Разрешить клиенту PPPoE на данном порту подключиться к серверу, или запретить подключение. По умолчанию соединение запрещено. Если для клиента настроены некоторые параметры, отличные от значений по умолчанию, то при установке client disable соединения запрещаются (и разрывается существующее), но настроенные значения параметров сохраняются — в отличие от команды no pppoe в вышестоящем меню.

Если клиент PPPoE запущен, в системе создается интерфейс с именем родительского интерфейса и суффиксом ".0" например:

```
eth0.0    PPPoE клиент на порту eth0
s1.0      PPPoE клиент на порту s1 с установленным модулем расширения IM-ET10F
eth0.101.0 PPPoE клиент на VLAN 101, работающей на физическом порту eth0
```

На этом же родительском интерфейсе, параллельно с пакетами PPPoE, могут отправляться и приниматься обычные пакеты Ethernet или VLAN, соответственно. Формат, в котором будет отправлен пакет ("обычный" или PPPoE), целиком определяется маршрутизацией.

В системе могут одновременно работать несколько клиентов PPPoE, при условии, что каждый из них работает на отдельном интерфейсе Ethernet или VLAN.

**ПРИМЕЧАНИЕ** В данной версии NSG Linux клиенты PPPoE могут работать на любом числе интерфейсов Ethernet, но только на одном суб-интерфейсе VLAN.

**server** <имя>

Указание имени PPPoE-сервера, к которому нужно подключиться. Имя может состоять из букв и цифр и вводится в двойных кавычках или без них. Большие и маленькие буквы в имени различаются. По умолчанию имя сервера не указано (указана пустая строка " "). В этом случае клиенту разрешается подключиться к любому серверу, доступному в данной локальной сети.

**virtual-template** <номер>

Использовать для данного клиента шаблон (virtual-template) с указанным номером.

**ppp-log** { previous | current }

Просмотр журнала сеанса PPPoE. Ключевое слово *previous* выводит журнал последней завершенной попытки, *current* — текущей попытки. Во втором случае, повторяя ввод команды, можно проследить ход сеанса по мере его выполнения.

В ходе сеанса PPPoE весь вывод направляется также на сервер Syslog, если он включен. Для этого необходимо предварительно перейти в командную оболочку ОС Linux и выполнить команду:

```
syslogd
```

либо включить *syslogd* при помощи сценария, см. Часть 1. Просмотреть файл системного журнала можно также в командной оболочке ОС Linux следующей командой:

```
cat /var/log/messages
```

**discovery** Поиск доступных серверов PPPoE в данной локальной сети. Команда выполняется в течении 5 секунд. Пример вывода:

```
nsg(config-port-eth0)# pppoe discovery
```

```
Access-Concentrator: LinuxRH9
```

```
Got a cookie: 43 19 cf a8 b3 0b 07 2f ce ea 76 00 e1 14 d5 1e db 20 00 00
```

```
AC-Ethernet-Address: 00:0c:6e:41:51:b5
```

```
-----
```

```
Access-Concentrator: NSGbasicSW
```

```
Got a cookie: e8 49 c1 49 5b 94 0a d3 8c 21 d8 ef 99 5c b4 95
```

```
AC-Ethernet-Address: 00:09:56:10:05:97
```

```
-----
```

```
Access-Concentrator: CISCO
```

```
Got a cookie: cb 13 b7 11 c0 6a 92 ff fb bf 42 ea 3f 8f 5f 03
```

```
AC-Ethernet-Address: 00:02:16:66:7b:40
```

```
-----
```

```
Access-Concentrator: NSGLinux
```

```
Got a cookie: 96 79 58 ff 45 3e 8f 1b f8 af 75 3e 5b 20 90 b6 00 00 00 76
```

```
AC-Ethernet-Address: 00:09:56:12:00:fe
```

```
-----
```

Здесь в первой строке указаны имена серверов (Access-Concentrator), в третьей строке — их MAC-адреса. В случае, если ни один из серверов не отозвался на процедуру *discovery*, будет выведено сообщение:

```
pppoe: Timeout waiting for PADO packets
```

Параметр *discovery* является командой и не сохраняется в энергонезависимой памяти.

**show** Просмотр статуса и статистики соединения PPPoE. Пример вывода:

```
nsg(config-port-s1)# pppoe show
```

```
15: s1.0: <POINTOPOINT,MULTICAST,NOARP,UP> mtu 1500 qdisc pfifo_fast qlen 3
```

```
link/ppp
```

```
inet 17.0.0.2 peer 17.0.0.1/32 scope global s1.0
```

```
RX: bytes  packets  errors  dropped  overrun  mcast
```

```
1062997  2715    0       0        0        0
```

```
RX errors: length  crc    frame  fifo    missed
```

```
0         0      0      0      0      0
```

```
TX: bytes  packets  errors  dropped  carrier  collsns
```

```
758       19      0       0        0        0
```

```
TX errors: aborted  fifo    window  heartbeat
```

```
0         0      0       0        0
```

Здесь в первой строке важно состояние интерфейса (ключевое слово "UP"). В третьей показаны адреса — локальный (inet) и удаленный (peer) — полученные от сервера в результате согласования.

### §2.6.4. Настройка клиента PPTP

Протокол PPTP предназначен для туннелирования пакетов PPP через сеть IP при помощи механизма GRE. Настройка клиента PPTP описана в Части 4 данного Руководства.

### §2.6.5. Настройка локального списка пользователей

Список имен и паролей пользователей для аутентификации редактируется из меню (config-nsg)# с помощью следующих команд:

username <имя> password <пароль>

Создать пользователя и установить пароль для него.

no username <имя>

Удалить пользователя.

Пары "имя–пароль" создаются и уничтожаются одновременно для всех способов аутентификации. Данный список используется двояким образом:

— При аутентификации устройства NSG как клиента на удаленной системе.

— При аутентификации удаленных клиентов, подключающихся к устройству NSG, если в шаблоне PPP-интерфейса указан способ аутентификации local.

### §2.6.6. Настройка клиента RADIUS

Клиент RADIUS позволяет производить аутентификацию, авторизацию и учет работы пользователей с помощью централизованного сервера при доступе по протоколу PPP и его производным (PPTP, PPPoE). Управление клиентом RADIUS производится в меню (config-nsg)#.

radius Включение клиента RADIUS и переход в меню его настройки.

no radius Выключение клиента RADIUS. (Эквивалентно radius host 0.0.0.0 и установке остальных параметров в значения по умолчанию.)

Дальнейшая настройка производится в меню (config-radius)#:

host <ip-адрес>

Определение IP-адреса удаленного сервера RADIUS.

auth-port <udp-port>

Номер порта UDP, используемого для аутентификации на сервере. Значение по умолчанию — 1812.

acct-port <udp-port>

Номер порта UDP, используемого для отсылки статистики на сервер. Значение по умолчанию — 1813.

key <строка>

Ключ для шифрования аутентификационной информации, используемый сервером и клиентом RADIUS. Значение параметра — строка символов, используемая для шифрования. Эта же строка должна быть задана в конфигурации сервера RADIUS для данного клиента. Строка не может содержать пробелы.

timeout <1...86400>

Время ожидания (в секундах) перед повторной передачей пакета, если на предыдущую посылку не получен ответ от сервера. Значение по умолчанию — 60.

retry <1...1000>

Количество повторных попыток аутентификации (в случае отсутствия ответов). Десятичное значение. Значение по умолчанию — 10. Если за указанное число попыток ответ от сервера не получен, клиенту будет отказано в подключении к устройству NSG.

**ПРИМЕЧАНИЕ** При аутентификации через RADIUS, если пользователь определил имя устройства (командой hostname), следует ввести вручную соответствие указанного имени и одного из IP-адресов устройства. Для этого нужно добавить соответствующую строку в файл /etc/hosts.

## §2.7. Просмотр состояния и статистики портов

Команда отображения статистики имеется в меню большинства сетевых устройств (в терминах Linux), присутствующих в системе, включая:

- физические порты
- виртуальные каналы Frame Relay
- виртуальные сети VLAN
- туннели IP-over-IP (GRE), IP-over-X.25, PPTP
- мосты
- интерфейсы клиента PPPoE

Исключением в данной версии являются динамически создаваемые интерфейсы сервера PPPoE.

Формат команды:

show

show statistics checkpoint 0

Вывод состояния, числа изменений состояния UP/DOWN, и текущей статистики объекта, начиная от момента старта устройства или данного объекта. Две вышеуказанные команды являются синонимами. Формат вывода частично варьируется в зависимости от типа объекта.

Для объектов, представляющих собой IP-интерфейсы, в статистике учитывается только IP-трафик.

Для портов учитывается трафик канального уровня (в т.ч. заголовки канального уровня, пакеты *keepalive*, пакеты LCP и др.). Дополнительные байты физического уровня (бит-стаффинг и др.) в статистике не учитываются.

show statistics checkpoint <1..15> [ set | unset ]

Вывод статистики относительно указанной контрольной точки. Опциональные команды *set* и *unset* устанавливают и удаляют контрольную точку, соответственно.

Нулевой точкой для сбора статистики, которая присутствует всегда, является момент старта устройства (или объекта). Помимо нее, для каждого объекта может быть установлено до 15 дополнительных контрольных точек.

show statistics checkpoint all unset

Удаление всех установленных контрольных точек.

## Приложение 2—А. Режимы синхронизации последовательных интерфейсов

### §2—А.1. Синхронные интерфейсы DCE/DTE

Физические синхронные интерфейсы DCE/DTE аппаратуры NSG, как и ряда других производителей, оснащены специальными разъемами, для которых нет общепринятой схемы распайки выводов (в данном случае это DBH–26f). Поэтому они не имеют какого-либо определенного аппаратного типа. Такие интерфейсы всегда используются со специальными кабелями, имеющими на одном конце фирменный разъем, а на другом — один из стандартных разъемов с распайкой выводов DCE или DTE. Удобно считать, что интерфейсный модуль вместе с кабелем представляет собой единое целое и только как целое имеет тот или иной тип.

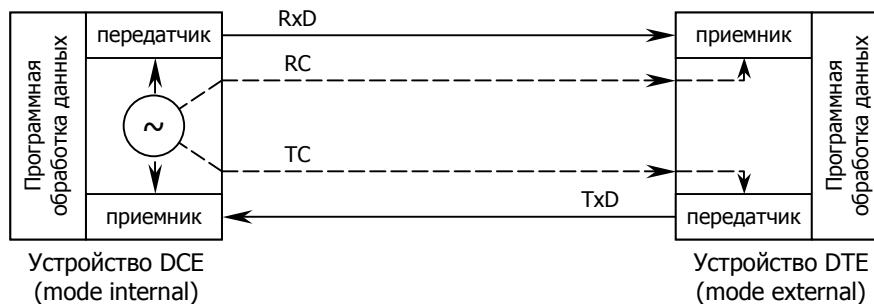
Для соединения с другими синхронными устройствами используются соответствующие интерфейсные кабели и переходники NSG:

Интерфейс	Кабель DTE		Кабель DCE	
	Обозначение	Разъем на кабеле для подключения к DCE	Обозначение	Разъем на кабеле для подключения к DTE
V.24	CAB–V24/D25/MT	DB–25m	CAB–V24/D25/FC	DB–25f
V.35	CAB–V35/M34/MT	M–34m	CAB–V35/M34/FC	M–34f
V.35 TTC	CAB–V35/M34/MT/T	M–34m	CAB–V35/M34/FC/T	M–34f
RS–530	CAB–530/D25/MT	DB–25m	CAB–530/D25/FC	DB–25f
X.21	CAB–X21/D15/MT	DB–15m	CAB–X21/D15/FC	DB–15f

Все кабели DTE оснащены разъемом "папа" для подключения к DCE и имеют в названии индекс "M", а все кабели DCE — разъемом "мама" и имеют индекс "F". Это правило соблюдается в устройствах большинства производителей, представленных на рынке. Однако оно не является абсолютным, поэтому в каждом конкретном случае следует обратить внимание на документацию для каждого устройства. Именно для этой цели в документации, как правило, полностью приводится назначение контактов разъемов.

Для соединения двух устройств NSG друг с другом напрямую (без модемов) необходимо использовать два кабеля, DTE и DCE, соединенных последовательно (см. пример ниже в данном разделе). Подробное описание разъемов и распайка кабелей NSG приведены в документе: *Мультипротокольные маршрутизаторы и коммутаторы пакетов NPS–7e, NSG–500, NX–300, NSG–800. Модули расширения и интерфейсные кабели.*

В классической схеме "терминал-модем" роль источника синхронизации всегда возлагается на устройство DCE. Его приемник и передатчик синхронизируются от внутреннего генератора, а приемник и передатчик устройства DTE — от сигналов Receive Clock (RC), Transmit Clock (TC) интерфейса. Таким образом, скорость передачи в линии определяется устройством DCE в соответствии с его параметром baudrate. Для устройства DTE параметр baudrate является справочным.



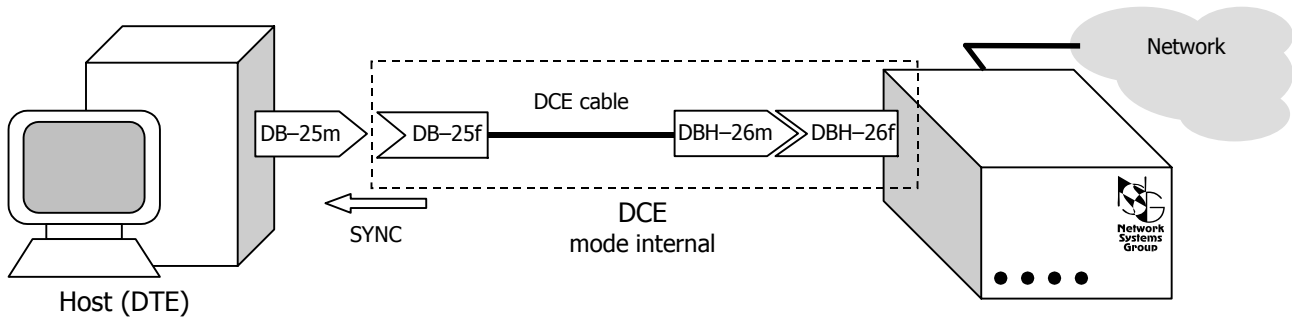
Синхронизация в классической схеме DTE/DCE

Сетевые устройства, в отличие от модемов и прикладных компьютеров, не имеют столь однозначной роли. В каждом отдельно взятом соединении могут как служить источниками/получателями трафика (т.е. являться устройствами DTE), так и передавать трафик других устройств (т.е. выполнять функции DCE). Если два сетевых устройства соединены друг с другом напрямую, то всегда одно из них играет роль DTE, другое — DCE.

Аппаратный тип порта NSG однозначно определяется, как уже говорилось выше, подключенным к нему кабелем. Программная настройка синхронизации (параметр mode) должна соответствовать выбранному аппаратному типу. Два порта WAN, соединенные друг с другом синхронной линией, должны иметь асимметричные значения параметра mode — internal для одного и external для другого. Значение mode internal соответствует интерфейсу с аппаратным типом DCE, mode external — интерфейсу с аппаратным типом DTE.

**а) Работа в качестве DCE:**

Для подключения ПК или сервера к устройству NSG используется кабель DCE (например, DBH-26m/DB-25f). В этом случае физический интерфейс устройства NSG вместе с кабелем имеет аппаратный тип DCE.

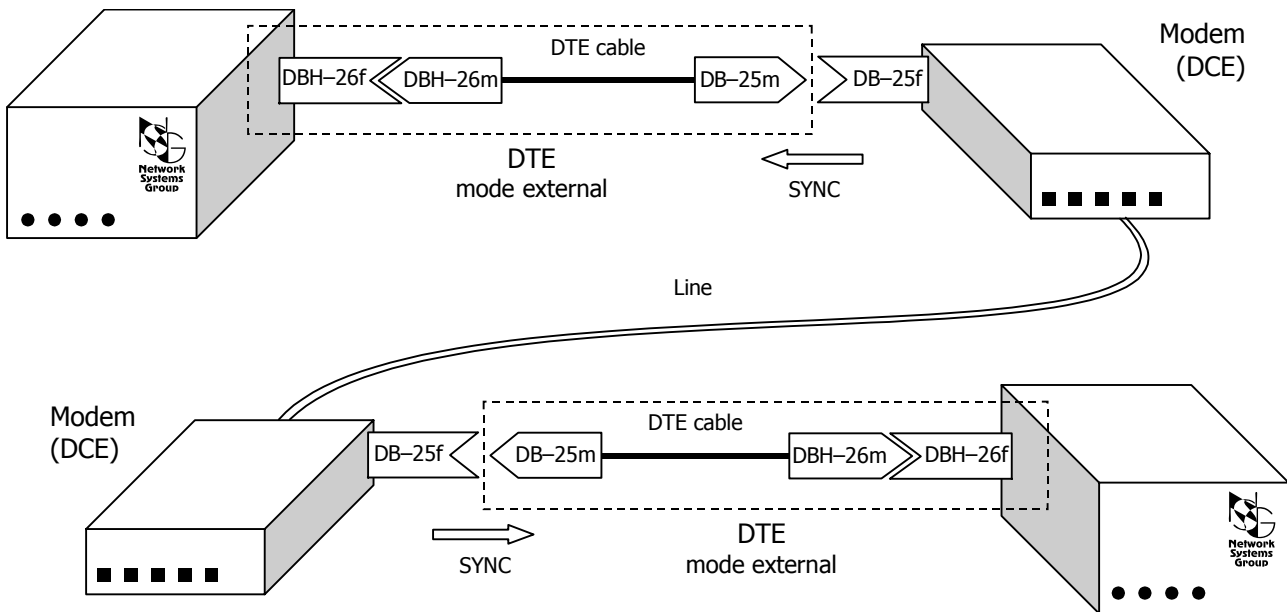


Подключение сервера (DTE) к устройству NSG по синхронному каналу

**ВНИМАНИЕ** При непосредственном соединении с хостом по синхронному каналу источником синхронизации является устройство NSG и для данного физического интерфейса устанавливается mode int. Хост должен быть сконфигурирован для синхронизации от внешнего источника.

**б) Работа в качестве DTE:**

Для подключения устройства NSG к модему или другому устройству передачи данных используется кабель DTE (например, DBH-26m/DB-25m). В этом случае физический интерфейс устройства NSG вместе с кабелем имеет аппаратный тип DTE.

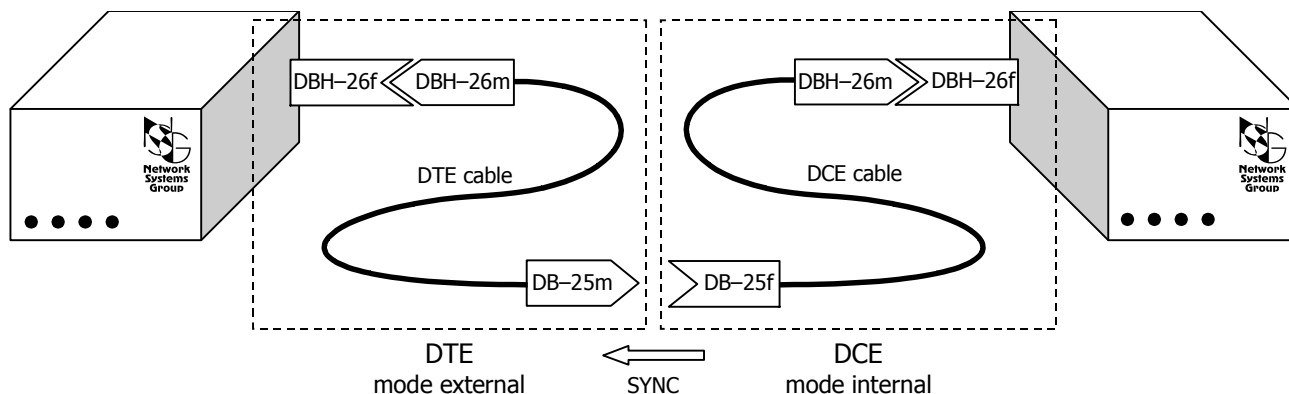


Модемное соединение между двумя устройствами NSG

**ВНИМАНИЕ** В данной схеме источником синхронизации для каждого из устройств NSG является соединенный с ним модем. В этом случае для физического интерфейса NSG устанавливается mode ext. (Обратите внимание на направление стрелок SYNC на рисунках.)

### в) Соединение двух устройств:

Для прямого соединения двух устройств NSG друг с другом необходимо использовать два кабеля: DTE и DCE. Интерфейс одного из устройств вместе с кабелем принимает аппаратный тип DTE, другого — DCE. Аналогичным образом соединяются два устройства различных производителей, например, NSG и Cisco.



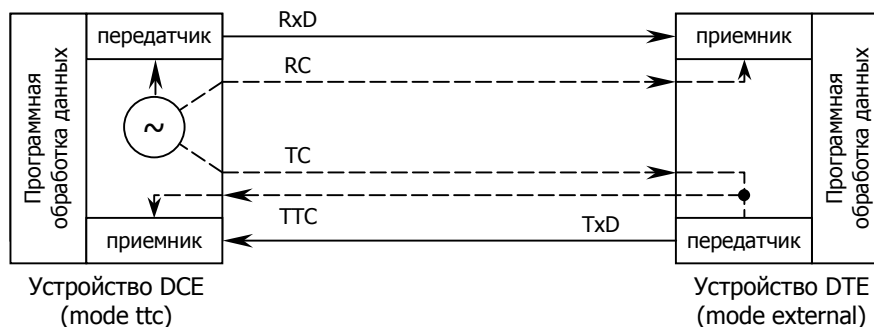
Прямое соединение между двумя устройствами NSG

**ВНИМАНИЕ** При прямом синхронном соединении двух устройств NSG для одного из них необходимо установить `mode external`, для другого — `mode internal` в строгом соответствии с типами подключенных к ним кабелей.

**ВНИМАНИЕ** Прямое соединение часто используется для того, чтобы проверить работоспособность некоторой конфигурации в лабораторных условиях, прежде чем устанавливать устройства на удаленных площадках и соединять их при помощи модемов. В этом случае при переносе устройств на удаленную площадку, помимо смены кабеля, необходимо проследить, чтобы для бывшего устройства DCE был изменен источник синхронизации: с `mode internal` на `mode external`.

## §2–А.2. Режим TTC

Режим TTC позволяет частично компенсировать искажения сигналов, возникающие при передаче по длинным кабелям, и благодаря этому увеличить скорость и/или дальность соединения. В этом режиме, в отличие от исходной схемы DCE/DTE, приемник устройства DCE синхронизируется не от внутреннего генератора, а от сигнала TTC (Transmit Clock from DTE), возвращенного устройством DTE.



Синхронизация в режиме TTC

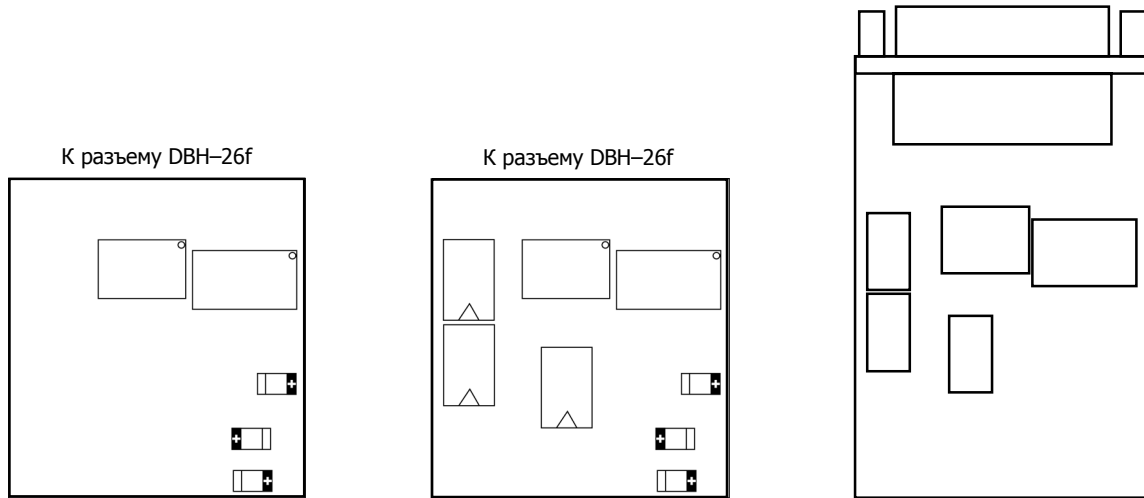
Использовать режим TTC возможно только в том случае, если он поддерживается всеми тремя составляющими последовательного интерфейса, а именно:

- Устройство DTE возвращает полученный сигнал TC обратно в виде сигнала TTC
- Устройство DCE допускает отдельную синхронизацию приемника и передатчика
- В кабеле, соединяющем два устройства, имеется дополнительный провод или пара проводов (в зависимости от типа интерфейса) для передачи сигнала TTC. Как правило, режим TTC используется для соединения двух устройств на больших расстояниях (десятки метров), поэтому кабели изготавливаются заказчиком по месту.

В устройствах NSG для работы в режиме TTC DCE, если этот режим поддерживается аппаратно, необходимо установить `mode ttc` вместо `mode internal`. Для работы в режиме TTC DTE специальных программных настроек не требуется — если в интерфейсном модуле предусмотрен сигнал TTC, он присутствует на выходе всегда; вопрос только в том, задействованы ли эти контакты в кабеле.

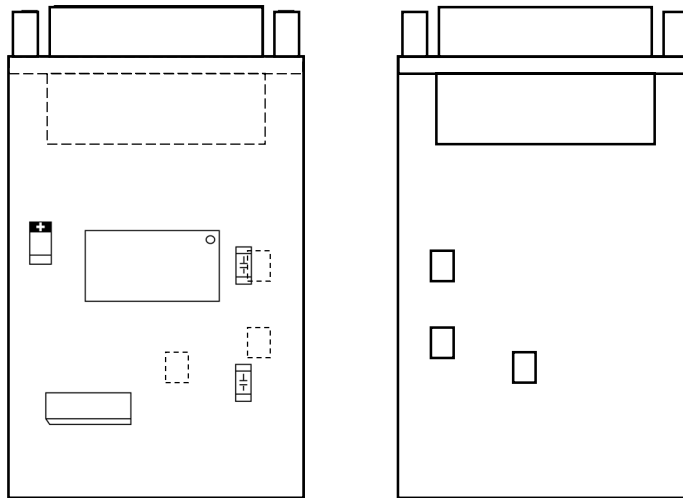
В режиме TTC могут использоваться следующие типы интерфейсных модулей NSG:

**Интерфейс V.24** — модули IM-V24, IM-V35, IM-V35-2 выпуска до 2003 г. (с 2 или 5 микросхемами, соответственно). Могут работать только в качестве TTC DTE, при условии изготовления специального кабеля. Модули более поздних выпусков (с одной большой микросхемой) сигнал TTC не обеспечивают.



Интерфейсные модули обр. до 2003 г.: IM-V24, IM-V35 (вид сверху), IM-V35-2 (вид снизу)

**Интерфейс V.35** — модули IM-V35-2 выпуска 2003 г. и позже (1 большая микросхема на верхней стороне, 3 маленькие планарные на нижней). Могут работать в качестве TTC DTE и TTC DCE, при условии использования специальных кабелей с индексом "Т" (CAB-V35/M34/MT/T и CAB-V35/M34/FC/T, соответственно), либо изготовления кабелей по месту. Гарантированная скорость в этом режиме — до 8 Мбит/с при длине кабеля 15 м.



Интерфейсный модуль IM-V35-2 обр. 2003 г. (вид сверху, вид снизу)

Если для подключения использован кабель DCE без сигнала TTC (CAB-V35/M34/FC), то установка `mode ttc` равнозначна `mode internal`.

Некоторые из более ранних модификаций IM-V35, IM-V35-2 также поддерживают работу в одном или обоих режимах TTC, однако привести для них исчерпывающую классификацию затруднительно ввиду их многочисленности.

**Интерфейс RS-530** — модуль IM-530 может использоваться в качестве TTC DTE, при условии изготовления специального кабеля.

**Интерфейс X.21** — режим TTC не предусмотрен спецификацией данного интерфейса.

Подробная информация о всех типах физических интерфейсов NSG, их модификациях и соответствующих им интерфейсных кабелях приведена в документе:

*Мультипротокольные маршрутизаторы и коммутаторы пакетов NSG. Модули расширения и интерфейсные кабели.*



## Приложение 2–Б. Примеры настройки PPP и PPPoE

### §2–Б.1. Согласование IP-адресов

Во всех нижеприведенных примерах предполагается, для наглядности, что на одной стороне асинхронного соединения находится устройство NSG с NSG Linux, на другой — с базовым программным обеспечением. Приведены конфигурации и состояния обоих устройств.

Пример 1. Каждая система знает свой адрес и не знает адреса другой системы.

```
!
nsg
  virtual-template 2
    ip address 15.0.0.1
!
IP:02 ADM:UP  NAME:""
  IADR:15.0.0.2 MASK:255.0.0.0 BRC:1 MTU:1500 ACCT:0 NAT:NO
  TY:PPP PO:2 SL:NO PAPER:0 PAPA:NO CHAPR:0 CHAPA:NO SCRIPT:0
  AC:NO PC:NO VJ:NO VJC:NO BSDC:NO KEEP:0 HOLD:0
  AM:00000000 ACCL:NO DOD:NO DTR:1 RNAME:"" RADR:0.0.0.0
```

В результате каждое устройство работает со своим адресом:

```
ppp0 Link encap:Point-Point Protocol
      inet addr:15.0.0.1 P-t-P:15.0.0.2 Mask:255.255.255.255
      UP POINTOPOINT RUNNING NOARP MULTICAST MTU:1500 Metric:1
      RX packets:13 errors:0 dropped:0 overruns:0 frame:0
      TX packets:13 errors:0 dropped:0 overruns:0 carrier:0
      collisions:0 txqueuelen:3
      RX bytes:70 (70.0 iB) TX bytes:186 (186.0 iB)

015.255.255.255 255.255.255.255 015.000.000.002 00 00 999 0000
015.000.000.000 255.255.255.255 015.000.000.002 00 00 999 0000
015.000.000.002 255.255.255.255 015.000.000.002 00 00 999 0000
015.000.000.000 255.000.000.000 015.000.000.002 00 02 999 0000
```

Пример 2. Система Linux не знает свой адрес и готова принять его от удаленной системы. Система с базовым ПО знает адрес для Linux (параметр RADR).

```
!
nsg
  virtual-template 2
    ppp ipcp accept-address yes
!
IP:02 ADM:UP  NAME:""
  IADR:15.0.0.2 MASK:255.0.0.0 BRC:1 MTU:1500 ACCT:0 NAT:NO
  TY:PPP PO:2 SL:NO PAPER:0 PAPA:NO CHAPR:0 CHAPA:NO SCRIPT:0
  AC:NO PC:NO VJ:NO VJC:NO BSDC:NO KEEP:0 HOLD:0
  AM:00000000 ACCL:NO DOD:NO DTR:1 RNAME:"" RADR:15.0.0.3
```

В результате система Linux работает на адресе, который прислала система NSG:

```
ppp0 Link encap:Point-Point Protocol
      inet addr:15.0.0.3 P-t-P:15.0.0.2 Mask:255.255.255.255
      UP POINTOPOINT RUNNING NOARP MULTICAST MTU:1500 Metric:1
      RX packets:6 errors:0 dropped:0 overruns:0 frame:0
      TX packets:6 errors:0 dropped:0 overruns:0 carrier:0
      collisions:0 txqueuelen:3
      RX bytes:48 (48.0 iB) TX bytes:76 (76.0 iB)
```

Аналогичный результат будет, если Linux знает свой адрес, но готов принять адрес от удаленного абонента:

```
!
nsg
  virtual-template 2
    ip address 15.0.0.1
    ppp ipcp accept-address yes
!
```

Пример 3. Linux знает свой адрес и знает адрес удаленного абонента. Удаленный абонент не знает никаких адресов и готов принять (ACCL:YES) адрес, динамически назначаемый системой Linux. (В базовом ПО не разрешено оставлять IADR в значении 0.0.0.0, поэтому устанавливается фиктивный адрес 192.168.0.1.)

```
!
nsg
  virtual-template 2
    ip address 15.0.0.1
    peer ip address 15.0.0.4
!
IP:02 ADM:UP NAME:""
IADR:192.168.0.1 MASK:255.0.0.0 BRC:1 MTU:1500 ACCT:0 NAT:NO
TY:PPP PO:2 SL:NO PAPR:0 PAPA:NO CHAPR:0 CHAPA:NO SCRIPT:0
AC:NO PC:NO VJ:NO VJC:NO BSDC:NO KEEP:0 HOLD:0
AM:00000000 ACCL:YES DOD:NO DTR:1 RNAME:"" RADR:0.0.0.0
```

В результате система с базовым ПО работает на адресе, полученном от системы Linux:

```
ppp0 Link encap:Point-Point Protocol
inet addr:15.0.0.1 P-t-P:15.0.0.4 Mask:255.255.255.255
UP POINTOPOINT RUNNING NOARP MULTICAST MTU:1500 Metric:1
RX packets:19 errors:0 dropped:0 overruns:0 frame:0
TX packets:19 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:3
RX bytes:124 (124.0 iB) TX bytes:240 (240.0 iB)

015.000.000.004 255.255.255.255 192.168.000.001 00 00 999 0000
015.000.000.001 255.255.255.255 015.000.000.001 01 02 999 0000
```

Пример 4. Система Linux знает свой адрес и знает адрес удаленного абонента. Удаленный абонент знает свой адрес и не готов принять (ACCL:NO) адрес, назначаемый системой Linux. Остальная конфигурация аналогична предыдущему примеру.

В результате система с базовым ПО работает на своем адресе (192.168.0.1) и знает удаленного абонента (Linux) как 15.0.0.1.

```
015.000.000.001 255.255.255.255 015.000.000.001 01 02 999 0000
192.255.255.255 255.255.255.255 192.168.000.001 00 00 999 0000
192.000.000.000 255.255.255.255 192.168.000.001 00 00 999 0000
192.000.000.000 255.000.000.000 192.168.000.001 00 02 999 0000
192.168.000.001 255.255.255.255 192.168.000.001 00 00 999 0000
```

При этом для устройства Linux удаленная система NSG все равно остается под адресом 15.0.0.4:

```
ppp0 Link encap:Point-Point Protocol
inet addr:15.0.0.1 P-t-P:15.0.0.4 Mask:255.255.255.255
UP POINTOPOINT RUNNING NOARP MULTICAST MTU:1500 Metric:1
RX packets:19 errors:0 dropped:0 overruns:0 frame:0
TX packets:19 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:3
RX bytes:124 (124.0 iB) TX bytes:240 (240.0 iB)
```

## §2–Б.2. Аутентификация

**ПРИМЕЧАНИЕ** Аутентификация в общем случае не связана с распределением адресов (см. выше), поэтому в качестве примеров выбраны некоторые распространенные случаи.

**Пример 1.** Система Linux работает как сервер, знает свой адрес, адрес подключаемого абонента, и требует от него его аутентификации при подключении. Метод аутентификации — локальный, протокол — PAP.

```
!
nsg
  username basile password p0upKiNe
  virtual-template 2
    ip address 15.0.0.1
    peer ip address 15.0.0.4
    ppp authentication pap local
!

IP:02 ADM:UP  NAME:"basile"
      IADR:192.168.0.1 MASK:255.0.0.0 BRC:1 MTU:1500 ACCT:0 NAT:NO
      TY:PPP PO:2 SL:NO PAPR:0 PAPA:YES CHAPR:0 CHAPA:NO SCRIPT:0
      AC:NO PC:NO VJ:NO VJC:NO BSDC:NO KEEP:0 HOLD:0
      AM:00000000 ACCL:YES DOD:NO DTR:1 RNAME:"" RADR:0.0.0.0

Manager: w s ip:2
Manager: d x pap
PAP Passwords Table
#      <client>      <server>      <secret>      [<IP address> ...]
~~~~~
PAP:01  basile          *          p0upKiNe
```

Существенные детали:

- На устройстве Linux локальный список пользователей формируется командой `username ... password ...`
- Для устройства с базовым ПО имя интерфейса (имя клиента) установлено параметром `NAME` интерфейса. Пароль берется из таблицы PAP (см. команду `A X PAP...`). Аутентификация себя как клиента разрешена (параметр `PAPA:YES`).
- Для подключения с пользовательского ПК имя и пароль указываются в окне `dialup`-подключения.

**Пример 2.** Система Linux работает как сервер аналогично предыдущему примеру, метод аутентификации — удаленный через RADIUS, протокол — CHAP.

```
!
hostname NSG-Linux
!
nsg
  radius host 10.0.0.2
  radius auth-port 1812
  radius acct-port 1813
  radius key nsg
  radius retry 3
  radius timeout 10
  virtual-template 2
    ip address 15.0.0.1
    peer ip address 15.0.0.4
    ppp authentication chap radius
!

IP:02 ADM:UP  NAME:"mike"
      IADR:192.168.0.1 MASK:255.0.0.0 BRC:1 MTU:1500 ACCT:0 NAT:NO
      TY:PPP PO:2 SL:NO PAPR:0 PAPA:NO CHAPR:0 CHAPA:YES SCRIPT:0
      AC:NO PC:NO VJ:NO VJC:NO BSDC:NO KEEP:0 HOLD:0
      AM:00000000 ACCL:YES DOD:NO DTR:1 RNAME:"NSG-Linux" RADR:0.0.0.0
```

Существенные детали:

- Параметр RNAME содержит имя удаленной системы, к которой осуществляется подключение. В данном примере имя NSG-Linux должно быть установлено в качестве hostname в системе Linux.
- Настройки системы Linux для работы с требуемым сервером RADIUS производятся до настроек virtual-template. Удаленный сервер RADIUS должен быть сконфигурирован и запущен.
- Для подключения с пользовательского ПК следует указать "использовать безопасный/зашифрованный пароль" (в терминах клиента удаленного доступа Microsoft Windows).
- При установке в Linux своего hostname необходимо добавить в файл /etc/hosts разрешение этого имени на IP-адрес одного из своих интерфейсов. (Под этим адресом устройство Linux должно значиться в базе удаленного сервера RADIUS в качестве клиента.)

Пример 3. Система Linux работает как клиент, которого просят аутентифицировать себя по протоколу PAP. После этого он сам назначает IP-адрес в соединении PPP.

```
!
hostname linux
!
nsg
  username linux password xunil
  virtual-template 2
    ip address 15.0.0.1
    peer ip address 15.0.0.4
    ppp sent-username linux
!
IP:02 ADM:UP NAME:""
  IADR:192.168.0.1 MASK:255.0.0.0 BRC:1 MTU:1500 ACCT:0 NAT:NO
  TY:PPP PO:2 SL:NO PAPR:1 PAPA:NO CHAPR:0 CHAPA:NO SCRIPT:0
  AC:NO PC:NO VJ:NO VJC:NO BSDC:NO KEEP:0 HOLD:0
  AM:00000000 ACCL:YES DOD:NO DTR:1 RNAME:"" RADR:0.0.0.0
Manager: d x pap
PAP:02 linux * xunil
```

Существенные детали:

- По умолчанию настройки virtual-template предполагают разрешение отвечать на запросы аутентифицировать себя, используя протоколы MS-CHAP v2, MS-CHAP, CHAP и PAP.
- Если в virtual-template имя пользователя не указано, то на запрос аутентификации отправляется строка (имя, пароль), в которой имя пользователя совпадает со значением hostname (в данном случае linux). Если строки с таким именем не оказывается в локальном списке пользователей, то запрос на аутентификацию отвергается.

Пример 4. Система Linux работает как клиент, которого просят аутентифицировать себя по протоколу CHAP.

```
!
hostname linux
!
nsg
  username linux password xunil
  virtual-template 2
    ip address 15.0.0.1
    peer ip address 15.0.0.4
!
IP:02 ADM:UP NAME:""
  IADR:192.168.0.1 MASK:255.0.0.0 BRC:1 MTU:1500 ACCT:0 NAT:NO
  TY:PPP PO:2 SL:NO PAPR:0 PAPA:NO CHAPR:1 CHAPA:NO SCRIPT:0
  AC:NO PC:NO VJ:NO VJC:NO BSDC:NO KEEP:0 HOLD:0
  AM:00000000 ACCL:YES DOD:NO DTR:1 RNAME:"" RADR:0.0.0.0
Manager: d x chap
CHAP:02 linux * xunil
```

**Пример 5.** Система Linux работает как клиент, которого просят аутентифицировать себя. Удаленная система предлагает сначала CHAP — Linux его отвергает (ppp chap refuse), затем предлагает PAP — Linux соглашается.

```
!
nsg
  virtual-template 2
    ip address 15.0.0.1
    peer ip address 15.0.0.4
    ppp refuse-auth chap
!
IP:02 ADM:UP  NAME:""
IADR:192.168.0.1 MASK:255.0.0.0 BRC:1 MTU:1500 ACCT:0 NAT:NO
TY:PPP PO:2 SL:NO PAPR:1 PAPA:NO CHAPR:1 CHAPA:NO SCRIPT:0
AC:NO PC:NO VJ:NO VJC:NO BSDC:NO KEEP:0 HOLD:0
AM:00000000 ACCL:YES DOD:NO DTR:1 RNAME:"" RADR:0.0.0.0
```

**ПРИМЕЧАНИЕ** Аналогичной командой можно запретить отвечать на запросы PAP, но продолжать отвечать на запросы CHAP и других протоколов аутентификации (ppp refuse-auth pap).

**Пример 6.** Взаимная аутентификация абонентов. Linux запрашивает аутентификацию по CHAP и проверяет локально. Удаленная система NSG — запрашивает по PAP и проверяет локально.

```
!
hostname linux
!
nsg
  username linux password xunil
  username mike password dan
  virtual-template 2
    ip address 15.0.0.1
    peer ip address 15.0.0.4
    ppp authentication chap local
!
IP:02 ADM:UP  NAME:"mike"
IADR:192.168.0.1 MASK:255.0.0.0 BRC:1 MTU:1500 ACCT:0 NAT:NO
TY:PPP PO:2 SL:NO PAPR:1 PAPA:NO CHAPR:0 CHAPA:YES SCRIPT:0
AC:NO PC:NO VJ:NO VJC:NO BSDC:NO KEEP:0 HOLD:0
AM:00000000 ACCL:YES DOD:NO DTR:1 RNAME:"linux" RADR:0.0.0.0
Manager: d x pap
PAP Passwords Table
PAP:02 linux * xunil
Manager: d x chap
CHAP:01 mike * dan
```

### §2–Б.3. Дополнительные возможности протокола PPP

Передача клиенту адреса DNS. Установка MTU. Периодическое зондирование канала при помощи механизма LCP Echo.

```
!
nsg
  virtual-template 2
    ip mtu 1400
    ip address 15.0.0.1
    keepalive 15 retry 4
    peer ip address 15.0.0.4
    ppp lcp dns 194.67.1.8
!
```

Если ответ на запросы LCP Echo не получен 4 раза подряд (т.е. соединение неработоспособно более 1 мин.), то соединение будет разорвано, интерфейс рестартован, и предпринята попытка установить соединение заново.

**ПРИМЕЧАНИЕ** Параметр `keepalive` определяет только интервал между отсылками своих запросов `lcp-echo-request`. На приходящие запросы `lcp-echo-request` порт отправляет ответы `lcp-echo-reply` в любом случае.

## §2–Б.4. Подключение клиентов по протоколу PPPoE

Устройство под управлением NSG Linux используется в качестве сервера PPPoE.

Пример 1. Подключение клиентов без аутентификации.

```
!  
nsg  
  virtual-template 1  
    ip address 14.0.0.1  
    peer ip address 14.0.0.10  
    exit  
  vpdn-group 1  
    protocol pppoe  
    pppoe limit 120  
    pppoe name mike2  
    virtual-template 1  
    exit  
  port eth0  
    vpdn-group 1  
!
```

Пример 2. Подключение клиентов с аутентификацией PAP (локально)

```
!  
nsg  
  username Zorro password Zorro  
  username user1 password pass1  
  username mike password pmike  
  virtual-template 1  
    ip address 14.0.0.1  
    peer ip address 14.0.0.10  
    ppp authentication pap local  
    exit  
  vpdn-group 1  
    protocol pppoe  
    pppoe limit 120  
    pppoe name mike2  
    virtual-template 1  
    exit  
  port eth0  
    vpdn-group 1  
!
```

Пример 3. Подключение клиентов с аутентификацией CHAP на удаленном RADIUS-сервере.

```
!  
nsg  
  radius host 10.0.0.2  
  radius auth-port 1812  
  radius acct-port 1813  
  radius key nsg  
  radius timeout 10  
  radius retry 3  
  virtual-template 1  
    ip address 14.0.0.1  
    peer ip address 14.0.0.10  
    ppp authentication chap radius  
    exit  
  vpdn-group 1  
    protocol pppoe  
    pppoe limit 120  
    pppoe name mike2  
    virtual-template 1  
  exit  
  port eth0  
    vpdn-group 1  
!
```

## §2–Б.5. Подключение к серверам PPPoE

Настройка устройства NSG в качестве клиента PPPoE. IP-адрес назначается динамически, аутентификация в данных примерах не используется.

```
!
nsg
  port eth0
    pppoe client enable
    pppoe server <имя_сервера>
  exit
```

Настройка устройства Cisco в качестве сервера PPPoE:

```
!
vpdn enable
!
vpdn-group 1
  accept-dialin
  protocol pppoe
  virtual-template 2
  pppoe limit per-mac 20
  local name CISCO
!
interface FastEthernet0/0
  pppoe enable
!
interface Virtual-Template 2
  ip address 16.0.0.1 255.0.0.0
  peer default ip address pool TEST
!
ip local pool TEST 16.0.0.2 16.0.0.20
```

Настройка устройства NSG с ПО NSG Linux в качестве сервера PPPoE:

```
!
nsg
  virtual-template 1
    peer ip address 19.0.0.2
    ip address 19.0.0.1
  exit
  vpdn-group 1
    pppoe limit per-mac 20
    protocol pppoe
    virtual-template 1
    local name NSGLinux
  port eth0
    vpdn-group 1
  exit
exit
```

Настройка устройства NSG с базовым ПО в качестве сервера PPPoE:

```
S P PO:00 TY:ETH ADM:UP IF:TP MODE:AUTO SP:100000000 NAME:"" ADDR:00.09.56.10.05.97.
S P ET:01 PO:0 TY:PPP NAME:"RTEMS" IP:ALL
S P IP:00 ADM:UP NUM:01 NAME:""
S P IP:01 ADM:UP NAME:"NSGbasic" IADR:17.0.0.1 MASK:255.0.0.0 TY:PPP PO:AUTO RADR:17.0.0.2
```

Настройка Linux Red Hat 9 (пакет Roaring Penguin PPPoE Version 3.5) в качестве сервера PPPoE:

```
pppoe-server -I eth0 -C Linux -L 18.0.0.1 -R 18.0.0.2
Содержимое обязательного файла /etc/ppp/pppoe-server-options
# PPP options for the PPPoE server
# LIC: GPL
lcp-echo-interval 10
lcp-echo-failure 2
```

