



**Мультипротокольные  
маршрутизаторы  
NSG  
Программное обеспечение NSG Linux**

**Руководство пользователя  
Часть 1  
Общесистемная конфигурация**

Версия программного обеспечения 1.0 build 1

Обновлено 14.09.2007

Москва 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

Первая часть содержит общее описание системы, описание параметров, относящихся к устройству в целом, а также руководство по установке и замене программного обеспечения. Кроме того, в данной части приведены практические указания по использованию командного языка системы и настройка агента SNMP.

Вопросы индивидуальной настройки физических портов и организации трафика средствами канального уровня (Ethernet bridging, VLAN, коммутация пакетов Frame Relay, организация PPP-доступа) рассмотрены в Части 2. 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 под управлением NSG Linux 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 1. Общесистемная конфигурация

§1.1. Введение .....	4
§1.1.1. Назначение и общая характеристика системы.....	4
§1.1.2. Структура программного обеспечения.....	4
§1.1.3. Документация .....	5
§1.1.4. Функциональные возможности.....	5
§1.2. Начало работы с системой .....	7
§1.2.1. Подключение к устройству через консольный порт .....	7
§1.2.2. Подключение к устройству по Telnet .....	8
§1.2.3. Подключение к устройству по сети X.25 .....	8
§1.2.4. Структура командного меню.....	9
§1.2.5. Правила ввода команд и параметров.....	10
§1.2.6. Особенности команд меню (config-nsg)# .....	13
§1.2.7. Завершение работы с устройством .....	13
§1.2.8. Обозначения и соглашения.....	14
§1.3. Общесистемные настройки .....	15
§1.3.1. Идентификация аппаратной части (одноплатные устройства) .....	15
§1.3.2. Идентификация ведомых плат в многоплатных устройствах .....	16
§1.3.3. Установка имени устройства.....	17
§1.3.4. Парольная защита устройства.....	17
§1.3.5. Установка системного времени.....	17
§1.3.6. Запуск и остановка ресурсоемких системных служб.....	17
§1.3.7. Использование служб SSH .....	18
§1.3.8. Сохранение конфигурации .....	19
§1.3.9. Перезагрузка устройства.....	19
§1.3.10. Просмотр и резервирование текущей конфигурации .....	19
§1.3.11. Восстановление заводской конфигурации .....	19
§1.3.12. Сценарий автозапуска и выполнение команд Linux.....	20
§1.3.13. Настраиваемая светодиодная индикация .....	21
§1.4. SNMP-управление .....	23
§1.4.1. Настройка агента SNMP .....	23
§1.4.2. Настройка SNMP community .....	24
§1.5. Установка и модернизация NSG Linux .....	25
§1.5.1. Аппаратные требования и файлы ПО.....	25
§1.5.2. Установка расширения энергонезависимой памяти.....	26
§1.5.3. Компоненты NSG Linux для различных аппаратных конфигураций .....	27
§1.5.4. Сценарии установки и модернизации NSG Linux .....	29
§1.5.5. Подготовка к установке U-Boot: вход в меню системного загрузчика.....	30
§1.5.6. Установка U-Boot через порт Fast Ethernet .....	31
§1.5.7. Установка U-Boot через консольный порт .....	33
§1.5.8. Установка и модернизация NSG Linux средствами U-Boot.....	35
§1.5.9. Установка и модернизация NSG Linux средствами Revision Utility.....	36
§1.5.10. Переустановка загрузчика U-Boot средствами U-Boot .....	38
§1.5.11. Резервирование и восстановление конфигурации.....	39
§1.5.12. Использование устройств NSG-900 с базовым программным обеспечением NSG .....	40
§1.6. Участие пользователей в развитии программы NSG Linux .....	41
Приложение 1–А. Поддерживаемые стандарты и спецификации .....	42
Приложение 1–Б. Команды загрузчика U-Boot.....	44
Приложение 1–В. Аварийная переустановка U-Boot на NSG-700.....	50
Приложение 1–Г. Корректировка команд при переходе на новую версию программного обеспечения .....	52

## §1.1. Введение

### §1.1.1. Назначение и общая характеристика системы

Программное обеспечение NSG Linux предназначено для мультипротокольных маршрутизаторов NSG и реализует современные технологии IP-сетей, такие как VPN, VLAN, протоколы динамической маршрутизации RIP2, OSPF, BGP, механизмы QoS, туннелирование, PPPoE, коммутацию канального уровня Ethernet (bridging). Основное назначение устройств NSG состоит в построении сетей коммутации пакетов на основе протоколов IP, Frame Relay, X.25 и территориально-распределенных сетей Ethernet.

Программное обеспечение NSG Linux является штатным для следующих серий маршрутизаторов NSG:

- NSG-900 на основе процессора Motorola MPC862
- NSG-700 на основе процессора Atmel AT91RM9200 180 МГц
- NSG-1000 на основе процессоров VIA Technologies с архитектурой x86

а также может использоваться на устройствах NSG-800 на основе процессоров Motorola MPC855T/860, при наличии основной энергонезависимой памяти (Flash ROM) объемом 4 МБ и расширенной энергонезависимой памяти (модуль Disc-on-Chip или FLEX). Требования к аппаратным платформам и процедура установки NSG Linux подробно рассмотрены в разделе 1.5.

**ПРИМЕЧАНИЕ** Расширенный набор функциональных возможностей X.25 и Frame Relay, включая редко используемые стандартные механизмы и фирменные расширения NSG, реализован в базовом программном обеспечении NSG для устройств серий NSG-500, NX-300, NSG-800. Программное обеспечение от серии NSG-800 может также использоваться на аналогичных устройствах серии NSG-900.

В состав системы входят как программные модули, свободно распространяемые на принципах Open Source, так и собственные разработки компании NSG, некоторые из которых могут являться коммерческими продуктами. Программное обеспечение, в соответствии с общепринятой практикой, поставляется в виде "как есть" (*as is*).

### §1.1.2. Структура программного обеспечения

Программное обеспечение NSG Linux представляет собой интегрированный комплекс на основе ОС Linux, предназначенный для выполнения задач маршрутизации и коммутации пакетов IP, Frame Relay, X.25 на аппаратной платформе устройств NSG. В состав комплекса входят следующие основные программные компоненты:

- Ядро Linux, необходимым образом доработанное и скомпилированное для работы на процессорах Motorola PowerPC, ARM9 либо x86, в зависимости от типа устройства.
- Драйверы для интерфейсных модулей и других специфических аппаратных компонент NSG, интегрированные в ядро.
- Набор прикладных пакетов и приложений, предназначенных для маршрутизации/коммутации пакетов, фильтрации, приоритизации трафика и решения других задач. Наиболее существенными из этих компонент являются IPtables, Zebra, FreeSWAN, BusyBox. Большинство прикладных компонент доработано специалистами NSG для корректной работы в составе единого программно-аппаратного комплекса, с общими средствами конфигурирования и мониторинга.
- Обработчики протоколов X.25 и Frame Relay, разработанные либо доработанные компанией NSG.
- Командный интерпретатор vtysh, входящий в состав пакета Zebra.
- Расширения vtysh, разработанные специалистами NSG. С помощью этих расширений производится конфигурирование низкоуровневых драйверов NSG и приложений, не относящихся к пакету Zebra.

Настройка устройств, работающих под управлением NSG Linux, производится при помощи интерфейса командной строки, который обеспечивается vtysh. Предлагаемый им язык команд приближен к популярному набору команд компании Cisco Systems (однако не идентичен ему). Благодаря этому, знание операционной системы Linux *не является* обязательным для администрирования устройств NSG. Значительная часть команд является общей для многих моделей маршрутизаторов различных производителей и не требует дополнительного изучения; остальные, как правило, следуют логике и синтаксису де-факто сложившегося стандарта. Развитая система подсказок способствует быстрому самообучению пользователей.

Пользователи, обладающие необходимыми знаниями в области операционной системы Linux, могут конфигурировать устройства средствами ОС Linux при помощи командной оболочки ash, входящей в состав пакета Linux-утилит BusyBox. При этом для ряда программных компонент доступны расширенные варианты и режимы конфигурации, выходящие за пределы основного набора команд.

Кроме того, пользователи, знакомые с ОС Linux, могут рассматривать маршрутизаторы NSG как универсальную Linux-машину (со специфическими аппаратными возможностями). В частности, пользователь может устанавливать на эту машину дополнительные программные компоненты, доступные в исходных кодах или разработанные самостоятельно.

### §1.1.3. Документация

Данное Руководство пользователя описывает пользователя режимы и параметры конфигурации, реализованные в основной командной оболочке NSG Linux. Руководство не является исчерпывающим и описывает преимущественно команды, которые являются специфическими для оборудования NSG, а также особенности использования NSG Linux на аппаратном обеспечении NSG. В дополнение к данному Руководству, пользователю рекомендуется ознакомиться, по мере необходимости, со следующими документами:

Подробное описание пакета Zebra, являющегося составной частью NSG Linux, представлено в документе:

*Kunihiro Ishiguro. GNU Zebra. A routing software package for TCP/IP networks.*

Расширенные возможности маршрутизации и управления трафиком, предоставляемые ОС Linux, рекомендуются к использованию только опытными пользователями. Эти возможности подробно описаны в документе:

*Bert Hubert et al. Linux Advanced Routing & Traffic Control HOWTO.*

Оба документа доступны на FTP-сервере компании NSG по адресу:

<ftp://ftp.nsg.net.ru/pub/nsg-linux/doc/>

Документация по другим продуктам, использованным при разработке NSG Linux, доступна на Web-сайтах их разработчиков, соответственно. Описания команд и основных приложений ОС Linux имеются в общей документации по ОС Linux. Наиболее распространенным форматом для их исчерпывающего описания являются так называемые страницы руководств — *man pages* — входящие в состав большинства дистрибутивов Linux и доступные на многих интернет-ресурсах, например:

<http://www.opennet.ru/man.shtml>

### §1.1.4. Функциональные возможности

В данной версии NSG Linux поддерживаются следующие типы физических интерфейсов:

- Традиционные последовательные интерфейсы DTE/DCE:  
V.24 (RS-232), V.35, V.36 (RS-449), RS-485, RS-530, RS-530A, X.21
- Встраиваемые модемы для физических линий:  
SHDSL-bis, SHDSL, SDSL, IDSL, MDSL 400/768/1168, SRM
- Встраиваемые аналоговые модемы ТФОП:  
V.92, V.34
- Встраиваемые модемы для сотовых сетей:  
GSM/GPRS/EDGE (900/1800 МГц)  
CDMA 2000 1x/EV-DO (450 МГц)
- Структурированные интерфейсы сетей цифровой плезиохронной иерархии (PDH)  
Fractional E1, E1 *drop-and-insert*
- Неструктурированные интерфейсы сетей цифровой плезиохронной иерархии (PDH)  
E0\* (64 Кбит/с), E12\* (2048 Кбит/с)
- Интерфейсы проводных и беспроводных локальных сетей:  
Ethernet 10Base-T, 10/100Base-T, 100Base-F  
Bluetooth 1.1 (2,4 ГГц)
- Интерфейс USB 2.0 Full Speed (12 Мбит/с, при наличии программной поддержки подключаемых устройств в ядре Linux, либо драйверов, доступных в исходных кодах)
- Консольный порт

Поддержка интерфейсов Channelized E1, сигнализации E1 PRI и ISDN BRI планируется в ближайших версиях. Поддержка интерфейсных модулей IM-C1И, контроллеров дискретного ввода-вывода, а также внешних USB-устройств сторонних производителей будет реализована по мере необходимости.

---

\* Терминология согласно рекомендации ИТУ-T G.703 в редакции 2001 г.; в предыдущих редакциях — G.703.1 и G.703.6, соответственно. Иногда именуется также DS0 и просто G.703, соответственно.

В данной версии NSG Linux реализованы следующие программные возможности:

### Стек TCP/IP

- Маршрутизация: BGP, OSPF, RIP2, RIP, static
- Протоколы канального уровня: Ethernet, Cisco-HDLC, PPP, Multilink PPP\*, SLIP, Frame Relay
- Фильтрация пакетов
- NAT
- QoS (traffic shaping, traffic policing, ToS/DiffServ)
- Балансировка исходящего трафика между несколькими соединениями
- Статистика по IP-интерфейсам
- Множественные IP-адреса для IP-интерфейсов
- ARP проху и static ARP\*
- Ping и traceroute

### Приложения TCP/IP

- FTP клиент\*
- TFTP клиент\*
- Telnet клиент, сервер, Reverse Telnet
- SSH/SSL сервер, клиент\*, Reverse SSH\*
- DHCP сервер, ретранслятор
- DNS клиент
- SNMP клиент\*
- NetFlow агент
- Принт-сервер

### Ethernet

- Работа в режиме прозрачного моста (IEEE 802.1d)
- Протокол Spanning Tree (IEEE 802.1d)
- VLAN (IEEE 802.1q)
- Мост Ethernet-over-Frame Relay и Ethernet-over-IP (Cisco-совместимые Bridge Groups)
- Мост/удаленный порт Ethernet-over-SDSL

### Virtual Private Networking (VPN)

- IPsec
- Согласование протоколов и ключей: автоматическое (IKE), ручное
- Аутентификация IPsec: MD5, SHA-1

### Стек X.25

- X.25 SVC
- Логические типы: DTE, DCE
- Маршрутизация вызовов X.25: фиксированная, по адресу источника, по адресу назначения, по полю данных
- Фильтрация вызовов X.25

### Стек Frame Relay

- Frame Relay PVC
- Управляющие протоколы: CCITT, ANSI
- Механизмы QoS: CIR/BC/BE
- Логические типы: DTE, DCE
- Инкапсуляция IP: IETF, Cisco

### Мультипротокольные возможности

- IP-over-Frame Relay
- IP-over-X.25
- PPP-over-Ethernet (сервер, клиент)
- PPTP (клиент)
- GRE tunneling (IP-over-IP, Ethernet bridge-over-IP, Frame Relay-over-IP, Generic HDLC-over-IP)
- Туннелирование IP-in-IP (Linux-compatible)
- X.25-over-TCP/IP (XOT)
- Мультиплексирование синхронного пакетного трафика во Frame Relay

### Аутентификация, авторизация и статистика

- Статистика портов и интерфейсов
- Сценарии аутентификации (клиент)
- PAP (клиент и сервер)
- CHAP (клиент и сервер)
- MS-CHAP (клиент)
- MS-CHAP v2 (клиент)
- Локальная авторизация (сервер)
- RADIUS (клиент)
- Сбор статистики в формате NetFlow

### Средства управления и диагностики

- Консольный порт
- Telnet
- SSH
- X.25 PAD
- Cisco-подобная командная оболочка
- Командная оболочка Linux
- SNMP v1, MIB II (только стандартная MIB)
- Программный шлейф для кольцевого теста синхронных каналов WAN
- Сервер syslog\*
- Резервирование и восстановление конфигурации
- Локальная модернизация программного обеспечения через TFTP и Xmodem
- Настраиваемая светодиодная индикация

**ПРИМЕЧАНИЕ** Функциональные возможности, отмеченные звездочкой (\*), в данной версии NSG Linux доступны только средствами командной оболочки Linux. Помимо них, поддерживаются также другие функции и приложения, имеющиеся в ОС Linux.

## §1.2. Начало работы с системой

### §1.2.1. Подключение к устройству через консольный порт

Все устройства NSG, предназначенные для работы под управлением NSG Linux, оснащены выделенным или разделяемым консольным портом. (Если консольный порт является разделяемым, то по умолчанию он настроен для работы именно в режиме консольного.) Для получения доступа к конфигурации устройства через консольный порт необходимо:

1. Присоединить COM-порт компьютера к консольному порту устройства кабелем, входящим в комплект устройства.
2. Запустить на ПК терминальную программу и установить в ней следующие параметры для данного COM-порта:

Baud Rate	9600
Parity	None
Data Bits	8
Stop Bits	1
Flow control	None

3. Нажать клавишу Enter. Система выводит подсказку:

```
nsg login:
```

В качестве имени пользователя следует ввести `nsg`. После этого будет запрошен пароль; по умолчанию пароль не установлен и не запрашивается. В целях безопасности рекомендуется установить уникальный секретный пароль для каждого устройства. (Подробнее об установке паролей см. п.1.3.4.)

При правильном вводе пароля системное приглашение принимает вид:

```
nsg>
```

В данном режиме доступны только команды диагностики и просмотра статистики. Для перехода в режим конфигурирования (привилегированный режим) следует ввести команду `enable`; после этого системное приглашение принимает вид:

```
nsg#
```

Пользователи, обладающие необходимыми знаниями в области операционной системы Linux, могут конфигурировать устройства средствами командной оболочки ОС Linux. Для входа в режим консольного терминала Linux необходимо при входе в систему ввести имя пользователя `root` и пустой пароль. (Впоследствии пароль рекомендуется изменить средствами ОС Linux.) Системное приглашение в этом случае имеет вид:

```
#
```

Для перехода из командной оболочки Linux в основное командное меню следует после системного приглашения Linux (`#`) ввести команду `vtsh`. Для перехода из основного меню в командную оболочку Linux следует ввести команду `start-shell` в меню `enable` (см. ниже).

**ВНИМАНИЕ** Для подключения терминала к консольному порту устройств NSG необходимо использовать кабель, прилагаемый к устройству. Консольные порты устройств различных серий имеют различное назначение контактов (и, в некоторых случаях, тип разъема) и несовместимы друг с другом. Типы консольных кабелей NSG и их распиновка приведены в Руководствах пользователя по соответствующим устройствам.

**ПРИМЕЧАНИЕ** Выделенный консольный порт устройств NSG-900 не поддерживает аппаратное управление потоком и предназначен преимущественно для целей управления. Использование этого порта для передачи данных не предусмотрено.

**ПРИМЕЧАНИЕ** В результате некоторых команд на консоль могут выводиться дополнительные служебные сообщения, которые оказываются на экране *после* приглашения к вводу следующей команды. В этом случае следующую команду можно набирать после сообщения, либо нажать клавишу Enter для повторного вывода приглашения.

### §1.2.2. Подключение к устройству по Telnet

Для доступа по Telnet устройство предварительно должно быть сконфигурировано для работы в сети IP. Первоначальная конфигурация устройства осуществляется только средствами консольного порта. В простейшем случае, для доступа к устройству по локальной сети Ethernet через встроенный порт Fast Ethernet необходимо назначить этому интерфейсу IP-адрес и маску подсети при помощи команд:

```
nsg> enable
nsg# configure terminal
nsg(configure)# nsg
nsg(configure-if)# port eth0 ip address <ip-адрес>/<длина маски>
```

Если доступ к устройству осуществляется через другой порт (например, Frame Relay), необходимо полностью настроить этот порт, включая IP-адрес и маску подсети. Описание команд, используемых для настройки отдельных компонент, приведено в Части 2 и Части 3 данного Руководства.

После того, как устройство сконфигурировано и к нему возможен доступ по сети IP (в частности, успешно проходит *ping*), для обращения к нему необходимо набрать на удаленном компьютере команду *telnet* либо ввести в клиенте Telnet команду *open*, после которых следует IP-адрес, назначенный устройству NSG. Дальнейшая работа полностью аналогична управлению через консольный порт. Подсказки для ввода имени пользователя и пароля, а также системные приглашения для ввода команд полностью идентичны в обоих случаях.

**ПРИМЕЧАНИЕ** Одновременно к устройству могут иметь доступ несколько пользователей по Telnet, X.25 и/или через консольный порт. При этом только один из них может работать в режиме конфигурирования устройства посредством основной командной оболочки; остальным разрешается только просматривать параметры конфигурации и статистику работы устройства.

### §1.2.3. Подключение к устройству по сети X.25

Для доступа с терминала, подключенного к удаленному PAD, устройство предварительно должно быть сконфигурировано для работы в сети X.25. Первоначальная конфигурация устройства осуществляется только средствами консольного порта. В простейшем случае, для доступа к устройству по сети X.25 необходимо сконфигурировать синхронный порт X.25, назначить устройству некоторый адрес X.121 и создать маршрут, оканчивающийся непосредственно на данном устройстве, например:

```
nsg> enable
nsg# configure terminal
nsg(config)# nsg
nsg(config-nsg)# card s1 im-v35
nsg(config-nsg)# port s1
nsg(config-port-s1)# baudrate ...
nsg(config-port-s1)# mode ...
nsg(config-port-s1)# encapsulation x25
nsg(config-port-s1)# lapb ...
nsg(config-port-s1)# x25 ...
nsg(config-port-s1)# exit
nsg(config-nsg)# x25 route add destination 1234567 local
```

Подробно о конфигурации параметров физического (*baudrate*, *mode*), канального (*lapb*) и пакетного (*x25*) уровней X.25 см. Часть 2 и Часть 5 данного Руководства, о маршрутизации — Часть 5. На всех промежуточных узлах сети X.25 должен быть описан маршрут для пакетов с адресом назначения (*called address*) 1234567, ведущий на данное устройство.

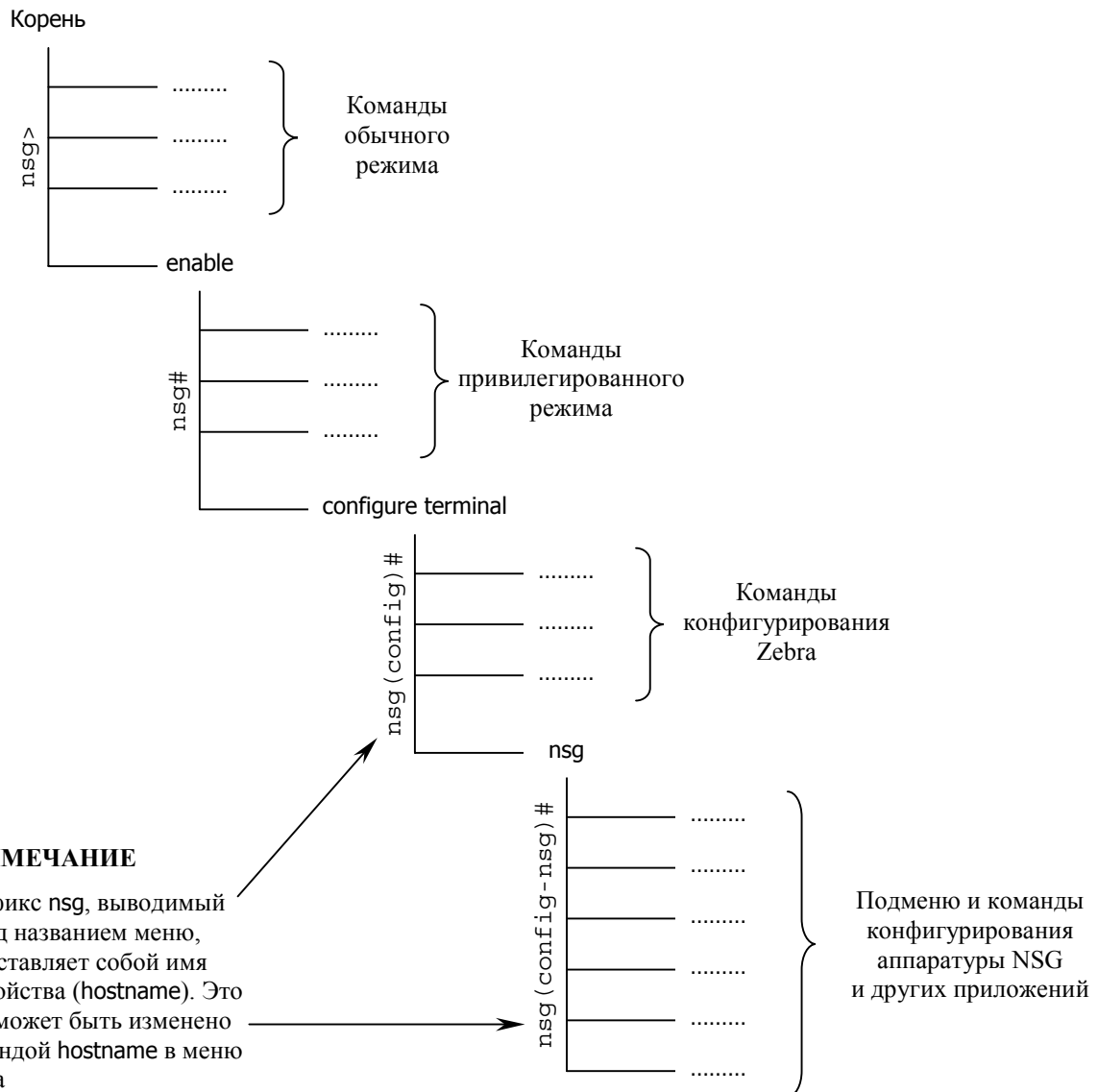
Для управления устройством NSG следует, в данном случае, послать с удаленного PAD вызов по адресу 1234567. После установления соединения на терминал будет выведено приглашение для ввода имени пользователя (*nsg* либо *root*) и пароля. Дальнейшая работа с устройством полностью идентична управлению через консольный порт или по Telnet.

**ПРИМЕЧАНИЕ** Одновременно к устройству могут иметь доступ несколько пользователей по Telnet, X.25 и/или через консольный порт. При этом только один из них может работать в режиме конфигурирования устройства; остальным разрешается только просматривать параметры конфигурации и статистику работы устройства.



### §1.2.4. Структура командного меню

Команды системы организованы в виде иерархических меню, относящихся к отдельным программным и аппаратным компонентам устройства. В общих чертах, меню имеет следующую структуру и следующий формат системных приглашений:



В обычном режиме работы пользователю доступны только базовые команды мониторинга (show), диагностики (ping, traceroute) и Telnet-клиент (telnet). В привилегированном режиме доступен расширенный набор команд, в том числе относящихся к конфигурации устройства в целом (например, команда write для записи конфигурации в энергонезависимую память).

В режиме configure terminal осуществляется настройка параметров маршрутизации, которая осуществляется при помощи пакета Zebra; помимо оригинальных команд Zebra, в это же меню добавлен новый узел nsq для доступа к специфическим параметрам маршрутизаторов NSG.

В подменю NSG осуществляется настройка аппаратной части маршрутизатора (шасси, физических портов), а также приложений, не входящих в пакет Zebra.

Во всех меню, наряду со специфическими командами, присутствуют следующие общие команды:

- no <команда> Отмена параметров, установленных указанной командой, и установка параметров по умолчанию.
- list Вывод списка всех команд меню (кроме (config-nsq)# и его дочерних узлов).
- exit Выход в предыдущее меню.
- quit Выход в предыдущее меню (кроме (config-nsq)# и его дочерних узлов).
- end Выход в меню привилегированного режима (enable).

### §1.2.5. Правила ввода команд и параметров

Работа с системой производится в режиме командной строки. В частности, вышеприведенный формат приглашения (nsg>) означает, что пользователь находится в режиме просмотра конфигурации (Read Only). Чтобы перейти в режим изменения конфигурации, необходимо ввести команду enable; после этого системное приглашение меняет вид:

```
nsg> enable
nsg#
```

Знак # в конце приглашения показывает, что пользователь находится в привилегированном режиме (с правами Read-Write). Название меню выводится в скобках в тексте системного приглашения, например:

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

В данном случае пользователь находится в меню конфигурации физического порта. Описание команд системы приведено в последующих разделах данного Руководства.

Все команды и параметры в данной версии NSG Linux вводятся только в нижнем регистре. Верхний регистр допускается только для символьных значений (имен, паролей, описаний объектов), причем в этом случае символы в верхнем и в нижнем регистре считаются различными.

Для вывода короткого списка команд текущего меню следует нажать клавишу ? (вопросительный знак). Для вывода списка команд со всеми параметрами следует набрать команду list.

Все команды и ключевые слова могут вводиться полностью или сокращаться до такой степени, которая позволяет однозначно идентифицировать их. Например, строка со не может быть идентифицирована однозначно, но строка con равносильна команде configure. Если начало команды введено однозначно, то нажатие клавиши TAB генерирует оставшуюся часть ключевого слова, а второе нажатие TAB выводит список параметров для этой команды. При этом после названия команды выводится пробел. Если для данной команды допустимо единственное значение параметра, то это значение сразу подставляется в командную строку. Примеры:

```
nsg> sh<TAB>
nsg> show <TAB>
bridge      debugging  interface  ip          table      version
nsg>

nsg# con<TAB>
nsg# configure <TAB>
nsg# configure terminal
nsg(config)#
```

Если набранное пользователем начало соответствует нескольким командам, то первое нажатие TAB дополняет общую часть этих команд, причем курсор ставится непосредственно после последнего символа, без пробела, чтобы можно было продолжить ввод. Второе нажатие TAB выводит список возможных окончаний. Пример:

```
nsg(config)# ro<TAB>
nsg(config)# route<TAB>
route-map   router
```

Команда может иметь один или несколько параметров. Чтобы получить список параметров, которые могут быть введены в данной позиции, следует нажать клавишу ?, например:

```
nsg(config-port-s1)# frame-relay <?>
display      Display information about this menu item
dlci         Set Frame Relay DLCI parameters
end          End current mode and down to ENABLE mode
exit         Exit current mode and down to previous mode
intf-type    Use DTE-DCE interface type LMI
lmi          Use NONE-ANSI-CCITT type LMI
n391         Full status polling counter
n392         LMI error threshold
n393         LMI monitored event count
t391         Link integrity verification polling timer
t392         Polling verification timer

nsg(config-port)# frame-relay
```

После имени параметра следует значение данного параметра, причем оно записывается в различном формате в зависимости от типа параметра. Чтобы просмотреть список или тип допустимых значений, следует ввести после параметра вопросительный знак.

**Перечислимый параметр.** В этом случае параметр может принимать только одно из нескольких фиксированных значений, например:

```
nsg(config-nsg)# port s1 encapsulation <?>
    cisco-hdlc      Cisco-HDLC (dflt)
    frame-relay     Frame Relay
    ppp             Point to Point Protocol
    raw-hdlc        Generic HDLC
    x25             X.25
```

(Current value = cisco-hdlc )

При вводе перечислимых значений, так же как при вводе команд и параметров, можно ограничиться несколькими начальными символами, определяющими его однозначно. Можно после этих символов ввести знак табуляции, чтобы закончить ввод. Если набранная часть не позволяет однозначно определить выбранное значение (в частности, если не введено ни одного символа), то по знаку табуляции выводится весь набор возможных продолжений, например:

```
nsg(config-nsg)# port s1 encapsulation <TAB>
cisco-hdlc frame-relay ppp raw-hdlc x25
```

**Числовой параметр.** Значением параметра может быть целое число из некоторого диапазона. Допустимый диапазон указывается в подсказке, выводимой с помощью клавиши ?, например:

```
nsg(config-interface-virtual-template-1)# keepalive <?>
<0 – 100> (dflt=10)
```

(Current value = 10 )

**Числовой параметр либо "no".** Значением параметра может быть либо целое число из некоторого диапазона, либо ключевое слово no. Значение no указывает, что данный механизм не используется, отключается, удаляется и т.п. Пример:

```
(config-dlci-81)# route port s1 17      Создать PVC между текущим виртуальным каналом Frame Relay
                                         (DLCI=81) и указанным каналом (DLCI=17) на указанном порту (s1).
(config-port-s3)# route port s1 no      Создать PVC между двумя портами по принципу "порт-на-порт" (с
                                         инкапсуляцией HDLC общего вида, без разделения на виртуальные
                                         каналы).
(config-dlci-81)# route no no           Удалить существующий PVC
```

**IP-адрес.** Подсказка для данного типа параметров имеет вид:

```
nsg(config-virtual-template-1)# ip address <?>
A.B.C.D
```

Значением параметра может являться только IP-адрес в десятичной дотовой нотации, например, 123.145.167.189. Если IP-адрес не задан в конфигурации устройства, то, как правило, он имеет значение 0.0.0.0.

**IP-префикс.** Подсказка для данного типа параметров имеет вид:

```
nsg(config-nsg)# port s1 ip address <?>
A.B.C.D/M
```

Значением параметра может являться только так называемый *префикс* — совокупность IP-адреса и маски, определяющая некоторое непрерывное пространство IP-адресов. IP-адрес вводится в десятичной дотовой нотации, вместо маски указывается ее длина (от 0 до 32 бит). Пример: 192.168.1.0/24. Значения префиксов по умолчанию, как правило — 0.0.0.0/0.

**MAC-адрес.** Значением параметра является последовательность из 6 байт, записанных в шестнадцатеричном виде и разделенных двоеточиями, например:

```
port eth0 mac-address 00:09:56:00:2B:67
```

**Адрес X.121.** В параметрах, имеющих смысл адреса или шаблона адресов X.121, допускаются десятичные цифры от 0 до 9. Также допускаются, в зависимости от контекста, следующие подстановочные символы:

```
*      Любое число любых цифр, в том числе и их отсутствие
?      Любая одна и только одна цифра
[...]  Любая одна и только одна цифра из перечисленных в квадратных скобках, например [123]
        — либо 1, либо 2, либо 3
```

Максимальная длина адреса X.121 — 15 цифр; маски, содержащей подстановочные символы — 31 цифра.

**Символьный параметр — слово (WORD).** К параметрам такого типа относятся административные имена различных объектов, пароли пользователей и т.п. Допускаются любые латинские буквы (верхний и нижний регистр различаются), десятичные цифры, а также следующие спецсимволы: точка (.), дефис (-), подчеркивание (\_), прямой слэш (/) и амперстед (@). Никакие другие символы, в том числе пробел и кавычки, не допускаются. Максимальная длина зависит от конкретного параметра.

**Символьный параметр — строка (STRING).** К параметрам такого типа относятся описания объектов (description), реквизиты SNMP (location и contact), сценарии для PPP-соединений по асинхронным линиям. Строка может содержать любые символы, с особыми правилами ввода для следующих символов:

- вопросительный знак (?) — вводится последовательным нажатием клавиш \ и ? (при этом обратный слэш преобразуется в вопросительный знак "на лету")
- двойная кавычка (") — если кавычка должна содержаться по существу (в зависимости от объекта, использующего данную строку), ее следует записывать в виде следующей escape-последовательности: \"
- обратный слэш (\) — если он необходим, его следует записывать в виде следующей escape-последовательности: \\

Строковое значение всегда заключается в двойные кавычки; однако если строка не содержит пробелов, то ее можно вводить и без кавычек. Максимальная длина строки зависит от конкретного параметра (при этом каждый символ escape-последовательностей считается отдельно). Дальнейшая интерпретация строкового значения и дополнительные правила ввода для определенных спецсимволов определяются конкретным объектом, которому адресован данный параметр (например, для сценария соединения таким объектом будет программа установления PPP-соединений).

Командный интерфейс устройства имеет возможность редактирования командной строки. Значения функциональных клавиш приведены ниже.

CTRL-B, Left	перевести курсор на один символ влево
CTRL-F, Right	перевести курсор на один символ вправо
ALT-B	перевести курсор влево до пробела
ALT-F	перевести курсор вправо до пробела
CTRL-A, Home	перевести курсор на начало строки
CTRL-E, End	перевести курсор на конец строки
CTRL-D, Del	удалить символ, на котором стоит курсор
CTRL-H, BS	удалить символ перед курсором
CTRL-W	удалить символы слева от курсора до пробела
ALT-D	удалить символы справа от курсора до пробела
CTRL-U	удалить символы слева от курсора до начала строки
CTRL-K	удалить символы справа от курсора до конца строки
CTRL-T	поменять местами символ, на котором стоит курсор, с предыдущим
CTRL-C	отменить ввод текущей строки и перейти на следующую строку
CTRL-N, Down	получить следующую строку из предыстории
CTRL-P, Up	получить предыдущую строку из предыстории
CTRL-Z	выйти из текущего подменю в корневое меню привилегированного режима
Enter	выполнить набранную команду
TAB	автоматически дополнить ввод команды

Клавиши, указанные во второй колонке, действуют если посылаемые ими коды соответствуют указанным в первой колонке. По этой причине они могут не работать на отдельных типах терминалов или программах эмуляции терминала, в которых им назначены другие коды.

### §1.2.6. Особенности команд меню (config-nsg)#

Узел меню (config-nsg)# содержит команды, являющиеся специфическими для продуктов NSG. По сравнению с общим командным языком vtysh, работа в этом разделе меню имеет следующие особенности:

- Набор доступных команд и параметров в каждом внутреннем подменю может генерироваться динамически в зависимости от параметров, установленных в других подменю, или определенных автоматически при старте системы. Например, возможные параметры команды card (выбор физического интерфейса в каждом из разъемов расширения) зависят от типа шасси, установленного командой chassis. Зависимости между пунктами меню приведены в описаниях команд.
- Допускается последовательно вводить команды нескольких подменю, вложенных друг в друга, например:

```
(config-nsg)# port s1 baudrate 2048000
```

При этом после выполнения команды пользователь остается в исходном (старшем) меню. Таким образом, приведенный выше пример эквивалентен следующей последовательности команд:

```
(config-nsg)# port
(config-port)# s1
(config-s1)# baudrate 2048000
(config-s1)# exit
(config-port)# exit
```

- В каждом меню имеется команда display, используемая со следующими параметрами:

display configuration	Вывести те параметры данного узла меню и его подузлов, которые ранее были изменены пользователем.
display all	Вывести все параметры данного узла меню и его подузлов, в т.ч. значения, установленные по умолчанию.
display path	Вывести путь к данному узлу меню.
display version	Вывести версию программного обеспечения.

- Для краткости, в системном приглашении указывается только config и название самого младшего подменю. Полностью узнать текущее положение в системе меню можно с помощью команды display path, например:

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

- Команда no имеет ограниченное применение и используется преимущественно для удаления объектов, созданных пользователем ранее (dcli, vlan, ip, community и т.п.). В остальных случаях для отмены сделанных настроек следует повторить соответствующие команды с "нулевыми" параметрами — значениями по умолчанию, нулями (напр., 0.0.0.0), no или none.

- Не поддерживаются команды list и quit.

### §1.2.7. Завершение работы с устройством

Для завершения сеанса работы с устройством следует набрать в корневом меню (в обычном или привилегированном режиме) команду quit или exit.

**ВНИМАНИЕ** Если в ходе работы были внесены какие-либо изменения в конфигурацию устройства, то их необходимо сохранить перед выходом сеанса конфигурации. В противном случае сделанные изменения будут потеряны при следующей перезагрузке устройства. Для сохранения конфигурации в энергонезависимой памяти используется команда write file.

### §1.2.8. Обозначения и соглашения

В описаниях команд, приведенных в последующих разделах данного Руководства, используются следующие обозначения:

{ значение1   значение2   ... }	При вводе команды необходимо выбрать одно из перечисленных значений.
<значение1...значение2>	При вводе команды необходимо выбрать числовое значение в указанном диапазоне.
параметр <значение>	При вводе команды необходимо задать числовое или текстовое значение указанного параметра.
[параметр <значение>]	Параметр не является обязательным.

Команды, являющиеся общими для всех подменю (? , list, exit и т.п.), отдельно не рассматриваются.

Операции, требующие последовательного перехода в несколько подменю внутри (config-nsg)#, могут описываться как поочередно по каждому подменю:

*Настройка объекта X производится в меню (config-x)#, содержащем следующие команды:*

.....  
 ууу    *Переход к операции Y*

*Настройка дочернего объекта Y производится в меню (config-y)#, содержащем следующие команды:*

.....  
 zzz    *Переход к операции Z*

*Команда Z имеет следующие параметры:*

<параметр1>    .....

<параметр2>    .....

так и сразу в форме одной команды:

*Операция XYZ производится в меню (config-x)# командой:*

ууу zzz <параметр>

Как правило, вторая (сокращенная) форма описания используется в случаях, когда каждое из дочерних подменю содержит только 1–2 специфических команды (не считая общих команд exit, display и т.п., а также списка дочерних объектов). Если меню содержат большее число существенных параметров, они рассматриваются поочередно. На практике оба способа ввода команд различаются только тем, в каком месте меню оказывается пользователь после их выполнения.

## §1.3. Общесистемные настройки

### §1.3.1. Идентификация аппаратной части (одноплатные устройства)

Для корректной работы устройства необходимо, в первую очередь, правильно указать тип шасси и интерфейсных модулей. При старте системы программное обеспечение пытается автоматически определить их с максимально возможной точностью, однако однозначно сделать это удастся не во всех аппаратных конфигурациях. В таких случаях для уточнения конфигурации необходимо вручную выбрать используемые аппаратные компоненты.

Идентификация производится из меню (config-nsg)# при помощи следующих команд:

```
chassis { шасси1 | шасси2 | ... }
```

Тип шасси. Команда необходима только для тех устройств, которые не определяются однозначно. Набор допустимых параметров соответствует возможным типам шасси, которые могут подпадать под результаты автоматического определения.

```
card { s1 | s2 | ... } <модуль>
```

Тип интерфейсного модуля, установленного в разъем расширения с указанным номером.

Допустимы следующие имена модулей:

```
empty (по умолчанию)
im-et10
im-2e1, im-ce1, im-e1, im-703, im-703-64
im-2shdsl-bis, im-shdsl-bis, im-idsl, im-mdsl, im-sdsl, im-shdsl, im-srm
im-rs485, im-rs530, im-v24, im-v35, im-x21
uim-cdma, um-usb, um-et100, um-hdlc
```

Число параметров вида sN определяется выбранным типом шасси, а набор возможных интерфейсных модулей — типом шасси и номером разъема расширения.

**ВНИМАНИЕ** Модули IM-2E1, IM-2E1-S идентифицируются одной командой `card s4 im-2e1` либо `card s2 im-2e1` (для четырех- и двухпортовых шасси, соответственно). Попытка описать их как два отдельных модуля IM-E1, IM-E1-S приводит к критической ошибке ПО.

Допустимые значения параметров можно просмотреть при помощи команды "?". Никакие другие значения, помимо перечисленных в справочной информации, не допускаются. Например, если некоторый тип интерфейсных модулей не поддерживается данным разъемом данного шасси, то это значение не будет предлагаться системой, а если пользователь, тем не менее, попытается установить его, то команда будет проигнорирована.

По умолчанию, если тип модуля не определен, устанавливается значение `empty`, т.е. предполагается, что разъем не оснащен интерфейсным модулем. В этом случае он не может быть использован и не выводится командой `display`. Проверить выбранные типы интерфейсных модулей можно с помощью команды `display`.

Если тип устройства (например, NSG-900/16A) или интерфейсного модуля определен системой однозначно, то команды `chassis` и `card sN` допускают только единственное значение параметра и носят формальный характер. Их можно использовать только для проверки правильности автоматического определения аппаратных компонент.

#### Примечания по отдельным типам шасси и модулей:

Для устройства NSG-800/16A автоматически различаются следующие аппаратные модификации:

nsg800-8a	<i>hardware ver.1.1</i> (асинхронные порты расположены с краю задней панели), 8-портовая модификация
nsg800-16a	<i>hardware ver.1.1</i> , 16-портовая модификация
nsg800-8am	<i>hardware ver.1.2</i> и старше (асинхронные порты расположены в центре), 8-портовая модификация
nsg800-16am	<i>hardware ver.1.2</i> и старше, 16-портовая модификация

Основное различие между модификациями NSG-800/16A *h/w ver.1.1* и *1.2* состоит в возможности использования модулей IM-xE1-x и IM-xSHDSL/bis. Подробную информацию по данному вопросу см. в руководстве пользователя NSG-800/16A.

Для устройства NSG-900/16A различаются две конфигурации — 8- и 16-портовая.

Для устройств NSG-900/4WL, NSG-900/2WL, NSG-800/4WL и NSG-800/WL тип шасси может определяться полностью либо с точностью до серии, в зависимости от модификации материнской платы. Значение `nsg800-2wl` соответствует модификации устройства NSG-800 с двумя разъемами расширения.

Поддержка интерфейсных модулей IM-CE1-S (IM-CE1) в данной версии NSG Linux не реализована. Имена этих модулей в команде `card` зарезервированы для последующего применения.

Для модулей, выпускавшихся в конструктивном исполнении NSG Тип 1 и Тип 2 (например, IM-V35 и IM-V35-2), либо в нескольких модификациях для различных аппаратных версий шасси (IM-E1, IM-E1-S, IM-E1-SS) используется одно и то же символическое имя.

Для модулей UIM-CDMA, UIM-EVDO, установленных в шасси NSG-700 либо во вспомогательные разъемы расширения 1, 2 устройства NSG-1000, необходимо указывать типа `uim-cdma`. Во всех остальных случаях модемы для сотовых сетей и проводных сетей ТФОП рассматриваются системой как отдельные устройства, подключенные по интерфейсу V.24 (RS-232), поэтому для них устанавливается тип `im-v24`.

### §1.3.2. Идентификация ведомых плат в многоплатных устройствах

Для увеличения числа синхронных портов в NSG Linux предусмотрено каскадирование нескольких устройств в одно целое. В качестве ведомых устройств стека могут использоваться шасси NSG-900/4WL и карты IC6e-4W (с точки зрения NSG Linux, эти устройства идентичны), на которые установлена специальная версия *slave* программного обеспечения. Максимальное число ведомых устройств в одном стеке — 4, т.е. дополнительно к портам ведущего устройства (карты) может быть подключено 4–16 портов. Для обозначения ведомых карт в устройствах NSG используется термин *tier* (ярус).

При работе в составе стека вся маршрутизация и коммутация пакетов, а также конфигурирование всего стека как целого, осуществляется на ведущем устройстве (карте). Для каскадирования используется коммутатор Ethernet. Порты ведомых устройств (карт) могут использоваться только для работы синхронных протоколов: Cisco-HDLC, PPP, Frame Relay, X.25, Raw-HDLC. Установка асинхронных и Ethernet-модулей не предусмотрена.

Частным примером стека является устройство NSG-900/8WL, включающее в себя две карты IC6e-4W и коммутатор Fast Ethernet. Одна из этих карт (нижняя) является ведущей, другая (верхняя) — ведомой. Возможна также конфигурация, при которой обе карты являются ведомыми при некотором другом устройстве NSG. Настройка стека состоит из следующих двух этапов:

1. Определение MAC-адреса ведомой карты. Необходимо подключиться к консоли *ведомой* карты и использовать следующую последовательность команд:

```
nsg> enable
nsg# configure terminal
nsg(configure)# port eth0
nsg(configure-if)# display all
```

Последняя команда выводит все параметры порта. Искомый MAC-адрес указывается в виде

```
mac-address 00:09:56:xx:xx:xx
```

2. Подключение ведомой карты к ведущей. Настройка выполняется из меню `(config-nsg)#` ведущей карты.

```
tier { a | b | c | d }
no tier { a | b | c | d }
```

Подключение ведомой карты и вход в ее меню и отключение ведомой карты, соответственно. В данной версии NSG Linux предусмотрено до 4 ведомых карт (устройств) с символическими именами `a`, `b`, `c`, `d`. Дальнейшая настройка производится в меню ведомой карты `(config-tier-N)#`.

Меню `(config-tier-N)#` содержит следующие команды:

```
description <строка>
```

Текстовое описание данной ведомой карты для целей администрирования. Максимальная длина — до 255 знаков. Если строка содержит пробелы, ее необходимо заключить в кавычки.

```
remote-mac-address <xx:xx:xx:xx:xx>
```

MAC-адрес ведомой карты.

```
port <имя> Имя порта Ethernet, используемого для каскадирования.
```

Параметры `remote-mac-address` и `port` являются обязательными. Если они не указаны или указаны некорректно, работа с ведомой картой невозможна.

После настройки ведомой карты в системе появляются физические интерфейсы и порты с именами вида `a-s1` ... `a-s4`. Первая часть имени порта указывает на ведомую карту, вторая — на локальный номер разъема расширения на этой карте. Далее необходимо указать тип физического интерфейса с помощью команды `card` (см. предыдущий параграф).

**ПРИМЕЧАНИЕ** При подключении ведомых карт аппаратная идентификация ведущей карты (устройства) не изменяется. Например, в устройстве NSG-900/8WL ведущая карта всегда идентифицируется как `nsg900-4wl`.



**Пример** конфигурации порта a-s1 для устройства NSG-900/8WL:

```
!
nsg
  tier a
    remote-mac-address 00:09:56:12:34:56
    port eth0
    exit
  card a-s1 im-v24
  port a-s1
    baudrate 128000
    mode external
    .....
```

### §1.3.3. Установка имени устройства

Сетевое имя устройства (hostname) выводится в начале системной подсказки во всех меню, а также используется при аутентификации и SNMP-управлении. По умолчанию устройства имеют имя nsg. Назначение сетевого имени осуществляется из меню следующей командой:

```
hostname <имя>
```

Назначение сетевого имени устройства. Имя должно быть непустым и не может содержать пробелы.

### §1.3.4. Парольная защита устройства

Для управления паролями, необходимыми для входа в основную командную оболочку (vtysh) и в командную оболочку ОС Linux, используются следующие команды в меню конфигурации Zebra (config)#:

```
password-nsg
no password-nsg
```

Установка/удаление пароля для пользователя *nsg*. Команды работают в диалоговом режиме. После получения соответствующих подсказок необходимо ввести старый пароль (если он был установлен ранее), новый пароль (если его следует установить), подтверждение нового пароля.

```
password-root
no password-root
```

Установка/удаление пароля для пользователя *root*. Команды аналогичны описанным выше, но изменяют пароль для входа непосредственно в командную оболочку ОС Linux.

Пароли, вводимые пользователем, не показываются на экране и не включаются в конфигурацию, выводимую командой *write* (см. ниже). После изменения пароля необходимо сохранить конфигурацию устройства (см. следующий параграф).

**ВНИМАНИЕ!** **ДЛЯ ПРЕДОТВРАЩЕНИЯ НЕСАНКЦИОНИРОВАННОГО ДОСТУПА К УСТРОЙСТВУ НЕОБХОДИМО УСТАНОВИТЬ УНИКАЛЬНЫЙ СЕКРЕТНЫЙ ПАРОЛЬ И ДЛЯ ПОЛЬЗОВАТЕЛЯ *nsg*, И ДЛЯ ПОЛЬЗОВАТЕЛЯ *root*. ПОМНИТЕ: ТЕ, КТО ХОЧЕТ ПРОНИКНУТЬ НА ВАШЕ УСТРОЙСТВО, ОБЫЧНО ЧИТАЮТ ДОКУМЕНТАЦИЮ ГОРАЗДО ВНИМАТЕЛЬНЕЕ ВАС!**

### §1.3.5. Установка системного времени

Для установки и проверки системного времени используются следующие команды в меню (config-nsg)#:

```
clock set hh:mm:ss dd.mm.yyyy
```

Установить системное время и дату, в указанном формате.

```
clock show
```

Вывести текущие показания системных часов.

### §1.3.6. Запуск и остановка ресурсоемких системных служб

Ряд компонент системы, потребляющих значительные вычислительные ресурсы и/или оперативную память, реализован в виде отдельных служб, которые могут быть включены или выключены по усмотрению пользователя. К таким компонентам относятся:

```
bgp      Поддержка протокола динамической маршрутизации BGP.
ospf     Поддержка протокола динамической маршрутизации OSPF.
rip      Поддержка протоколов динамической маршрутизации RIP 1 и RIP 2.
```

Запуск и остановка этих компонент выполняется в меню (config-nsg)# следующей командой:

```
services <имя_службы> { enable | disable }
```

где <имя\_службы> может принимать одно из перечисленных выше значений.

- enable Разрешить запуск указанной службы при следующей перезагрузке устройства.
- disable Остановить указанную службу, удалить ее параметры из конфигурации устройства, и запретить ее запуск при следующей перезагрузке.

**ВНИМАНИЕ** Вышеперечисленные службы глубоко интегрированы в систему, поэтому для их запуска, после команды `enable`, необходима перезагрузка устройства; начинать настройку этих служб можно *только после* перезагрузки. При отключении службы все параметры настройки, относящиеся к ней, удаляются из конфигурации устройства при очередной команде `write file` и не могут быть восстановлены.

По умолчанию, все указанные службы включены. Данная команда позволяет пользователю отключить неиспользуемые службы. Рекомендуется сформировать список необходимых служб непосредственно при первичной установке устройства NSG в сетевое решение, с тем, чтобы впредь изменять его только в случае существенной реорганизации сети. После этого следует сохранить конфигурацию, перезагрузить устройство, и затем приступить к дальнейшей настройке.

### §1.3.7. Использование служб SSH

Для безопасного удаленного доступа к устройству возможно (и рекомендуется) использовать протокол SSH2. Управление встроенным сервером SSH2 производится в меню (config-nsg)# следующей командой:

```
services ssh { enable | disable | keygen }
```

Параметры команды:

- keygen Сгенерировать ключи для SSH.
- enable Стартовать сервер SSH (предварительно должны быть сгенерированы ключи).
- disable Остановить сервер SSH.

Генерация ключей требует значительные вычислительные ресурсы и занимает продолжительное время (на младших моделях — до 15 минут), поэтому после первого ввода команды запрашивается подтверждение. После повторного ввода команда запускается на исполнение. Данная операция производится однократно и в случае необходимости может быть повторена. Не рекомендуется выполнять ее на устройстве, работающем в реальной сети, в периоды высокой нагрузки.

Параметр `keygen` в действительности является командой и не сохраняется в энергонезависимой памяти. Сгенерированные ключи, как и другие настройки, следует сохранить при помощи команды `write file` (см. §1.3.8).

После создания ключей к устройству возможно подключиться клиентом SSH2 по порту 22. Для входа на устройство можно использовать имя и пароль пользователей `root`, `nsg`, а также другие имена и пароли, перечисленные в файле `/etc/passwd` (данный файл может просматриваться и редактироваться средствами ОС Linux). Как частный случай, средствами ОС Linux возможно создать пользователя, который при входе в систему будет прозрачно проклячаться на один из асинхронных портов, т.е. реализовать Reverse SSH (аналогично Reverse Telnet).

**ПРИМЕЧАНИЕ** Доступ по SSH возможен только для тех пользователей, для которых установлен непустой пароль.

Для запуска клиента SSH необходимо перейти в командную оболочку ОС Linux и ввести команду:

```
ssh <ip-адрес> [порт] [доп.опции]
```

По умолчанию, служба SSH использует порт TCP 22. Для отключения от удаленной стороны следует ввести управляющую последовательность `"~."` (тильда и точка). Подробно обо всех дополнительных опциях команды `ssh` см. *man pages* к ОС Linux.

Как частный случай, возможно реализовать вход с некоторого терминального устройства, подключенного к асинхронному порту NSG и поддерживающего механизм сценариев (скриптов), на удаленный сервер SSH.

### §1.3.8. Сохранение конфигурации

Полученная конфигурация устройства должна быть сохранена в энергонезависимой памяти для того, чтобы эти изменения стали постоянными. Запись конфигурации производится в меню конфигурации Zebra (config)# либо в корневом меню привилегированного режима командой:

```
write file
```

**ВНИМАНИЕ** Изменения конфигурации, не сохраненные в энергонезависимой памяти, будут утрачены после следующей перезагрузки устройства.

**ПРИМЕЧАНИЕ** Команда write file подразумевает, что в конфигурации устройства были произведены какие-либо изменения, поэтому она доступна только после того, как пользователь хотя бы однажды войдет в меню configure terminal.

### §1.3.9. Перезагрузка устройства

Для программной перезагрузки устройства необходимо перейти в меню привилегированного режима (см. п.1.2.4) и ввести команду:

```
reload
```

Для аппаратной перезагрузки устройства следует нажать защищенную кнопку Reset, либо выключить и включить питание.

### §1.3.10. Просмотр и резервирование текущей конфигурации

Для просмотра полной конфигурации устройства следует использовать команду

```
write terminal
```

Данная команда генерирует сценарий конфигурации, содержащий различия между текущей конфигурацией и конфигурацией по умолчанию. Команда доступна в меню привилегированного режима и в меню конфигурирования Zebra. Для резервирования текущей конфигурации следует сохранить ее вывод в виде script-файла. Для восстановления конфигурации следует восстановить конфигурацию по умолчанию (см. ниже), а затем загрузить данный script-файл.

**ПРИМЕЧАНИЕ** При вводе и выводе длинных строк возможны проблемы, связанные с настройками пользовательского терминала (автоперенос строк, дополнение CR/LF и т.п.). В этом случае строки длиной более 79 символов рекомендуется исключить из script-файла и вводить отдельно вручную.

Внутри узла команд NSG доступны также следующие команды для просмотра конфигурации текущего объекта и всех его вложенных объектов:

display configuration	Вывести те параметры данного узла меню и его подузлов, которые ранее были изменены пользователем.
display all	Вывести все параметры данного узла меню и его подузлов, в т.ч. значения, установленные по умолчанию.
display path	Вывести путь к данному узлу меню.
display version	Вывести версию программного обеспечения.

### §1.3.11. Восстановление заводской конфигурации

Начиная с версии NSG Linux 1.0 *build 0*, для установки значений всех параметров по умолчанию следует использовать меню NSG Linux Revision Utility на этапе загрузки системы. Операция выполняется только с локального терминала, подключенного к консольному порту. В ходе процедуры загрузки на экран выводится меню:

```
Welcome to NSG Linux Revision Utility!
```

```
1) continue booting
2) update software
3) boot with default config
```

```
timeout='25', default_choice='1'
Make your choice:
```

После этого делается пауза продолжительностью 25 сек. В течение этого времени необходимо нажать клавишу 3, после чего устройство стартует с заводской конфигурацией. При этом на устройствах без

расширенной энергонезависимой памяти заводская конфигурация загружается в качестве текущей, но не записывается; в памяти сохраняется прежняя конфигурация. Для того, чтобы полностью удалить пользовательскую конфигурацию, следует записать текущую (заводскую) конфигурацию в энергонезависимую память командой `write file` (см. выше). Если же, наоборот, ее необходимо сохранить, то следует, не выполняя команду `write file`, перезагрузить устройство в обычном режиме.

На устройствах, оснащенных расширенной энергонезависимой памятью, выдается дополнительный запрос для подтверждения операции. После этого заводская конфигурация записывается непосредственно в энергонезависимую память, и устройство стартует уже с этой конфигурацией.

Если в течение 25 сек. не нажата никакая клавиша, то устройство продолжает загружаться с существующей пользовательской конфигурацией.

Удаление пользовательской конфигурации возможно также средствами загрузчика `uboot`, запускаемого при аппаратном старте системы. Подробно о работе с `uboot` см. Приложение 1–Б.

### §1.3.12. Сценарий автозапуска и выполнение команд Linux

При включении или перезагрузке устройства можно произвести дополнительную конфигурацию системы средствами ОС Linux. Для этой цели предназначена вспомогательная команда `script`, доступная из меню (`config-nsg`)#. Команда используется следующим образом:

```
script add <номер> "<команда>"
```

Создать/изменить запись с указанным номером в таблице сценариев. Номер записи может находиться в пределах от 1 до 1000. Текст команды может иметь длину до 255 символов и содержать любую команду ОС Linux. В частности, допускается определять переменные, используемые в последующих командах, и т.п.

Поскольку команда обычно содержит пробелы, она заключается в кавычки.

```
script exec {<номер> | -1 | all }
```

Выполнить одну команду с указанным номером; если значение параметра равно `-1` или `all` — выполнить последовательно все команды.

```
script del <номер>
```

Удалить команду с указанным номером из таблицы сценариев.

**ВНИМАНИЕ** Созданная таблица сценариев, как и другие настройки, должна быть сохранена в энергонезависимой памяти устройства с помощью команды `write file`.

Команды, содержащиеся в таблице сценариев, исполняются последовательно в порядке их нумерации. Исполнение стартового сценария происходит после старта системы и загрузки всей остальной конфигурации из энергонезависимой памяти. С помощью данной команды можно, в частности, стартовать дополнительные службы, не запускаемые по умолчанию, либо, наоборот, остановить службы, запущенные по умолчанию, но не требуемые в данном случае.

**ВНИМАНИЕ** Команда `script exec` предназначена только для исполнения команд конфигурирования. Она запускает указанный сценарий (или последовательность сценариев) на исполнение и ждет завершения его работы. По этой причине нельзя включать в сценарий бесконечные циклы, команды, ожидающие ввода с терминала и т.п. — они приведут к остановке работы командной оболочки.

Еще одна особенность выполнения сценариев связана с тем, что команда `script` ожидает сигнала о завершении исполнения команды. Если исполняемая команда запускает некоторый демон Linux, то в сценарий необходимо включить команду, выполняющую какой-нибудь фиктивный вывод, например:

```
script add 1 "echo OK; syslogd"
```

**ВНИМАНИЕ** При запуске демонов (или других программ, работающих в фоновом режиме) необходимо перенаправить их вывод в некоторый файл или в `null`. Вывод в стандартное устройство (`stdout`) для таких процессов невозможен.

На уровне ОС Linux команды автозапуска, по умолчанию, содержатся в файле `/etc/rc.sh`. При необходимости данный файл можно просмотреть или изменить средствами ОС Linux, например, с помощью редактора `vi`. Команда `script` позволяет, в некоторой степени, откорректировать результаты исполнения данного файла и предназначена, в первую очередь, для пользователей, не имеющих навыков работы в среде ОС Linux.

При запуске дополнительных служб с помощью команды `script` необходимо иметь в виду, что данная команда выполняется *после* загрузки всей остальной конфигурации. Таким образом, если в конфигурации имеются команды, относящиеся к такой службе, то при загрузке устройства они будут проигнорированы (поскольку на этот момент служба еще не стартовала), а затем служба будет запущена с настройками, принятыми по умолчанию. Для корректной загрузки таких служб следует внести изменения в файл `/etc/rc.sh`.

Пример использования сценария: настройка DNAT (Destination NAT, или виртуальные сервера) средствами IPtables. Физический порт `s1` подключен к внешней сети, порт `eth0` — к внутренней сети с FTP-сервером. Требуется обеспечить доступ к этому серверу из внешней сети (в обоих режимах FTP — активном и пассивном).

```
!
nsg
  port s1
    ip address 123.145.167.189/30
    nat masquerade
  exit
  port eth0
    ip address 10.0.0.1/8 anycast 0.0.0.0
  exit
script add 1 "iptables -t nat -A PREROUTING -p TCP -i s1 --dport 20 -j DNAT --to-destination 10.0.0.2:20"
script add 2 "iptables -t nat -A PREROUTING -p TCP -i s1 --dport 21 -j DNAT --to-destination 10.0.0.2:21"
script exec -1
```

### §1.3.13. Настраиваемая светодиодная индикация

На устройствах, оснащенных программируемыми светодиодными индикаторами, каждый из этих индикаторов позволяет отображать состояние программных и аппаратных объектов, выбранных пользователем. Устройства NSG-700 *h/w ver.3* и старше оснащены двумя программируемыми индикаторами. Устройства NSG-900 оснащены 2 или 4 индикаторами, в зависимости от модели. В отдельных устройствах и модификациях могут применяться двухцветные светодиоды (красно-зеленые).

Для управления индикаторами используются следующие команды в меню (`config-nsg`)#:

```
led { l1 | l2 | ... } client <имя>
```

Выбор индикатора и объекта, отображаемого им. Допустимые имена индикаторов соответствуют указанным на корпусе устройства. (Для устройств NSG-900 они могут быть указаны в формате Port 1 ... Port 4). Для двухцветных индикаторов используются имена вида `l1`, `l2`, ... , соответствующие зеленому цвету, и `l1-r`, `l2-r`, ... , соответствующие красному цвету.

Имена объектов-клиентов записываются в следующем формате:

<имя>-if IP-интерфейсы (в т.ч. туннели) с указанным именем

<имя>-proto протоколы 2 уровня (Frame Relay, Cisco-HDLC и др.) на указанных портах

<имя>-phy физические интерфейсы (или суб-интерфейсы структурированных интерфейсов)

Включенный индикатор означает, что выбранный объект находится в состоянии UP, выключенный — DOWN. Мигание индикатора означает прием-передачу данных через указанный объект.

Имена объектов-клиентов можно вводить вручную, либо выбирать из списка, который генерируется системой автоматически. Текущий список возможных клиентов можно просмотреть командой `led ... client ?`, однако он включает только объекты, существующие в системе в данный момент. Для объектов, порождаемых динамически, или для тех, которые пользователь только планирует создать позже, имя необходимо вводить вручную.

При выборе объектов-клиентов необходимо учитывать особенности работы конкретных протоколов. Например, для протокола PPP, если согласование IP-адресов завершается неудачей, последовательно разрываются PPP-соединение (2 уровня) и физическое соединение (1 уровня). Таким образом, состояние, при котором 1 и 2 уровень подняты, а 3 — нет, сохраняется очень короткое время. Если есть подозрение, что процедура установления соединения обрывается именно в это фазе, то при мониторинге клиента `...-proto` или `...-phy` следует установить минимальное время обновления показаний (см. ниже) и судить о достижении этой фазы по результатам нескольких попыток — в лучшем случае будет одна кратковременная вспышка.

**ПРИМЕЧАНИЕ** В данной версии NSG Linux индикация реализована для всех IP-интерфейсов и для частичного набора портов и физических интерфейсов.

```
led { l1 | l2 | ... } client { test-on | test-off | test-blinking | test-sos }
```

Тестирование индикаторов. В качестве клиента могут быть указаны следующие тесты:

`test-on` индикатор включен постоянно

`test-off` индикатор выключен постоянно

`test-blinking` индикатор мигает через каждый такт опроса

`test-sos` индикатор обрабатывает сигнал SOS азбукой Морзе: 3 коротких вспышки, 3 длинных, 3 коротких, пауза

```
led refresh <5 ... 6000>
```

Интервал обновления показаний индикаторов, в сотых долях секунды. Значение по умолчанию — 10 (1 раз в секунду).

**ПРИМЕЧАНИЕ** Для двухцветных светодиодов при одновременном включении красного и зеленого сегментов образуется оранжевый цвет, который легко спутать с красным. Рекомендуется избегать настроек, при которых они могут быть включены одновременно, либо предварительно протестировать возможные комбинации и убедиться, что они различимы на глаз.

Пример конфигурации (устройство NSG–700):

```
!  
nsg  
  card s1 uim-cdma  
  port s1 encapsulation ppp  
  .....  
  tunnel pptp 1  
  .....  
  led l1 client s1-if  
  led l2 client pptp1-if
```

В данном случае индикатор L1 включен при наличии IP-соединения (и, следовательно, PPP- и физического соединения) с сетью оператора CDMA, индикатор L2 — при наличии PPTP-туннеля до удаленного узла корпоративной сети.

## §1.4. SNMP-управление

### §1.4.1. Настройка агента SNMP

В программное обеспечение устройств NSG Linux входит встроенный агент SNMP. Для мониторинга и управления данными устройствами могут использоваться любые стандартные платформы сетевого управления на основе SNMP. Собственно агент SNMP обеспечивает доступ к устройству как к объекту управления, а также определяет субъекты этого управления (SNMP communities).

Для корректной работы агента SNMP необходимо, в первую очередь, настроить протокол SMUX. Этот протокол обеспечивает взаимодействие агента с остальными компонентами системы. Настройка осуществляется из основного меню конфигурации (config)# при помощи следующих команд:

```
smux peer 1.3.6.1.2.1.4.24 dan
no smux peer 1.3.6.1.2.1.4.24 dan
```

Включение/выключение протокола SMUX.

Настройка собственно агента осуществляется из меню (config-nsg)# при помощи следующих команд:

```
snmp-agent running { enable | disable }
```

Включение/выключение агента SNMP. По умолчанию агент выключен (disable).

```
snmp-agent community <имя-community>
```

```
no snmp-agent community <имя-community>
```

Создание/настройка и удаление управляющего сообщества (*community*) SNMP. Первая команда создает *community* с указанным номером, если оно не существует, и осуществляет вход в меню редактирования *community* (см. следующий параграф). Имя *community* должно быть непустой строкой и не может содержать пробелы.

Если в системе не определено ни одного *community*, то SNMP-агент не будет отвечать ни на чьи запросы.

```
snmp-agent location <строка>
```

```
snmp-agent contact <строка>
```

Установка параметров идентификации данного узла: местонахождения и ответственного специалиста (переменные *sysLocation* и *sysContact* согласно RFC-1213). Если данные строки должны содержать пробелы, то их следует вводить в кавычках.

По умолчанию значениями обеих переменных являются пустые строки. Чтобы обнулить существующее значение переменной, следует ввести в данной команде две кавычки подряд ("").

Третья идентификационная переменная — имя устройства (*hostname*) устанавливается из вышестоящего меню (config)# командой *hostname* (см. п. 1.3.3).

```
snmp-agent traps { enable | disable }
```

Разрешить/запретить SNMP-агенту генерацию Trap-сообщений. По умолчанию Trap-сообщения запрещены (disable).

```
snmp-agent host <ip-адрес>
```

```
no snmp-agent host <ip-адрес>
```

Добавление и удаление станции с указанным IP-адресом в/из списка станций, которым рассылаются Trap-сообщения. Следует подчеркнуть, что выбор станций осуществляется на основе индивидуальных IP-адресов, а не диапазонов адресов.

По умолчанию список рассылки пустой. При пустом списке Trap-сообщения не посылаются.

```
snmp-agent trap-source <ip-адрес>
```

Установка IP-адреса, который будет указываться во всех Trap-сообщениях в качестве исходного адреса. Если такой адрес не определен (значение данного параметра 0.0.0.0), то в каждом сообщении указывается IP-адрес того интерфейса, через который оно посылается; это режим по умолчанию. Для удаления ранее установленного адреса следует ввести нулевой адрес.

```
snmp-agent logging { none | short | detail }
```

Управление объемом отладочной информации SNMP:

*none* Информация не выводится

*short* Выводится минимальный объем информации (IP-адреса станций, от которых получены запросы, аварийные сообщения и т.п.)

*detail* Выводится полная информация обо всех полученных запросах и отправленных сообщениях SNMP в шестнадцатеричном виде

Отладочная информация выводится в файл /var/log/snmpd.log, который можно просмотреть и обработать средствами Linux.

**ВНИМАНИЕ** Файл `snmpd.log` находится на виртуальном диске в оперативной памяти устройства, поэтому при перезагрузке или выключении питания он не сохраняется. Кроме того, чрезмерно большой объем этого файла, который может быть достигнут в режиме `detail`, может негативно сказаться на работе системы. Максимальный объем файла — около 15 МБ.

### §1.4.2. Настройка SNMP community

Управляющие сообщества (*communities*) являются субъектами SNMP-управления. Сообщество состоит из одной или более управляющих станций (*managers*), объединенных некоторым идентификатором — именем *community*.

Для перехода в меню настройки *community* используется команда следующего вида:

```
(config-nsg)# snmp-agent community <имя-community>
```

При этом, если *community* с указанным именем не существует, оно будет создано. Меню настройки *community* содержит в системном приглашении имя данного *community*, например:

```
(config-snmp-agent-community-public)#
```

(для сообщества `public`) и содержит следующие команды:

```
manager <ip-префикс>
```

```
no manager <ip-префикс>
```

Добавление и удаление указанного множества IP-адресов в/из данного *community*. IP-префикс состоит из адреса сети и длины маски (через слэш). Одно *community* может содержать несколько диапазонов адресов, например:

```
manager 10.0.0.0/8
```

```
manager 192.168.1.1/32
```

— включить в текущее управляющее сообщество все станции из сети `10.x.x.x` и одну станцию с адресом `192.168.1.1`

Если допустимые адреса управляющих станций определены, то агент SNMP будет выполнять запросы только от этих станций. Если для данного *community* не определена ни одна запись *manager*, то допускаются обращения от станций с произвольными адресами.

```
access { ro | rw }
```

Установить права доступа для данного *community*:

```
ro — исполняются только запросы на чтение (Read-Only)
```

```
rw — исполняются запросы на чтение и запись (Read-Write)
```

При создании *community* по умолчанию устанавливается значение `ro`.

```
view
```

Команда зарезервирована для последующего использования.



## §1.5. Установка и модернизация NSG Linux

### §1.5.1. Аппаратные требования и файлы ПО

Программное обеспечение NSG Linux является штатным для всех устройств серий NSG-900, NSG-700 и NSG-1000, а также может опционально устанавливаться на мультипротокольные маршрутизаторы серии NSG-800. Начиная с версии NSG Linux 1.0 *build 0*, поддерживаются следующие аппаратные конфигурации:

— **Устройства с процессором Motorola MPC 862/860/855T и Flash ROM 8 МБ:**

NSG-900/2WL	NSG-800/WL (1- и 2-портовые модификации)
NSG-900/4WL	NSG-800/4WL
NSG-900/16A (16- и 8-портовые модификации)	NSG-800/16A (16- и 8-портовые модификации)

Из продуктов серии NSG-800 к данной конфигурации относятся NSG-800 *h/w ver.2.3*, а также отдельные партии других устройств NSG-800, поставленные с Flash ROM 8 МБ.

Для данной аппаратной конфигурации поставляется специальная сборка NSG Linux, из которой исключены отдельные громоздкие компоненты — в данной версии, SSH. Сборка состоит из одного файла `nsg900-linux-sumo.bin`. Рекомендуется приобрести для данных устройств недорогой модуль расширения энергонезависимой памяти FLEX (FLash EXtender) и установить полную версию.

— **Устройства с процессором Motorola MPC 862/860/855T, Flash ROM 4 или 8 МБ и установленным модулем расширения энергонезависимой памяти Disc-on-Chip (DoC) либо FLEX:**

NSG-900/2WL	NSG-800/WL (1- и 2-портовые модификации)
NSG-900/4WL	NSG-800/4WL
NSG-900/16A (16- и 8-портовые модификации)	NSG-800/16A (16- и 8-портовые модификации)

Для данных конфигураций поставляется полная сборка NSG Linux, состоящая из двух файлов:

`nsg900-linux-core.bin` и `nsg900-linux-doc.tar.gz` — для системы с модулем DoC  
`nsg900-linux-core.bin` и `nsg900-linux-flex.tar.gz` — для системы с модулем FLEX

— **Устройства с процессором Atmel AT91RM9200 и Flash ROM 8 МБ:**

NSG-700/4AU, NSG-700/8A, NSG-700/12A (первая серийная партия *h/w ver.2*, выпуска IV кв. 2006 г.)

Для данной конфигурации поставляется специальная сборка NSG Linux без поддержки SSH. Сборка состоит из одного файла `nsg700-linux-sumo8.bin`. Такое название файла использовано в порядке исключения и только в данной версии — чтобы разрешить конфликт с полной сборкой для устройств с Flash ROM 16 МБ, составляющих основную часть выпуска. В последующих версиях NSG Linux для данного устройства будет необходимо расширение энергонезависимой памяти с помощью USB Flash, а сборка будет состоять из двух файлов `nsg700-linux-core.bin` и `nsg700-linux-usbflash.tar.gz`.

— **Устройства с процессором Atmel AT91RM9200 и Flash ROM 16 МБ или более:**

Вся серия NSG-700, кроме первой партии

Для данной конфигурации текущая версия NSG Linux поставляется полностью и состоит из одного файла `nsg700-linux-sumo.bin`. Предполагается, что размер Flash ROM достаточен для дальнейшего развития NSG Linux, и такой формат поставки сохранится в обозримой перспективе.

— **Устройства NSG-900/maxU (прежнее название — NSG-800/maxU), NSG-1000.**

Для данных устройств в настоящей версии поставляются отдельные сборки NSG Linux. Набор файлов для последующих версий будет указан позже.

Помимо этого, для работы NSG Linux необходим новый системный загрузчик U-Boot, поставляемый в виде файла:

`u-boot-nsg900.bin` — для устройств NSG-900, NSG-800  
`u-boot-nsg700.bin` — для устройств NSG-700

Взаимодействие различных компонент программного обеспечения рассмотрено в п.1.5.3. Установка или обновление U-Boot обязательно при установке данной версии NSG Linux на любые устройства NSG-700, NSG-800, NSG-900, в том числе:

- NSG-800, NSG-900, ранее оснащенные базовым программным обеспечением NSG
- NSG-800, NSG-900, ранее оснащенные версиями NSG Linux 0.x.x
- поставленные ранее с версиями NSG Linux 1.0.0 beta x

Все вышеперечисленные файлы можно загрузить с Web-сайта компании по адресам:

<http://www.nsg.ru/nsg-linux/binary/>  
<ftp://ftp.nsg.net.ru/pub/nsg-linux/binary/>

Для устройств NSG–800, NSG–900 версия системного загрузчика (NSG Boot) и объем RAM выводятся на консоль при старте системы (см. п.1.5.7). В NSG Boot v5.4 или старше выводится также объем Flash ROM. Устройства, оснащенные NSG Boot более ранних версий, всегда имеют Flash ROM 2 МБ. Пример вывода:

```

=== NSG Boot Version 5.4 (Dec 9 2002) ===           ← версия системного загрузчика
=== Copyright (C) 2002 by Network Systems Group Ltd. ===
EXTCLK:4 MHz
DRAM: 32 MB                                         ← размер RAM
Load boot...OK
Check boot...OK
Start boot
Flash info:
  Manufacturer: AMD
  Device: AM29LV033C
  Size: 4096 Kb                                     ← размер Flash ROM

Test Real Time Clock ...OK
Date: 09:27:54 Thu,17.04.03
Press <Esc> and <Enter> for Boot Setup...Continue

```

Устройства NSG–800 ранних партий с объемом Flash ROM 2 МБ не поддерживаются ни при каких условиях. Модернизация процессорного ядра устройств NSG (увеличение объема Flash ROM, обновление системного загрузчика и т.п.) не предусмотрена. Реализация NSG Linux на платформах NPS–7e, NSG–500, NX–300 также не представляется возможной.

Для устройств, оснащенных загрузчиком U-Boot (в т.ч. NSG–700 во всех случаях), при старте системы на консоль также выводятся версия загрузчика, объем RAM и ROM:

```

U-Boot 1.1.4 (Apr 24 2007 - 11:31:19) (NSG build 1) ← версия системного загрузчика

U-Boot code: 23FC0000 -> 23FD926C BSS: -> 23FF6204
RAM Configuration:
Bank #0: 20000000 64 MB                             ← размер RAM
Flash: 16 MB                                         ← размер Flash ROM
.....

```

### §1.5.2. Установка расширения энергонезависимой памяти

Если модуль расширения энергонезависимой памяти приобретен пользователем отдельно для модернизации имеющегося устройства, то его установка производится самостоятельно следующим образом:

**Для модулей DoC (Disc-on-Chip) или FLEX (FLash EXtender) устройствах NSG–800 или NSG–900 (кроме NSG–900/maxU, NSG–900/8WL)**

1. Выключить питание устройства и отключить его от сети.
2. Открыть верхнюю крышку устройства, следуя указаниям Руководства пользователя по данному устройству.
3. Не вынимая процессорную плату из материнской, сориентировать устройство так, чтобы разъемы консольного порта и Fast Ethernet были справа.
4. Взять плату DoC или FLEX так, чтобы название платы на её верхней стороне читалось нормально (не вверх ногами).
5. Вставить плату DoC или FLEX в процессорную плату в крайнее левое положение. С правой стороны на процессорной плате должны остаться свободные пары контактов. Эти контакты не используются.
6. Установить верхнюю крышку устройства и зафиксировать ее винтами.

**Для модулей DoC (Disc-on-Chip) в устройствах NSG–900/maxU, NSG–900/8WL:**

1. Выключить питание устройства и отключить его от сети.
2. Извлечь процессорный модуль (NSG–900/maxU) или ведущую плату (NSG–900/8WL), следуя указаниям Руководства пользователя по данному устройству.
3. Определить местонахождение гнезда для микросхемы Disc-on-Chip.
4. Сориентировать микросхему таким образом, чтобы ее ключ (скошенный угол) находился с той же стороны, что и ключ на гнезде (выемка на поперечном элементе).
5. Вставить контакты микросхемы в гнездо. Визуально проверить совпадение всех контактов и зажимов гнезда, отсутствие сдвигов и перекосов.
6. Вставить микросхему в гнездо до упора.
7. Установить процессорный модуль или плату на место.

**Для модулей USB Flash в устройствах NSG–700:**

1. Выключить питание устройства и отключить его от сети.
2. Открыть верхнюю крышку устройства, следуя указаниям Руководства пользователя по данному устройству.
3. Вставить USB Flash во внутренний разъем USB.
4. Установить верхнюю крышку устройства и зафиксировать ее винтами.

**§1.5.3. Компоненты NSG Linux для различных аппаратных конфигураций**

Расположение компонент NSG Linux и другого программного обеспечения в энергонезависимой памяти варьируется в зависимости от модели шасси и наличия в нем дополнительных модулей и устройств памяти.

**а) Устройства NSG–900, NSG–800**

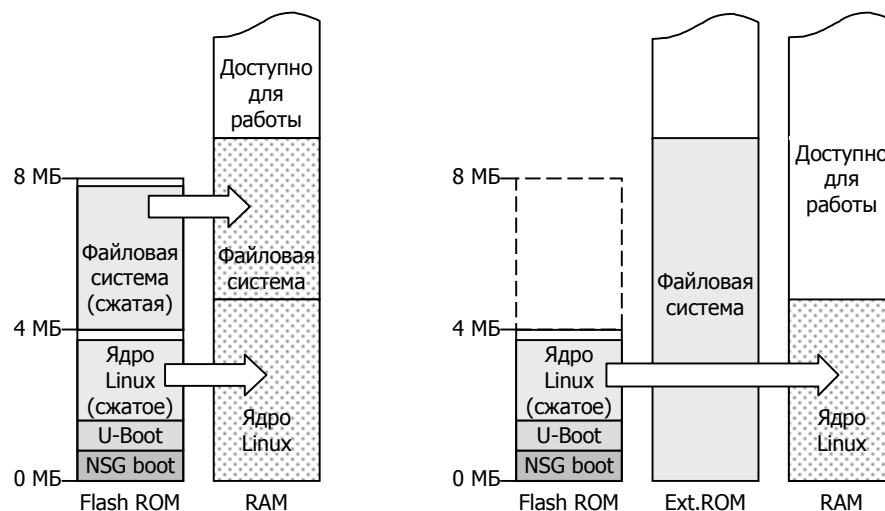
В устройствах NSG–900, NSG–800 компоненты располагаются в следующем порядке:

- Системный загрузчик NSG Boot. Данный загрузчик не требуется для работы NSG Linux, однако сохранен для совместимости с базовым программным обеспечением NSG. С его помощью производится установка расширенного загрузчика U-Boot при переходе с базового ПО или с версий NSG Linux 0.x.x. Возможна также переустановка U-Boot или установка базового ПО вместо NSG Linux.

**ВНИМАНИЕ** При установке и переустановке программного обеспечения посредством NSG Boot стирается и перезаписывается весь объем Flash ROM, за исключением самого загрузчика. Никакие другие компоненты программного обеспечения и конфигурации не сохраняются.

Сам загрузчик защищен от стирания и записи и не может быть поврежден в процессе установки ПО. Это гарантирует возможность восстановления работоспособности устройства при любых нарушениях в ходе выполнения данной процедуры.

- Загрузчик с расширенными возможностями U-Boot и его конфигурация. Предназначен для первичной установки NSG Linux и его запуска. Содержит некоторые ключевые параметры, передаваемые ядру NSG Linux при запуске (например, скорость консольного порта). Может также использоваться для переустановки и модернизации NSG Linux, удаления пользовательской конфигурации и других сервисных операций.
- Ядро NSG Linux и набор основных утилит— в виде сжатого архива. Под эту компоненту отводится вся оставшаяся часть Flash ROM до границы 4 МБ. Такое разбиение ROM обусловлено необходимостью единой схемы распределения памяти для устройств с размером Flash ROM 4 и 8 МБ, с расширением энергонезависимой памяти или без него. В настоящее время время указанная область памяти используется не полностью, что создает резерв для дальнейшего развития ядра NSG Linux на платформе NSG–800. С другой стороны, такое разбиение приводит к дальнейшему ограничению объема прикладной части NSG Linux на устройствах без расширенной памяти.



**Распределение памяти в устройствах NSG–800, NSG–900 с расширением энергонезависимой памяти (справа) и без него (слева)**

- Остальную часть NSG Linux составляет файловая система, в которой хранятся приложения и конфигурация. В зависимости от наличия расширенной энергонезависимой памяти, файловая система располагается следующим образом:

- В устройствах с модулем DoC или FLEX — в развернутом виде на указанном модуле. При этом для работы системы доступен наибольший объем оперативной памяти.

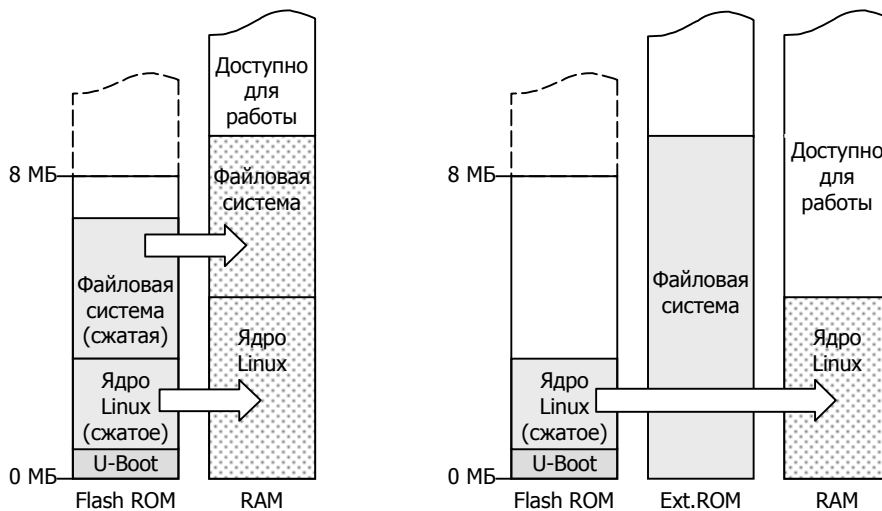
- В устройствах с Flash ROM 8 МБ без расширения — в виде сжатого архива в старших 4 МБ ROM. При запуске файловая система распаковывается на виртуальный диск, который создается в оперативной памяти устройства, и с него запускается на исполнение. Следует учитывать, что виртуальный диск может занимать существенную часть оперативной памяти (до 16 МБ), ограничивая, таким образом, память, доступную для работы системы. Это может быть особенно заметно при работе ресурсоемких приложений, таких как протокол BGP.

## б) Устройства NSG-700

В серии NSG-700 штатным загрузчиком является U-Boot, устанавливаемый на устройство при его изготовлении. С помощью загрузчика может также производиться переустановка и модернизация NSG Linux, удаление пользовательской конфигурации и другие сервисные операции. Загрузчик защищен от перезаписи и не может быть поврежден в процессе установки основного ПО.

После загрузчика располагаются ядро Linux и файловая система — то и другое в сжатом виде. Суммарный объем ПО в данной версии укладывается в 8 МБ Flash ROM. При запуске файловая система распаковывается на виртуальный диск, который создается в оперативной памяти устройства, и с него запускается на исполнение. Следует учитывать, что виртуальный диск может занимать существенную часть оперативной памяти (до 16 МБ), ограничивая, таким образом, память, доступную для работы системы. Это может быть особенно заметно при работе ресурсоемких приложений, таких как протокол BGP.

В устройствах, оборудованных расширением энергонезависимой памяти (USB Flash либо USB HDD), в основную Flash ROM может устанавливаться только ядро NSG Linux, а развернутая файловая система — в расширенной памяти. При этом для работы системы доступен больший объем оперативной памяти.

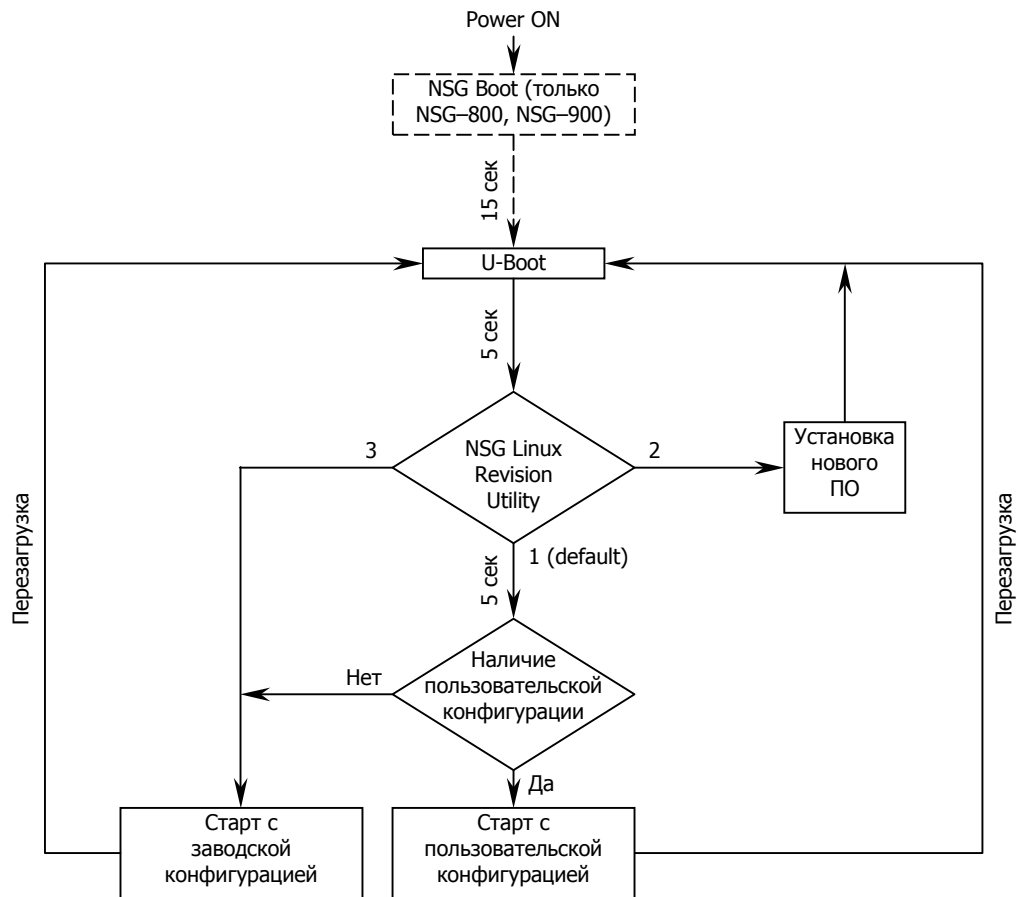


**Распределение памяти в устройствах NSG-700  
с расширением энергонезависимой памяти (справа) и без него (слева)**

При загрузке системы программное обеспечение работает в следующей последовательности:

1. Загрузчик NSG Boot. После запуска делается пауза 15 сек., во время которой возможно нажать клавиши Esc и затем Enter для входа в меню загрузчика.
2. Загрузчик U-Boot. После запуска делается пауза 5 сек., во время которой возможно нажать любую клавишу для входа в меню загрузчика. Если за это время не нажата никакая клавиша, и если после U-Boot найдено ядро NSG Linux, то запускается основное программное обеспечение.
3. Программа NSG Linux Revision Utility, входящая в состав основного программного обеспечения. После запуска делается пауза 5 сек., во время которой возможно выбрать одну из возможных опций: стартовать систему с пользовательской конфигурацией, обновить/доустановить программное обеспечение, либо стартовать с заводской конфигурацией. Если за 5 секунд не выбрана никакая опция, система стартует с пользовательской конфигурацией. Если пользовательская конфигурация отсутствует, система стартует с заводской конфигурацией.
4. Если выбран пункт 2 (обновление ПО), то производится установка новой версии ядра и/или файловой системы с помощью последовательности интерактивных меню, и затем — перезагрузка системы.
5. Основное программное обеспечение NSG Linux.

В устройствах NSG-800, NSG-900 при включении питания данная последовательность выполняется полностью. В устройствах NSG-700 процедура начинается с пункта 2. При перезагрузке системы (как аппаратной — кнопкой Reset, так и программной — командами reload в основной командной оболочке или reboot в ОС Linux) процедура также выполняется, начиная с пункта 2. График загрузки показан на рисунке на следующей странице.



Последовательность загрузки и перезагрузки NSG Linux

#### §1.5.4. Сценарии установки и модернизации NSG Linux

Установка и модернизация NSG Linux может производиться различными способами, в зависимости от поставленной задачи и исходных условий.

##### а) Штатная переустановка или установка новой версии NSG Linux на устройстве с NSG Linux v1.0 build 0 и старше

Производится при установке новой версии NSG Linux, а также при повреждении существующей версии. Процедура состоит из нескольких этапов, выполняемых с помощью NSG Linux Revision Utility. Пользовательская конфигурация при данной процедуре сохраняется.

##### б) Нештатная переустановка или установка новой версии NSG Linux на устройстве с NSG Linux v1.0 build 0 и старше

Требуется только при повреждении существующей версии NSG Linux Revision Utility и/или ядра. Процедура состоит из следующих этапов:

- Установка NSG Linux полностью (для устройств с Flash ROM 8 МБ без расширения), либо только ядра (для устройств с модулем DoC или FLEX). Выполняется с помощью загрузчика U-Boot.
- Установка файловой системы (только для устройств с модулем DoC или FLEX). Выполняется с помощью NSG Linux Revision Utility.

Пользовательская конфигурация при данной процедуре сохраняется, если только пользователь не удаляет ее преднамеренно средствами U-Boot.

##### в) Полная установка или переустановка U-Boot и NSG Linux

Производится при первичной установке NSG Linux на устройства NSG-800, NSG-900 с базовым ПО NSG, при переходе на версию 1.0 build 0 или старше с предыдущих версий NSG Linux, а также при повреждении загрузчика U-Boot. Процедура установки состоит из следующих этапов:

- Установка расширенного загрузчика U-Boot с помощью системного загрузчика NSG Boot.
- Установка NSG Linux полностью (для устройств с Flash ROM 8 МБ без расширения), либо только ядра (для устройств с модулем DoC или FLEX). Выполняется с помощью загрузчика U-Boot.
- Установка файловой системы (только для устройств с модулем DoC или FLEX). Выполняется с помощью NSG Linux Revision Utility.

**ВНИМАНИЕ** При установке и переустановке программного обеспечения посредством NSG Boot стирается и перезаписывается весь объем Flash ROM, за исключением самого загрузчика. Никакие другие компоненты программного обеспечения и конфигурации не сохраняются.

**ВНИМАНИЕ** При переходе на новую версию возможны исключения из принципа обратной совместимости, т.е. отдельные элементы существующей конфигурации могут быть удалены или изменены в новой версии. Перед модернизацией необходимо внимательно ознакомиться со списком несовместимых параметров конфигурации, приведенным в Приложении 1–Г.

#### г) Переустановка U-Boot средствами самого U-Boot

Производится в исключительных случаях для установки новой версии загрузчика U-Boot на устройствах NSG–700 и последующих серий, не оснащенных загрузчиком NSG Boot. Также может производиться на устройствах NSG–800, NSG–900 для обновления одного только U-Boot.

Пользовательская конфигурация при данной процедуре сохраняется, если только пользователь не удаляет ее преднамеренно средствами U-Boot.

#### д) Аварийная переустановка U-Boot средствами низкоуровневого загрузчика

Производится на устройствах NSG–700 в самых исключительных случаях, когда загрузчик U-Boot поврежден, например, в процессе переустановки его собственными средствами. Для восстановления работоспособности устройства используется встроенный низкоуровневый аппаратный загрузчик и утилита, которая принимает двоичный файл U-Boot и записывает его в энергонезависимую память.

#### е) Установка и переустановка NSG Linux на устройстве NSG–1000

Для устройства NSG–1000 поставляется отдельная сборка NSG Linux. Процедура установки программного обеспечения является специфической для данного устройства и описана в Руководстве пользователя NSG–1000. Пользовательская конфигурация при данной процедуре сохраняется.

Управление установкой программного обеспечения производится локально через консольный порт. Файлы нового ПО загружаются, как правило, через порт Fast Ethernet по протоколу TFTP. Исключением является полная установка на устройства NSG–800, NSG–900 с версией системного загрузчика NSG Boot ниже 7.0. Данные версии не поддерживают загрузку по TFTP, поэтому для них используется загрузка U-Boot через консольный порт по протоколу Xmodem. В остальных случаях загрузка через консольный порт возможна, но нецелесообразна по причине большого размера файлов и низкой скорости.

Удаленная модернизация программного обеспечения по сети в данной версии NSG Linux предусмотрена только для устройства NSG–1000.

Подробное описание каждого из этапов установки приведено в последующих параграфах.

### §1.5.5. Подготовка к установке U-Boot: вход в меню системного загрузчика

Системный загрузчик (NSG Boot) на устройствах NSG–800, NSG–900 представляет собой вспомогательную программу, постоянно хранящуюся в энергонезависимой памяти и управляющую запуском остального программного обеспечения. С его помощью на устройства может быть установлено и переустановлено как базовое программное обеспечение, так и NSG Linux в любой последовательности.

Системный загрузчик хранится в отдельной области энергонезависимой памяти, недоступной для записи, и сохраняет работоспособность в любом случае, если не повреждена аппаратная часть устройства. Изменение, удаление или повреждение NSG Boot средствами, доступными пользователю, невозможно.

Если системный загрузчик недоступен средствами, описанными в данном разделе, восстановление работоспособности устройства возможно только в заводских условиях в компании NSG.

Для работы с меню загрузчика требуется асинхронный терминал. Далее предполагается, что в качестве асинхронного терминала используется персональный компьютер с COM-портом.

**ВНИМАНИЕ** Для подключения терминала к консольному порту устройств NSG необходимо использовать кабель, прилагаемый к устройству. Консольные порты устройств различных серий имеют различное назначение контактов (и, в некоторых случаях, тип разъема) и несовместимы друг с другом. Типы консольных кабелей NSG и их распайка приведены в Руководствах пользователя по соответствующим устройствам.

1. Отключить питание устройства. Присоединить COM-порт компьютера к консольному порту устройства кабелем, входящим в комплект устройства.
2. Запустить на ПК терминальную программу и установить в ней следующие параметры для данного COM-порта:

Baud Rate	9600
Parity	None (отсутствует)
Data Bits	8
Stop Bits	1
Flow Control	None

3. Включить питание устройства. На экране ПК появится подсказка для входа в загрузчик:

```

===      NSG Boot  Version #.# (xxx ## ####)      ===
=== Copyright (C) ##### by Network Systems Group Ltd. ===
DRAM: ## MB
Load boot...OK
Check boot...OK
Start boot
Test Real Time Clock ...OK
Date: ##:##:## xxx,##.##.##
Press <Esc> and <Enter> for Boot Setup...

```

4. Нажать на клавиатуре компьютера клавишу Esc, затем клавишу Enter. На экране компьютера появится меню загрузчика и приглашение:

```

*** Boot Setup ***
 1. Read time
 2. Set time
 3. Update software
 4. Set current partition
 5. Test DRAM
 6. Test LED
 7. Cold Start
 8. Continue
>

```

Пункты меню выбираются цифровыми клавишами на клавиатуре компьютера. Пункты 1 и 2 производят просмотр и установку системного времени. Пункт 3 предназначен для модернизации программного обеспечения (см. следующий параграф). Пункт 4 зарезервирован для дальнейшего использования. Пункты 5 и 6 осуществляют тестирование оперативной памяти и светодиодов устройства при помощи очевидных текстовых подсказок и меню. Пункт 7 выполняет процедуру "холодный старт" и не относится к NSG Linux.

Для выхода из меню загрузчика надо либо выбрать в меню пункт 8 (Continue), либо выключить и включить питание для перезагрузки устройства. Если через 30 секунд не будет выбран какой-либо пункт меню, устройство автоматически переходит к загрузке и запуску программного обеспечения в обычном режиме (что равносильно выбору пункта 8).

### §1.5.6. Установка U-Boot через порт Fast Ethernet

В устройствах NSG-800, NSG-900, оснащенных версией системного загрузчика NSG Boot 7.0, файл с новым программным обеспечением может быть загружен через порт Fast Ethernet по протоколу TFTP, при локальном управлении через консольный порт. Операция производится в следующей последовательности:

1. Сохранить файл u-boot-nsg900.bin на сервере TFTP. Сервер должен быть доступен из локальной сети, в которой находится устройство NSG, непосредственно либо через сеть IP.
2. Подключить встроенный порт Fast Ethernet устройства к локальной сети.
3. Войти в меню загрузчика (см. п.1.5.5).
4. Выбрать в меню пункт 3 (Update software). Появится меню выбора способа загрузки:

```

+++ Update Software +++
 1. Xmodem
 2. TFTP
Select protocol[2]>

```

Для приёма нового программного обеспечения по протоколу TFTP следует нажать клавишу 2 или Enter. После этого системный загрузчик последовательно предложит установить минимальный набор параметров, необходимый для получения файла с сервера. Параметры, отмеченные звёздочкой в квадратных скобках ([\*]), являются обязательными; для необязательных параметров вместо звёздочки показывается значение по умолчанию.

Server address[\*]:

Необходимо ввести IP-адрес сервера TFTP в десятичной дотовой нотации.

Server port[69]:

Номер порта UDP, на котором работает сервер TFTP. По умолчанию используется стандартный порт 69.

IP address[\*]:

Необходимо ввести IP-адрес, который будет временно назначен устройству NSG для данного сеанса TFTP.

Gateway address[0.0.0.0]:

Если сервер TFTP находится вне локальной сети, необходимо ввести IP-адрес шлюза, через который проходит маршрут к нему. Если устройство NSG и сервер TFTP находятся в одной локальной сети, указывать шлюз не требуется.

File name[\*]: u-boot-nsg900.bin

Необходимо ввести имя файла, который должен быть запрошен с сервера.

Таким образом, устройство NSG должно находиться в одной физической локальной сети либо с самим сервером TFTP, либо с указанным шлюзом. Временный IP-адрес устройства может быть любым, однако на сервере и шлюзе должен быть известен маршрут к этому адресу. Рекомендуется выбирать его в пределах адресного пространства существующей сети.

**ВНИМАНИЕ** Ввод каждого из вышеперечисленных параметров TFTP завершается нажатием клавиши Enter. Время для ввода параметров ограничено 15 секундами. Если за это время ответ на очередной запрос загрузчика не введен, загрузчик возвращается в исходное меню.

- После ввода имени файла начинается загрузка указанного файла с сервера. Загрузка сопровождается диагностическими сообщениями. По окончании загрузки выводится сообщение:

```
Image received. Checking...
```

Проверяется контрольная сумма полученного файла. Если она совпадает, то после многоточия выводится OK. При неудачном завершении появится сообщение об ошибке. Обновление ПО прерывается, при этом в энергонезависимой памяти устройства останутся без изменений прежняя версия ПО и текущая конфигурация. Например, при приеме программного обеспечения, предназначенного для другого устройства, появится сообщение:

```
Image received. Checking...Failure
!!! Invalid header
```

- После успешного приема нового программного обеспечения будет выведено сообщение о его версии и запрошено подтверждение для замены прежней версии в энергонезависимой памяти устройства:

```
Label: NSG u-boot
*** ATTENTION ***
Now image will be written into non-volatile memory
Old image will be erased!
Are you sure ?
```

На этом этапе можно отказаться от модернизации, если нажать клавишу N. Появится сообщение:

```
!!! Image updating aborted
```

Процедура обновления прерывается, при этом в энергонезависимой памяти устройства останутся без изменений прежняя версия программного обеспечения и текущая конфигурация.

Чтобы подтвердить запись новой версии в энергонезависимую память, следует нажать клавишу Y. Появится сообщение:

```
*** ATTENTION: Do not power off or reset until image updating finished ***
```

и начнется стирание энергонезависимой памяти.

**ВНИМАНИЕ** Категорически запрещается отключать питание устройства и нажимать кнопку Reset во время стирания или записи энергонезависимой памяти. Несоблюдение этого требования может привести к повреждению микросхем энергонезависимой памяти и, как следствие, к утрате работоспособности устройства.

После завершения стирания появится сообщение:

```
Erasing...OK
```

Это означает, что прежняя версия программного обеспечения и прежняя конфигурация удалены из энергонезависимой памяти устройства. После завершения стирания начнется запись нового программного



обеспечения из оперативной в энергонезависимую память устройства. Во время записи будет выводиться индикатор хода операции.

При успешном завершении записи появится сообщение:

```
Writing...OK
Updating successfully finished.
```

и приглашение системного загрузчика (>). Для продолжения работы следует нажать клавишу 8, после чего запускается загрузчик U-Boot:

```
>Continue
Current Partition #0
Label:NSG u-boot
Load...OK
Start.

U-Boot 1.1.4 (Nov 8 2006 - 11:20:38)
```

```
.....
Hit any key to stop autoboot: 5
```

Дальнейшая процедура установки NSG Linux описана в п.1.5.8.

### §1.5.7. Установка U-Boot через консольный порт

Установка U-Boot через консольный порт требуется только для устройств NSG-800, NSG-900 ранних выпусков с версиями системного загрузчика NSG Boot ниже 7.0, не поддерживающими загрузку файлов по TFTP. Данная процедура также может использоваться в случае отсутствия доступного сервера TFTP.

Терминальная программа должна поддерживать протокол Xmodem.

1. Сохранить файл u-boot-nsg900.bin на локальном ПК.
2. Войти в меню загрузчика (см. п.1.5.5).
3. Выбрать в меню пункт 3 (Update software). Если устройство оснащено системным загрузчиком версии 7.0 или старше, появится меню выбора способа загрузки:

```
+++ Update Software +++
  1. Xmodem
  2. TFTP
Select protocol[2]>
```

Для приема нового программного обеспечения по протоколу Xmodem следует нажать клавишу 1. (По умолчанию в версии загрузчика 7.0 и старше предполагается TFTP.) В более ранних версиях системного загрузчика это меню отсутствует.

4. Выбрать скорость консольного порта для загрузки:

```
+++ Update Software +++
  0. 9600
  1. 19200
  2. 38400
  3. 57600
  4. 115200
  5. 230400
Select speed[0]>
```

Объем передаваемого файла невелик (макс. 128 КБ), поэтому выбор скорости не играет в данном случае существенной роли. Если выбрана скорость, отличная от 9600 Кбит/с, появится сообщение вида:

```
!!! Current rate will be set to 115200 bps !!!
*** Change baud rate on your terminal and press <Enter> ***
```

Необходимо изменить в настройках терминальной программы скорость COM-порта на 115200 бит/с. (При этом для некоторых терминальных программ необходимо разорвать соединение и установить его заново.) После этого следует нажать клавишу Enter, появится сообщение:

```
+++ Your current baud rate is 115200 bps +++
Please, start XModem sender. XModem receiving started...
```

Устройство готово к приёму нового программного обеспечения по протоколу Xmodem.

**ВНИМАНИЕ** После вывода вышеуказанного сообщения устройство ожидает начала передачи по протоколу Xmodem 1K в течение 15 или 20 сек (в зависимости от версии загрузчика), и затем по обычному протоколу Xmodem в течение 15 или 20 сек. Если за это время передача файла не будет начата, загрузчик возвращается в исходное меню.

5. Инициировать в терминальной программе передачу (*upload*) файла *u-boot-nsg900.bin* по протоколу Xmodem. Получаемый файл сохраняется в оперативной памяти устройства. После того, как файл принят, появится сообщение о результате. При успешном завершении операции оно выглядит следующим образом:

```
Image received. Checking...OK
```

При неудачном завершении появится сообщение об ошибке. Обновление ПО прерывается, при этом в энергонезависимой памяти устройства останутся без изменений прежняя версия ПО и текущая конфигурация. Например, при приёме программного обеспечения, предназначенного для другого устройства, появится сообщение:

```
Image received. Checking...Failure
!!! Invalid header
```

7. После успешного приёма нового программного обеспечения будет выведено сообщение о его версии и запрошено подтверждение для замены прежней версии в энергонезависимой памяти устройства:

```
Label: NSG u-boot
*** ATTENTION ***
Now image will be written into non-volatile memory
Old image will be erased!
Are you sure ?
```

На этом этапе можно отказаться от модернизации, если нажать клавишу N. Появится сообщение:

```
!!! Image updating aborted
```

Процедура обновления прерывается, при этом в энергонезависимой памяти устройства останутся без изменений прежняя версия программного обеспечения и текущая конфигурация.

Чтобы подтвердить запись новой версии в энергонезависимую память, следует нажать клавишу Y. Появится сообщение:

```
*** ATTENTION: Do not power off or reset until image updating finished ***
```

и начнётся стирание энергонезависимой памяти.

**ВНИМАНИЕ** Категорически запрещается отключать питание устройства и нажимать кнопку Reset во время стирания или записи энергонезависимой памяти. Несоблюдение этого требования может привести к повреждению микросхем энергонезависимой памяти и, как следствие, к утрате работоспособности устройства.

После завершения стирания появится сообщение:

```
Erasing...OK
```

Это означает, что прежняя версия программного обеспечения и прежняя конфигурация удалены из энергонезависимой памяти устройства. После завершения стирания начнётся запись нового программного обеспечения из оперативной в энергонезависимую память устройства. Во время записи будет выводиться индикатор хода операции.

При успешном завершении записи появится сообщение:

```
Writing...OK
Updating successfully finished.
```

и приглашение системного загрузчика (>). Для продолжения работы следует нажать клавишу 8, после чего запускается загрузчик U-Boot:

```
>Continue
Current Partition #0
Label:NSG u-boot
Load...OK
Start.

U-Boot 1.1.4 (Nov  8 2006 - 11:20:38)

.....
Hit any key to stop autoboot:  5
```

Дальнейшая процедура установки NSG Linux описана в п.1.5.8.

### §1.5.8. Установка и модернизация NSG Linux средствами U-Boot

Для входа в меню загрузчика U-Boot необходимо нажать любую клавишу в течение 5 сек. после появления подсказки:

Hit any key to stop autoboot: <секунды>

U-Boot представляет собой загрузчик с широкими программными возможностями. Для просмотра списка команд можно использовать команду `help` или `?`, для получения справки по конкретным командам — `help <команда>`, для просмотра установленных переменных окружения — команду `printenv`. Подробное описание команд U-Boot и используемых переменных окружения приведено в Приложении 1–Б.

**ВНИМАНИЕ** В отдельных случаях для установки новой версии NSG Linux может быть необходимо предварительно обновить загрузчик U-Boot. Критерии для определения такой необходимости, файл с новой версией U-Boot и детальная инструкция по выполнению этой операции поставляются вместе с очередной версией NSG Linux. Общий ход процедуры описан в п.1.5.10.

Загрузка файлов производится, как правило, посредством протокола BOOTP/DHCP. Устройство рассылает по локальной сети широковещательный запрос, в ответ на который оно получает назначенный ему IP-адрес, IP-адрес сервера и другие параметры IP-окружения. После этого происходит загрузка требуемого файла по TFTP с назначенного таким образом сервера.

Если загрузка будет производиться непосредственно по TFTP (т.е. без использования DHCP), то используются адреса, назначенные по умолчанию:

— Клиент (т.е. устройство NSG) — 192.168.0.1  
— Сервер — 192.168.0.250

Эти адреса хранятся в переменных окружения `ipaddr` и `serverip`, соответственно. При необходимости они могут быть изменены пользователем.

Для устройств, не оснащенных расширенной энергонезависимой памятью, на данном этапе производится загрузка NSG Linux полностью. Для устройств, оснащенных модулями DoC, FLEX, USB Flash или USB HDD, загружается только ядро и набор основных утилит, после чего файловая система доустанавливается средствами NSG Linux Revision Utility (см. следующий параграф). Загрузка NSG Linux средствами U-Boot производится в следующем порядке:

1. Сохранить на сервере BOOTP или TFTP требуемый файл программного обеспечения в зависимости от модели и аппаратной конфигурации устройства NSG:

— NSG–800, NSG–900 без расширенной энергонезависимой памяти:	<code>nsg900-linux-sumo.bin</code>
— NSG–800, NSG–900 с расширенной энергонезависимой памятью:	<code>nsg900-linux-core.bin</code>
— NSG–700 с Flash Rom 16 МБ без расширенной энергонезависимой памяти:	<code>nsg700-linux-sumo.bin</code>
— NSG–700 с Flash Rom 8 МБ без расширенной энергонезависимой памяти:	<code>nsg700-linux-sumo8.bin</code>
— NSG–700 с расширенной энергонезависимой памятью:	<code>nsg700-linux-core.bin</code>

Сервер должен быть доступен из локальной сети, в которой находится устройство NSG, непосредственно либо через сеть IP.

2. Если загрузка будет производиться без использования протокола BOOTP, то назначить серверу IP-адрес 192.168.0.250, маску 255.255.255.0 и удостовериться, что никакое другое устройство в сети не использует адрес 192.168.0.1. В противном случае следует установить переменные окружения в соответствии с фактической конфигурацией сети:

```
setenv ipaddr <ip-адрес устройства NSG>
setenv serverip <ip-адрес сервера TFTP>
```

3. Подключить встроенный порт Fast Ethernet устройства к локальной сети.

4. Если производится установка NSG Linux средствами U-Boot поверх существующей версии, очистить энергонезависимую память при помощи одной из следующих команд:

— Для устройств NSG–700 с Flash Rom 8 МБ:	<code>run nsgerase_sumo8</code>
— Для всех других устройств без расширенной энергонезависимой памяти:	<code>run nsgerase_sumo</code>
— Для устройств с расширенной энергонезависимой памятью:	<code>run nsgerase_core</code>

При полной установке на устройствах NSG–800, NSG–900 данные команды не требуются, поскольку вся свободная часть Flash ROM стирается системным загрузчиком NSG Boot в любом случае.

5. Загрузить и записать в энергонезависимую память файл программного обеспечения при помощи одной из следующих команд, соответственно:

- Для устройств NSG–700 с Flash Rom 8 МБ:
 

```
bootp ${nsgupdate_sumo8}
tftpboot ${nsgupdate_sumo8}
```
- Для всех других устройств без расширенной энергонезависимой памяти:
 

```
bootp ${nsgupdate_sumo}
tftpboot ${nsgupdate_sumo}
```
- Для устройств с расширенной энергонезависимой памятью:
 

```
bootp ${nsgupdate_core}
tftpboot ${nsgupdate_core}
```

Имя файла и начальный адрес в памяти, специфические для данного шасси, установлены в переменных окружения `nsgupdate_sumo` и `nsgupdate_core`, соответственно.

После ввода команды начинается прием и запись файла, сопровождаемые выводом индикатора процесса в виде знаков `#`. Один знак соответствует 5120 байт, одна строка (65 знаков) — 325 КБ, что позволяет оценить долю файла, загруженную на текущий момент. По завершении загрузки выводится сообщение:

```
done
Bytes transferred = <число байт>
```

и системное приглашение U-Boot (`>`).

**ПРИМЕЧАНИЕ** В процессе загрузки могут иметь место превышения времени ожидания для очередного пакета, которые отображаются на экране в виде символа "Т" и пробела. Это не является ошибкой и не сказывается на целостности полученного файла. Данные предупреждения следует игнорировать.

**ВНИМАНИЕ** Категорически запрещается отключать питание устройства и нажимать кнопку `Reset` во время стирания или записи энергонезависимой памяти. Несоблюдение этого требования может привести к повреждению микросхем энергонезависимой памяти и, как следствие, к утрате работоспособности устройства.

#### 6. Ввести команду:

```
reset
```

Устройство рестартует, и начинается загрузка NSG Linux. По завершении процесса выводится меню NSG Linux Revision Utility:

```
Welcome to NSG revision utility!
1) continue booting
2) update software
3) boot with default config
timeout='25', default_choice='1'
```

#### 7. Для устройств без расширенной энергонезависимой памяти установка NSG Linux завершена. Следует выбрать пункт 1 или 3 (при отсутствии прежней пользовательской конфигурации будет использоваться конфигурация по умолчанию) или дождаться срабатывания таймера. После этого запускается непосредственно основное программное обеспечение и выводится приглашение для входа в систему:

```
nsg login:
```

Для устройств с расширенной энергонезависимой памятью следует выбрать пункт меню 2 и приступить к установке файловой системы.

### §1.5.9. Установка и модернизация NSG Linux средствами Revision Utility

Применительно к данной версии NSG Linux, средствами Revision Utility производится только установка или модернизация файловой системы в расширенной энергонезависимой памяти устройств NSG–800, NSG–900, а также восстановление заводской конфигурации устройства. В остальных случаях модернизацию программного обеспечения следует производить с помощью загрузчика U-Boot.

Процедура установки файловой системы выполняется следующим образом.

1. Сохранить на сервере TFTP требуемый файл программного обеспечения: `nsg900-linux-doc.tar.gz` или `nsg900-linux-flex.tar.gz`, в зависимости от типа установленного модуля расширения энергонезависимой памяти.
2. Подключить встроенный порт Fast Ethernet устройства к локальной сети.
3. Включить или перезагрузить устройство NSG и дождаться появления меню NSG Linux Revision Utility:

```
Welcome to NSG revision utility!
1) continue booting
2) update software
3) boot with default config
timeout='25', default_choice='1'
```

4. Для установки или модернизации файловой системы выбрать пункт 2. Выбор должен быть произведен в течение 25 секунд, в противном случае загрузка устройства продолжается нормальным образом. После выбора пункта 2 выводится следующее меню:

```
Select Flash extension type:
```

- ```
1) DoC module
2) FLEX module
0) exit
```

```
timeout='25', default_choice='0'
```

5. Выбрать тип расширенной энергонезависимой памяти (DoC или FLEX) в зависимости от физически установленного модуля. Если вход в данное меню произошел по ошибке, следует выбрать пункт 0, в этом случае загрузка устройства будет продолжена нормальным образом.
6. После выбора типа модуля выполняются служебные операции для подготовки его к записи программного обеспечения. По завершении этого этапа выводится приглашение:

```
Preparing for tftp download...
```

```
Enter IP addresses and file name as follows:
```

```
<client_ip> <netmask> <server_ip> <doc.tar.gz_file>
```

```
Example (all arguments needed):
```

```
192.168.0.1 255.255.255.0 192.168.0.250 nsg900-linux-doc.tar.gz
```

```
timeout='600', default_choice='192.168.0.1 255.255.255.0 192.168.0.250 nsg900-linux-doc.tar.gz'
```

```
Enter parameters:
```

7. После этого приглашения необходимо ввести строку, содержащую:
- адрес, назначаемый устройству NSG для загрузки файла по TFTP
  - маску подсети
  - адрес сервера TFTP
  - имя загружаемого файла

Если все параметры соответствуют установленным по умолчанию, в ответ на приглашение можно просто нажать клавишу Enter. В противном случае необходимо ввести все четыре параметра в указанном порядке.

8. Вновь установленные параметры выводятся на экран для контроля. Если все они введены правильно, то на следующее приглашение:

```
Are these parameters correct? [yes/no]:
```

```
timeout='30', default_choice='yes'
```

необходимо ввести ответ `yes` или нажать клавишу Enter. После этого начнется непосредственно загрузка и установка программного обеспечения.

**ВНИМАНИЕ** Категорически запрещается отключать питание устройства и нажимать кнопку Reset во время стирания или записи энергонезависимой памяти. Несоблюдение этого требования может привести к повреждению микросхем энергонезависимой памяти и, как следствие, к утрате работоспособности устройства.

Если какой-либо из вышеприведенных параметров введен неправильно, то следует ввести ответ `no`. В этом случае будет повторно предложено ввести параметры. Отказаться от установки на данном этапе можно только посредством перезагрузки устройства.

9. В случае положительного ответа файл с программным обеспечением загружается на устройство и распаковывается в расширенную энергонезависимую память. По завершении процедуры выводится обычное приглашение для входа в систему:

```
nsg login:
```

Устройство полностью готово к работе.

Прежняя конфигурация устройства при модернизации программного обеспечения средствами NSG Revision Utility сохраняется. Для восстановления заводской конфигурации, если это необходимо, следует выбрать на шаге 3 пункт меню 3. В этом случае будет запрошено подтверждение:

```
Reset configuration to factory defaults? [yes/no]:
```

Если в ответ введено `yes`, то на устройствах с расширенной энергонезависимой памятью существующая конфигурация будет стерта и заменена конфигурацией по умолчанию; после этого система стартует уже с заводской конфигурацией. На устройствах, оснащенных только основным Flash ROM, заводская конфигурация загружается в оперативную память, но не записывается в ROM; чтобы сделать это изменение постоянным, необходимо сохранить ее командой `write file`, в противном случае исходная конфигурация остается в неприкосновенности.

При ответе `no` загрузка устройства будет продолжена нормальным образом.

### §1.5.10. Переустановка загрузчика U-Boot средствами U-Boot

Переустановка загрузчика U-Boot средствами самого U-Boot является штатной операцией и производится только в исключительных случаях, когда обновление U-Boot необходимо для установки новой версии основного программного обеспечения NSG Linux. Если такие ситуации возникают по мере развития программного обеспечения NSG, то очередная версия NSG Linux комплектуется, помимо обычных файлов, новым загрузчиком для соответствующей аппаратной платформы и детальной инструкцией по его установке. Процедура обновления U-Boot в целом одинакова для всех устройств, однако конкретные параметры — адреса и номера сегментов Flash ROM, названия файлов — являются специфическими для каждой платформы. В нижеприведенном примере рассматривается обновление U-Boot на устройстве NSG-700 с объемом Flash ROM 16 МБ.

Обновление U-Boot производится в следующем порядке:

1. Войти в меню U-Boot и посмотреть заводской MAC-адрес(а) устройства:

```
u-boot> printenv
.....
ethaddr=00:09:56:XX:XX:XX
.....
```

Полученный адрес необходимо записать на бумаге, чтобы восстановить его впоследствии.

2. Если для загрузки используются статические IP-адреса, установленные переменными окружения `ipaddr` и `serverip`, то рекомендуется проверить доступность TFTP-сервера, на котором находятся требуемые файлы:

```
u-boot> ping 192.168.0.250
host 192.168.0.250 is alive
```

3. Снять программную защиту от записи с областей, в которых расположены загрузчик и его конфигурации:

```
u-boot> protect off 10000000 1003ffff
Un-Protected 2 sectors
```

4. Стереть загрузчик и его конфигурацию:

```
u-boot> erase 10000000 1003ffff
.... done
Erased 2 sectors
```

Одновременно, в зависимости от конкретной ситуации, может быть целесообразно стереть конфигурацию устройства, хранящуюся в последующих ячейках памяти вплоть до `1007ffff`.

5. Записать новый загрузчик во Flash ROM (например, по TFTP):

```
u-boot> tftpboot 10000000 u-boot-nsg700.bin
TFTP from server 192.168.0.250; our IP address is 192.168.0.1
Filename 'u-boot-nsg700.bin'.
Load address: 0x10000000
Loading: #####
done
Bytes transferred = 103020 (1926c hex)
```

**ВНИМАНИЕ** Категорически запрещается отключать питание устройства и нажимать кнопку `Reset` во время стирания или записи энергонезависимой памяти. Несоблюдение этого требования может привести к повреждению микросхем энергонезависимой памяти и, как следствие, к утрате работоспособности устройства.

6. Рестартовать устройство:

```
u-boot> reset
```

и снова зайти в меню загрузчика. После рестарта работает уже новая версия загрузчика.

7. Восстановить заводской MAC-адрес, сохраненный на шаге 1:

```
u-boot> setenv ethaddr 00:09:56:0E:XX:XX
```

8. Проверить правильность установки MAC-адреса:

```
u-boot> printenv
```

В случае ошибки повторить шаг 7.

## 9. Сохранить конфигурацию загрузчика

```
u-boot> saveenv
Saving Environment to Flash...
Un-Protected 2 sectors
Erasing Flash...
.. done
Erased 2 sectors
Writing to Flash... done
Protected 2 sectors
```

## 10. Рестартовать устройство:

```
u-boot> reset.
```

На этом процедура обновления U-Boot завершена, и можно переходить к обновлению основного ПО.

В случае аварийного завершения данной операции, например, отключения электропитания на этапе стирания и записи U-Boot, доступ к устройству будет потерян безвозвратно. Для восстановления работоспособности устройства в этом случае необходима специальная процедура установки U-Boot средствами низкоуровневого аппаратного загрузчика, выполняемая, как правило, в заводских условиях. При отсутствии такой возможности следует руководствоваться инструкцией, приведенной в Приложении 1–В.

### §1.5.11. Резервирование и восстановление конфигурации

Для вывода текущей конфигурации устройства используется команда

```
(config)# write terminal
```

Команда находится в меню привилегированного режима и генерирует сценарий конфигурации для всех параметров, текущие значения которых отличны от настроек по умолчанию. Чтобы получить готовый конфигурационный файл (скрипт), следует направить вывод команды в файл или скопировать его из буфера терминальной программы, клиента Telnet. Полученный файл рекомендуется отредактировать, удалив все строки до первого и после последнего восклицательных знаков.

Для восстановления конфигурации следует подключиться к устройству с помощью эмулятора терминала или клиента Telnet и загрузить в него конфигурационный скрипт командой "передать текстовый файл".

Резервирование и восстановление конфигурации устройства можно также осуществлять средствами ОС Linux. Для этого следует либо подключиться к системе с именем root, либо в основной командной оболочке набрать команду start-shell в меню привилегированного режима.

Конфигурация устройства представляет собой набор текстовых файлов, хранящихся в директории /etc. Для резервирования следует сохранить ее на удаленном хосте при помощи FTP, TFTP или NFS. Примеры:

```
cd /
tar -czf myconfig.tar.gz /etc
tftp 192.168.0.2 -p -l nsgconfig.tar.gz -r nsgconfig.tar.gz
rm nsgconfig.tar.gz

mount -t nfs -o nolock 192.168.0.2:/config_backups /mnt/nfs
cd /
tar -czf /mnt/nfs/ nsgconfig.tar.gz /etc
```

Для восстановления конфигурации из резервной копии следует загрузить и распаковать соответствующий файл любым из перечисленных способов. Примеры:

```
cd /
tftp 192.168.0.2 -g -l nsgconfig.tar.gz -r nsgconfig.tar.gz-X.Y.Z
tar -xzf nsgconfig.tar.gz
rm nsgconfig.tar.gz

mount -t nfs -o nolock 192.168.0.2:/config_backups /mnt/nfs
cd /
tar -xzf /mnt/nfs/ nsgconfig.tar.gz
```

**ПРИМЕЧАНИЕ** Для упорядочения архива конфигураций, файлы с резервными копиями рекомендуется именовать в соответствии с административными именами устройств, установленными командой hostname.

### §1.5.12. Использование устройств NSG–900 с базовым программным обеспечением NSG

Следующие модели серии NSG–900 могут использоваться под управлением как NSG Linux, так и базового программного обеспечения от аналогичных моделей серии NSG–800:

- NSG–900/2WL
- NSG–900/4WL
- NSG–900/16A (16- и 8-портовые модификации)

В случае необходимости любой из двух типов ПО может быть установлен пользователем самостоятельно, заменен на альтернативный или модернизирован в любое время. Поскольку набор функциональных возможностей, предлагаемых базовым программным обеспечением и NSG Linux, в настоящее время различается, пользователи могут самостоятельно выбирать из двух вариантов наиболее подходящий для своих специфических задач.

При установке программного обеспечения от NSG–800 следует использовать нижеперечисленные файлы в строгом соответствии с моделью шасси и наличием интерфейсных модулей IM–xE1–x:

| Наличие модулей IM–E1–x | Модель             |                    |                       |                        |
|-------------------------|--------------------|--------------------|-----------------------|------------------------|
|                         | NSG–900/4WL        | NSG–900/2WL        | NSG–900/16A (8-порт.) | NSG–900/16A (16-порт.) |
| нет                     | NSG800_4WL.bin     | NSG800_2WL.bin     | NSG800_8AM.bin        | NSG800_16AM.bin        |
| IM–E1–x                 | NSG800_4WL_1e.bin  | NSG800_2WL_1e.bin  | NSG800_8AM_1e.bin     | NSG800_16AM_1e.bin     |
| IM–2E1–x                | NSG800_4WL_2e.bin  | NSG800_2WL_2e.bin  | NSG800_8AM_2e.bin     | NSG800_16AM_2e.bin     |
| IM–CE1–x                | NSG800_4WL_1ce.bin | NSG800_2WL_1ce.bin | NSG800_8AM_1ce.bin    | NSG800_16AM_1ce.bin    |

Версия программного обеспечения должна быть не ниже 8.2.1.

Процедура установки программного обеспечения для серии NSG–800 описана в документе:

*Мультипротокольные маршрутизаторы и коммутаторы пакетов NPS–7e, NSG–500, NX–300, NSG–800. Базовое программное обеспечение. Руководство пользователя. (Части 1–9.)*

Замена или модернизация программного обеспечения не влечет за собой ограничения в гарантии на устройства или технической поддержке.

**ПРИМЕЧАНИЕ** При установке базового программного обеспечения устройства NSG–900 программно идентифицируются как аналогичные модели из серии NSG–800.



## §1.6. Участие пользователей в развитии программы NSG Linux

Пользователи, обладающие необходимыми знаниями и навыками в области программирования для ОС Linux, могут самостоятельно реализовать специфические программные возможности, требуемые для решения их задач, а также внести свой вклад в развитие проекта в целом. С этой точки зрения, устройства NSG под управлением NSG Linux можно рассматривать как Linux-машину общего вида, на которой, наряду с коммуникационным программным обеспечением NSG, исполняются дополнительные приложения пользователя.

Вместе с готовым программным обеспечением NSG Linux компания NSG предлагает кросс-платформенный инструментарий разработчика. С его помощью пользователи могут самостоятельно разрабатывать специфические приложения для своих нужд, а также переносить на эту платформу программные продукты, доступные в исходных кодах, при условии, что такое их использование не нарушает права интеллектуальной собственности третьих лиц. В частности, пользователи могут переносить на устройства NSG имеющиеся у них программные продукты собственной разработки, продукты, распространяемые с открытым кодом, и продукты, законно приобретенные пользователями у сторонних разработчиков с правом дальнейшего изменения.

Приложения пользователя рекомендуется устанавливать в файловую систему, расположенную в развернутом виде на устройствах расширения энергонезависимой памяти (DoC или FLEX, USB Flash или USB HDD, в зависимости от модели шасси). При этом следует иметь в виду, что память типа FLEX и USB Flash предназначена, в основном, для хранения приложений. Для часто перезаписываемых пользовательских данных, таких как статистика, журналы ввода-вывода и т.п., следует использовать модуль DoC, выдерживающий большее число циклов стирания-записи, или жесткий диск.

В устройствах, не оснащенных расширенной энергонезависимой памятью, рабочая файловая система располагается на виртуальном диске в оперативной памяти. Приложения пользователя следует устанавливать в директорию `/etc`. После установки необходимо выполнить команду (в командной оболочке ОС Linux):

```
savecfg
```

По данной команде вся файловая система упаковывается в один архив и записывается в энергонезависимую память — при условии, что ее объем достаточен для хранения дополнительных компонент.

Вопрос об участии сторонних разработчиков в задачах, связанных с развитием ядра NSG Linux и протокольных компонент системы, решается на основе индивидуальных соглашений.

Инструментарий разработчика и документация для самостоятельной разработки компонент NSG Linux доступны на сайте компании по адресам:

```
http://www.nsg.ru/nsg-linux/ELDK/  
ftp://ftp.nsg.net.ru/pub/nsg-linux/ELDK/
```

## Приложение 1–А. Поддерживаемые стандарты и спецификации

### Физические интерфейсы

|                                    |                                                                                                                                                                                                                                                                                                                                                                                                            |      |             |      |                                  |      |                                 |     |                              |       |                          |           |                                       |
|------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------|-------------|------|----------------------------------|------|---------------------------------|-----|------------------------------|-------|--------------------------|-----------|---------------------------------------|
| DTE/DCE                            | ITU–T V.24, V.35, V.36<br>TIA/EIA RS–232, RS–485, RS–449, RS–530, RS–530A<br>ITU–T X.21                                                                                                                                                                                                                                                                                                                    |      |             |      |                                  |      |                                 |     |                              |       |                          |           |                                       |
| Аналоговые модемы ТЧ               | ITU–T V.92, V.34, V.32bis, V.32, V.23, V.22bis, V.22, V.21<br>Lucent Technologies & Rockwell Semiconductor Systems K56Flex<br>Bell 212, 103<br>ITU–T V.44, V.42bis, V.42<br>Microcom MNP 5, MNP 10                                                                                                                                                                                                         |      |             |      |                                  |      |                                 |     |                              |       |                          |           |                                       |
| xDSL и модемы для физических линий | <table> <tr> <td>IDSL</td> <td>ITU–T G.961</td> </tr> <tr> <td>MDSL</td> <td>фирменная спецификация Level One</td> </tr> <tr> <td>SDSL</td> <td>фирменная спецификация Conexant</td> </tr> <tr> <td>SRM</td> <td>фирменная спецификация Mitel</td> </tr> <tr> <td>SHDSL</td> <td>ITU–T G.991 (Annex A, B)</td> </tr> <tr> <td>SHDSL-bis</td> <td>ITU–T G.991 (Annex A, B, F, G), G.994</td> </tr> </table> | IDSL | ITU–T G.961 | MDSL | фирменная спецификация Level One | SDSL | фирменная спецификация Conexant | SRM | фирменная спецификация Mitel | SHDSL | ITU–T G.991 (Annex A, B) | SHDSL-bis | ITU–T G.991 (Annex A, B, F, G), G.994 |
| IDSL                               | ITU–T G.961                                                                                                                                                                                                                                                                                                                                                                                                |      |             |      |                                  |      |                                 |     |                              |       |                          |           |                                       |
| MDSL                               | фирменная спецификация Level One                                                                                                                                                                                                                                                                                                                                                                           |      |             |      |                                  |      |                                 |     |                              |       |                          |           |                                       |
| SDSL                               | фирменная спецификация Conexant                                                                                                                                                                                                                                                                                                                                                                            |      |             |      |                                  |      |                                 |     |                              |       |                          |           |                                       |
| SRM                                | фирменная спецификация Mitel                                                                                                                                                                                                                                                                                                                                                                               |      |             |      |                                  |      |                                 |     |                              |       |                          |           |                                       |
| SHDSL                              | ITU–T G.991 (Annex A, B)                                                                                                                                                                                                                                                                                                                                                                                   |      |             |      |                                  |      |                                 |     |                              |       |                          |           |                                       |
| SHDSL-bis                          | ITU–T G.991 (Annex A, B, F, G), G.994                                                                                                                                                                                                                                                                                                                                                                      |      |             |      |                                  |      |                                 |     |                              |       |                          |           |                                       |
| Цифровая плезиохронная иерархия    | ITU–T G.703, G.704, G.706, G.732, G.823                                                                                                                                                                                                                                                                                                                                                                    |      |             |      |                                  |      |                                 |     |                              |       |                          |           |                                       |
| GSM/GPRS/EDGE                      | ETSI GSM Phase 2/2+ 900/1800 МГц                                                                                                                                                                                                                                                                                                                                                                           |      |             |      |                                  |      |                                 |     |                              |       |                          |           |                                       |
| CDMA                               | CDMA 1x MC RTT (IS–2000) 450 МГц<br>CDMA 1xEV-DO (IS–856) 450 МГц                                                                                                                                                                                                                                                                                                                                          |      |             |      |                                  |      |                                 |     |                              |       |                          |           |                                       |
| Bluetooth                          | Bluetooth v1.1 2,4 ГГц                                                                                                                                                                                                                                                                                                                                                                                     |      |             |      |                                  |      |                                 |     |                              |       |                          |           |                                       |
| Universal Serial Bus               | USB 2.0                                                                                                                                                                                                                                                                                                                                                                                                    |      |             |      |                                  |      |                                 |     |                              |       |                          |           |                                       |

### Сети Ethernet

|                                            |                                                    |
|--------------------------------------------|----------------------------------------------------|
| Ethernet 10Base–T, 10/100Base–T, 100Base–F | IEEE 802.3                                         |
| Режим моста и протокол Spanning Tree       | IEEE 802.1d                                        |
| VLAN                                       | IEEE 802.1q                                        |
| Мост Ethernet-over-IP (GRE)                | IETF RFC–1701, RFC–1702, фирм. спец. Cisco Systems |
| Мост Ethernet-over-Frame Relay             | фирменная спецификация Cisco Systems               |
| Мост/удаленный порт Ethernet-over-SDSL     | фирменная спецификация NSG                         |

### Стек Frame Relay

|                                                                              |                                            |
|------------------------------------------------------------------------------|--------------------------------------------|
| Коммутация Frame Relay (PVC)                                                 | } ITU–T X.36, X.76                         |
| Интерфейсы DTE/DCE/NNI                                                       |                                            |
| Механизмы QoS (CIR, BC, BE)                                                  |                                            |
| Управляющие протоколы                                                        | ITU–T Q.933 Annex A<br>ANSI T1.617 Annex D |
| Мультиплексирование синхронного пакетного трафика общего вида во Frame Relay | фирменная спецификация NSG                 |
| Мультиплексирование синхронного пакетного трафика общего вида в туннели GRE  | фирменная спецификация NSG                 |

### Стек X.25

|                        |               |
|------------------------|---------------|
| Коммутация X.25 (SVC)  | ITU–T X.25    |
| Адресация X.25         | ITU–T X.121   |
| X.25-over-TCP/IP (XOT) | IETF RFC–1613 |

**Стек TCP/IP**

|                                |                                                     |
|--------------------------------|-----------------------------------------------------|
| IPv4                           | IETF RFC-791                                        |
| IP over Ethernet               | IETF RFC-894, RFC-1042                              |
| IP over Cisco-HDLC             | фирменная спецификация Cisco Systems                |
| IP over PPP                    | IETF RFC-1332, RFC-1661                             |
| IP over SLIP                   | IETF RFC-1055                                       |
| IP over Frame Relay            | IETF RFC-1490, фирменная спецификация Cisco Systems |
| IP over X.25                   | IETF RFC-1356                                       |
| TCP                            | IETF RFC-793                                        |
| UDP                            | IETF RFC-768                                        |
| ICMP                           | IETF RFC-792                                        |
| GRE                            | IETF RFC-1701, RFC-1702                             |
| IP-in-IP                       | стандарт де-факто для ОС Linux                      |
| PPP over Ethernet              | IETF RFC-2516                                       |
| PPTP                           | IETF RFC-2637                                       |
| RIPv1                          | IETF RFC-1058                                       |
| RIPv2                          | IETF RFC-2453                                       |
| OSPF                           | IETF RFC-2328                                       |
| BGP                            | IETF RFC-1771, 1965, 1997, 2798, 2858, 2842         |
| NAT                            | IETF RFC-1631                                       |
| ARP                            | IETF RFC-826                                        |
| DHCP (сервер, ретранслятор)    | IETF RFC-2131, RFC-2132                             |
| Telnet (клиент и сервер)       | IETF RFC-854                                        |
| FTP (клиент)                   | IETF RFC-959                                        |
| TFTP (клиент)                  | IETF RFC-1350                                       |
| SNMPv1, MIB II                 | IETF RFC-1156, RFC-1157, RFC-1158                   |
| SNTP (клиент)                  | IETF RFC-1361                                       |
| Виртуальные частные сети (VPN) | IETF RFC-2401 ... 2409                              |
| Принт-сервер                   | Raw Socket over TCP/IP (HP JetDirect)               |

**Аутентификация, авторизация и статистика**

|                              |                                              |
|------------------------------|----------------------------------------------|
| PAP (клиент и сервер)        | IETF RFC-1661                                |
| CHAP (клиент и сервер)       | IETF RFC-1661, RFC-1994                      |
| MS-CHAP, MS-CHAP v2 (клиент) | фирменная спецификация Microsoft Corporation |
| RADIUS (клиент)              | IETF RFC-2138, RFC-2139                      |
| NetFlow                      | фирменная спецификация Cisco Systems         |

## Приложение 1–Б. Команды загрузчика U-Boot

Загрузчик U-Boot представляет собой минимизированную сборку Linux, предназначенную для установки и запуска основного программного обеспечения. С помощью U-Boot возможна также переустановка некорректно работающего ПО и очистка конфигурации устройства (в последнем случае при запуске основного ПО устанавливается заводская конфигурация устройства).

Широкие возможности, предоставляемые U-Boot, актуальны преимущественно на этапе разработки программного обеспечения. Далее они используются в заранее сконфигурированном режиме. Конечному пользователю в любом случае следует выполнять эти операции в строгом соответствии с инструкциями и руководствами NSG. По этой причине данное Приложение носит вводный характер; его основная цель — помочь любопытствующему пользователю понять суть предписываемых ему действий. Исчерпывающая информация о U-Boot доступна на сайте данного проекта <http://www.denx.de>.

Доступ к U-Boot возможен только через консольный порт устройства. Для входа в меню U-Boot необходимо перезагрузить устройство любым способом (выключением/включением питания, кнопкой Reset или программно) и дождаться вывода подсказки, например:

```
U-Boot 0.1.4 (NSG build 1) Dec 26 2006 - 13:31:58
U-Boot code: 23FC0000 -> 23FD926C BSS: -> 23FF6204
RAM Configuration:
Bank #0: 20000000 64 MB
Flash: 16 MB
In: serial
Out: serial
Err: serial
PHY 0: NSG700_ETH detected
Hit any key to stop autoboot: 5
```

Подсказка выводится на консоль в течение 5 секунд, за которые следует нажать любую клавишу. Если ни одна клавиша за это время не нажата, начинается загрузка основного программного обеспечения. Если нажата какая-либо клавиша, запускается интерпретатор командной строки U-Boot и выводится системное приглашение:

```
u-boot>
```

**ПРИМЕЧАНИЕ** В версиях U-Boot *NSG build 2* и старше, если установлена переменная окружения `silent=yes` (см. ниже), никакие сообщения U-Boot не выводятся. В этом случае для входа в меню U-Boot также необходимо нажать любую клавишу в течение 5 секунд после рестарта.

Команды U-Boot делятся на несколько функциональных групп. При вводе команд следует обратить внимание на несколько общих правил:

- Для ввода команды достаточно набрать один или несколько первых символов, определяющих ее однозначным образом — например, `help` можно сократить до `h`.
- Автодополнение по нажатию клавиши TAB работает как для команд, так и для переменных окружения, используемых как параметры команд.
- Результат действия некоторых команд может зависеть от установленных значений переменных окружения.
- Все числовые значения вводятся в шестнадцатеричном виде.
- Для редактирования командной строки следует пользоваться только клавишей Backspace.
- В одной строке могут вводиться несколько команд, разделенных точкой с запятой (;).
- Нажатие клавиши Enter сразу после подсказки приводит к повторному исполнению предыдущей команды.

### Справочные команды

`help` или ? Вывод списка команд.

`help [ команда ]`  
Вывод подробной информации об указанной команде.

`version` Вывод номера версии U-Boot и даты его генерации:  

```
u-boot> version
U-Boot 1.1.4 (Apr 20 2007 - 16:30:53) (NSG build 1)
```

`coninfo` Вывод информации о подключенных консольных устройствах, установленных для них флагах и направлении потоков ввода-вывода.

flinfo [ номер\_банка ]

Вывод информации об объеме Flash ROM и его состоянии по секторам. Для каждого сектора могут быть указаны следующие атрибуты:

RO Сектор защищен от записи (см. команду protect).  
E Сектор стерт и готов для записи.

Во всех устройствах NSG, оснащенных U-Boot, используется только один банк памяти, поэтому команды flinfo и flinfo 1 эквивалентны, а другие номера банков недопустимы.

iminfo <стартовый сектор>

Вывод информации о программном обеспечении, которое хранится, начиная с указанного адреса.

При этом проверяется контрольная сумма файла ПО и выводится его заголовок.

Номер (шестнадцатеричный) стартового сектора для хранения основного ПО для конкретных устройств NSG записан в переменных окружения, управляющих стиранием, установкой и запуском ПО — nsgupdate..., nsgerase..., bootcmd и т.п.; подробнее см. команду printenv.

imls

Просмотр начальных байт всех секторов и вывод информации обо всех экземплярах программного обеспечения, найденных на границах секторов.

### Примеры вывода информации.

u-boot> flinfo

Bank # 1: CFI conformant FLASH (16 x 16) Size: 16 MB in 128 Sectors

Erase timeout 16384 ms, write timeout 1 ms, buffer write timeout 4096 ms, buffer size 32

Sector Start Addresses:

|             |             |            |            |            |
|-------------|-------------|------------|------------|------------|
| 10000000 RO | 10020000 RO | 10040000   | 10060000 E | 10080000   |
| 100A0000    | 100C0000    | 100E0000   | 10100000   | 10120000   |
| 10140000    | 10160000    | 10180000   | 101A0000   | 101C0000   |
| 101E0000    | 10200000    | 10220000   | 10240000   | 10260000   |
| 10280000    | 102A0000    | 102C0000   | 102E0000   | 10300000   |
| 10320000    | 10340000    | 10360000   | 10380000   | 103A0000   |
| 103C0000    | 103E0000    | 10400000   | 10420000   | 10440000   |
| 10460000    | 10480000    | 104A0000   | 104C0000   | 104E0000   |
| 10500000    | 10520000    | 10540000   | 10560000   | 10580000   |
| 105A0000    | 105C0000    | 105E0000   | 10600000   | 10620000   |
| 10640000    | 10660000    | 10680000   | 106A0000   | 106C0000   |
| 106E0000    | 10700000 E  | 10720000 E | 10740000 E | 10760000 E |
| 10780000 E  | 107A0000 E  | 107C0000 E | 107E0000 E | 10800000 E |
| 10820000 E  | 10840000 E  | 10860000 E | 10880000 E | 108A0000 E |
| 108C0000 E  | 108E0000 E  | 10900000 E | 10920000 E | 10940000 E |
| 10960000 E  | 10980000 E  | 109A0000 E | 109C0000 E | 109E0000 E |
| 10A00000 E  | 10A20000 E  | 10A40000 E | 10A60000 E | 10A80000 E |
| 10AA0000 E  | 10AC0000 E  | 10AE0000 E | 10B00000 E | 10B20000 E |
| 10B40000 E  | 10B60000 E  | 10B80000 E | 10BA0000 E | 10BC0000 E |
| 10BE0000 E  | 10C00000 E  | 10C20000 E | 10C40000 E | 10C60000 E |
| 10C80000 E  | 10CA0000 E  | 10CC0000 E | 10CE0000 E | 10D00000 E |
| 10D20000 E  | 10D40000 E  | 10D60000 E | 10D80000 E | 10DA0000 E |
| 10DC0000 E  | 10DE0000 E  | 10E00000 E | 10E20000 E | 10E40000 E |
| 10E60000 E  | 10E80000 E  | 10EA0000 E | 10EC0000 E | 10EE0000 E |
| 10F00000 E  | 10F20000 E  | 10F40000 E | 10F60000 E | 10F80000 E |
| 10FA0000 E  | 10FC0000 E  | 10FE0000 E |            |            |

В данном случае (устройство NSG-700) Flash ROM имеет объем 16 МБ и содержит 128 секторов по 128 КБ каждый. В первых 2 секторах, защищенных от записи, размещается сам U-Boot и его переменные окружения. В третьем (по адресу 10040000) хранится конфигурация устройства. Четвертый также зарезервирован под хранение конфигурации, но пока он свободен. Начиная с пятого сектора (10080000), располагается основное ПО. Последние сектора (более половины) свободны.

**ПРИМЕЧАНИЕ** Размер сектора связан с объемом Flash ROM, поэтому на устройствах с различным размером Flash ROM одним и тем же адресам могут соответствовать различные номера секторов. По этой причине в командах стирания, записи и т.п. рекомендуется использовать именно адреса, поскольку они не зависят от архитектуры Flash ROM.

Информация о программном обеспечении, размещенном по адресу 10080000:

```

u-boot> imi 10080000
## Checking Image at 10080000 ...
Image Name: Linux Multiboot-Image
Image Type: ARM Linux Multi-File Image (gzip compressed)
Data Size: 6689041 Bytes = 6.4 MB
Load Address: 20008000
Entry Point: 20008000
Contents:
Image 0: 1145630 Bytes = 1.1 MB
Image 1: 5543397 Bytes = 5.3 MB
Verifying Checksum ... OK

```

Не разбираясь предварительно в размещении различных компонент, эту же информацию можно было бы получить командой `imls`.

Этот же адрес указывается в качестве стартового в переменных окружения:

```

u-boot> pri
bootcmd=bootm 0x10080000
.....
nsgerase_core=erase 0x10080000 0x102FFFFFF
nsgupdate_core=0x10080000 nsglinux-arm-core.bin
nsgerase_sumo=erase 0x10080000 0x107FFFFFF
nsgupdate_sumo=0x10080000 nsglinux-arm-sumo.bin
.....

```

### Команды для работы с переменными окружения

`printenv` [ переменная [ переменная... ] ]

Вывести значения всех установленных переменных окружения, или нескольких перечисленных.

`setenv` <переменная> <значение>

Установить значение для указанной переменной.

В частности, значением переменной может быть произвольная команда или последовательность команд; в последнем случае разделителем является последовательность `\;`. Такая переменная является аналогом скрипта, или batch-файла.

`saveenv`

Сохранить установленные значения переменных окружения. До этого момента все изменения, производимые командой `setenv`, происходят только в оперативной памяти. Для сохранения сделанных изменений необходимо снять блокировку с соответствующего сектора Flash ROM, стереть этот сектор, записать его заново и установить защиту от записи.

`run` [ переменная [ переменная... ] ]

Выполнить команды, содержащиеся в указанных переменных окружения.

С помощью команды `run` выполняется, в частности, очистка Flash ROM перед установкой новой версии основного программного обеспечения:

```

u-boot> pri
nsgerase_core=erase 0x10080000 0x102FFFFFF
nsgerase_sumo=erase 0x10080000 0x107FFFFFF
u-boot> run nsgerase_core
или
u-boot> run nsgerase_sumo

```

Все аппаратно-зависимые константы (стартовый адрес и т.п.) в данном случае содержатся в предустановленных переменных окружения, поэтому для конечного пользователя процедура установки ПО выполняется одними и теми же командами на любой платформе.

### ВНИМАНИЕ

При изменении переменных окружения следует обратить особое внимание на то, чтобы их имена были набраны без ошибки. В противном случае, поскольку имена могут быть произвольными, вместо изменения требуемой переменной в системе создается другая, бесполезная, переменная с похожим именем, например, `ethadr` вместо `ethaddr`. Однако, с точки зрения командного языка, данное действие не является ошибкой, поэтому никакое аварийное сообщение не может быть выведено.

**Команды для работы с оперативной памятью**

mtest [ начальный\_адрес [ конечный адрес [ код ]]

Тестирование заданной области RAM (запись и чтение указанного кода длиной 4 байта). По умолчанию, тестируется вся память несколькими последовательными кодами. Для завершения теста следует нажать клавиши CTRL-C.

Рекомендуется использовать данную команду без параметров, при этом тестируется вся память, кроме области, занятой самим U-Boot. При ручном указании диапазона адресов может быть затронута область, в которой размещается U-Boot, что приведет к краху системы.

К данной группе относятся также команды base, crc32, cmp, cp, md, mm, mw, nm, loop. Эти команды предназначены для разработчиков. Конечному пользователю применять их не следует, кроме исключительных ситуаций по явному указанию службы поддержки.

**Команды для работы с Flash ROM**

flinfo ...

Вывод информации о Flash ROM — см. выше.

erase <начальный адрес> <конечный адрес>

erase <банк>:<начальный сектор >[–<конечный сектор>]

erase bank <номер банка>

erase all

Стирание указанной области Flash ROM. Сектора, в случае ввода команды во второй нотации, нумеруются, начиная с нуля.

Пример: для распределения энергонезависимой памяти, приведенного выше, конфигурация основного ПО устройства может быть принудительно очищена любой из двух команд:

```
erase 10040000 1007FFFF
```

```
erase 1:2–3
```

protect { on | off } <начальный адрес> <конечный адрес>

protect { on | off } <банк>:<начальный сектор >[–<конечный сектор>]

protect { on | off } bank <номер банка>

protect { on | off } all

Установка и снятие защиты от стирания для указанной области Flash ROM. Сектора, в случае ввода команды во второй нотации, нумеруются, начиная с нуля.

Пример: если для распределения энергонезависимой памяти, приведенного выше, потребуется модернизация самого U-Boot, то перед записью новой версии необходимо удалить старую, а для этого — разрешить ее удаление:

```
protect off 1:0–1
```

```
erase 1:0–1
```

Если в диапазоне памяти, указанном для стирания, находятся защищенные от записи сектора, то они останутся в неизменном виде, о чем выводится соответствующее предупреждение.

**Команды для загрузки и запуска программного обеспечения**

bootm [ адрес [ аргументы ]]

Запустить программное обеспечение, находящееся по указанному адресу, с указанными аргументами, и передать ему управление.

boot или bootd

Выполнить команду загрузки и запуска программного обеспечения, установленную по умолчанию переменной окружения bootcmd.

Пример: для распределения энергонезависимой памяти, приведенного выше, основное ПО запускается командой:

```
bootm 10080000
```

Эта же строка содержится в переменной bootcmd:

```
u-boot> pri
```

```
bootcmd=bootm 0x10080000
```

bootp <адрес> <имя файла>

Загрузить файл с программным обеспечением, разместить его в памяти устройства (оперативной или энергонезависимой) по указанному адресу, и передать ему управление. Сначала при помощи протокола DHCP запрашивается адрес устройства и адрес сервера, после чего указанный файл загружается с сервера при помощи протокола TFTP.

`tftpboot <адрес> <имя файла>`

Загрузить файл с программным обеспечением по протоколу TFTP, разместить его в памяти устройства (оперативной или энергонезависимой) по указанному адресу, и передать ему управление. В отличие от предыдущей команды, IP-адреса устройства NSG и сервера TFTP в данном случае устанавливаются статически переменными окружения `ipaddr` и `serverip`, соответственно.

`rarpboot <адрес> <имя файла>`

Загрузить файл с программным обеспечением, разместить его в памяти устройства (оперативной или энергонезависимой) по указанному адресу, и передать ему управление. Адрес устройства и адрес сервера устанавливаются при помощи протокола RARP, после чего указанный файл загружается с сервера при помощи протокола TFTP.

**ВНИМАНИЕ** Перед исполнением вышеуказанных команд область Flash ROM, предназначенная для записи принимаемого файла, должна быть очищена командой `erase`.

Аргументы вышеуказанных команд могут быть заданы в одной переменной окружения, которая вызывается в формате `#{имя переменной}`, например:

```
bootp ${nsgupdate_sumo}
tftpboot ${nsgupdate_core}
```

Помимо IP-адресов, действие этих команд зависит от переменных окружения `autoload` (загружать файл по TFTP) и `autostart` (запустить полученный файл на исполнение). По умолчанию обе переменные имеют значение `yes`. Значения по предписывают U-Boot, соответственно, только запросить сетевую конфигурацию, либо только принять файл и записать его в память.

### Другие команды

`echo <строка>`

Вывести указанную строку на консоль.

`ping <ip-адрес>`

Послать пакет ICMP ECHO REQUEST на указанный IP-адрес. Это простейшая реализация *ping*, предназначенная только для того, чтобы проверить доступность удаленного сервера. Посылается не более 2 пакетов, время ожидания ответа — 5 секунд. Рекомендуется выполнить данную процедуру перед загрузкой программного обеспечения по TFTP или NFS.

`reset`

Перезагрузить устройство.

Следующие команды предназначены для использования в скриптах, а также для исполнения отдельных приложений в среде U-Boot: `autoscr`, `exit`, `go`, `itest`, `sleep`, `test`. Эти команды адресованы разработчикам. Конечному пользователю применять их не следует, кроме исключительных ситуаций по явному указанию службы техподдержки.

### Переменные окружения

Для настройки U-Boot и основного программного обеспечения NSG Linux существенное значение имеют следующие переменные окружения, хранящиеся в конфигурации U-Boot.

`bootargs=root=/dev/rd/0 init=/linuxrc rw nsg=verbose`

Строка параметров, передаваемая ядру основного ПО NSG Linux при его загрузке.

`bootcmd=bootm 0xНННННННН`

Способ и адрес загрузки основного ПО, используемые по умолчанию. Адрес может варьироваться в зависимости от конкретной модели шасси.

`bootdelay=5`

Задержка перед запуском основного ПО, в секундах. В течение этого времени пользователь может нажать любую клавишу для входа в меню U-Boot. При необходимости можно увеличить этот интервал, или сократить его до 1–2 сек. Не следует устанавливать его равным нулю, поскольку в этом случае восстановить доступ к U-Boot штатными средствами самого U-Boot будет невозможно.

`baudrate=9600`

Скорость консольного порта. Это значение передается в основное ПО и используется для работы выделенного консольного порта и разделяемого порта с инкапсуляцией `console`.

`silent=yes`

Режим загрузки без вывода сообщений U-Boot и NSG Linux в консольный порт. (Начиная с версии U-Boot *NSG build 2*.) При любом непустом значении данной переменной сообщения не выводятся; этот режим предназначен для случая, когда консольный порт используется для передачи данных и подключённое оборудование может пытаться по-своему интерпретировать сообщения загрузчика и отвечать на них. Для восстановления вывода сообщений следует установить данной переменной пустое значение (`setenv silent <Enter>`).



ethaddr=00:09:56:xx:xx:xx

Заводской MAC-адрес встроенного порта Ethernet. Это значение передается в основное ПО. Адрес устанавливается один раз после установки или переустановки U-Boot, впоследствии изменять его запрещается. Изменение MAC-адреса возможно в ходе работы основного ПО, заводское значение при этом сохраняется.

**ВНИМАНИЕ** При переустановке U-Boot необходимо предварительно выписать заводское значение MAC-адреса, установить новую версию, а затем ввести это значение заново. В противном случае возможен конфликт MAC-адресов между несколькими устройствами NSG в одной локальной сети.

Если в устройстве имеется несколько встроенных портов Ethernet, то их MAC-адреса хранятся в переменных ethaddr1, ethaddr2 и т.д. В NSG Linux эти порты получают имена eth1, eth2, ... соответственно.

ipaddr=192.168.0.1

serverip=192.168.0.250

IP-адреса встроенного порта Ethernet и удаленного сервера TFTP для загрузки программного обеспечения без использования механизмов DHCP или RARP. При получении IP-адресов посредством DHCP или RARP текущие значения данных переменных изменяются, а также может быть определен ряд других параметров IP и TFTP (gatewayip, bootfile, filesize и т.п.).

nsgerase\_core=erase 0xHHHHHHHH 0xGGGGGGGG

nsgerase\_sumo=erase 0xHHHHHHHH 0xGGGGGGGG

Макрокоманды для стирания Flash ROM (для систем с расширенной энергонезависимой памятью и без нее, соответственно).

nsgupdate\_core=0xHHHHHHHH nsgXXX-linux-core.bin

nsgupdate\_sumo=0xHHHHHHHH nsgXXX-linux-sumo.bin

Параметры команд \*boot\* для загрузки и запуска основного ПО.

В четырех вышеприведенных командах платформенно-зависимые параметры (адреса, имена файлов) предустановлены в зависимости от конкретной модели шасси. Изменять эти значения пользователю не следует. Благодаря этому команды для стирания ПО и установки новой версии выглядят одинаково на всех типах шасси.

stdin=serial

stdout=serial

stderr=serial

Описание стандартных потоков ввода-вывода. В данном случае все они связаны с консолью.

loads\_echo=1

Включение/выключение режима эхо при приеме файла командой loads через последовательный порт.

Помимо вышеперечисленных, в системе могут быть определены также другие переменные окружения. Если они определены статически при разработке, то изменять их пользователю не следует, кроме как по явному указанию службы техподдержки.

## Приложение 1–В. Аварийная переустановка U-Boot на NSG–700

Устройства NSG–700 поставляются со штатным системным загрузчиком U-Boot, который допускает, в частности, обновление себя самого своими собственными средствами. Подробно о данной операции см. п.1.5.10.

В случае аварийного завершения операции устройство становится недоступным для пользователя и для восстановления его работоспособности необходима специальная процедура первичной установки U-Boot средствами низкоуровневого аппаратного загрузчика, описываемая ниже. Она производится, как правило, в заводских условиях при изготовлении устройства. Для конечного пользователя эта процедура является нештатной и прибегать к ней следует исключительно в чрезвычайных случаях, приведших к утрате U-Boot.

Процедура выполняется в 3 этапа:

- Загрузка утилиты loader.bin средствами низкоуровневого аппаратного загрузчика, встроенного в микросхема процессора.
- Загрузка U-Boot средствами loader.bin и его запись в энергонезависимую память.
- Переустановка конфигурации U-Boot.

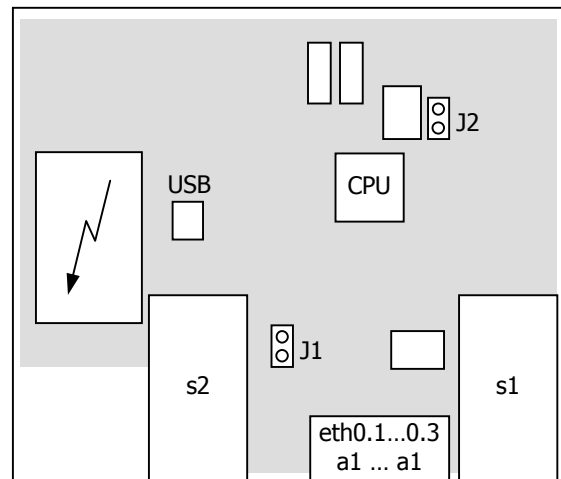
Подробное описание процедуры приведено ниже.

1. Загрузить файл loader.bin с Web-сайта NSG или с CD-ROM, прилагаемого к устройству, и сохранить его на локальном ПК.
2. Выключить питание устройства и отключить его от сети.
3. Открыть верхнюю крышку устройства, следуя указаниям Руководства пользователя по данному устройству.
4. Снять технологическую перемычку J1 (см. рисунок).

### Назначение перемычек на шасси NSG–700

|    |            |                               |
|----|------------|-------------------------------|
| J1 | замкнута   | Старт загрузчика из Flash ROM |
|    | разомкнута | Старт аппаратного загрузчика  |
| J2 | замкнута   | Перезапись U-Boot разрешена   |
|    | разомкнута | Перезапись U-Boot запрещена   |

По умолчанию, устройство поставляется с обеими установленными перемычками.



5. Присоединить COM-порт компьютера к консольному порту устройства кабелем, входящим в комплект устройства.
6. Запустить на ПК терминальную программу и установить в ней следующие параметры для данного COM-порта:

```

Baud Rate      115200
Parity         None (отсутствует)
Data Bits      8
Stop Bits      1
Flow Control    None
  
```

7. Включить питание устройства и дождаться появления запроса на передачу файла по Xmodem:
 

```

CCC...
      
```
8. Инициировать в терминальной программе передачу (*upload*) файла loader.bin по протоколу Xmodem. Получаемый файл сохраняется в оперативной памяти устройства и запускается на исполнение. При этом снова выводится запрос на передачу файла по Xmodem:
 

```

Send file by Xmodem:
CCC...
      
```
9. Инициировать в терминальной программе передачу (*upload*) файла u-boot-nsg700.bin по протоколу Xmodem. Получаемый файл сохраняется в оперативной памяти устройства. По завершении приема выводится служебное меню:

```
[E]rase flash
[0]program u-boot (offset 0x00000)
.....
[C]ontinue
```

10. Нажать клавишу 0 (ноль). Полученный образ U-Boot будет записан в энергонезависимую память. По завершении записи снова выводится запрос на передачу файла по Xmodem:

```
Erase sector 0 Ok
Erase sector 1 Ok
Write flash
Offset 0x0000
Ok

Send file by Xmodem:
CCC...
```

11. Выключить питание устройства.  
 12. Установить перемычку J1.  
 13. Включить питание устройства, войти в U-Boot и убедиться, что он функционирует.  
 14. Проверить MAC-адрес устройства, записанный в переменной окружения ethaddr:

```
printenv
```

Если переменные окружения также уничтожены и исходный MAC-адрес неизвестен и не был нигде сохранен, то впредь в последние 2 байта следует записывать последние 4 цифры заводского номера устройства, преобразованные из десятичной в шестнадцатеричную форму. Пример: для устройства с заводским номером 06120089 следует установить MAC-адрес 00:09:56:0E:00:59 (0089dec = 0059hex).

15. Поскольку конфигурация U-Boot, с большой долей вероятности, может быть повреждена ранее вместе с самим U-Boot, или несовместима с новой версией программного обеспечения, следует стереть ее и восстановить конфигурацию по умолчанию для новой версии U-Boot. Для этого необходимо снять программную защиту от записи с области, в которой расположена конфигурация U-Boot:

```
u-boot> protect off 10020000 1003ffff
Un-Protected 1 sector
```

16. Стереть конфигурацию:

```
u-boot> erase 10020000 1003ffff
.... done
Erased 1 sectors
```

17. Рестартовать устройство:

```
u-boot> reset
```

и снова зайти в меню загрузчика. После рестарта восстанавливается заводская конфигурация новой версии загрузчика.

18. Восстановить заводской MAC-адрес, сохраненный ранее:

```
u-boot> setenv ethaddr 00:09:56:0E:XX:XX
```

19. Проверить правильность установки MAC-адреса:

```
u-boot> printenv
```

В случае ошибки повторить шаг 18.

20. Сохранить конфигурацию загрузчика

```
u-boot> saveenv
Saving Environment to Flash...
Un-Protected 1 sectors
Erasing Flash...
.. done
Erased 1 sectors
Writing to Flash... done
Protected 1 sectors
```

21. Выключить питание устройства и отключить от него все кабели.

22. Установить верхнюю крышку устройства и зафиксировать ее винтами.

## Приложение 1–Г. Корректировка команд при переходе на новую версию программного обеспечения

Разработчики программного обеспечения уделяют особое внимание обратной совместимости новых версий со старыми. Однако в отдельных случаях такую совместимость не удастся сохранить, поскольку дальнейшее развитие ПО требует изменения синтаксиса некоторых команд или их переноса в другое меню. В этом случае конфигурация существующего устройства может оказаться неработоспособной в новой версии ПО, и ее необходимо откорректировать вручную. Несовместимые изменения команд приведены в таблице.

| Объекты конфигурации                                                                                         | Было в предыдущей версии                                                                                                                                                                                                                                                                                                                                         | Стало в текущей версии                                                                                                                                                                                                                                                                        |
|--------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>NSG Linux 0.x.x, 1.0 build 0beta</b>                                                                      |                                                                                                                                                                                                                                                                                                                                                                  |                                                                                                                                                                                                                                                                                               |
| <i>Все объекты</i> / <i>Исходный синтаксис</i>                                                               |                                                                                                                                                                                                                                                                                                                                                                  |                                                                                                                                                                                                                                                                                               |
| <b>NSG Linux 1.0 build 0</b>                                                                                 |                                                                                                                                                                                                                                                                                                                                                                  |                                                                                                                                                                                                                                                                                               |
| Общая структура ПО                                                                                           | Изменен системный загрузчик, схема размещения ПО в энергонезависимой памяти, процедуры установки и обновления, имена файлов для установки. Для устройств NSG–900 с Flash ROM 8 МБ рекомендуется установка расширения энергонезависимой памяти (DoC или FLEX), для NSG–800 с Flash ROM 4 МБ — обязательна. Устройства NSG–800 с Flash ROM 2 МБ не поддерживаются. |                                                                                                                                                                                                                                                                                               |
| VPN (IPsec)                                                                                                  | !<br>access-list N permit ip ...<br>crypto transform-set M<br>ah ...<br>esp ...<br>crypto map YY I { ipsec-manual   ipsec-isakmp }<br>crypto isakmp key ...<br>interface ZZZ<br>crypto map MMM<br>clear crypto sa ...<br>Поддерживаются защита данных 3DES/DES/none и аутентификация MD5/SHA/none в любом сочетании                                              | !<br>nsg<br>access-list ext-ip N<br>add K permit ip ...<br>crypto transform-set XXX esp ...<br>crypto map YY I<br>method { ipsec-manual   ipsec-isakmp }<br>crypto isakmp key ...<br>port PPP<br>crypto map MMM<br>crypto clear sa ...<br>Поддерживаются только алгоритмы 3DES+MD5 и 3DES+SHA |
| Шаблон PPP-интерфейса, имя и пароль пользователя                                                             | !<br>username XXX password YYY<br>nsg<br>interface virtual-template N<br>ppp pap sent-username XXX<br>ppp XXX refuse                                                                                                                                                                                                                                             | !<br>nsg<br>username XXX password YYY<br>virtual-template N<br>ppp sent-username XXX<br>ppp refuse-auth XXX YYY ...                                                                                                                                                                           |
| Клиент RADIUS                                                                                                | !<br>radius-server host X.X.X.X acct-port Y auth-port Z<br>radius-server key AAA<br>radius-server timeout SEC<br>radius-server retransmit N                                                                                                                                                                                                                      | !<br>nsg<br>radius host X.X.X.X<br>radius acct-port Y<br>radius auth-port Z<br>radius key AAA<br>radius timeout SEC<br>radius retry N                                                                                                                                                         |
| Сервер PPPoE                                                                                                 | !<br>vpdn-group N<br>pppoe limit per-mac N<br>.....<br>interface XXX pppoe { enable   disable }                                                                                                                                                                                                                                                                  | !<br>nsg<br>vpdn-group N<br>pppoe limit N<br>.....<br>port XXX vpdn-group { N   no }                                                                                                                                                                                                          |
| <b>NSG Linux 1.0 build 0</b> для NSG–1000, NSG–900/maxU<br><b>NSG Linux 1.0 build 1</b> для других устройств |                                                                                                                                                                                                                                                                                                                                                                  |                                                                                                                                                                                                                                                                                               |
| Порт и физический интерфейс                                                                                  | Все параметры 1–3 уровней собраны в одном меню port:<br>!<br>nsg<br>port XXX<br>baudrate NNNN<br>encapsulation YYY<br>ip address A.B.C.D/M<br>.....                                                                                                                                                                                                              | Параметры физического уровня вынесены в отдельное меню phy:<br>!<br>nsg<br>phy XXX<br>baudrate NNNN<br>.....<br>port XXX<br>encapsulation YYY<br>ip address A.B.C.D/M<br>.....                                                                                                                |