Version 7.5.0
Rev 20.07.01

Часть III

СОДЕРЖАНИЕ

3.	ИСПОЛЬЗОВАНИЕ ПРОТОКОЛОВ И СЛУЖБ	3.4
3.1	Подключение терминального оборудования по протоколу PAD	3.4
3.1.1	Формат команд протокола X.28	3.5
3.1.1.1	Команда установки соединения	3.5
3.1.1.2	Команда разрыва соединения	3.6
3.1.1.3	Команда определения состояния линии	3.6
3.1.1.4	Команда прерывания	3.6
3.1.1.5	Команда сброса	3.7
3.1.1.6	Команда установки параметров порта	3.7
3.1.1.7	Команда установки и просмотра параметров порта	3.7
3.1.1.8	Команда просмотра параметров порта	3.7
3.1.1.9	Команда получения идентификатора порта	3.8
3.1.1.10	Команда установки и просмотра параметров удаленного порта	3.8
3.1.1.11	Команда просмотра параметров удаленного порта	3.8
3.1.1.12	Команда установки профиля порта	3.9
3.1.1.13	Команда вызова строки автоподстановки	3.9
3.1.1.14	Команда установки соединения с модулем Manager	3.9
3.1.2	Параметры профиля X.3	3.10
3.1.3	Стандартные профили	3.13
3.2	Подключение терминального оборудования по протоколу PPP	3.14
3.2.1	Использование таблицы сценариев	3.16
3.2.2	Принципы аутентификации для PPP интерфейса	3.17
3.3	Подключение терминального оборудования по протоколу ASYNC	3.19
3.3.1	Работа асинхронного порта в прозрачном режиме	3.19
3.3.2	Работа асинхронного порта в протокольном режиме	3.19
3.3.2.1	Локальная аутентификация для асинхронного порта	3.22
3.3.2.2	Удаленная аутентификация с использованием RADIUS и TACACS+	3.22
3.3.3	Пример использования способов аутентификации	3.24
3.4	Эмуляция абонентского оборудования (AntiPAD)	3.24
3.4.1	Назначение AntiPAD	3.24
3.4.2	Настройка AntiPAD	3.25
3.4.3	Пример настройки AntiPAD	3.28
3.5	Многоканальный асинхронный порт (МАП)	3.29
3.5.1	Организация обмена информации и структура пакета	3.29
3.5.2	Настройка МАП	3.29

3.5.3	Управление потоком данных	3.29
3.5.4	Пример программирования	3.30
3.6	Трансляция IP-адресов (NAT)	3.33
3.6.1	Назначение и принцип трансляции IP-адресов	3.33
3.6.2	Механизм трансляции адресов	3.34
3.6.3	Таблицы трансляции адресов	3.35
3.7	Сбор учетной информации	3.37
3.7.1	Биллинг X.25	3.37
3.7.2	Формат записи учетной информации	3.38
3.8	Служба XOX (X.25 Over X.25)	3.39
3.8.1	Организация виртуальных линий в публичной сети X.25	3.39
3.8.1.1	Назначение XOX	3.39
3.8.1.2	Организация XOX в NSG устройствах	3.40
3.8.1.3	Параметры HX сервера	3.41
3.8.1.4	Параметры HX станции	3.41
3.8.1.5	Пример настройки XOX службы	3.44
3.9	Логические порты	3.47
3.9.1	Трафик генератор	3.47
3.9.2	Эхо-порт	3.47
3.10	Служба SNMP	3.48
3.10.1	Рассылка TRAP-сообщений	3.48
3.11	WEB-управление. Служба HTTP	3.50
3.11.1	Назначение	3.50
3.11.1.1	Ограничения на использование WEB-браузеров	3.50
3.11.1.2	Ресурсы, потребляемые HTTP-сервером	3.50
3.11.2	Запуск и останов HTTP-сервера	3.51
3.11.3	Работа с устройством используя WEB-управление	3.52
3.11.3.1	Подключение к устройству	3.52
3.11.3.2	Процедура аутентификации	3.52
3.11.3.3	Структура главной страницы	3.53
3.11.3.4	Структура управляемых объектов	3.54
3.11.4	Работа с управляемым объектом	3.55
3.11.5	Влияние на процесс Manager	3.58

3. ИСПОЛЬЗОВАНИЕ ПРОТОКОЛОВ И СЛУЖБ

3.1 Подключение терминального оборудования по протоколу PAD

Рекомендация X.28 ITU-T определяет интерфейс между оборудованием, работающим в асинхронном (символьном) режиме и устройством PAD (Packet Assembler/Disassembler).

Работая с PAD, пользователь находится в одном из двух возможных режимов:

— **КОМАНДНЫЙ РЕЖИМ;**
— **РЕЖИМ ПЕРЕДАЧИ ДАННЫХ.**

В КОМАНДНОМ РЕЖИМЕ пользователь может установить и изменить параметры ПРОФИЛЯ порта, установить или разорвать логическое соединение, а также выполнять другие команды, описанные ниже.

В РЕЖИМЕ ПЕРЕДАЧИ ДАННЫХ (пользователь находится после успешной установки логического соединения) происходит обмен информацией. Информация, приходящая от пользователя посимвольно, группируется в пакеты данных и направляется через сеть пакетной коммутации удаленному абоненту. Информация, приходящая от удаленного абонента в виде пакета данных, разбирается PAD и посимвольно передается в терминальное устройство пользователя.

Находясь в РЕЖИМЕ ПЕРЕДАЧИ ДАННЫХ, пользователь может временно перейти в КОМАНДНЫЙ РЕЖИМ, для чего следует ввести СИГНАЛ ВНИМАНИЕ. Перейдя таким образом в КОМАНДНЫЙ РЕЖИМ, пользователь может выполнить любую команду, после чего автоматически вернется в РЕЖИМ ПЕРЕДАЧИ ДАННЫХ.

Для ввода команд пользователь использует буквы латинского алфавита (верхний или нижний регистр), а также цифры.

С помощью команд пользователь может выполнить следующие действия:

— установить логическое соединение с удаленным абонентом (<сетевой адрес>);
— разорвать логическое соединение с удаленным абонентом (CLR);
— определить состояние линии (STAT);
— послать пакет ПРЕРЫВАНИЕ (INT);
— послать пакет СБРОС (RESET);
— установить параметры ПРОФИЛЯ локального порта (SET, SET?);
— просмотреть параметры ПРОФИЛЯ локального порта (PAR?);
— получить идентификатор порта (ID);
— установить параметры ПРОФИЛЯ порта удаленного абонента (RSET?);
— просмотреть параметры ПРОФИЛЯ порта удаленного абонента (RPAR?).

Команды **CLR, RESET, INT, RSET?, RPAR?** можно использовать только после установки логического соединения с удаленным абонентом. При неправильном вводе команды или ее параметров выводится сервисный сигнал **ERR.**

Ниже приведены команды пользователя и реакции на них со стороны PAD.

3.1.1 Формат команд протокола X.28

При работе с портом типа PAD, пользователь (или программа) может использовать команды протокола X.28. Часть команд является стандартными (например CLR, INT и др..) остальные реализованы для удобства работы (ID, AD, PROF, MN).

Набор и выполнение команд возможно только в командном режиме работы порта. При этом могут использоваться как прописные так и строчные буквы.

Если в качестве команды ввести знак '**?**' или "**help**" то будет выведена подсказка:

```
To access to Manager type 'MN'
```

Команда MN (обращение к модулю Manager) приведена п.3.1.1.14.

3.1.1.1 Команда установки соединения

Для установки логического соединения следует ввести команду:

[<услуги> -] <сетевой адрес> [<данные>]

<услуги> — поле услуг, передаваемых в пакете вызов (Call). Это поле не является обязательным. В этом поле могут быть определены следующие услуги: идентификатор пользователя сети, реверсивная оплата, быстрый выбор с ограничением на ответ, закрытая группа пользователей, исходящая закрытая группа пользователей. Услуги отделяются от поля <сетевой адрес> знаком "-". Если определено несколько услуг, то они отделяются друг от друга запятой.

Форматы услуг:

идентификатор пользователя сети: **N <строка>**
где <строка> - один или более символов кроме <SP>, <DEL>, "-", "+".
Реверсивная оплата: **R**
Быстрый выбор: **F**
Быстрый выбор с ограничением на ответ: **Q**
Закрытая группа пользователей: **G <индекс>**
где <индекс> - одна или две десятичные цифры.
Исходящая закрытая группа пользователей: **O <индекс>**
где <индекс> - одна или две десятичные цифры.

<сетевой адрес> — до 15 десятичных цифр (без пробелов), определяет адрес назначения. Параметр обязательный.

<данные> — поле данных пользователя, передаваемых в пакете вызов (Call). Это поле не является обязательным. На первой позиции

поля ставится символ D или P. За этим символом следуют данные пользователя длиной до 12 символов; при использовании услуги "быстрый выбор". Длина строки данных до 124 символов.

Пример: NABC,R-32205765DThis_is_data

В сеть коммутации пакетов будет послан пакет **вызов** с вызываемым (Called) адресом 32205765 и полем данных "This_is_data".

При успешно установленном соединении пользователю будет выдан сервисный сигнал `COM`.

Если соединение по каким-либо причинам не установлено, то будет выдан сервисный сигнал `CLR`.

Примечание: В качестве вызывающего адреса (Calling Address) в пакете вызов будет использован параметр AD данного порта.

3.1.1.2 Команда разрыва соединения

Формат команды РАЗРЫВ СОЕДИНЕНИЯ:

CLR

При вводе команды РАЗРЫВ СОЕДИНЕНИЯ произойдет разрыв соединения (если оно было ранее установлено). В качестве подтверждения будет выдан сервисный сигнал `CLR CONF`

3.1.1.3 Команда определения состояния линии

Формат команды ОПРЕДЕЛИТЬ СОСТОЯНИЕ ЛИНИИ:

STAT

При установленном логическом соединении будет выведен сервисный сигнал

`ENGAGED`

Если соединение не установлено, выводится

`FREE`

3.1.1.4 Команда прерывания

Формат команды **ПРЕРЫВАНИЕ** (Interrupt):

INT

При вводе команды **ПРЕРЫВАНИЕ** по установленному логическому каналу посылается пакет прерывание (Interrupt Packet). При получении пакета прерывание от удаленного абонента PAD выведет сервисный сигнал **INT** на терминальное устройство пользователя.

3.1.1.5 Команда сброса

Формат команды СБРОС (Reset): **RESET**

При вводе команды СБРОС по установленному логическому каналу посылается пакет **сброс** (Reset Packet). При получении пакета **сброс** от удаленного абонента PAD выведет сервисный сигнал **RESET** на терминальное устройство пользователя.

3.1.1.6 Команда установки параметров порта

Формат команды установки параметров порта:

SET xxx:yyy

xxx — номер параметра; yyy — значение параметра.

Параметры порта и допустимые значения приведены в п. 3.2. При вводе одной командой значений для нескольких параметров пары xxx:yyy отделяются друг от друга запятой или пробелом.

Например:

SET 2:0,3:2,9:4

Устанавливает следующие значения параметров для порта:

Параметр	Значение	Примечание
2	0	эхо запрещено
3	2	отправка пакетов по CR
9	4	вставка 4-х NUL после CR

3.1.1.7 Команда установки и просмотра параметров порта

Формат команды установки и просмотра параметров порта:

SET? xxx:yyy

xxx — номер параметра;
yyy — значение параметра.

Параметры порта и допустимые значения приведены в п. 3.2. При вводе одной командой значений для нескольких параметров пары xxx:yyy отделяются друг от друга запятой или пробелом. Например:

SET? 2:1 4:0 10:0 15:1

В ответ на введенную команду PAD выведет:

```
PAR 2:1 4:0 10:0 15:1
```

3.1.1.8 Команда просмотра параметров порта

Формат команды просмотра параметров порта:

PAR? xxx

xxx — номер параметра.
Параметры порта и допустимые значения приведены в п. 3.2. При вводе одной командой нескольких номеров параметров номера отделяются друг от друга запятой или пробелом. Например:

PAR? 2,6,7

В ответ на введенную команду PAD выведет:

```
PAR 2:xxx 6:yyy 7:zzz
```

где **xxx, yyy** и **zzz** - значения параметров порта с номерами 2, 6 и 7 соответственно. При вводе команды PAR без номеров параметров пользователь получит значения всех 19 параметров профиля порта.

3.1.1.9 Команда получения идентификатора порта

Формат команды получения идентификатора порта:

ID

В ответ на введенную команду будет выведен сервисный сигнал:

```
NSG PAD VERSION *.*(date) Port #xxx
```

где **x** — номер порта.

3.1.1.10 Команда установки и просмотра параметров удаленного порта

Формат команды установки и просмотра параметра удаленного порта:

RSET? xxx:yyy

xxx — номер параметра;
yyy — значение параметра.

Параметры порта и допустимые значения приведены в п. 3.2. При вводе одной командой значений для нескольких параметров пары xxx:yyy отделяются друг от друга запятой или пробелом. Например:

RSET? 2:1 4:0 10:0 15:1

В ответ на введенную команду PAD выведет:

```
RPAR 2:1,4:0,10:0,15:1
```

3.1.1.11 Команда просмотра параметров удаленного порта

Формат команды просмотра параметров удаленного порта:

RPAR? xxx

xxx — номер параметра.

Параметры порта и допустимые значения приведены в п. 3.2. При вводе

одной командой нескольких номеров параметров номера отделяются друг от друга запятой или пробелом. Например:

RPAR? 2,6,7

В ответ на введенную команду PAD выведет:

```
RPAR 2:xxx,6:yyy,7:zzz
```

где xxx, yyy и zzz - значения параметров порта с номерами 2, 6 и 7 соответственно. При вводе команды **RPAR** без номеров параметров пользователь получит значения всех 19 параметров профиля порта.

3.1.1.12 Команда установки профиля порта

Формат команды установки профиля порта:

PROF <номер профиля>

<номер профиля> — число от 0 до 7, определяющее профиль.
После ввода команды **PROF** изменяется значение сразу 19 параметров порта в соответствии с выбранным профилем. Номера профилей и значения параметров, им соответствующих, приведены в п. 3.3.

3.1.1.13 Команда вызова строки автоподстановки

Перед использованием этой команды, указанная строка должна быть определена в устройстве командой **S A AD** (п.2.2.6.1). Строка автоподстановки может содержать любые символы, которые будут интерпретироваться как обычная команда PAD, введенная с терминала пользователем.

Вызов строки автоподстановки осуществляется командой:

ADn

в которой "n" указывает номер строки автоподстановки (от 0 до 7).

В системе допустимо рекурсивное использование строк автоподстановки, когда содержимое строки автоподстановки есть обращение к другой команде вызова строки автоподстановки.

3.1.1.14 Команда установки соединения с модулем Manager

Формат команды:

MN

При выполнении данной команды будет произведена попытка установки соединения с модулем Manager. Если в момент обращения модуль Manager не занят другим пользователем то произойдет подключение и будет выведено сообщение:

```
 COM
Password:
```

Далее пользователь должен ввести пароль (при вводе не отображается).

Если в момент обращения модуль Manager занят другим пользователем, то будет выведен сервисный сигнал:

```
Manager is already connected to "port_number chanel_number"
CLR DTE C:0 D:0 - Call cleared by remote device
```

Примечание: Для обращения к модулю Manager с локального порта типа PAD (или с Telnet-станции типа PAD) дополнительных записей в таблице маршрутизации не требуется.

3.1.2 Параметры профиля X.3

Каждый параметр (всего 19) имеет свой десятичный номер и значение.

Значение параметра может быть установлено командой **SET** или **SET?** (см. п.п. 3.1.1.6, 3.1.1.7), а также с помощью команды **S P** модуля MANAGER (см. п. 4.2.2). Совокупность параметров порта называется ПРОФИЛЬ порта. ПРОФИЛЬ порта может быть выбран командой **PROF** (см. п. 3.1.1.12).

Приведенные ниже значения параметров определены стандартом X.3 CCITT.

1. СИГНАЛ ВНИМАНИЕ (Recall character)

0 — не используется
1 — символ DLE (CTRL-P)
2..127 — символ (ASCII), имеющий данный код.

2. ЭХО (Echo)

0 — эхо запрещено
1 — эхо разрешено

Примечание: Символы, определенные как СИГНАЛ ВНИМАНИЕ, символы редактирования (см. параметры 16, 17, 18) и управления потоком XON/XOFF не отображаются на терминале даже при разрешенном “ЭХО”.

3. СИГНАЛ ОТПРАВКИ (Forwarding characters)

0 — отсутствует (т. е. пакет отправляется по другим причинам [по длине или тайм-ауту])
1 — любой графический символ
2 — символ CR
4 — символы ESC, BEL, ENQ, ACK
8 — символы CAN, DEL, DC2
16 — символы ETX, EOT
32 — символы VT, HT, LF, FF
64 — любой символ с кодом менее 32, кроме упомянутых выше

Примечание: Все другие значения параметра из диапазона от 1 до 127 являются комбинацией (объединением) нескольких перечисленных значений.

4. ТАЙМ-АУТ (Idle timer delay)

0 — механизм отправки пакета не используется
1..255 — значение тайм-аута в двадцатых долях секунды

Примечание: При включенном режиме редактирования (параметр 15:1) данный параметр игнорируется.

5. ПОДЧИНЕННОЕ УСТРОЙСТВО (Ancillary Device Control)

Управление со стороны СРП (PAD) потоком данных, идущих от DTE.
0 — не производится
1 — используются XON/XOFF только при режиме передачи данных
2 — используются XON/XOFF при режиме передачи данных и команд

6. СЕРВИСНЫЙ СИГНАЛ (Control of PAD service signal)

0 — сигнал не выводится
1 — выводится "*" как признак готовности СРП (PAD) принять команду от DTE

7. РЕАКЦИЯ НА СИГНАЛ РАЗРЫВА ЛИНИИ (Break signal operation)

0 — действий не производится
1 — послать пакет ПРЕРЫВАНИЕ (INTERRUPT)
2 — послать пакет СБРОС (RESET)
4 — послать пакет индикация Break
8 — переход в командный режим
16 — очистка данных, готовых к отправке в DTE

Примечание: Все другие значения параметра из диапазона от 1 до 31 являются комбинацией (объединением) нескольких перечисленных значений.

8. ЗАПРЕТ ВЫВОДА (Discard Output)

0 — нормальная доставка данных
1 — вывод данных в DTE запрещен

9. ВСТАВКА СИМВОЛОВ ПОСЛЕ CR (Padding after Carriage Return)

Значение параметра определяет количество символов NUL, вставляемых СРП (PAD) после отправки CR в DTE. Параметр обычно используется при подключении медленного оборудования (например, принтера).

10. СВЕРТКА СТРОКИ (Line Folding)

0 — механизм не используется
1..255 — вставка CR LF после указанного числа графических символов, выведенных на DTE

11. СКОРОСТЬ ПЕРЕДАЧИ (Binary Speed)

0 — 110 бит/с
1 — 134 бит/с
2 — 300 бит/с
3 — 1200 бит/с
4 — 600 бит/с
5 — 75 бит/с
6 — 150 бит/с
7 — 1800 бит/с
8 — 200 бит/с
9 — 100 бит/с
10 — 50 бит/с
11 — 1200 бит/с
12 — 2400 бит/с
13 — 4800 бит/с
14 — 9600 бит/с
15 — 19200 бит/с
16 — 48000 бит/с
17 — 56000 бит/с
18 — 64000 бит/с

12. УПРАВЛЕНИЕ ПОТОКОМ (Flow Control)

Управление со стороны DTE потоком символов, идущих от СРП (PAD).
0 — управление не производится
1 — используются XON/XOFF при режиме передачи данных

13. ВСТАВКА СИМВОЛА ПЕРЕВОД СТРОКИ (Linefeed Insertion)

0 — нет вставки;
1 — вставка LF после CR, направленного в DTE;
2 — вставка LF в поток данных после CR, приходящего от DTE;
4 — вставка LF после эхо-CR, направленного в DTE.

Примечание: Все другие значения параметра из диапазона от 1 до 7 являются комбинацией (объединением) нескольких перечисленных значений.

14. ВСТАВКА СИМВОЛОВ ПОСЛЕ LF (Padding after Linefeed)

Значение параметра определяет количество символов NUL, вставляемых СРП (PAD) после отправки LF в DTE. Параметр обычно используется при подключении низкоскоростного оборудования (например, принтера).

15. РЕДАКТИРОВАНИЕ (Editing)

0 — редактирование не производится;
1 — редактирование производится.

16. УДАЛЕНИЕ СИМВОЛА (Character Delete)

Значение параметра определяет код символа, используемого для удаления предыдущего символа.

17. УДАЛЕНИЕ СТРОКИ (Line Delete)

Значение параметра определяет код символа, используемого для удаления введенной (но еще не отправленной) строки.

18. ПОВТОР СТРОКИ (Line Display)

Значение параметра определяет код символа, используемого для вывода в DTE введенной (но еще не отправленной) строки.

19. ТИП СИМВОЛА РЕДАКТИРОВАНИЯ (Editing PAD Service Signal)

0 — символы редактирования не посылаются на DTE;
1 — символы редактирования как для печатающего устройства "\" — уничтожение символа "XXX" — уничтожение строки;
2 — символы редактирования как для терминала: BS, SP, BS — уничтожение символа, n-раз (BS, SP, BS) — уничтожение строки длиной n.

3.1.3 Стандартные профили

Профилем порта называется совокупность 19 параметров, определяющих его функционирование. Набор параметров и их возможные значения определены в Рекомендации X.3 CCITT и приведены в п. 6.2 данного руководства. В зависимости от приложения, на практике часто используются некоторые фиксированные наборы значений параметров, называемые СТАНДАРТНЫЕ ПРОФИЛИ.

Четыре наиболее употребимых профиля устанавливаются командой "фабричные установки" и индексируются номерами 0, 1, 2 и 3.

Профиль 0 используется при терминальном обмене (абонент-абонент) или при диалоговом режиме с некоторым сервером сети, например с почтой. Профили 1, 2 и 3 (прозрачные профили) используются для пересылки файлов с помощью протоколов XMODEM, YMODEM, ZMODEM, KERMIT и других.

Ниже приведены значения параметров, устанавливаемых при выборе соответствующего профиля командой **PROF** (см. п. 3.1.12).

Название параметра	Профиль 0	Профиль 1	Профиль 2	Профиль 3
1. СИГНАЛ ВНИМАНИЕ	1	0	0	0
2. ЭХО	1	0	0	0
3. СИГНАЛ ОТПРАВКИ	2	127	0	0
4. ТАЙМ-АУТ	0	0	1	1
5. ПОДЧИНЕННОЕ УСТР.	0	0	0	0
6. СЕРВИСНЫЙ СИГНАЛ	1	1	1	0
7. РЕАКЦИЯ НА РАЗРЫВ ЛИНИИ	2	8	8	0
8. ЗАПРЕТ ВЫВОДА	0	0	0	0
9. ВСТАВКА ПОСЛЕ CR	0	0	0	0
10. СВЕРТКА СТРОКИ	0	0	0	0
11. СКОРОСТЬ ПЕРЕДАЧИ	—	—	—	—
12. УПРАВЛЕНИЕ ПОТОКОМ	1	0	0	0

13. ВСТАВКА СИМВОЛА LF	6	0	0	0
14. ВСТАВКА ПОСЛЕ LF	0	0	0	0
15. РЕДАКТИРОВАНИЕ	1	0	0	0
16. УДАЛЕНИЕ СИМВОЛА	8	0	0	0
17. УДАЛЕНИЕ СТРОКИ	24	0	0	0
18. ПОВТОР СТРОКИ	2	0	0	0
19. ТИП РЕДАКТИРОВАНИЯ	2	0	0	0

Примечание: скорость передачи (параметр 11) используется только для чтения и при команде **PROF** значения не меняет.

3.2 Подключение терминального оборудования по протоколу PPP

Подключение терминального оборудования (пользователя) через протокол PPP предполагает установление некоторого логического соединения и дальнейшую работу по нему.

Использование PPP-соединения позволяет получить:
- надежную передачу IP-трафика (контроль ошибок);
- проведение процедур аутентификации PAP, CHAP (на этапе подключения);
- уменьшение объема передаваемой информации за счет компрессии заголовков пакетов (Van Jacobson) и компрессии данных;
- назначение пользователю динамического IP-адреса и передача ему адресов DNS.

Настройка порта устройства NSG предполагает следующее:

- настройка параметров физического порта (п.2.2.1.7)
- настройка IP-интерфейса (п.2.2.3.2 и п.2.2.3.3)
- дополнение таблицы сценариев SCRIPT (п.2.2.7) - опционально
- дополнение таблицы паролей PAP,CHAP (п.2.2.7) - опционально
- настройка параметров DNS у интерфейса IP:0 - опционально

IP-интерфейс (типа PPP) связан с соответствующим физическим портом устройства (типа ASYNC_PPP) при помощи параметра PO:(номер порта) (п.2.2.3.2). Если в процессе работы устройства требуется изменить параметры функционирования порта, то после изменений требуется сначала рестартовать порт

W S PO:n

Затем следует рестартовать IP-интерфейс командой:

W S IP:m

При изменении параметров IP-интерфейса достаточно рестартовать данный интерфейс.

После переинициализации интерфейса (или рестарта всего устройства), установка соединения проходит несколько стадий (фаз); переход к каждой последующей осуществляется только при успешном завершении предыдущей (или отсутствия оной). Начало процедур установки соединения зависит от значения параметров SL: и DOD: (п.2.2.3.3).

Фазы установки соединения:

— **Выполнение начального сценария (опционально)**

Часто используется для подготовки к работе подключенного к порту модема, выполнение дозвона и входа в удаленую систему. Процедура подразумевает серию сообщений, посылаемых портом в модем и контроль откликов последнего.

Для выполнения этой фазы параметр SCRIPT соответствующего интерфейса должен быть отличен от 0 (п.2.2.3.3) и в таблице сценариев должна быть установлена соответствующая запись. Формат записи приведен в п.3.2.1.

— **Согласование параметров линии (Line Control Protoclol)**

На данном этапе происходит согласование основных параметров создаваемого соединения:

- размер MTU;
- особенности передачи управляющих символов (параметр AM:);
- использование или отсутствие механизмов сжатия полей адреса (параметр AC:) и протокола (параметр PC:) пакета PPP (п.2.2.3.3);
- использование процедур сжатия поля данных;
- наличие и тип процедур аутентификации (см.ниже п.3.2.2)

— **Выполнение процедур аутентификации (PAP, CHAP)**

Выполнение этих процедур предназначено для того, чтобы разрешить работу с данным интерфейсом устройства только ограниченному кругу пользователей. Протокол аутентификации является асимметричным, т.е. имеется сторона запрашивающая пароль (сервер) и сторона обязанная ответить на запрос (клиент).

Если IP-интерфейс устройства NSG работает в качестве сервера, то должен быть установлен как минимум один из следующих параметров:

PAPR:YES (если не требуется шифрования пароля)
CHAPR:YES (шифрование требуется)

Если IP-интерфейс устройства NSG работает в качестве клиента, (т.е. сам подключается к стороне, работающей как сервер) то должен быть установлен как минимум один из следующих параметров:

PAPA:YES (если не требуется шифрования пароля)
CHAPA:YES (шифрование требуется)

Формат записей таблицы паролей PAP/CHAP, а так же описание принципов аутентификации приведено в п.3.2.2.

— **Согласование и установка параметров IP (IP Control Protocol)**

На данном этапе происходит согласование основных параметров работы протокола IP:

- согласование IP-адресов

  Каждая из участвующих сторон может работать со своим (предустановленным) IP-адресом или же в данном этапе получить IP-адрес от удаленной стороны.

  В данной процедуре со стороны устройства NSG значимыми являются параметры IP-интерфейса ACCL:, RADR:, IADR: (п.2.2.3.2 и п.2.2.3.3)
- использование или отсутствие механизмов сжатия полей пакета IP, IP-заголовка (параметр VJ:) и идентификатора соединения (параметр VJC)
- согласование адресов DNS (устанавливаются параметрами DNS1 и DNS2 у интерфейса IP:0)

— **Согласование параметров процедур компрессии данных (Compression Control Protocol)**

На данном этапе происходит согласование параметров процедур сжатия данных. В устройстве NSG предусмотрен только один алгоритм компрессии данных, а именно BSD-компрессия (п.2.2.3.3).

Примечание: Используя механизмы компрессии следует иметь в виду, что они значительно расходуют ресурсы устройства NSG (в первую очередь оперативную память).

3.2.1 Использование таблицы сценариев

Часто до инициации PPP соединения необходимо установить связь с модемом. Для этого используется таблица сценариев. Если в параметрах PPP интерфейса значение параметра SCRIPT не равно 0, то будет выполняться сценарий. Этот сценарий находится в таблице сценариев в строке с номером,

указанным в параметре SCRIPT. Сценарий состоит из пар записей «ожидание-посылка» разделенных пробелами.

Запись представляет собой последовательность символов в кавычках или без них.

Например сценарий **ogin: "ppp" ssword: hello2u2**

означает, что программа будет ожидать с асинхронной линии последовательность «**ogin:**». Когда эта последовательность символов будет получена, в линию будет послана строка «**ppp**»(без кавычек!) и программа будет ожидать последовательность «**ssword:**» и так до конца сценария. Заметим, что каждая посылаемая последовательность символов дополняется символом <CR>. Если в качестве записи ожидания стоит пустая запись (""), то программа ничего не ждет и сразу переходит к посылке следующей записи. Концом сценария является конец строки в таблице сценариев. Если строка в таблице кончается символом '\', то следующая строка таблицы будет продолжением этого сценария.

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

"" ATZ OK ATDT5551212 CONNECT "" TIMEOUT 10 ogin: account

тайм-аут будет уменьшен до 10 секунд перед ожиданием строки «login:»

В качестве записи ожидания может быть записана последовательность пар строк «ожидание-посылка» разделенных дефисом '-'. Например в сценарии

TIMEOUT 5 OK-AT-OK ATDT5551212 TIMEOUT 60 CONNECT

если в течении 5 секунд не будут получены символы «OK», то будет послано «AT» и снова ожидаться «OK». Если же «OK» в течении 5 секунд будет получено то «AT» посылаться не будет, а сразу пошлется «ATDT5551212».

3.2.2 Принципы аутентификации для PPP интерфейса

Аутентификация основана на паролях, которые находятся в таблице паролей **PAP** или **CHAP**. Обе таблицы имеют одинаковый формат и хранят пароли для нескольких сочетаний сервера(сторона, запрашивающая пароль) и клиента (отвечающая сторона). Заметим, что PPP интерфейс может выступать как в роли сервера, так и в роли клиента.

Каждая строка таблицы состоит по крайней мере из 3 слов в порядке клиент, сервер, пароль. Любые следующие слова в строке рассматриваются как IP адреса, допустимые для данного клиента.

Если в строке ровно 3 слова, то для клиента допустимы любые IP адреса. Чтобы запретить все IP адреса используйте «-».

Слово «*» в качестве имени клиента или сервера означает любое имя.

Когда программа ищет пароль, она выбирает строку с минимальным количеством звездочек.

В таблице паролей хранятся как пароли, используемые для аутентификации

удаленного пользователя, так и пароли посылаемые в удаленный сервер для аутентификации себя. Выбор пароля из таблицы основан на локальном имени (параметр NAME) и удаленном имени (параметр RNAME).

Подключение удаленного пользователя, используя протокол PAP происходит в следующем порядке:

- устройство NSG имеет настройки IP-интерфейса:

  IP:m ADM:UP TY:PPP NAME:"IP_Name" PAPR:YES ....

  запрашивает пользователя об имени и пароле
- пользователь отвечает, присылая имя (например "User_id") и пароль ("User_pass")
- в таблице PAP находится строка, в которой первое поле (клиент) совпадает со значением "User_id", например

  клиент сервер пароль

  PAP:n User_id IP_Name User_pass
- второе поле записи (сервер) "IP_Name" сравнивается с именем данного IP-интерфейса (параметр NAME:)
- в случае совпадения производится последнее сравнение пароля, присланного пользователем (User_pass) и записи в строке PPP.

В случае совпадения при последнем сравнении процесс поиска заканчивается - пользователь аутентифицирован.

Для аутентификации себя (как клиента) программа ищет строку со значением клиента (первое слово в строке) равного локальному имени (параметр NAME) и со значением сервера (второе слово в строке) равного удаленному имени (параметр RNAME). Пароль из этой строки посылается удаленной стороне.

Если при аутентификации удаленного пользователя в таблице найдена строка с пустым паролем («»), то принимается любой пароль от удаленной стороны.

В таблице паролей **CHAP** поиск пароля проводится следующим образом:

- Для аутентификации удаленного пользователя программа ищет строку со значением клиента (первое слово в строке) равного имени определенного в сообщение CHAP (CHAP-Response message) и со значением сервера(второе слово в строке) равного локальному имени (параметр NAME).
- Для аутентификации себя (как клиента) программа ищет строку со значением клиента (первое слово в строке) равного локальному имени(параметр NAME) и со значением сервера (второе слово в строке) равного имени определенному в сообщение CHAP (CHAP-Challenge message). Пароль из этой строки посылается удаленной стороне.

3.3 Подключение терминального оборудования по протоколу ASYNC

Работа порта типа ASYNC может происходить в одном из двух режимов:
- работа без использования какого-либо протокола (прозрачный режим)
- работа в протокольном режиме PAD или PPP, используя один из способов аутентификации.

3.3.1 Работа асинхронного порта в прозрачном режиме

Работа порта в прозрачном режиме предполагает передачу дуплексного асинхронного потока без какой-либо обработки. Символы, получаемые портом, группируются в пакет.

Условия отсылки пакета:
- за время, равное времени приема одного символа, не будет принят очередной символ из порта;
- достижение максимального размера пакета.

Принимаемые для данного асинхронного порта пакеты помещаются в буфер, и из этого буфера посимвольно выводятся в порт. Настройка асинхронного порта в прозрачном режиме:

PO:n TY:ASYNC SP:57600 AU:0 LG:128

Параметр LG: определяет максимальный размер пакета.

Данный порт может быть скоммутирован (используя постоянные виртуальные цепи) со следующими типами объектов:

- Frame Relay - станция (тип TY:ASYNC);
- Telnet - станция (тип TY:ASYNC);
- физический порт (тип TY:ASYNC), работающий в прозрачном режиме.

Примечание: При коммутации асинхронного порта и Frame Relay-станции возможны потери данных из-за отсутствия механизмов управления потоком в сети Frame Relay.

Для избежания этого следует ограничить соответствующий канал (DLCI) (используя CIR, BC, BE) скоростью, не превышающей значение параметра SP: асинхронного порта.

3.3.2 Работа асинхронного порта в протокольном режиме

В отличие от других режимов работы, этот режим предполагает динамическое назначение первичного протокола (PAD или PPP) в зависимости от результатов аутентификации, подключившегося в данный момент пользователя.

Аутентификация пользователя может происходить локально (то есть вся информация о пользователях установлена в параметрах конфигурации устройства), так и удаленно (когда устройство обращается за информацией к некоторому удаленному серверу).

И в том и в другом случае начало работы порта после подключения пользователя происходит одинаково:

- порт выводит строку (login:), предлагая пользователю ввести имя (идентификатор);
- пользователь отвечает на запрос, завершая строку символом <CR> (возврат каретки);
- порт выводит строку (password:), предлагая пользователю ввести пароль;
- пользователь вводит строку-пароль (не отображается на экране).

> ***Примечание:*** Ввод имени и пароля может осуществляться как в терминальном окне (вручную), так и использованием некоторого скрипта (сценария), выполняемого на компъютере пользователя.

Получив имя и пароль пользователя, система производит аутентификацию, в соответствии со способом, на который ссылается параметр AU: данного порта.

Например:
настройки порта <m>:
PO:<m> TY:ASYNC AU:<n>
настройки способа аутентификации:
AU:<n> TY:<тип способа аутентификации> <параметры>

Описание параметров для настройки различных способов аутентификации приведено в п.2.2.8. Пример использования некоторых способов аутентификации приведен в п.3.3.x.x.

В случае локальной аутентификации (TY:LOCAL) и удаленной аутентификации (TY:TACACS+) протокол PAD или PPP определяется суффиксом в имени пользователя. Суффикс - это часть имени после точки.

Если, например, с этого порта набрать:

login: user1.ppp
Password: psw1 то начнется PPP сессия.

Если набрать:

login: user1.pad
Password: psw1 то начнется PAD сессия.

Если суффикс не набирать:

login: user1
Password: psw1 то также начнется PAD сессия.

У порта типа ASYNC имеются все параметры порта типа PAD, однако они не выводятся по команде D P PO:<m> . Это параметры используются только в том случае, когда после аутентификации порту назначен протокол PAD. Чтобы их посмотреть или изменить, надо временно назначить этому порту TY:PAD.

Пример:

```
Manager: d p po:3

PO:03  TY:ASYNC     SP:9600    AU:1 ACCT:YES
Manager: s p po:3 ty:pad

PO:03 TY:PAD IF:V24 SP:9600 AF:8N1 CO:NO AP:NO RP:NO AC:NO CM:NO
      LG:128 MB:NO CD:YES BI:0 AD:NO PT:"*" MS:""
     1:1  2:1  3:2  4:0  5:0  6:1  7:2  8:0  9:0  10:0  11:14
     12:1  13:4  14:0  15:1  16:8  17:24  18:2  19:2
Manager: s p po:3 prof:2

PO:03 TY:PAD IF:V24 SP:9600 AF:8N1 CO:NO AP:NO RP:NO AC:NO CM:NO
       LG:128 MB:NO CD:YES BI:0 AD:NO PT:"*" MS:""
      1:0  2:0  3:0  4:1  5:0  6:1  7:8  8:0  9:0  10:0  11:14
       12:0  13:0  14:0  15:0  16:0  17:0  18:0  19:0
Manager: s p po:3 ty:async

PO:03  TY:ASYNC     SP:9600    AU:1 ACCT:YES
Manager: w f
Manager: w s po:3
```

Теперь при входе через Login PAD будет работать в прозрачном профиле.

Для одновременной работы нескольких пользователей (при подключении их через порты ASYNC, используя протокол PPP, в системе должно быть сконфигурировано столько же свободных IP-интерфейсов типа PPP. Параметр порт (PO:) у этих интерфейсов должен быть установлен в значение "AUTO".

3.3.2.1 Локальная аутентификация для асинхронного порта

При локальной аутентификации вся информация о пользователях установлена в параметрах конфигурации устройства.

Последовательность проверки имени и пароля подключившегося пользователя пояснено на следующем примере.

Пусть устройство NSG имеет настройки:

способ аутентификации - AU:n TY:LOCAL ID:"LA_Name"
настройка порта - PO:m TY:ASYNC AU:<n>
строка в таблице PAP - PAP:k User_id LA_Name User_pass

Получив от пользователя имя (например "User_id") и пароль ("User_pass") производятся следующие действия:

— в таблице PAP находится строка, в которой первое поле (клиент) совпадает со значением "User_id"
— второе поле записи (сервер) "LA_Name" сравнивается с идентификатором способа аутентификации (параметр ID:)
— в случае совпадения производится сравнение пароля, присланного пользователем (User_pass) и записи в строке PAP.

В случае совпадения при последнем сравнении процесс поиска заканчивается - пользователь аутентифицирован.

Выбор протокола (PAD или PPP) при подключения пользователя происходит на основании имени пользователя (ответ после запроса "login:")

3.3.2.2 Удаленная аутентификация с использованием RADIUS и TACACS+

При подключении пользователя оба способа аутентификации посылают запрос некоторому удаленному серверу (RADIUS или TACACS). В ответе, посылаемом RADIUS или TACACS-сервером, указывается возможность работы пользователя и некоторые параметры (протокол, IP-адрес и пр.).

При аутентификации через RADIUS используется обмен UDP-пакетами. Настройка параметров при использовании системы RADIUS описана в п.2.2.8.3

Для того, чтобы сессия открылась как PPP, RADIUS сервер должен в ответе прислать атрибуты:

Service-Type = Framed-User
Framed-Protocol = PPP

Для того, чтобы сессия открылась как PAD, RADIUS сервер должен в ответе прислать атрибуты:

Service-Type = Login-User
Login-Service = Rlogin

При других значениях этих атрибутов никакая сессия открываться не будет. При открытии PPP сессии будут восприниматься следующие атрибуты:

Framed-IP-Address
Framed-Compression
Idle-Timeout

Остальные атрибуты игнорируются.

При аутентификации через TACACS+ используется TCP-соединение. Настройка параметров при использовании системы TACACS+ описана в п.2.2.8.4

Выбор протокола (PAD или PPP) при подключения пользователя происходит на основании имени пользователя (ответ после запроса «login:»)

Во фрагменте файла TACACS-сервера для следующих пользователей допустимы следующие протоколы и параметры:

```
пользователь   пароль   протоколы     параметры
igor            gar     pad, ppp   40.30.20.10  (ip-адрес
                                          для  ppp-сессии)
mike            mike          pad        -
boba            boba          ppp   40.30.20.10  (ip-адрес
                                          для  ppp-сессии)
```

```
user = igor {
  login = cleartext gar
  service = pad { }
  service = ppp protocol = ip {
    addr = 40.30.20.10
  }
}

user = mike {
  login = cleartext mike
  service = pad { }
}

user = boba {
  login = cleartext boba
  service = ppp protocol = ip {
    addr = 40.30.20.11
  }
}
```

3.3.3 Пример использования способов аутентификации

Способы аутентификации могут быть заданы например следующим образом:

```
S P AU:01  TY:RADIUS  IADR:10.0.0.17  TO:10  RT:3  ID:"NX-300_"  SN:2
       SADR:10.0.0.10   KEY:"nsgKey"
       SADR1:10.0.0.12   KEY1:"keyPPP"
S  P  AU:02   TY:LOCAL   ID:"au02"
S  P  AU:03   TY:LOCAL   ID:"au03"
S  P  AU:04   TY:NO_AUTH
```

Если в маршрутизаторе есть порты следующих типов:

```
PO:04   TY:ASYNC      SP:57600    AU:1
PO:05   TY:ASYNC      SP:57600    AU:2
PO:06   TY:ASYNC      SP:57600    AU:3
PO:07   TY:ASYNC      SP:57600    AU:4
PO:08   TY:ASYNC      SP:57600    AU:4
```

то пользователь, входящий в маршрутизатор через порт 4, будет аутентифицироваться через RADIUS, причем если сервер с адресом 10.0.0.10 не работает, запрос будет направлен в сервер с адресом 10.0.0.12.

Пользователи, входящие в маршрутизатор через порт 5 или порт 6 будут аутентифицироваться локально. Если, например, таблица паролей имеет следующий вид:

```
PAP  Passwords  Table
#    <client> <server>    <secret> [<IP address> ...]
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
01   igor      au03         psw
02   boba      au02         secr
03   mike      au02         mypsw
04   borya     *            ""
```

то пользователь igor может войти только через порт 6, пользователи boba и mike могут войти только через порт 5, а пользователь borya может войти через любой порт (5 или 6) и без пароля.

Через порты 7 и 8 в маршрутизатор не сможет войти ни один клиент.

3.4 Эмуляция абонентского оборудования (AntiPAD)

3.4.1. Назначение AntiPAD

В сетях X.25 порты с протоколом X.28 (PAD) традиционно служат для подключения оконечного асинхронного оборудования. Однако бывают случаи, когда оконечное оборудование с синхронным X.25 выходом необходимо состыковать с асинхронной линией X.28, предоставляемой провайдером. В этом случае может помочь устройство NSG с функцией AntiPAD.

AntiPAD - это разновидность порта типа PAD, который можно подключать к стандартному PAD порту и транслировать через эту связку пакеты X.25.

Линия PAD - AntiPAD не может в полном объеме выполнять функции стандартной синхронной линии X.25, поэтому она имеет следующие ограничения:

- может быть установлено только одно логическое соединение;
- транслируются только пакеты CALL, CALL CONFIRMATION, CLEAR, DATA.
- для вызова (пакет CALL) идущего в направлении AntiPAD - PAD могут быть переданы только вызываемый адрес (called address) и данные пользователя (call user data);
- для вызова (пакет CALL) идущего в направлении PAD - AntiPAD может быть передан только вызывающий адрес (calling address);
- пакеты CLEAR приводят только к падению физического сигнала на линии;
- при передаче пакета данных теряется информация содержащаяся в заголовке пакета (M-bit, Q-bit, D-bit);
- при передаче данных не гарантируется их сборка в пакеты той же длины какими они были изначально.

Несмотря на эти ограничения AntiPAD может успешно выполнять функции по установке/разрыву соединений и передаче данных для некоторых конкретных приложений (См. пример ниже). Если AntiPAD использовать вместе со службой XOX (см. 4.5), то все вышеперечисленные ограничения снимаются.

3.4.2 Настройка AntiPAD

Порт, настроенный как AntiPAD работает следующим образом:

1. При приходе вызова (пакет CALL) AntiPAD формирует командную строку в соответствии со стандартом X.28 (адрес назначения в символьном виде) и передает ее по асинхронной линии в PAD, к которому он подключен. В командной строке перед адресом может быть вставлена строка (см. параметр PREF) и/или добавлены данные пользователя после адреса (см. параметр UD).

2. AntiPAD ожидает от PADa последовательность символов, сигнализирующую об установке соединения. Эта последовательность задается параметром CACF (обычно CACF="COM\r\n"). При получении данной последовательности AntiPAD посылает пакет CALL CONFIRMATION в сторону абонента, выдавшего CALL. Если во время ожидания на линии падает физический сигнал, AntiPAD воспринимает это как разрыв соединения и посылает пакет CLEAR в сторону абонента, выдавшего CALL.

3. После установки соединения AntiPAD работает в режиме обычного PADa с профилем 1:0, 2:0, 3:0, 4:n, 5:0, 6:0, 7:0, 8:0, 9:0,10:0, 12:0, 13:0, 14:0, 15:0, 16:0, 17:0, 18:0, 19:0, 20:0, 21:0, 22:0

4. Когда на AntiPAD приходит запрос на разрыв соединения (пакет CLEAR), он роняет физический сигнал на линии на 2 секунды и переходит в командный режим.

5. В командном режиме AntiPAD может получить пакет CALL и далее см. п.1 и одновременно ожидает от PADa последовательность символов сигнализирующую об установке соединения. Эта последовательность состоит из вызывающего адреса и строки которая задается параметром CALL (обычно CALL="COM\r\n"). При получении данной последовательности AntiPAD формирует пакет CALL в котором в качестве вызываемого адреса (called address) устанавливается значение параметра AD, а в качестве вызывающего адреса (calling address) устанавливается значение, полученное от PADa. Далее см. п. 3.

Чтобы связка PAD - AntiPAD работала, в первую очередь должен быть правильно настроен стандартный PAD, а именно:

- PAD должен ронять физический сигнал на линии при приходе пакета CLEAR из сети (со стороны PAD) и восстанавливать его через некоторое время;
- PAD должен воспринимать падение физического сигнала на линии как разрыв логического соединения;
- PAD должен воспринимать стандартные команды X.28;
- у PAD должен быть "прозрачный" профиль и он должен передавать сервисные сигналы в стандартном виде, т.е. профиль должен быть следующим 1:0,2:0,3:0,4:1,5:0,6:5,7:0,8:0,9:0,10:0,12:0,13:0,14:0,15:0,16:0,17:0, 18:0,19:0,20:0,21:0,22:0 (параметр 4 может быть любым отличным от нуля).

Для настройки порта в устройстве NSG в режиме AntiPAD необходимо выполнить команду:

S P PO:n TY:PAD IF:ANTI [<par>:<val>...]

где **n** - номер порта;
<par> - имя параметра;
<val> - значение параметра .

Параметры **SP, AF, LG, MB, BI** имеют тот же смысл и значение, как и для обычного порта типа PAD.

Новые параметры и параметры имеющие иной смысл описаны ниже.

Параметр PREF (Call Prefix)

Параметр **PREF** определяет последовательность символов, которая будет добавлена перед адресом в командную строку, которая формируется портом AntiPAD для посылки в подключенный к нему PAD.

Синтаксис: **PREF:<string>**
где **<string>** - последовательность символов в кавычках или без них.

Пример: **PREF:"c "**

Параметр CACF (Call Confirmation Pattern)

Параметр CACF определяет последовательность символов, которую будет ожидать AntiPAD со стороны PADa как сигнал о подтверждении установки соединения. После посылки командной строки вызова, AntiPAD будет игнорировать все символы, принимаемые от PADa, пока не встретит последовательность заданную как образец в параметре CACF. Все следующие за образцом символы будут уже восприниматься как данные.

Синтаксис: CACF:<string>
где <string> - последовательность символов в кавычках или без них.

Пример: CACF:"COM\r\n"

Параметр CALL (Call Pattern)

Параметр CALL определяет последовательность символов, которую будет ожидать AntiPAD как сигнал об установке соединения со стороны PADa.

В командном режиме AntiPAD игнорирует все символы, принимаемые от PADa, пока не встретит последовательность заданную как образец в параметре CALL. Десятичное число принятое перед этим образцом будет подставлено в качестве вызывающего адреса(calling address) в пакет вызова(CALL).

Все следующие за образцом символы будут уже восприниматься как данные.

Синтаксис: CALL:<string>
где <string> - последовательность символов в кавычках или без них.

Пример: CALL:"com\r\n"

Параметр AD (Address)

Параметр AD определяет вызываемый адрес(called address), который подставляется в пакет вызова(CALL), формируемый AntiPADом после получения сигнала об установке соединения от PADa (см. параметр CALL).

Синтаксис: AD:n
где n - X.121 адрес.

Пример: AD:25019999990002

Параметр UD (User Data adding)

Параметр UD определяет добавлять или нет данные пользователя в командную строку вызова передаваемую в PAD.

Синтаксис:
UD:YES - добавлять данные пользователя в командную строку;
UD:NO - не добавлять данные пользователя в командную строку;

Параметр DT (Clear Delay Time)

Параметр DT определяет временную задержку между падением сигнала на физической линии и посылкой пакета CLEAR.

Синтаксис: **DT:n**

где **n** - время задержки в секундах.

Пример: DT:10

Параметр 4 (Idle Timer Delay)

Параметр имеет тот же смысл и значение, как и для обычного порта типа PAD, только не должен принимать нулевое значение.

3.4.3 Пример настройки AntiPAD

Дано:

- банкомат с синхронным X.25 каналом;
- асинхронная X.28 (PAD) линия, предоставленная провайдером.

Необходимо чтобы банкомат устанавливал соединение с хостом через эту асинхронную линию. Адрес хоста в сети - 250199999909.

Берем NSG устройство, подключаем порт 1 к банкомату, порт 2 к асинхронной линии.

Производим следующие настройки NSG устройства:

S P PO:1 TY:X25 IF:V24 TE:DCE LC:8 SP:9600 FW:7 N2:9 LG:128 PW:2 T1:3 T2:180 AP:NO BI:0

S P PO:2 TY:PAD IF:ANTI SP:2400 AF:8N1 LG:128 MB:YES BI:0 AD:2222 4:10 DT:4 UD:NO PREF:"" CACF:"COM\r\n" CALL:"COM\r\n"

S R PR:1 ID:D RT:2222 PO:1
S R PR:2 ID:D RT:250199999909 PO:2

Примечания: 1. AntiPAD не может передать PADy calling address из пакета вызова приходящего от банкомата, поэтому он должен формироваться самим PADом.

2. Параметр DT не должен быть равен нулю, т.к. в случае падения сигнала на асинхронной линии, банкомат немедленно получит пакет CLEAR и тут же попытается установить новое соединение, а линия X.28 несколько секунд после падения сигнала может находиться в нерабочем состоянии.

3. Параметр MB=YES, т.к. пакет данных от хоста может превышать 128 байт.

4. Параметр 4 должен имет значение которое гарантирует, что пакет данных от хоста полностью прошел через асинхронную линию.

3.5 Многоканальный асинхронный порт (МАП)

3.5.1 Организация обмена информации и структура пакета

Многоканальный асинхронный порт (**МАП**) представляет собой расширение стандартных возможностей PAD-порта, что позволяет организовать обмен информацией сразу с **несколькими абонентами** по установленным логическим каналам сети X.25.

Передача данных осуществляется пакетами, которые формируются из асинхронного потока данных, путем использования SLIP-протокола (RFC-1055). Сформированные пакеты передаются по заранее установленным (проключенным) логическим каналам (Permanent Virtual Circuits (PVCs)).

В каждом пакете имеется 2-х байтный заголовок. Первый байт заголовка определяет номер логического канала, которому принадлежит данный пакет. Номер логического канала должен лежать в диапазоне от 1 до LCN-max, определенного для данного порта параметром LC.

Второй байт заголовка используется для указания типа передаваемой информации (команда/данные). Использование PVC предполагает передачу только пакетов данных, поэтому младший бит второго байта должен иметь нулевое значение. Остальные значения зарезервированы для дальнейшего использования (при работе по Switched Virtual Circuits (SVCs)).

При получении из МАП пакета, первый байт заголовка указывает номер логического канала, с которого он получен, второй байт равен 0. Если значение второго байта не ноль, то произошло:

- использование канала LCN вне диапазона 1..LCN-max;
- использование непроключенного канала;
- нарушение структуры пакета.

При передаче пакетов, длина которых превосходит максимальную длину пакета для канала X.25, коммутатор NSG использует механизм M-bit'a.

3.5.2 Настройка МАП

Определение порта МАП осуществляется командой Manager:

S P PO:n TY:PAD IF:MULTI SP:s LC:LCN-max

в которой:

n - номер порта МАП
s - скорость передачи (см. настройки PAD-порта)
LCN-max - максимальное число логических каналов МАП

3.5.3 Управление потоком данных

Если адаптер не успевает обрабатывать данные, он выдает команду XOFF для конкретного логического канала. После получения команды XOFF прикладная программа обязана прекратить передачу данных в адаптер по

данному логическому каналу. В противном случае возможна потеря данных. По другим каналам работу можно продолжать. Передачу данных можно возобновить после получения команды XON.

Управление потоком сигналом CTS остается. При падении сигнала CTS необходимо прекратить передачу данных по всем логическим каналом порта.

Команда XOFF представляет собой пакет следующего формата:

1-й байт - Номер логического канала;
2-й байт - XXXXXXX1 (младший бит =1, остальные произвольные);
3-й байт - 00010011 (hex 13);

Команда XON представляет собой пакет следующего формата:

1-й байт - Номер логического канала;
2-й байт - XXXXXXX1 (младший бит =1, остальные произвольные);
3-й байт - 00010001 (hex 11);

3.5.4 Пример программирования

Ниже приведены функции на языке C, которые реализуют прием и передачу пакетов по асинхронной линии (пример из RFC-1055).

```
/* SLIP special character codes
 */
#define END             0300    /* indicates end of packet */
#define ESC             0333    /* indicates byte stuffing */
#define ESC_END         0334    /* ESC ESC_END means END data byte */
#define ESC_ESC         0335    /* ESC ESC_ESC means ESC data byte */

/* SEND_PACKET: sends a packet of length "len", starting at
 * location "p".
 */
void send_packet(p, len)
        char *p;
        int len; {
  /* send an initial END character to flush out any data that may
   * have accumulated in the receiver due to line noise
   */
     send_char(END);
  /* for each byte in the packet, send the appropriate character
   * sequence
   */
       while(len--) {
```

```
            switch(*p) {
             /* if it's the same code as an END character, we send a
              * special two character code so as not to make the
              * receiver think we sent an END
             */
            case END:
                    send_char(ESC);
                    send_char(ESC_END);
                    break;
             /* if it's the same code as an ESC character,
              * we send a special two character code so as not
              * to make the receiver think we sent an ESC
             */
            case ESC:
                    send_char(ESC);
                    send_char(ESC_ESC);
                    break;
             /* otherwise, we just send the character
             */
            default:
                        send_char(*p);
                        }
            p++;
              }
        /* tell the receiver that we're done sending the packet
         */
        send_char(END);
        }
/* RECV_PACKET: receives a packet into the buffer located at "p".
 *      If more than len bytes are received, the packet will
 *      be truncated.
 *      Returns the number of bytes stored in the buffer.*/
int recv_packet(p, len)
       char *p;
       int len; {
       char c;
       int received = 0;
        /* sit in a loop reading bytes until we put together
        * a whole packet.
         * Make sure not to copy them into the packet if we
        * run out of room.
         */
        while(1) {
              /* get a character to process
```

```
 */
  c = recv_char();
 /* handle bytestuffing if necessary
 */
switch(c) {
 /* if it's an END character then we're done with
  * the packet
 */
case END:
        /* a minor optimization: if there is no
         * data in the packet, ignore it. This is
           * meant to avoid bothering IP with all
           * the empty packets generated by the
           * duplicate END characters which are in
         * turn sent to try to detect line noise.
        */
       if(received)
              return received;
          else
                 break;
 /* if it's the same code as an ESC character, wait
   * and get another character and then figure out
   * what to store in the packet based on that.
    */
case ESC:
       c = recv_char();
        /* if "c" is not one of these two, then we
         * have a protocol violation.  The best bet
         * seems to be to leave the byte alone and
         * just stuff it into the packet
        */
       switch(c) {
       case ESC_END:
              c = END;
              break;
       case ESC_ESC:
              c = ESC;
              break;
             }
 /* here we fall into the default handler and let
   * it store the character for us
    */
default:
       if(received < len)
```

```
                    p[received++] = c;
                }
            }
        }
```

3.6 Трансляция IP-адресов (NAT)

3.6.1 Назначение и принципы трансляции IP-адресов

Работа маршрутизатора заключается в том, чтобы для любого пакета, входящего с какого-либо интерфейса, определив маршрут, направить этот пакет с другого интерфейса. При этом, как правило, содержимое пакета не меняется. Информация о источнике и пункте назначения пакета содержится в адресной части заголовка пакета. Уникальность адресов и настройки всех маршрутизаторов Internet дают возможность доставить пакет от любого источника любому потребителю и обратно. Назначение адресов лежит в ведении провайдеров услуг Internet (ISP - Internet Servise Provider). Легально назначенные адреса являются глобально уникальными (далее в тексте называются **глобальными адресами**). Сети, использующие только глобальные адреса, называются далее в тексте "**внешние сети**".

Существует большое количество сетей, адреса в которых назначаются их администраторами произвольно. Адреса в таких сетях будем называть **локальными**. Далее в тексте такие сети называются "**внутренними сетями**" и они успешно работают в том случае, когда обмен информации осуществляется только в рамках данной сети (например внутри LAN-сети предприятия).

Проблемы возникают когда машина, принадлежащая внутренней сети (и имеющая локальный адрес), пытается получить доступ к ресурсу, расположенному во внешней сети. Дело в том, что зная адрес во внешней сети, машина пользователя формирует и направляет во внешнюю сеть запрос, подставляя в качестве обратного адрес свой (локальный) адрес. Ответ на такой запрос просто "не вернется" обратно во внутреннюю сеть.

Одно из решений этой проблемы заключается в преобразовании адресной части заголовка пакета, при выводе пакета во внешнюю сеть (см.рис.3.1). Этот механизм получил название Network Address Translation (трансляция сетевых адресов) - **NAT**.

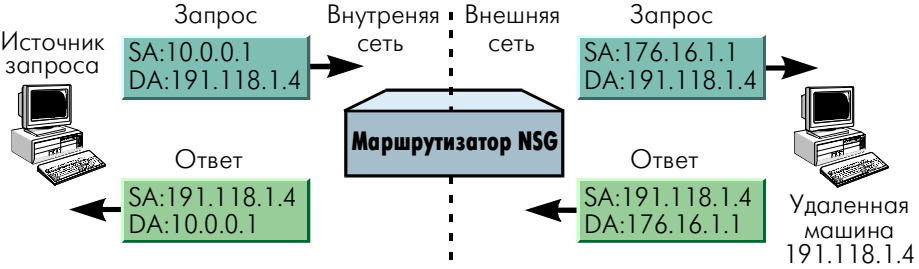

Рис.3.1 Схема трансляции заголовка

Запрос пользователя, имеющий в качестве адреса источника - адрес внутренней сети попадает в маршрутизатор. Адрес источника запроса - локальный (SA = 10.0.0.1), адрес получателя - глобальный (DA = 191.118.1.4). При выводе этого пакета с интерфейса, использующего NAT, адрес источника подменяется одним из глобальных адресов. Теперь адрес источника - глобальный (SA = 176.16.1.1).

Примечание: Для работы с внешними сетями, внутренняя сеть должна как минимум иметь один глобальный адрес.

Удаленная машина, обработав запрос, посылает ответ (SA = 191.118.1.4, DA = 176.16.1.1). Когда ответ на запрос приходит в маршрутизатор, происходит обратное преобразование, и пакет отправляется во внутреннюю сеть уже с исходным (локальным) адресом (SA = 191.118.1.4, DA = 10.0.0.1).

Преобразование (трансляция) выполняется только для пакетов, относящихся к протоколам ICMP, UDP и TCP.

3.6.2 Механизм трансляции адресов

С точки зрения выполнения процедур адресной трансляции IP-интерфейс может находится в одном из следующих режимов:

- не выполнять адресную трансляцию;
- выполнять адресную трансляцию для конкретных диапазонов внутренних (локальных) адресов;
- выполнять трансляцию всех выводимых пакетов.

В первом случае, параметр NAT:NO. Никаких преобразований в пакете не производится. Интерфейс работает в обычном режиме.

Во втором случае, параметр NAT:YES и таблица трансляции интерфейса (см.п.3.6.3) содержит как минимум одну запись. IP-интерфейс выполняет адресную трансляцию только для тех пакетов, которые находятся в диапазоне транслируемых адресов. Эти диапазоны (внутренние сети) и соответствующие им значения глобальных адресов определены в таблице трансляции. Остальные пакеты передаются "прозрачно" без какой-либо обработки.

В третьем случае, параметр NAT:YES и таблица трансляции интерфейса не содержит ни одной записи. IP-интерфейс выполняет адресную трансляцию для всех пакетов, отправляемых с него. Для преобразования будет использован адрес данного интерфейса.

При прохождении пакетов из внутренней сети (с локальными адресами) во внешнюю, каждый пакет (принадлежащий указанным выше протоколам) проходит обработку. Информация об этом пакете (или TCP-соединении) сохраняется в таблице. Эта информация используется для выполнения процедур обратного преобразования, при поступлении ответа (или пакета по установленному TCP-соединению) из внешней сети во внутреннюю. Одновременная работа

множества абонентов через один глобальный IP адрес достигается за счет динамического выделения отдельных идентификаторов пакетов (ICMP) и номеров портов (UDP/TCP) для каждого отправляемого во внешнюю сеть пакета.

Примечание: Для внешней сети устройства, находящиеся во внутренней сети и имеющие локальные адреса, недоступны.

Ниже приведен пример (см. рис. 3.2), в котором через один интерфейс выходят в Internet как станции внутренней сети, имеющие локальные адреса (используя трансляцию), так и станции внутренней сети, имеющие глобальные адреса.

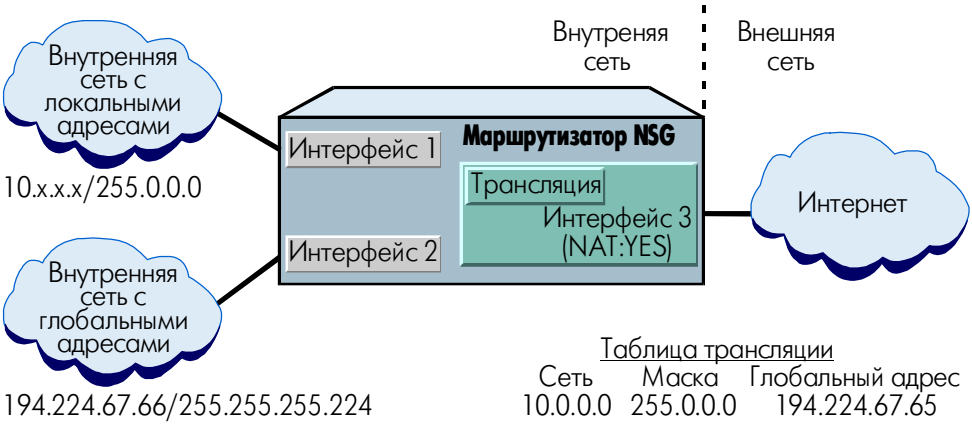

Рис.3.2 Трансляция конкретных диапазонов адресов

Если не транслируемые адреса являются глобальными, то соответствующие им станции внутренней сети будут "видимы" из внешней сети. Таким образом, через один и тот же интерфейс с NAT могут работать как станции с внутренними (локальными) адресами так и станции с внешними (глобальными) адресами.

В отличие от станций с внутренними адресами, станции с внешними адресами могут быть не только клиентами, но и серверами для станций из внешней сети.

3.6.3 Таблицы трансляции адресов

Таблица трансляции адресов формируется для каждого интерфейса отдельно. В ней каждая запись определяет внутреннюю сеть (диапазон локальных адресов - адрес/маска) и внешний (глобальный) адрес, подставляемый при процедуре трансляции.

Ниже приведен пример таблицы трансляции, определяющий для трех внутренних сетей (20.0.0.0/8, 30.0.0.0/8 и 40.0.0.0/8) соответственно три глобальных внешних адреса.

```
Manager: d n ip:2

  IP:2  EADR:174.74.56.10  IADR:20.0.0.0  MASK:255.0.0.0
  IP:2  EADR:174.74.56.11  IADR:30.0.0.0  MASK:255.0.0.0
  IP:2  EADR:174.74.56.12  IADR:30.0.0.0  MASK:255.0.0.0
```

Примечание 1: Один глобальный адрес можно использовать в качестве адреса трансляции для нескольких внутренних сетей.

Примечание 2: Допустимо использовать перекрывающиеся области внутренних (локальных) адресов, указывая им разные глобальные адреса. В этом случае глобальный адрес будет выбран по принципу "первый подходящий".

Примечание 3: В качестве транслируемого можно указать конкретный адрес, определив маску - MASK:255.255.255.255.

Для формирования таблицы трансляции используются команды - "S N ...", создать запись, удалить запись - "C N ..." и просмотреть всю таблицу - "D N ...". Синтаксис команд для формирования таблицы трансляции подробно описан в п.2.2.4.7 (глава 2).

Все записи таблицы трансляции наряду с другими параметрами сохраняются в энергонезависимой памяти устройства (при команде W F).

При добавлении новой записи трансляции она вносится в таблицу параметров и будет использована только после рестарта данного IP-интерфейса. При рестарте IP-интерфейса происходит активизация всех записей трансляции относящихся к данному IP-интерфейсу.

Все используемые в данный момент (динамические) записи трансляции можно посмотреть командой

D S NAT:n
где n-номер IP-интерфейса

Записи выводятся в начале таблицы.

```
Manager: dsnat:2
```

```
~~~~~~~~~~~ Current Address Translation Table ~~~~~~~~~~~~~~

External Address        Internal Net        Subnet Mask
194.67.244.230           10.0.0.0           255.0.0.0

# Prot  Internal IP/Port(ID) Local IP/Port(ID) Remote IP/Port
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
1  TCP    10.0.0.21  2044   194.67.244.230   59514   205.188.7.144   5190
2  TCP    10.0.0.21  2043   194.67.244.230  59521    194.67.35.191  80
3  TCP    10.0.0.19  1094   194.67.244.230  59046    208.239.159.17 80
4  TCP    10.0.0.19  1097   194.67.244.230  59047    208.239.159.17 80
5  TCP    10.0.0.15  2164   194.67.244.230  59051    212.5.91.195   110
6  TCP    10.0.0.15  2165   194.67.244.230  59052    194.67.23.76   110
7  TCP    10.0.0.15  2166   194.67.244.230  59053    213.180.193.87 110
8  TCP    10.0.0.21  2310   194.67.244.230  59054    213.180.194.130  80
9  TCP    10.0.0.21  2312   194.67.244.230  59056    194.67.35.195    80
10  TCP   10.0.0.21  2315   194.67.244.230  59059    213.180.194.113  80
11  TCP   10.0.0.21  2317   194.67.244.230  59060    213.180.194.131  80
12  TCP   10.0.0.21  2316   194.67.244.230  59061    213.180.194.130  80
13  TCP   10.0.0.21  2318   194.67.244.230  59062    213.180.194.113  80
14  TCP   10.0.0.21  2319   194.67.244.230  59063    213.180.194.131  80
15  TCP   10.0.0.19  1098   194.67.244.230  59064    208.239.159.17   80
```

3.7 Сбор учетной информации

3.7.1 Биллинг X.25

Этот тип биллинга предназначен для сбора учетной информации по виртуальным соединениям следующих объектов системы:

— портов типа X25, PAD;

— Frame Relay и Ethernet - станций типа X25;

— Telnet-станций типа PAD.

У каждого из перечисленных объектов имеется параметр BI: (Billing), указывающий на некоторый способ биллинга. Сбор и периодическая отсылка статистической информации на некоторый сервер будет производится в соответствии с параметрами указанного способа биллинга.

Учетная информация представляет собой символьные строки, которые передаются на некоторый сервер в сети X.25. Формат записи учетной информации приведен в п.3.7.2. Маршрутизатор периодически устанавливает соединение с сервером, передает по этому соединению учетную информацию и разрывает соединение. Сервером может служить принтер, терминал или ПК, подключенные к линии PAD сети X.25. Если маршрутизатор не смог установить соединение с сервером, он повторит попытку соединения через 10 секунд. После шести неудачных попыток маршрутизатор будет устанавливать соединение с запасным сервером, если он определен в параметрах способа биллинга.

Параметры, установленные для конкретного способа биллинга (см.2.2.9.2), определяют:

- количество и адреса серверов, куда производится отсылка информации

(параметры SN:, SADR: (и возможно SADR1, SADR2 ...));

- особенности пакета вызов (Call) при установки соединения с сервером (собственный адрес (параметр AD:), поле данных в пакете вызова (параметр CUD:))
- условие отсылки информации (параметры TO: и RN:);
- способ подтверждения информации (параметр DCM:);
- особенности подсчета объема информации (параметр SS: и SC:)

Если значение параметра BI:0 или значение указывает на несконфигурированный способ биллинга, например,

PO:<n> TY:X25 .... BI:<m>

BI:<m> TY:NO_BILL

то сбор и отсылка информации для этого объекта производится не будет.

3.7.2 Формат записи учетной информации

Запись учетной информации состоит из полей, разделенных запятыми. Каждая запись заканчивается символами <CR><LF>.

Формат записи представлен ниже:

Параметр	**Формат**	**Описание**
Node Name	ASCII string	Имя узла, с которого данная запись послана;
Sequence	Decimal	Последовательный номер записи;
Account Name	ASCII string	Пароль или NUI;
Connect Date	dd-mmm-yyyy	Дата установки соединения;
Connect Time	hh:mm:ss	Время установки соединения;
Disonnect Date	dd-mmm-yyyy	Дата разрыва соединения;
Disonnect Time	hh:mm:ss	Время разрыва соединения;
Source	ASCII string	Канал, с которого пришел пакет вызова (Call) (см. замечание 1);
Destination	ASCII string	Канал, на который направлен вызов (см. замечание 1);
Called Address	X.121 address (before)	Вызываемый адрес до трансляции адресов;
Called Address	X.121 address (after)	Вызываемый адрес после трансляции адресов;
Calling Address	X.121 address	Вызывающий адрес;
Packets Received	Decimal	Число принятых пакетов;
Packets Transmitted	Decimal	Число переданных пакетов;
Segments Received	Decimal	Число принятых сегментов;
Segments Transmitted	Decimal	Число переданных сегментов;

Characters Received	Decimal	Число принятых байтов (см. замечание 2);
Characters Transmitted	Decimal	Число переданных байтов (см. замечание 2);
Clear Cause Code	Hex	Код причины разрыва соединения;
Clear Diagnostic Code	Hex	Код диагностики разрыва соединения;
Facilities Used	Hex	Услуги, используемые в пакете вызова (Call): бит - значение 0 - Reverse Charging 1 - Fast Select 2 - Closed User Group 3 - Packet Negotiation 4 - Window Negatiation 5 - NUI;
General Information	Hex	Общая информация: бит - значение 0 - Соединение разорвано до подтверждения вызова 1 - не используется 2 - Рассоединение инициировано вызывающей стороной 3 - Рассоединение инициировано вызываемой стороной.

Замечание 1. Номер порта и номер логического канала указывается в символьном виде PO:n (CH:m).

Замечание 2. Если размер сегмента (параметр SS) не равен нулю и параметр SC равен «T», то общее число переданных (принятых) байтов вычисляется по формуле:

<общее число байтов>=<число сегментов>*<размер сегмента>+<число байтов>

Пример записи учетной информации:

```
NSG_Moscow,85634,NONE,25-OCT-1999,17:14:33,27-OCT-1999,03:57:02,
PO:3(CH:57),PO:5(CH:1),25067321,25067321,7701,3745,20,31,0,2345,6735,
00,00,02,04
```

3.8 Служба XOX (X.25 Over X.25)

3.8.1 Организация виртуальных линий в публичной сети X.25

3.8.1.1 Назначение XOX

Служба XOX предназначена для организации виртуальных линий между устройствами NSG через (публичную) сеть X.25. Суть XOX заключается в инкапсуляции пакетного уровня протокола X.25 в одно виртуальное соединение X.25 устанавливаемое между двумя устройствами NSG.

Возможные применения службы XOX:

— Оконечное устройство (банкомат) устанавливает логическое соединение (SVC) с хостом через публичную сеть и держит его в течение длительного времени, а обмен данными идет редко. XOX позволяет разрывать соединение в публичной сети при отсутствии данных и устанавливать его при возобновлении обмена между хостом и оконечным устройством (банкоматом). При этом ни хост ни оконечное устройство (банкомат) не видят этих разрывов. Таким образом XOX позволит снизить плату за время использования SVC в публичной сети.

— Через линию X.25 необходимо одновременно устанавливать SVC, число которых превышает максимальное число логических каналов для данной линии. XOX позволяет транспортировать любое число SVC через один логический канал установленный через (публичную) сеть X.25.

— Имеется оконечное оборудование с синхронным каналом X.25 для подключения к хосту через сеть, а со стороны сети может быть предоставлен только асинхронный канал X.28 (PAD). XOX позволяет транспортировать пакеты X.25 через асинхронный канал X.28 без потери какой либо информации. (см. также AntiPAD п. 3.5).

— Совокупность вышеперечисленных возможностей. Например, необходимо подключить два банкомата через RadioPAD (устройство для доступа к радиосети X.25), который имеет один асинхронный интерфейс X.28, при этом надо уменьшить затраты за время использования радиоканала. XOX позволяет сделать это.

3.8.1.2 Организация XOX в NSG устройствах

Служба XOX реализуется в NSG устройствах объектом который имеет аббревиатуру HX (Hidden X.25) и называется HX сервер.
В HX сервере может быть определено несколько HX станций, которые имеют обозначения HX:1 HX:2 HX:3 и т.д. Обозначение HX:0 используется для определения параметров, общих для всего HX сервера.

Каждая HX станция устанавливает логическое соединение с одной и только одной HX станцией в другом узле. Это логическое соединение будем называть виртуальной линией. Каждая виртуальная линия может содержать в себе по несколько обычных логических соединений.

Для установки параметров HX станции используется команда

S P HX:n <par>:<val> [<par>:<val> ...]

Для просмотра параметров HX станции используется команда

D P HX:n

Для рестарта HX станции используется команда

W S HX:n

где **n** - номер HX сервера;
<par> - имя параметра;
<val> - значение параметра.

Команда W S HX:0 рестартует весь HX сервер.

3.8.1.3 Параметры HX сервера

Параметры HX сервера задаются командой

S P HX:0 <par>:<val>

Параметр NUM (Number of HX servers)

Параметр определяет число HX станций и, соответственно, число виртуальных линий, которые могут быть установлены.

Синтаксис: **NUM:n**
где **n** - число HX станций.

Параметр LC (Number of Logical Channels)

Параметр определяет максимальное число логических каналов для всего HX сервера, т.е. суммарное число SVC по всем виртуальным линиям не может превысить это значение.

Синтаксис: **LC:n**
где **n** - число логических каналов.

3.8.1.4 Параметры HX станции

Параметр ADM (Administrative Status)

Параметр ADM разрешает или запрещает работу данной HX станции.

Синтаксис:
ADM:UP - разрешает работу HX станции
ADM:DOWN - запрещает работу HX станции

Параметр LADR (Local HX station Address)

Параметр LADR определяет X.121 адрес данной HX станции. Этот адрес будет подставляться в качестве вызывающего адреса(calling address) при установке

соединения с удаленной HX станцией.

Синтаксис: **LADR:<addr>**
где **<addr>** - X.121 адрес.
Пример: **LADR:250199999999909**

Параметр RADR (Remote HX station Address)

Параметр RADR определяет X.121 адреса удаленной HX станции.

Синтаксис: **RADR:<addr1>[:<addr2>]**
где **<addr1>** - X.121 адрес, который будет подставляться в качестве вызываемого адреса(called address) при установке соединения с удаленной HX станцией.
<addr2> - X.121 адрес удаленной HX станции. Это значение сравнивается с вызывающим адресом(calling address) во входящем пакете вызова (Incoming call), полученным от удаленной HX станции. Если эти адреса не совпадают, то входящий вызов будет отвергнут и соединение между HX станциями не установится.

Если значение <addr2> не определено, то оно приравнивается к значению <addr1>.

Пример: **LADR:250199999999901**
LADR:250199999999901:250199999999

Параметр IT (Idle Timer)

Параметр IT определяет время неактивности на виртуальной линии.
Если в течение этого периода времени не было принято или передано никаких данных то виртуальная линия между HX станциями будет разорвана(без разрыва логических соединений установленных через эту виртуальную линию).
Если IT=0, то виртуальная линия не будет разрываться по неактивности.
См. также замечание в описании параметра AT.

Синтаксис: **IT:n**
где **n** - время неактивности в секундах.

Параметр AT (Activity Timer)

Параметр AT определяет время активности виртуальной линии.
После установки соединения (виртуальной линии) с удаленной HX станцией, через период времени AT оно будет разорвано в любом случае(без разрыва логических соединений установленных через эту виртуальную линию), даже если есть данные для передачи.
Если AT=0, то виртуальная линия не будет разрываться.

Замечание: Реально разрыв соединения будет происходить через время несколько большее чем значение AT, т.к. после срабатывания таймера в удаленную HX станцию должен быть послан служебный пакет и получено подтверждение. Эта процедура необходима, чтобы избежать потерь данных в виртуальной линии. Это замечание касается также параметра IT.

Синтаксис: **AT:n**
где **n** - время активности в секундах.

Параметр CT (Call Delay Time)

Параметр CT определяет минимальное время которое должно пройти между разрывом виртуальной линии и последующей установкой соединения с удаленной HX станцией.

Синтаксис: **CT:n**
где **n** - время задержки в секундах.

Параметр HT (Holddown Timer)

Параметр AT определяет время, по истечении которого будет предпринята повторная попытка установки соединения с удаленной HX станцией, если предыдущая попытка не была успешной.

Синтаксис: **HT:n**
где **n** - время задержки в секундах.

Параметр RT (Retries)

Параметр RT определяет число попыток установки соединения с удаленной HX станцией. Если за это число попыток установить соединение не удается, то разрываются все логические соединения ранее установленные по данной виртуальной линии.

Синтаксис: **RT:n**
где **n** - число попыток.

Параметр PI (Packet Integrity Method)

Параметр PI определяет способ, которым достигается целостность пакетов, которые передаются по виртуальной линии между двумя HX станциями.

Синтаксис:
PI:MBIT - целостность пакетов обеспечивается применением M-bit процедуры. Этот способ применим, если логическое соединение реализующее виртуальную линию между двумя HX станциями является "чистым" X.25 соединением.
PI:SLIP - целостность пакетов обеспечивается применением протокола SLIP. Этот способ необходим, если логическое соединение реализующее

виртуальную линию между двумя HX станциями проходит через асинхронный канал X.28 (PAD), который не гарантирует сборку пакетов в их первоначальном виде. (см. также AntiPAD п. 3.5). Способ SLIP может использоваться и для "чистых" X.25 соединений, но он более медленный.

Параметр LG (Packet Length)

Параметр LG определяет максимальную длину в байтах поля данных пакетного(третьего) уровня протокола X.25, инкапсулированного в виртуальную линию между двумя HX станциями.

Синтаксис: **LG:n**
где **n** - длина пакета (16, 32, 64 ,128, 256, 512, 1024).

Параметр PW (Packet Window)

Параметр PW определяет величину окна для пакетного (третьего) уровня протокола X.25, инкапсулированного в виртуальную линию между двумя HX станциями.

Синтаксис: **PW:n**
где **n** - величина окна (1...7).

3.8.1.5 Пример настройки XOX службы

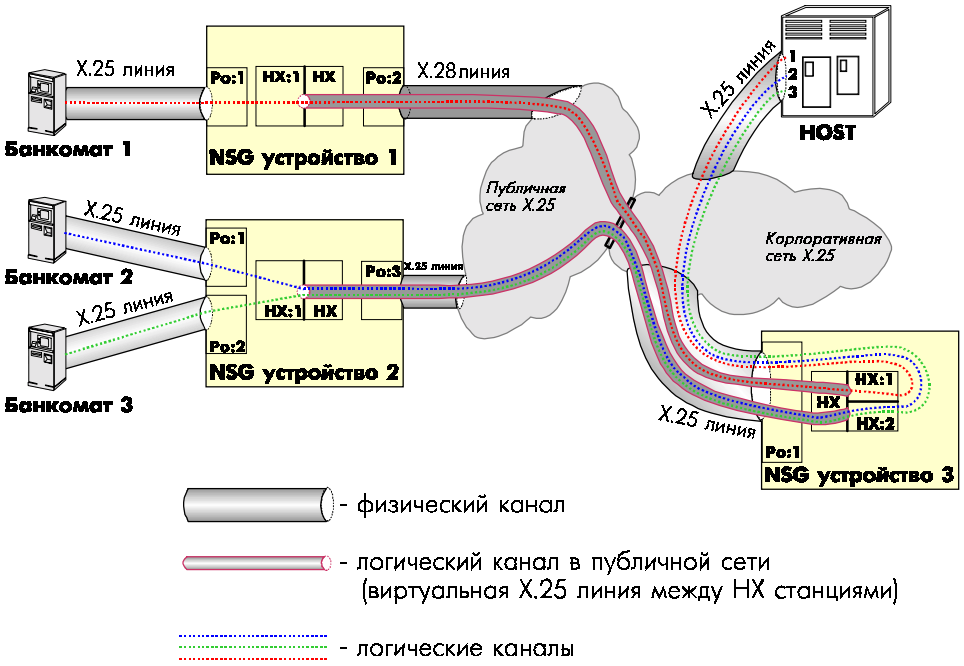

Рис 3.3 Настройка XOX службы

Дано (см. рис. 3.3):

— три банкомата с синхронными X.25 каналами, которые устанавливают логическое соединение с хостом и держат его в течение всего рабочего времени;
— к банкомату 1 подведена асинхронная X.28 линия по которой ограничено время жизни логического соединения;
— к банкоматам 2 и 3 подведена синхронная линия X.25;
— в публичной сети X.25 высокие тарифы на время соединения.

Необходимо: подключить банкоматы с минимальными затратами за время использования каналов.

Подключаем 3 NSG устройства по схеме, показанной на рисунке X.
Пусть имеем следующие адреса:
в публичной сети
адрес NSG-1 - 250199999901
адрес NSG-2 - 250199999902
адрес NSG-3 - 250199999903

в корпоративной сети
адрес хоста - 777707
адрес банкомата 1 - 777701
адрес банкомата 2 - 777702
адрес банкомата 3 - 777703

Настраиваем устройства следующим образом:

УСТРОЙСТВО NSG-1

```
S P PO:1 TY:X25 IF:V24 TE:DCE LC:8 SP:9600 FW:7 N2:9 LG:128 PW:2 T1:3 T2:180
AP:NO BI:0
S P PO:2 TY:PAD IF:ANTI SP:2400 AF:8N1 LG:128 MB:YES BI:0 AD:2222 4:1 DT:0
UD:NO PREF:"" CACF:"COM\r\n" CALL:"COM\r\n"

S P HX:0  NUM:1 LC:1
S P HX:1  ADM:UP LADR:250199999901 RADR:250199999903 IT:20 AT:110 CT:5
HT:5 RT:3 PI:SLIP LG:128 PW:2

S R PR:1 ID:D RT:2222          PO:HX
S R PR:2 ID:D RT:250199999903 PO:2
S R PR:3 ID:D RT:777707        PO:HX,1
S R PR:4 ID:D RT:777701        PO:1
```

УСТРОЙСТВО NSG-2

```
S P PO:1 TY:X25 IF:V24 TE:DCE LC:8 SP:9600 FW:7 N2:9 LG:128 PW:2 T1:3 T2:180
AP:NO BI:0
S P PO:2 TY:X25 IF:V24 TE:DCE LC:8 SP:9600 FW:7 N2:9 LG:128 PW:2 T1:3 T2:180
AP:NO BI:0
S P PO:3 TY:X25 IF:V24 TE:DTE LC:4 SP:EXT  FW:7 N2:9 LG:128 PW:2 T1:3 T2:180
AP:NO BI:0


S P HX:0  NUM:1 LC:2
S P HX:1  ADM:UP LADR:250199999902 RADR:250199999903 IT:30 AT:0 CT:2 HT:2
RT:3 PI:MBIT LG:128 PW:2

S R PR:1 ID:D RT:250199999902 PO:HX
S R PR:2 ID:D RT:250199999903 PO:3
S R PR:3 ID:D RT:777707       PO:HX,1
S R PR:4 ID:D RT:777702       PO:1
S R PR:4 ID:D RT:777703       PO:2
```

УСТРОЙСТВО NSG-3

```
S P PO:1 TY:X25 IF:V24 TE:DTE LC:4 SP:EXT  FW:7 N2:9 LG:128 PW:2 T1:3 T2:180
AP:NO BI:0


S P HX:0  NUM:2 LC:3
S P HX:1  ADM:UP LADR:250199999903 RADR:250199999901 IT:0 AT:0 CT:7 HT:7
RT:3 PI:SLIP LG:128 PW:2
S P HX:2  ADM:UP LADR:250199999903 RADR:250199999902 IT:0 AT:0 CT:3 HT:3
RT:3 PI:MBIT LG:128 PW:2

S R PR:1 ID:D RT:250199999903 PO:HX
S R PR:2 ID:D RT:250199999901 PO:1
S R PR:2 ID:D RT:250199999902 PO:1
S R PR:3 ID:D RT:777701       PO:HX,1
S R PR:4 ID:D RT:777702       PO:HX,2
S R PR:4 ID:D RT:777703       PO:HX,2
S R PR:4 ID:D RT:777707       PO:1
```

Примечания:

1. Обратите особое внимание: вызовы для установки логических соединений (виртуальных каналов) между HX станциями должны адресоваться на порт HX (без номера!) т.е. к HX серверу, а для установки

логических соединений между оконечными устройствами (банкоматами и хостом) должны адресоваться на порт HX,n (где n - номер соответствующей HX станции).

3.9 Логические порты

Логические порты устройства представляют собой программные процессы, функционирующие в рамках программного обеспечения устройства. Работа с логическими портами основана на принципах сетей пакетной коммутации X.25. То есть динамически или постоянно создается логический виртуальный канал, после чего, вплоть до разрыва соединения, происходит обмен данными.

Часть логических портов является подсистемами различных служб (IP, HX и пр.), другая часть (приведенные в данной главе) реализуют некоторые простейшие функции тестирования.

3.9.1 Трафик генератор

ТРАФИК-ГЕНЕРАТОР предназначен для проверки работоспособности программного обеспечения, а также для создания реальной нагрузки на физическую линию при тестировании канала.

ТРАФИК-ГЕНЕРАТОР представляет собой модуль программного обеспечения, который работает как обычный порт (**PO:TG**), только без подключения к реальной физической линии.

При появлении пакета "входящий вызов" (Incomming Call) на порт "TG" ТРАФИК-ГЕНЕРАТОР подтверждает установку соединения (Call Confirmation) и по установленному логическому каналу начинает непрерывно посылать пакеты данных.

Пакет данных содержит строку из 80 байт:

xxxxxx THE QUICK BROWN FOX JUMPS OVER THE LAZY DOG. 1234567890

где **xxxxxx** — номер посланного пакета.

Посылка пакетов происходит вплоть до разрыва логического соединения со стороны абонента.

Примечание: При реальной работе в сети не рекомендуется обращение к ТРАФИК-ГЕНЕРАТОРУ, так как это может привести к снижению пропускной способности.

3.9.2 Эхо-порт

ЭХО-ПОРТ предназначен для проверки работоспособности. Он представляет собой модуль программного обеспечения, который работает как обычный порт (**PO:EH**) только без подключения к реальной физической линии.

При появлении пакета "входящий вызов" (Incoming Call) на порт "EH", эхо-порт подтверждает установку соединения (Call Confirmation) и далее возвращает все пакеты данных, приходящие к нему по установленному логическому соединению.

3.10 Служба SNMP

SNMP-агент реализован в соответствии с рекомендацией RFC-1155, RFC-1157 (SNMP v.1).

Обмен данными между управляющей станцией и SNMP-агентом происходит на основе датаграммных посылок, используя протокол UDP (порты 161 и 162).

При получение запроса от управляющей станции (Get_Request, Set_Request) SNMP-агент проверяет:

- соответствие имени Community (в запросе) одной из сконфигурированных Community (в устройстве);
- принадлежность адреса управляющей станции диапазону адресов найденной Community;
- права доступа (при операции Set_Request).

После успешно пройденных проверок, SNMP-агент выполняет запрос и отсылает ответ (Get_Response) управляющей станции. Если соответствия не найдено, то фиксируется ситуация нарушение прав доступа, запрос игнорируется и (если сконфигурировано) отсылается TRAP-сообщение ("Authentication Failure").

База управляемых переменных реализована соответствии с рекомендацией RFC-1213 (MIB-II).

Поддерживаются следующие группы управляемых переменных:

- system (1);
- interfaces (2);
- ip (4);
- icmp (5);
- tcp (6);
- udp (7);
- transmission (10); (частично)
- snmp (11).

> ***Примечание:*** В группе transmission реализованы группы x25 (5), ethernet (7), lapb (16), ds1 (18), e1 (19), frame relay (32). В дальнейшем список поддерживаемых групп может изменяться.

3.10.1 Рассылка TRAP-сообщений

Рассылка Trap-сообщений осуществляется тем управляющим станциям, Community которых имеют:

- параметр MASK: 255.255.255.255 (т.е. параметр iadr определяет конкретный, а не групповой адрес);
- параметр TP: отличен от значения «NO».

При значении параметра TP:GE для управляющей станции будут посылаться следующие Trap-сообщения:

— **ColdStart** - в системе произошла переинициализация одного из физических портов перед которой была произведена запись конфигурации в энергонезависимую (Flash) память. (Т.е. возможно было произведено изменение конфигурации системы). Это сообщение посылается также при запуске/сбросе устройства или переинициализации всей IP-подсистемы (W S IP:0).

— **WarmStart** - в системе произошла переинициализация одного из физических портов без изменения конфигурации системы, записанного во Flash.

— **LinkUp/LinkDown** - передается в случае перехода одного из физических интерфейсов в состояние UP или DOWN соответственно. Номер интерфейса, у которого произошло изменение состояния протокола, передается в TRAP-сообщении параметром IfIndex.

Примечание: Номер физического интерфейса на единицу больше номера порта, к которому он относится. (Например: если получен TRAP LinkUp с IfIndex = 3, то это означает порт номер 2 устройства перешел в состояние UP).

При значении параметра TP:AU для управляющей станции будут посылаться следующие Trap-сообщения:

— **AuthenticationFailure** - передается в случае обнаружения SNMP-агентом несанкционированного обращения. Несанкционированным считается обращение, при котором в SNMP-запросе (GetRequest, GetNextRequest или SetRequest) обнаружено хотя бы одно нарушение:

- поле «Community name» отличается от всех, объявленных в устройстве, Community (параметр «name» для CO: 0..n)
- IP-адрес (Source Address) в запросе отличается от всех, объявленных в устройстве адресов управляющих станций (параметр «iadr» для CO:0..n, с учетом маски «mask»)
- IP-адрес и «Community name», указанные в запросе, не относятся к одной и той же записи CO.

— **EnterpriseSpecific** - TRAP-сообщения, которые сгенерированы SNMP-агентом при возникновении в системе некоторых существенных ситуаций:

- Trap = 0 (PRIVILEGE VIOLATION FOR SET REQUEST) - произошел запрос на изменение параметра системы от управляющей станции, для которой установлены привилегии Read Only (WR:NO);
- Trap = 1 (CONNECT TO MANAGER) - в системе произошло подключение пользователя к управляющему модулю «Manager»;
- Trap = 2 (DISCONNECT FROM MANAGER) - в системе произошло отключение пользователя от управляющего модуля «Manager»;

— Trap = 3 (REWRITE CONFIGURATION) - пользователь выполнил команду записи конфигурации «W F»;
— Trap = 4 (CONNECT TO TELNET STATION) - произошло подключение к Telnet-станции. Номер станции передается в Trap-сообщении;
— Trap = 5 (DISCONNECT FROM TELNET STATION) - произошло отключение от Telnet-станции. Номер станции передается в Trap-сообщении.

Примечание: список TRAP-сообщений типа EnterpriseSpecific может быть расширен.

3.11 WEB - управление. Служба HTTP

3.11.1 Назначение

HTTP-сервер предназначен для реализации функций удаленного управления (WEB-управление), используя для этой цели WEB-браузер.

WEB-управление предлагает практически тот же набор функций по управлению устройством, что и удаленная работа с консолью устройства (например используя Telnet). Сюда относятся задачи по изменению и контролю конфигурации устройства, получение информации о текущем статусе и статистике компонентов устройства, рестарт отдельных подсистем и служб и другие функции.

Управляемое устройство должно иметь активированный TCP/IP стек и как минимум один IP-интерфейс, через который будет осуществляться связь с устройством.

3.11.1.1 Ограничения на использование WEB-браузеров

Из-за отсутствия полной совместимости WEB-браузеров различных фирм круг поддерживаемых браузеров жестко ограничен. Поддерживается два наиболее популярных WEB-браузера:

— Internet Explorer версии 4.0 и выше;
— Netscape Navigator версии 4.0 и выше.

Осуществление WEB-управления из других браузеров не гарантируется.

Минимальное разрешение экрана - 800 x 600 точек.
Оптимальное разрешение экрана - 1024 x 768 точек.

3.11.1.2 Ресурсы, потребляемые HTTP-сервером

В процессе работы, как при запуске HTTP-сервера так и при выполнении очередного запроса управления, используется динамическая память системы.

При запуске HTTP-сервер занимает:

- область Heap - приблизительно 160 - 170 кБайт;
- область Stack - приблизительно 4 кБайт.

Если в момент запуска HTTP-сервера динамической памяти недостаточно, то (при включенном выводе DEBUG) будет выведено сообщение:

```
HTTP:  Unsufficient Memory (xxxx)
```

где xxxx - укажет какой именно памяти недостаточно.

При выполнении запроса управления HTTP-сервер дополнительно занимает:

- область Heap - приблизительно 5 кБайт;
- область Stack - приблизительно 20 кБайт.

Если в момент выполнения запроса динамической памяти недостаточно, то запрос не будет выполнен , а пользователю будет выведено в окно браузера сообщение:

```
500 Server Error
```

По мере освобождения ресурсов, HTTP-сервер снова будет отвечать на запросы пользователя.

3.11.2 Запуск и останов HTTP-сервера

Для запуска HTTP-сервера требуется установить параметр HTTP интерфейса IP:0 значение "YES".

```
S P IP:0 HTTP:YES
```

И рестартовать HTTP-сервер:

```
W S HTTP
```

Примечание: На устройства NSG с невысоким быстродействием (NPS-7, NSG-5xx) выполнение этой команды потребует 3-4 секунды.

Для останова HTTP-сервера требуется установить параметр HTTP интерфейса IP:0 значение "NO".

```
S P IP:0 HTTP:NO
```

И рестартовать HTTP-сервер:

```
W S HTTP
```

При останове, динамическая память (см.п.1.2) занимаемая HTTP-сервером, освобождается.

> ***Примечание:*** При включение (рестарте) устройства, запуск HTTP-сервера определяется значением параметра HTTP интерфейса IP:0, которое было сохранено в энергонезависимой памяти.

3.11.3 Работа с устройством используя WEB-управление

3.11.3.1 Подключение к устройству

На компьютере, с которого можно установить связь с устройством используя TCP/IP-протокол, пользователь запускает какой-либо браузер (см.п.3.11.1.1). Затем, в поле Location (Адрес) пользователь вводит строку:

http://<имя управляемого устройства>/

<имя управляемого устройства> - символьная строка, определяющая (посредством DNS протокола) IP-адрес управляемого устройства. Допустимо вместо имени сразу указывать IP-адрес одного из интерфейсов устройства.

Во время первого обращения к устройству пользователь проходит процедуру аутентификации (см.п.3.11.3.2), определяющую полномочия пользователя по управлению данным устройством.

После успешно пройденной процедуры аутентификации в браузер загружается главная страница, структура которой приведена в п.3.11.3.3.

3.11.3.2 Процедура аутентификации

При первом обращении к устройству пользователь должен аутентифицировать себя, введя свое имя и пароль в прелагаемом диалоговом окне. Имя и пароль (при вводе отображается в виде символов '*') представляют из себя строку символов.

> ***Примечание:*** В данной версии допустимо указывать любое имя пользователя, а в качестве пароля нужно ввести пароль для обращения к модулю Manager.

Если в процессе работы будет изменен пароль модуля Manager (например администратором подключенным к модулю Manager в терминальном режиме), то при очередном обращении из браузера, пользователю вновь придется пройти процедуру аутентификации.

Дальнейшее развитие WEB-управления предполагает обеспечение дополнительного разграничения выполняемых пользователем функций, в зависимости от имени и пароля пользователя, определенных на этапе аутентификации.

3.11.3.3 Структура главной страницы

После успешно пройденной процедуры аутентификации, пользователю будет выведена в окно браузера главная страница, которая будет находится пока пользователь не перейдет к какому-либо другому ресурсу Internet, сменив поле Location (Адрес).

Главная страница разделена на три поля (см.рис.3.4)

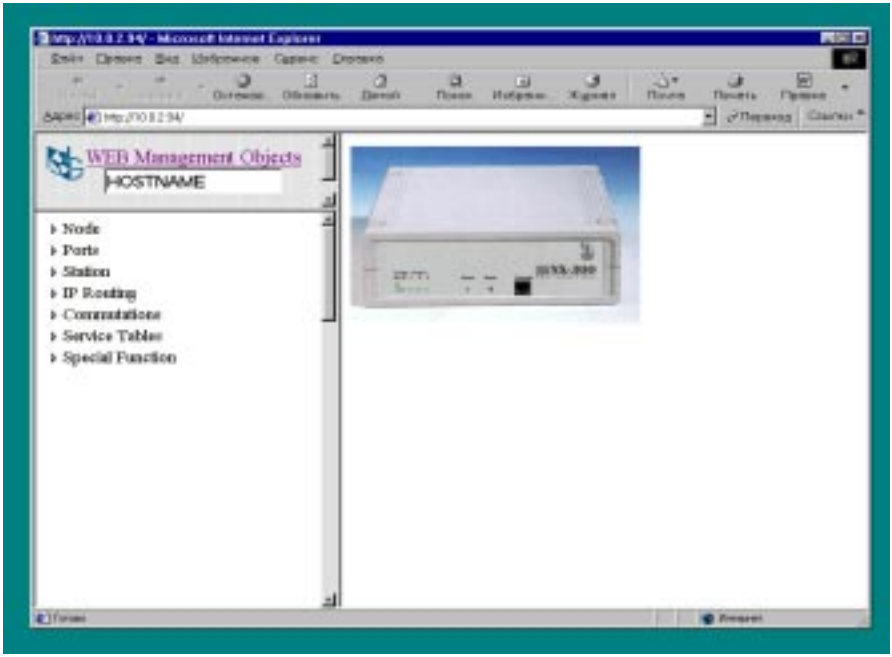

Рис 3.4 Структура главной страницы

Верхнее (левое) поле практически не меняется в процессе работы пользователя.

Поле содержит 3 элемента:

- Логотип компании NSG - картинка, представляющая гиперссылку на WEB-сайт компании;
- строка "WEB Management Objects" - гиперссылка, при нажатии на которую будет обновлен список управляемых объектов (см.п.3.11.3.4).
- Поле, содержащее имя устройства. Имя устройства - строка символов, назначаемая администратором (параметр HNAM). Данное поле бывает полезно при осуществлении WEB-управления несколькими NSG-устройствами. Оно показывает администратору с каким именно устройством он в данный момент работает.

Левое поле отображает управляемые объекты устройства, представленные в виде иерархического дерева. Описание работы пользователя при выборе управляемого объекта приведено в п.3.11.3.4.

Правое поле страницы (основное информационное поле) используется для ввода пользователем значений параметров управляемых объектов, а так же для вывода информации о работе устройства и его подсистем.

При выполнении отдельных операций (установка параметров, рестарт какой-либо компоненты и др..) поверх главной страницы выводится небольшое окно-подтверждение выполненной функции (успешное или неуспешное). Окно-подтверждение носит чисто информативный характер и пользователь уберет это окно щелкнув мышью на кнопке "OK" внутри окна, или же щелкнув мышью на любом другом объекте главной страницы.

3.11.3.4 Структура управляемых объектов

Под управляемым объектом следует понимать либо какой-то параметр функционирования системы (например параметры порта или станции, записи таблицы маршрутизации и.т.п), либо какое-то действие над объектом системы (рестарт порта, рестарт службы, сохранение конфигурации и.т.п).

В качестве управляемого объекта может выступать некоторый скалярный элемент (например параметры модуля Manager), либо одномерный массив однотипных элементов (например физические порты, станции и.т.п).

Все управляемые объекты объединены и представлены в виде иерархической структуры (см. рис. 3.5). Для удобства доступа к управляемым объектам, их совокупность представлена в виде Outline - интерфейса. В каждый момент времени на экране представлена только часть управляемых объектов, которая интересует пользователя.

Элемент структуры управляемых объектов, который является массивом или содержит в себе некоторое количество скалярных управляемых объектов и/или вложенных в него других элементов, находится на экране в сжатом виде либо в разжатом (представленном) виде.

В первом случае элемент структуры представлен только одной строкой, в которой указывается его название и признак '▶' указывающий, что данный элемент сжат.

Во втором случае, после названия элемента, выводится все вложенные в него скалярные элементы, массивы и элементы нижнего уровня (в сжатом виде). Перед названием разжатого элемента стоит признак '▼'. Щелчок левой кнопкой мыши на признаке, переводит элемент структуры в сжатое состояние и обратно.

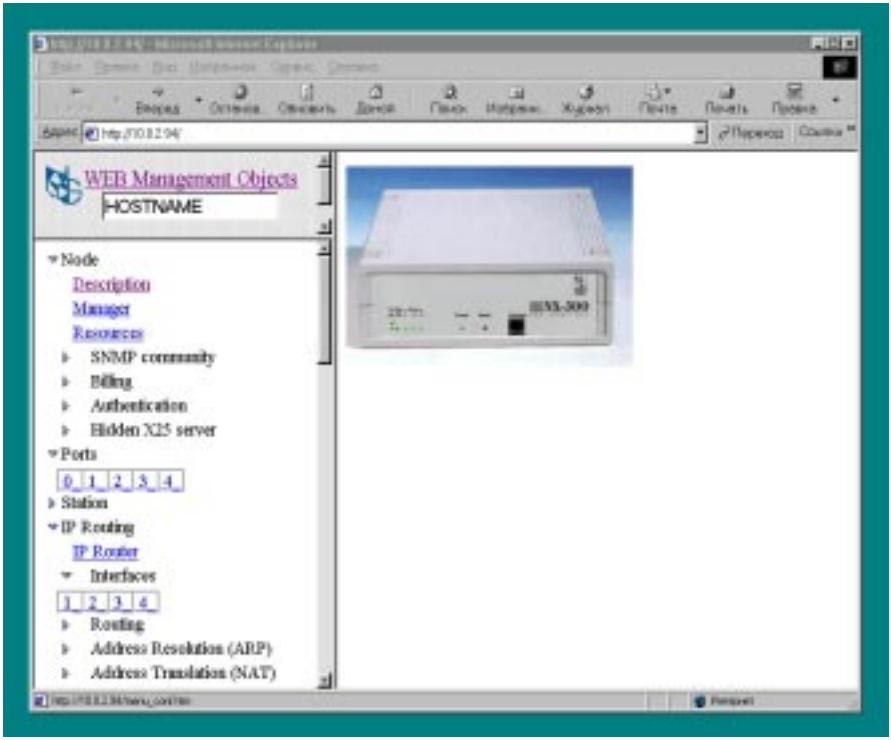

Рис 3.5 Структура управляемых объектов

3.11.4 Работа с управляемым объектом

Работа с управляемым объектом, предполагает выполнение одной (или нескольких) из перечисленных ниже функций:

- установка и контроль значений параметров;
- определение текущего статуса и статистики;
- рестарт управляемого объекта;
- очистка статистики и др.

Пользуясь левой кнопкой мыши укажите на управляемый объект. После этого в правое поле будет выведена страница, содержащая перечень параметров указанного объекта. (см.рис.3.6). Каждый параметр представлен в виде его названия и текущего значения. В поле названия параметра, в скобках, указывается его сокращенное обозначение (или аббревиатура), которое используется для данного параметра в модуле Manager.

Для изменения значения параметра необходимо поместить курсор в область значения и пользуясь клавиатурой и/или мышью ввести новое значение.

Под таблицей параметров располагаются кнопки, нажатие на которые приводит к выполнению перечисленных ранее функций над управляемым объектом.

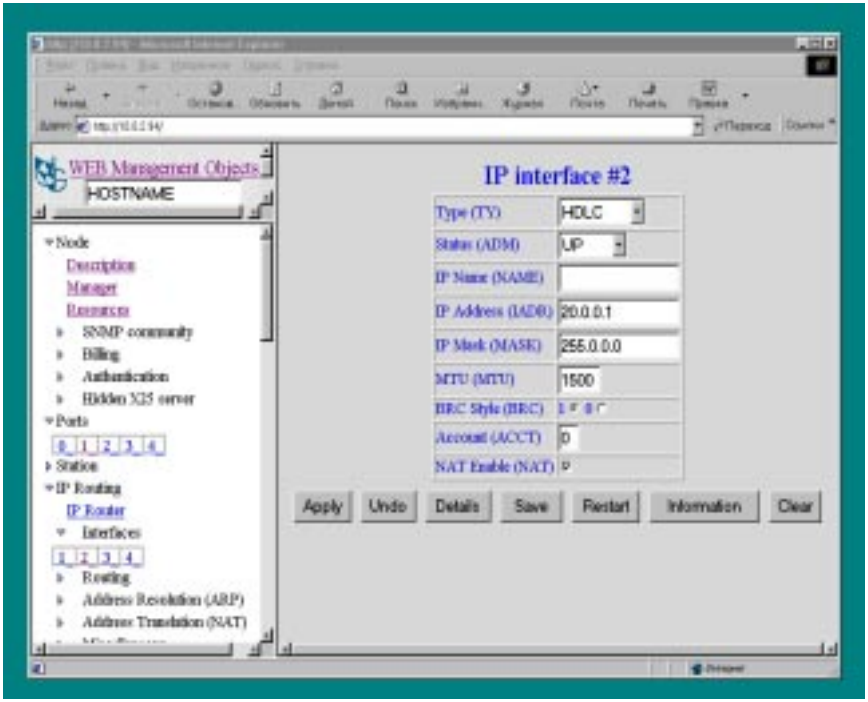

Рис 3.6 Установка параметров порта

Кнопка "**Apply**" - производит собственно отсылку установленных параметров в систему. Нажатие кнопки эквивалентно выполнению команды модуля Manager (Set Parameters) с теми значениями, который в данный момент находятся на экране. До нажатия кнопки "**Apply**" никакие действия по изменению параметров системы не производятся.

Перед отправкой заполненной формы в устройство, производится синтаксический разбор значений, введенных пользователем. Если при этом разборе обнаружена ошибка, то форма не отсылается, а пользователь получает сообщение о неправильном значении параметра.

> ***Примечание:*** Если пользователь поменял значение параметров, но на нажал кнопку 'Apply' , то при переходе к другому управляемому объекту будет выдано предупреждение.

Кнопка "**Undo**" - возвращает значение параметров к значениям на момент загрузки текущей страницы. Кнопкой "**Undo**" нельзя отменить сделанные изменения параметров системы, произведенные после нажатия кнопки '**Apply**'.

Кнопка "**Details**" - приводит к выводу в текущее окно дополнительных параметров управляемого объекта.

Кнопка "**Restart**" - приводит к выполнению рестарта управляемого объекта. Эквивалент выполнения команды "W S ..".

Примечание: В случае выполнения рестарта всего узла, пользователю выводится диалоговое окно-предупреждение. Если пользователь подтвердит рестарт нажатием кнопки "OK", то произойдет рестарт всего устройства в целом. Устройство при этом будет доступно через несколько секунд.

Кнопка "**Information**" - в окно выводится информация о статусе и статистике управляемого объекта. Эквивалент выполнения команды "D S ...".

Примечание: Информация об объекте выводится на момент выполнения запроса и автоматически не обновляется.

Кнопка "**Save**" - используется для сохранения текущей конфигурации в энергонезависимой памяти системы. Эквивалент выполнения команды W F (Write Flash).

Примечание: В системе выполнение команды W F сохраняет все текущие значения параметров.

В некоторых случаях информация хранится в виде таблиц состоящих из записей, например, таблица маршрутизации X.25, таблица статических IP-маршрутов, таблица PAP/CHAP - паролей и др. Информация из устройства считывается и выводится в окно также в виде таблицы, каждая строка которой представляет отдельную запись (см.рис.3.7).

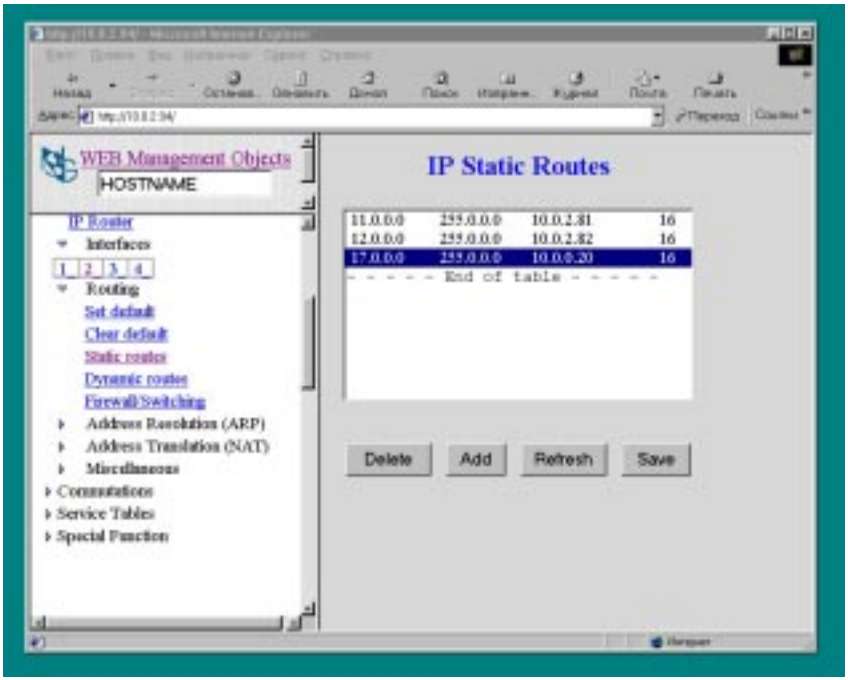

Рис 3.7 Установка статических IP-маршрутов

Над каждым элементом (записью) могут быть произведены приведенные ниже действия:

Кнопка "**Delete**" - уничтожить выделенный элемент.
Кнопка "**Edit**" - редактирование текущего выделенного элемента.
Кнопка **"Add"** - добавление (вставка) нового элемента таблицы перед выделенным (или в конец таблицы).
Кнопка "**Refresh**" - обновляет содержимое таблицы на экране.

Дополнительно может присутствовать кнопка "**Save**", действие которой описано выше.

3.11.5 Влияние на процесс Manager

WEB-управление обеспечивает возможность осуществлять администрирование устройство со стороны нескольких пользователей одновременно. При этом не исключены ситуации, когда совместное изменение конфигурации устройства может привести к нежелательным последствиям.

При работе какого-либо пользователя с модулем Manager допустимо обращение других клиентов, использующих WEB-управление, к данному узлу. При выполнении пользователем команд, требующих длительного выполнения, например, Trace Route (P R), Ping (P P), D S ... (с параметром UP:n) обращение других клиентов прервет выполнение этих команд.