ООО «Компания «АЛС и ТЕК»

SHDSL-16.EFM

РУКОВОДСТВО ПРОГРАММИСТА

ДРНК.405470.023ТО

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

Оглавление

ВВЕДЕНИЕ

Современные концентраторы DSL представляют собой оборудование нового поколения, позволяющее подключать абонентов к сети передачи данных, используя последние технологии, и имеющее сетевые интерфейсы, такие как Ethernet. DSLAM устанавливаются на стороне оператора связи и позволяют абонентам получать высокоскоростной доступ к сетям передачи данных, сохраняя при этом существующую инфраструктуру и доступ к ТфОП.

Требования, которые предъявляет потребитель к разным классам DSL-оборудования, существенно различаются. Имеют значение: надежность, размеры, плотность портов, потребляемая мощность. Использование медной проводки и простая процедура установки концентратора делают первоначальные вложения для создания сети доступа минимальными. Таким образом, использование концентраторов позволяет абонентам получать дополнительные виды услуг, а операторам — дополнительные виды дохода.

Настоящее руководство содержит сведения, необходимые для обеспечения действий программиста при работе с устройством «SHDSL-16.EFM».

1 ОБЩИЕ СВЕДЕНИЯ О СИСТЕМЕ

SHDSL-16.EFM - это мультиплексор SHDSL доступа, устанавливаемый на стороне поставщика услуг широкополосного доступа в сеть и обеспечивающий подключение абонентского оборудования по технологии SHDSL. К сети провайдера услуг DSLAM подключается через интерфейс Ethernet. Используя технологии SHDSL, этот IP DSLAM предоставляет провайдерам услуг экономичное решение для предложения пользователям различных сервисов с помощью таких функций, как управление полосой пропускания, приоритезация трафика и управление безопасностью потока данных.

2 ФУНКЦИОНАЛЬНОЕ НАЗНАЧЕНИЕ

Мультиплексор абонентского доступа SHDSL-16.EFM предоставляет возможность поставщику услуг широкополосного доступа в сеть подключать абонентов по медному кабелю с использованием существующих телефонных линий связи.

Устройство имеет 16 SHDSL-портов, каждый из которых обеспечивает скорость до 11.2Мбит/с (в зависимости от качества линии передачи) и два порта Ethernet обеспечивающих доступ к сети провайдера по медному кабелю (10/100Base-TX). Система управления устройства имеет несколько интерфейсов настройки, обеспечивающих полный контроль над функционированием устройства: текстовый командный интерфейс (CLI), доступный через порт RS-232 и по протоколу SSH, и специализированное ПО.

Отличительной особенностью устройства является полная совместимость с уже имеющимся оборудованием (абонентским комплектом АК-32). Мультиплексор предоставляет доступ абонентов не только к сети Ethernet, но и полнофункциональный доступ до сети ТФоП, совместно с устройством уплотнения абонентских каналов АЛС-АУ.

Полноценное функционирование комплекса обеспечивается наличием дополнительного оборудования: платы управления дистанционным питанием ПВДП, источником дистанционного питания ИДП-240/1,2, и устройством уплотнения абонентских каналов АЛС-АУ.

Помещение, в котором устанавливается SHDSL-16.EFM должно быть чистым и хорошо вентилируемым. Для работы устройства необходим блок БУН-21/6, который устанавливается в стандартную 19” стойку . Устройство работает от источника питания с напряжением 36 - 72 В.

3 ОПИСАНИЕ ФИЗИЧЕСКОЙ ЧАСТИ СИСТЕМЫ

Рисунок 2: Вид платыSHDSL-16.EFM

Рисунок 1: Вид передней панели SHDSL-16.EFM

Внешний вид SHDSL-16.EFM и изображение его лицевой панели приведены ниже:

На лицевой панели платы SHDSL-16.EFM располагаются следующие элементы управления:

- тумблер питания (положение вверх – питание включено, положение вниз – питание выключено);
- COM-порт для управления;
- 2 Uplink-порта Fast Ethernet для подключения сетевых интерфейсов;
- 2 порта USB;
- 2 порта Fast Ethernet локального управления;

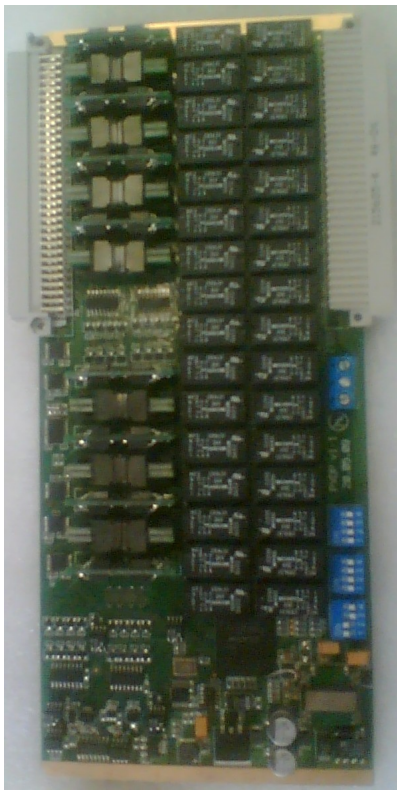

Рисунок 3: плата ПВДП

Модуль ввода дистанционного питания (ПВДП) предназначен для фильтрации и коммутации дистанционного питания, необходимого для запитки DSL линий платы SHDSL-16.EFM от источника дистанционного питания ИДП 240/1,2, и контроля параметров запитки каждой DSL линии (ток утечки, короткое замыкания, защитное отключение питания линии). Также модуль предоставляет возможность измерителю ИПАЛ, измерить параметры каждой DSL линии.

Модуль ПВДП может устанавливаться как непосредственно в кросс БУН-21 рядом с платой SHDSL-16.EFM, так и с обратной стороны в 96 контактный разъем кросс платы БУН-21 сзади платы SHDSL-16.EFM (рис 6, рис 7) Дистанционное питание от блока ИДП-240 подается на специальный разъем подачи ДП(рис 3 снизу от внешнего 96 контактного разъема). «Плюс» ДП подключается к верхнему контакту клемника, «минус» ДП к нижнему, средний контакт клемника остается незадействованным. Назначение контактов 96-контактного разъема совпадает с платой SHDSL-16.EFM и приведены в приложении 4.2.

4 КОНФИГУРИРОВАНИЕ СИСТЕМЫ

4.1. Подключение к устройству по протоколу Ethernet

Доступ по Ethernet необходим для конфигурирования устройства с помощью различных имеющихся интерфейсов. Для обеспечения их работоспособности, необходимо произвести действия, описанные ниже.

4.1.1. Настройка компьютера программиста

Для подключения к блоку при помощи протокола Ethernet необходимо, чтобы у ПК программиста был физический доступ до устройства через сеть Ethernet и правильно сделаны сетевые настройки операционной системы.

Для того, чтобы правильно настроить операционную систему на компьютере программиста, достаточно знать IP-адрес устройства. IP-адрес может быть различным, в зависимости от конфигурации устройства. Если заводская конфигурация не была изменена, то устройство будет иметь IP адрес 192.168.0.180.

После определения IP-адреса устройства необходимо проверить настройки сети на ПК, с которого будет осуществляться конфигурирование. Следует помнить, что связь между рабочей станцией и SHDSL-16.EFM может быть установлена только в том случае, когда они имеют IP-адреса из одной подсети.

К примеру: если на устройстве используется заводская конфигурация, то сетевой карте ПК может быть присвоен любой адрес, начиная с 192.168.0.1 и заканчивая 192.168.0.254, за исключением адреса самого SHDSL-16.EFM 192.168.0.180. Пример настройки сетевой карты в ОС Windows показан на рисунке ниже:

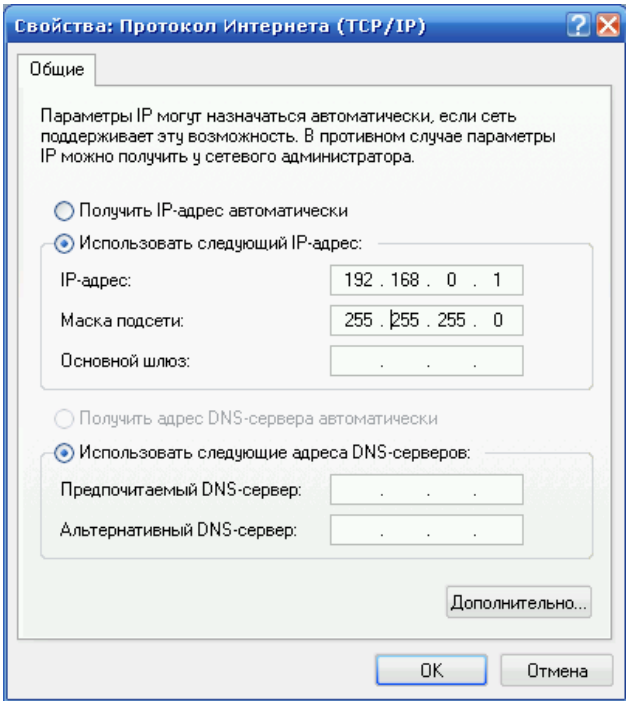

Рисунок 4: Установка IP-адреса для ПК

Проверить настройки IP-протокола и доступность устройства можно с помощью команды `ping`. Для этого нужно выполнить следующие действия (для ОС Windows и блока с загруженной заводской конфигурацией):

1. Выберите из меню «Пуск»: *Программы* → *Стандартные (Accessories)* → *Командная строка*.
2. В открывшемся окне введите команду `ping 192.168.0.180` и нажмите клавишу Enter.
3. Если на экране появилась надпись «Превышен интервал ожидания для запроса», то это означает, что SHDSL-16.EFM недоступен. В этом случае необходимо проверить настройки IP-протокола на ПК и подключения ПК к данному устройству.
4. В случаю появления ответов от SHDSL-16.EFM тестирование настроек IP и доступности блока можно считать успешным.

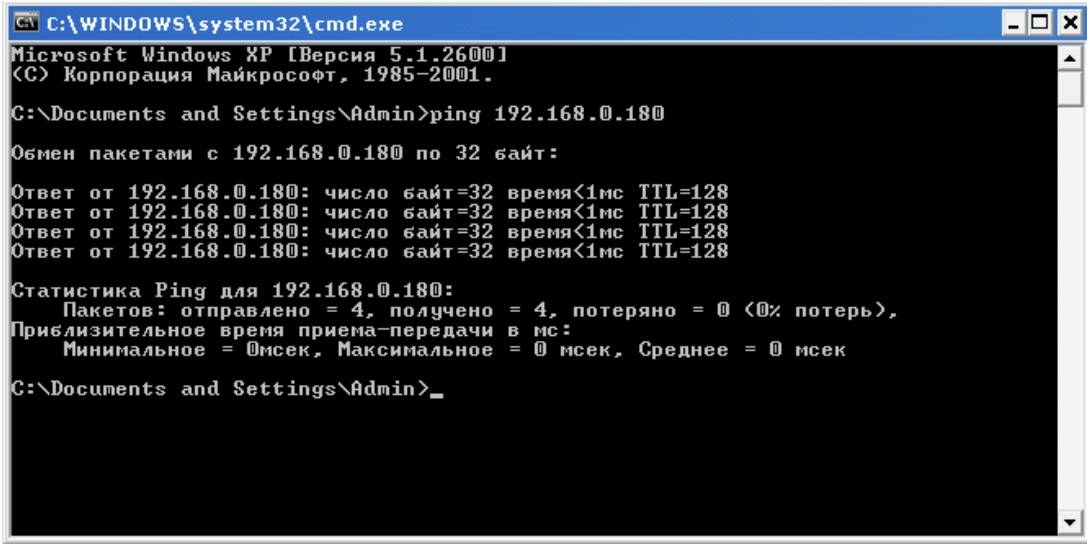

Рисунок 5: Использование команды ping

4.1.2. Настройка SSH клиента

Наличие SSH клиента у программиста необходимо, если используется конфигурирование через интерфейс командной строки CLI. Ниже представлен пример настройки SSH клиента Putty под ОС Windows. Подразумевается, что сетевые настройки компьютера оператора сделаны правильно согласно предыдущему пункту.

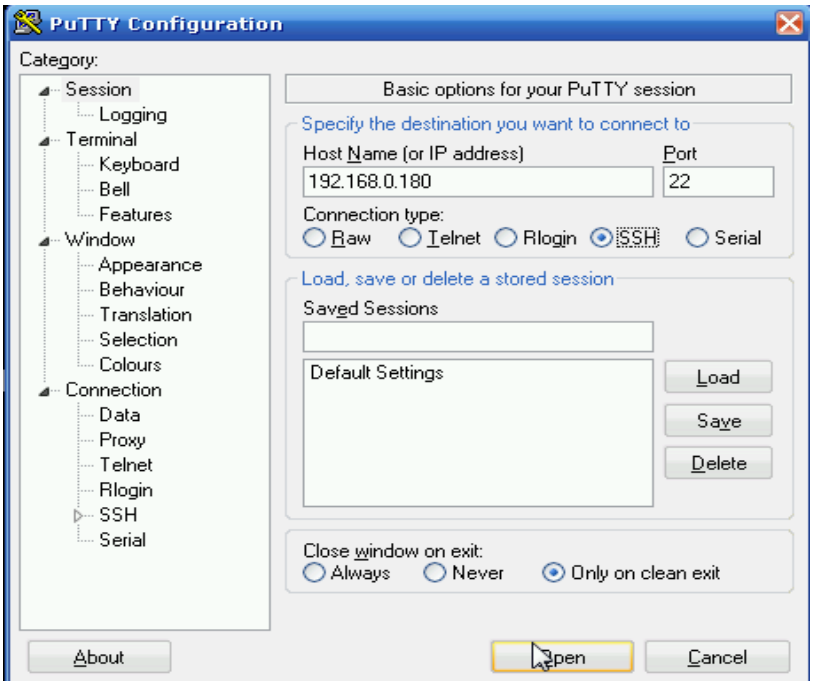

Рисунок 6: Использование клиента PuTTY

Таким образом, достаточно указать только IP-адрес устройства (порт подключения

стандартный).

После успешного подключения в окне терминала отобразится диалог входа в систему, где нужно ввести имя пользователя и пароль. В системе зарегистрировано 2 пользователя :

имя пользователя	пароль	командная оболочка
specadmin	alsitec	sh
superuser	123456	cli

Соответственно, чтобы получить доступ до командного интерфейса настройки CLI, необходимо войти под именем пользователя: superuser.

4.2. Конфигурирование устройства

4.2.1. Основные понятия

DSLAM (Digital Subscriber Line Access Multiplexer) – устройство, позволяющее организовать высокоскоростное подключение к интернет через телефонную линию. Обычно оно располагается на небольшом расстоянии от абонентов и соединяет несколько DSL абонентов с высокоскоростным интернет каналом, используя технологии мультиплексирования. Типовое изображение DSLAM на схемах показано на изображении.

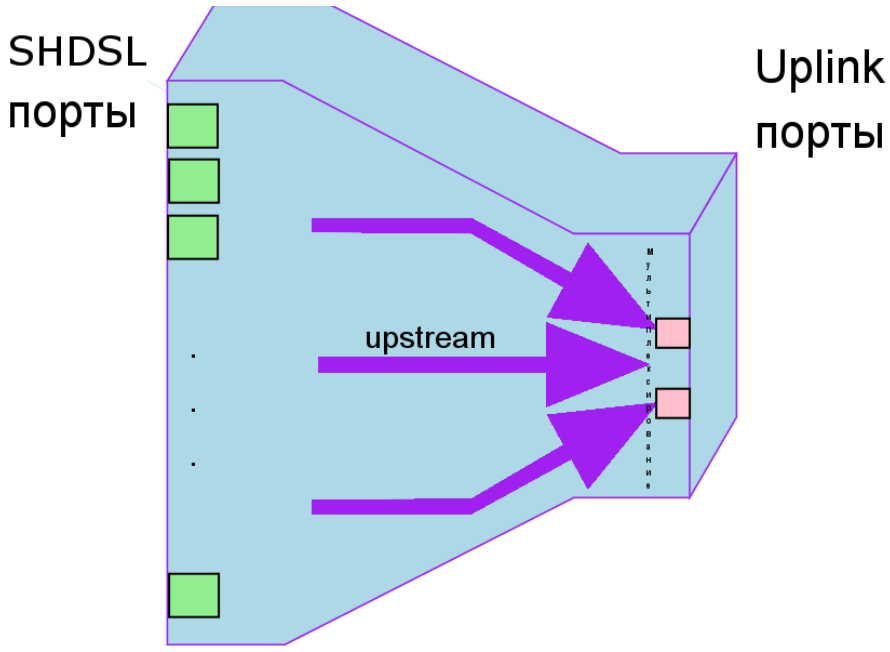

Иллюстрация 1.: Типовое изображение DSLAM на схемах

Upstream – направление трафика от абонента (SHDSL-порта) в сторону провайдера (UPLINK-порта);

Downstream – направление трафика от провайдера (UPLINK порта) в сторону абонента

(SHDSL порта);

Порт - представляют собой физические соединители и каналы аппаратных средств. В данном документе понятия порт и интерфейс отличаются. На рисунке изображен набор портов. В DSLAM бывает два типа портов:

- SHDSL порты;
- UPLINK порты.

Примерами портов может служить разъем Ethernet RJ-45;

Интерфейс – это логическая конструкция, которая обеспечивает протокол и сервисную информацию более высокого уровня, независимо от физических портов. Список интерфейсов, поддерживаемых SHDSL-16.EFM будет представлен позже.

Стек протоколов - набор взаимодействующих сетевых протоколов. Сами по себе протоколы почти никогда не работают. Каждый протокол отвечает только за определенную часть пакета. Обработав свою часть, прокол передает пакет следующему (более высокому) уровню. Таким образом пакет проходит весь стек от самого нижнего (физического), до самого высокого (пользовательского). Построение протоколов в стеки наглядно описывает модель OSI.

4.2.2. Концепция конфигурирования

4.2.2.1. Контексты

Контекст представляет собой определенную сетевую технологию или протокол, а именно DSLAM технологию (на базе стека протоколов SHDSL). Контекст может рассматриваться как виртуально отдельное оборудование внутри устройства. Например, в MSPU ОС SHDSL:

- контекст TDM содержит функции коммутации каналов;
- контекст DSLAM содержит функции, относящиеся к настройке DSLAM.

Контексты идентифицируются именем и содержат команды конфигурирования, которые связаны с технологией, которую они представляют. Раздельное конфигурирование позволяет поддерживать новые технологии сетевого уровня не усложняя методы конфигурирования существующих функций.

4.2.2.2. Интерфейсы, порты и привязки

Концепция *интерфейса* SHDSL-16.EFM в отличается от этого понятия в традиционных сетевых устройствах. Традиционно, термин интерфейс является синонимом *порта* или *разъема*, которые являются физическими объектами. В SHDSL-16.EFM, однако, интерфейс – это логическая конструкция, которая обеспечивает протокол и сервисную информацию более высокого уровня, независимо от физических портов и схем. Отделение интерфейса от объектов

физического уровня позволяет поддерживать многие расширенные функции, предлагаемые SHDSL-16.EFM.

Для активизации протоколов более высокого уровня, необходимо связать интерфейс с физическим портом или схемой. Эта ассоциация задается в SHDSL-16.EFM как *привязка (binding)*.

Порты и *схемы* в SHDSL-16.EFM представляют физические соединители и каналы аппаратных средств SHDSL-16.EFM. Конфигурирование порта или схемы включает параметры для физического уровня и уровня передачи данных, такие, как синхронизация линии, линейный код, управление доступом и т.д.

Для того, чтобы любые пользовательские данные могли передаваться через физический порт или схему, нужно связать этот порт или схему с интерфейсом контекста.

Привязки формируют ассоциацию между схемами или портами и интерфейсами, конфигурируемыми в контексте. Никакие пользовательские данные не могут передаваться на схему или порт, пока какой либо сервис более высокого уровня не сконфигурирован и не привязан к этим порту и схеме.

Проще говоря, пока интерфейсы и порты не будут объединены в стек протоколов, данные через устройство не пойдут. Именно конфигурация привязок определяет, как и куда пойдут данные.

В различных случаях связывание портов и интерфейсов производится либо снизу вверх, либо сверху вниз. В любом случае это делается командой *bind*.

4.2.2.3. Профили

Профили обеспечивают сокращение времени при конфигурировании. Они содержат определенные параметры, которые могут использоваться в множестве контекстов, интерфейсов или портов. Такая концепция позволяет избегать повторений групп команд конфигурирования, которые являются идентичными для многих элементов при конфигурировании.

Команды использования профилей формируют ассоциацию между профилями и контекстами, интерфейсами и портами. Когда профиль используется в контексте, все параметры этого профиля становятся активными внутри контекста. При изменении присоединенного к интерфейсу или контексту профиля, меняется и поведение присоединенных к нему контекстов или интерфейсов.

4.2.2.4. Пример использования концепции

Приведем пример использования данной концепции на основе DSLAM. Для начала необходимо описать стек протоколов, который возможен на этом устройстве.

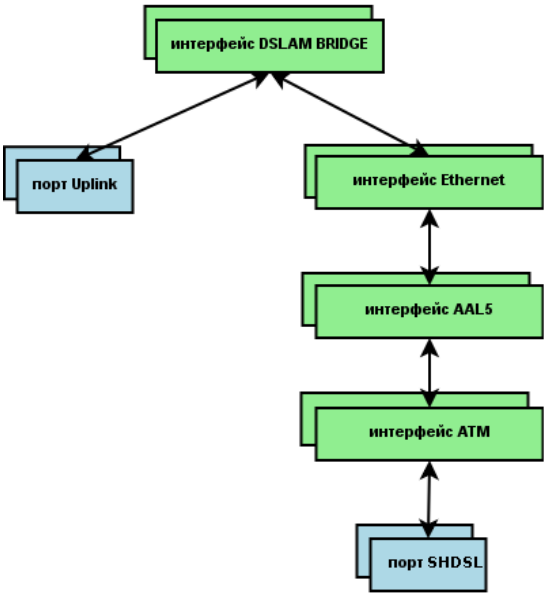

Иллюстрация 2.: Стек протокола для DSLAM для сценария EoA
(зеленые прямоугольники – интерфейсы, синие – порты, стрелки - привязки)

Сценарий EoA (Ethernet over ATM) – способ соединения интерфейсов в стек таким образом, чтобы пакет, принимающийся с SHDSL порта, воспринимается как ATM ячейки. Интерфейс AAL5 собирает ячейки с нужными полями VPI/VCI и собирает из ячеек AAL5 кадры. Эти кадры проверятся и из них выделяются ETHERNET кадры (путем отделения заголовка AAL5). Затем ETHERNET кадр попадает на интерфейс ETHERNET. Здесь происходит обработка ETHERNET заголовка. (снятие/навешивание VLAN и другие преобразования). После этого ETHERNET попадает на ETHERNET-мост, который, в зависимости от режима рассылает пакет либо всем портам, либо только на порт UPLINK. Пакеты идущие в обратную сторону, проходят те же этапы, но в обратном порядке. Таким образом, DSLAM в сценарии EoA работает как обычный L2 мост.

Следует отметить, что в данной схеме могут присутствовать несколько экземпляров портов UPLINK и SHDSL, интерфейсов ATM, AAL5, ETHERNET и DSLAM_BRIDGE. T.e. Каждый экземпляр порта SHDSL (например, shdsl1) должен быть привязан к своему экземпляру интерфейса ATM (например, atm1). Интерфейс ATM в свою очередь связан с интерфейсом AAL5 (или несколькими), и т.д., вплоть до DSLAM_BRIDGE.

Каждый экземпляр может иметь свои настройки для протокола, который он представляет. Например интерфейс ETHERNET включает в себя настойки, определяющие, какие VLAN метки должны быть повешены/сняты на данном интерфейсе и в какую сторону.

Интерфейсы – логические объекты, они могут создаваться и удаляться. Удалять ненужные

интерфейсы стоит хотя бы потому, чтобы не загромождать рабочую конфигурацию. Порты – физические объекты, их нельзя ни удалять ни добавлять.

Теперь рассмотрим пример, показывающий настройку двух портов для двух разных абонентов.

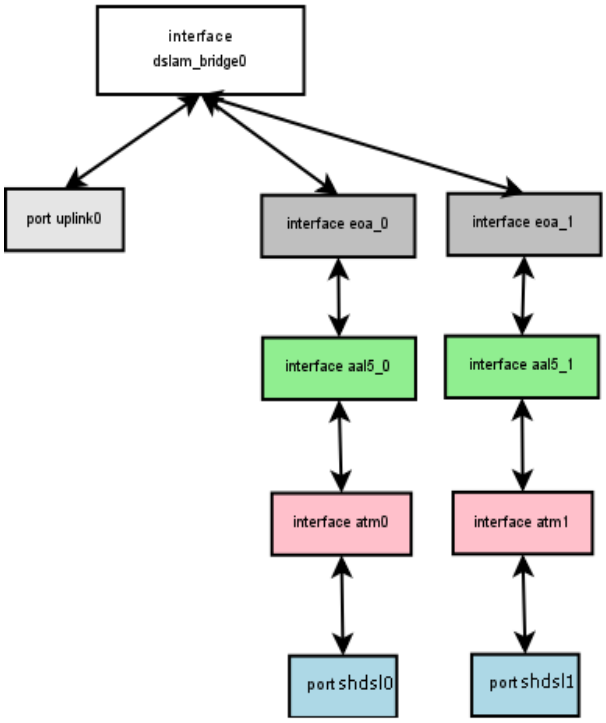

Иллюстрация 3.: Пример соединения 2 абонентских портов SHDSL с портом UPLINK по сценарию EoA

Как видно из представленной схемы, у портов только один общий интерфейс – DSLAM_BRIDGE. Это означает, что любой экземпляр интерфейса DSLAM_BRIDGE должен поддерживать несколько привязок «снизу».

Рассмотрим теперь более сложный пример: использование нескольких PVC на одном порту SHDSL.

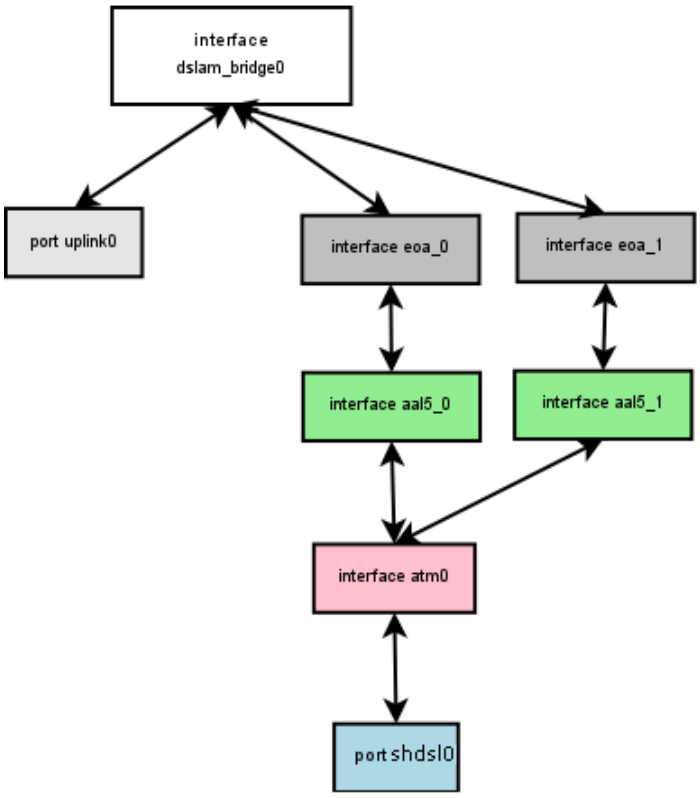

Иллюстрация 4.: Пример использования 2 PVC на одном SHDSL порту с дальнейшим разнесением по разным VLAN, но в один UPLINK порт.

Данная схема отличается от предыдущей тем, что здесь оба AAL5 интерфейса привязаны к одному и тому же интерфейсу ATM, а через него и к одному интерфейсу SHDSL. Вопрос, как определить, какая ATM ячейка куда должна пойти, определяется по заголовку ATM, а именно по полям PVC – VPI/VCI.

Как видно из последней схемы, интерфейс ATM должен поддерживать несколько привязок «сверху».

4.2.3. Конфигурирование устройства с помощью интерфейса командной строки CLI

В данный момент подразумевается, что у оператор уже получил доступ по протоколу SSH до устройства. Тогда на экране терминала будет отображаться приглашение к вводу команд. Оператору рекомендуется ознакомиться с общими правилами работы и командами CLI в соответствующей документации. В данном разделе содержится только краткая справочная информация, достаточная для создания типовой конфигурации. Более детальное описание команд и их параметров изложено в документации на CLI данного устройства.

4.2.3.1. Настройка соединения от SHDSL-порта к порту UPLINK.

Последовательность действий при настройке.

Основные задачи по конфигурированию контекста, связанных с ним интерфейсов и портов:

- Вход в контекст DSLAM;
- Создание интерфейса DSLAM_BRIDGE (как минимум с одним интерфейсом COMMUNICATION) и его активизация;
- Активизация интерфейса ATM;
- Создание, связывание и активизация интерфейса AAL5;
- Определение типа инкапсулированного в AAL5 протокола, его связывание и активизация;
- Связывание порта UPLINK и его активизация;
- Активизация порта SHDSL.

После входа в контекст и выполнения основных задач конфигурирования становится возможным конфигурирование дополнительных настроек интерфейсов.

По окончании данного раздела должен получиться работающий набор интерфейсов, изображенных на картинке:

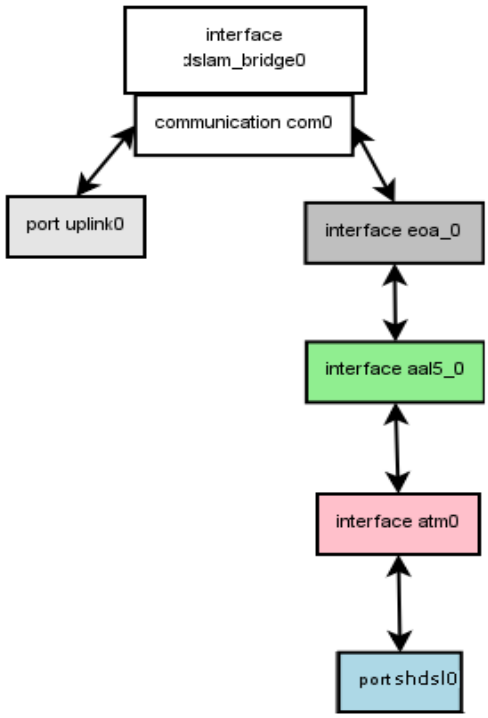

Иллюстрация 5.: Результат выполнения настройки по одному порту

Вход в контекст DSLAM.

Интерфейс командной строки (CLI) SHDSL-16.EFM имеет предварительно определенный контекст DSLAM, в котором содержатся настройки всех интерфейсов устройства. Поэтому перед непосредственным конфигурированием необходимо перейти в этот контекст.

Таблица 1.Последовательность действий для входа в контекст DSLAM

Шаг	Действие	Описание действия
1.	als$> **context dslam**	Переход в режим конфигурирования контекста DSLAM.

Создание и активация интерфейса DSLAM_BRIDGE.

Обычный DSLAM работает как коммутатор (L2 switch). Следовательно, ему нужен мост (bridge), который будет передавать пакеты с одного Ethernet-совместимого порта, подключенного к нему, на другой. С одной стороны в мост включен порт Uplink, который передает Ethernet-фреймы, с другой стороны – Ethernet-интерфейсы, которые передают данные от порта SHDSL. Интерфейс Bridge в DSLAM – это программно-аппаратный объект, и у него есть некоторые особенности, делающие его непохожим на обычный мост. Для того чтобы это подчеркнуть интерфейс называется Dslam_bridge.

Для создания и активизации нового моста необходимо выполнить следующие шаги:

Таблица 2. Последовательность действий для создания и активации интерфейса DSLAM_BRIDGE

Шаг	Действие	Описание действия
1.	**als(cntx-dslam)# interface dslam_bridge** br0	Создание и переход в режим конфигурирования нового мостового интерфейса `br0`.
2.	**als(interface)[dslam_bridge** br0**]# communication** com0	Создание интерфейса соединения (Communication) с именем `com0` внутри моста-мультиплексора `br0`. Данный тип интерфейсов активируется автоматически при создании моста. В дальнейшем именно через этот интерфейс будет проходить трафик между абонентским портом и портом Uplink.
3.	**als(interface)[dslam_bridge** br0**]# no shutdown**	Включение интерфейса `br0`.

Интерфейс Dslam_bridge может содержать в себе несколько объектов Communication. К одному интерфейсу Communication можно привязать только один порт Uplink. Это особенность аппаратной реализации. Таким образом, если в мост необходимо добавить несколько Uplink-портов, то для каждой привязки нужно создавать собственный Communication. Поясним рисунком:

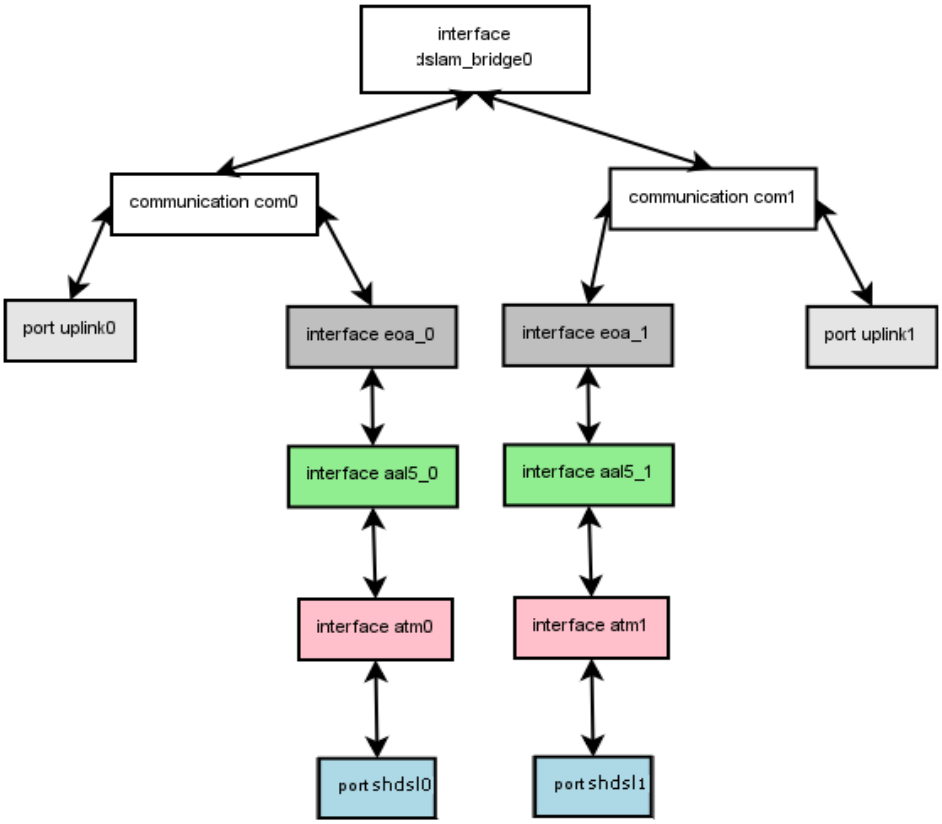

Иллюстрация 6.: Использование двух портов UPLINK

Пакеты, приходящие с порта SHDSL0 пойдут на UPLINK0. Пакеты, приходящие с порта SHDSL1 пойдут на UPLINK1. T.e. Мост жестко устанавливает соединение между SHDSL и UPLINK в upstream направлении, а не определяет это по mac learning алгоритму. Это и есть особенность аппаратной реализации моста.

Тем не менее, все остальные функции бриджа (например, STP, управление со всех портов по одному IP) выполняются для всех COMMUNICATION на уровне моста.

В простом случае не имеет смыла использование двух и более объектов COMMUNICATION, но такая возможность оставлена.

После этой операции схема приняла вид:

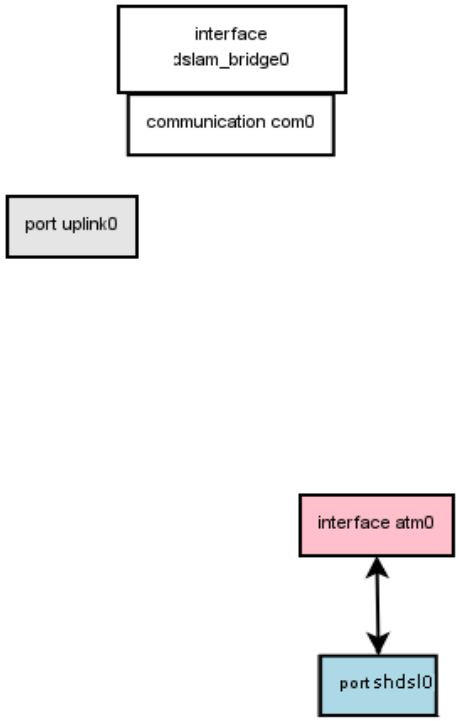

Иллюстрация 7.: Результат выполнения первого шага

Активация интерфейса ATM.

Для того чтобы начался прием ATM из порта SHDSL, необходимо активировать связанный с этим портом ATM-интерфейс.

Таблица 3.Последовательность действий для активации интерфейса ATM

Шаг	Действие	Описание действия
1.	**als(cntx-dslam)# interface atm** atm0	Переход в режим конфигурирования интерфейса `atm0`.
2.	**als(interface)[atm** atm0**]# no shutdown**	Включение интерфейса `atm0`.

Интерфейс ATM не надо связывать с портом SHDSL, т.к. эта связь постоянна (выполнена на аппаратном уровне). Каждому SHDSL-порту соответствует интерфейс ATM с тем же номером.

После того, как интерфейс был активизирован изменений в нашей схеме не произошо, т.к. не было добавлено ни интерфейсов, ни связок.

Создание, связывание и активация интерфейса AAL5.

Интерфейс AAL5 отвечает за выборку из интерфейса ATM ячеек с заданными значениями полей VPI/VCI и их сборку в пакеты Ethernet / IP / PPP (в зависимости от типа инкапсуляции).

Таблица 4.Последовательность действий для создания, связывания и интерфейса ATM

Шаг	Действие	Описание действия
1.	**als(interface)[atm** atm0**]# interface aal5** aal50	Создание нового интерфейса `aal50`, который реализует протокол AAL5. Эта команда также переводит CLI в режим конфигурирования созданного интерфейса.
2.	**als(interface)[aal5** aal50**]# bind** atm0	Связывание интерфейсов `aal50` и `atm0`.
3.	**als(interface)[aal5** aal50**]# no shutdown**	Включение интерфейса `aal50`.
4.	**als(interface)[aal5** aal50**]# encapsulation ethernet**	Создание интерфейса инкапсуляции типа Ethernet (для режима Ethernet через ATM) для данного AAL5 и переход в режим конфигурирования созданного интерфейса.

Таким образом, после выполнения описанных команд наша схема имеет вид:

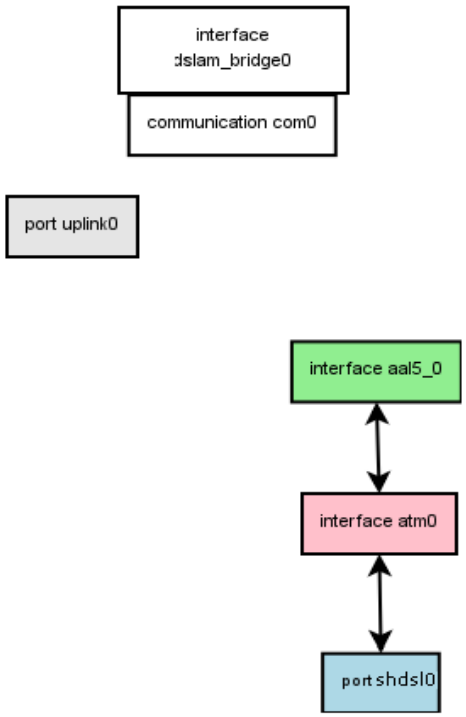

Иллюстрация 8.: Результат выполнения третьего шага

AAL5 интерфейс поддерживает различные типа инкапсулированных пакетов: ETHERNET/IP/PPP. Следующим шагом будет задание и настройка типа инкапсуляции.

Определение типа интерфейса инкапсуляции, его связывание и активация.

Интерфейс Encapsulation нужен для того, чтобы передавать дальше собранные интерфейсом AAL5 пакеты из ATM-ячеек, которые пришли с порта SHDSL, а также для приема пакетов, пришедших с порта Uplink, и передачи их на интерфейс AAL5.

Таблица 5.Последовательность действий для задания типа инкапсулирующего интерфейса, его связывания и активации

Шаг	Действие	Описание действия
1.	**als(interface)[aal5** aal50**]# encapsulation ethernet**	Создание интерфейса инкапсуляции типа Ethernet (для режима Ethernet через ATM) для данного AAL5 и переход в режим конфигурирования созданного интерфейса.
2.	**als(aal5)[encap ethernet]# bind** com0	Связывание интерфейса encapsulation с интерфейсом Communication `com0`.
3.	**als(aal5)[encap ethernet]# no shutdown**	Активация текущий интерфейса инкапсуляции.

В настоящее время SHDSL-16.EFM поддерживает только инкапсуляцию типа ETHERNET.

Команда создания типа инкапсуляции делает сразу несколько действий. Если у интерфейса AAL5 не было связанного с ним верхнего интерфейса, то такой интерфейс создается и с ним устанавливается связка. Если же интерфейс был указанного в команде типа, то эта команда просто переходит в режим его редактирования. Если тип не совпадает – команда возвращает ошибку.

Таким образом, после выполнения описанных команд наша схема имеет вид:

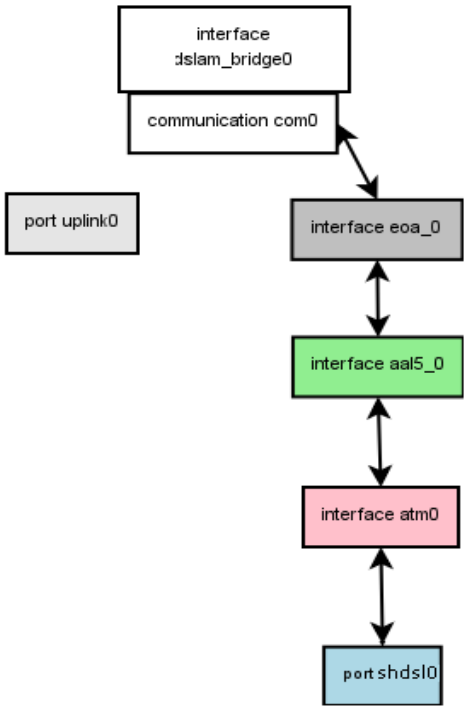

Иллюстрация 9.: Результат выполнения четвертого шага

Теперь, когда все необходимые интерфейсы и порты созданы, и пакеты могут проходить

от порта SHDSL до моста и обратно, осталось только связать и активировать порты Uplink и SHDSL.

Связывание и активация порта UPLINK.

Для того, чтобы пакеты могли приходить на мост со стороны Uplink-порта, его необходимо связать с этим мостом. Для этого нужно выполнить следующие действия:

Таблица 6.Последовательность действий для связывания и активации порта Uplink

Шаг	Действие	Описание действия
1.	**als(aal5)[encap ethernet]# port uplink** uplink0	Данная команда переводит пользователя в режим редактирования настроек порта `uplink0`.
2.	**als(port)[uplink** uplink0**]# bind** com0	Связывание порта `uplink0` с Communication `com0`.
3.	**als(port)[uplink** uplink0**]# no shutdown**	Включение текущего порта `uplink0`.

После выполнения описанных команд наша схема имеет вид:

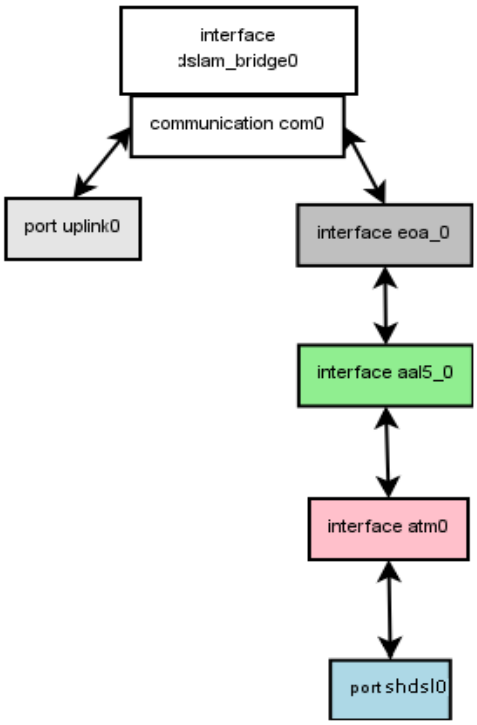

Иллюстрация 10.: Результат выполнения пятого шага

Фактически, мы уже получили требуемую структуру, но она пока работать не будет. Осталось сделать последнюю вещь – включить SHDSL порт.

Активация порта SHDSL.

Активация порта SHDSL не требуется, порт SHDSL включен на физическом уровне.

После установления соединения с модемом DSLAM начнет передавать пакеты от пользователя в сеть (upstream) и из сети к пользователю (downstream).

4.2.3.2 Стандартные варианты настройки DSLAM.

Введение.

Данный раздел содержит типовые варианты использования и настройки DSLAM и ссылки на конфигурации. Для лучшего понимания происходящего рекомендуется прочитать предыдущие разделы. Кроме того необходимо иметь знания в построении сетевой архитектуры. Тем не менее раздел написан доступно и имеются наглядные схематические примеры.

Простой способ подключения DSLAM (без VLAN).

В данной конфигурации DSLAM является мостом, пропускающим трафик без меток VLAN от абонента в сеть провайдера и наоборот. Тем не менее абоненты не смогут получить доступ к друг другу. Кроме того данная конфигурация будет позволять управление как со стороны сети провайдера, так и со стороны абонентов. Визуально это должны выглядеть следующим образом.

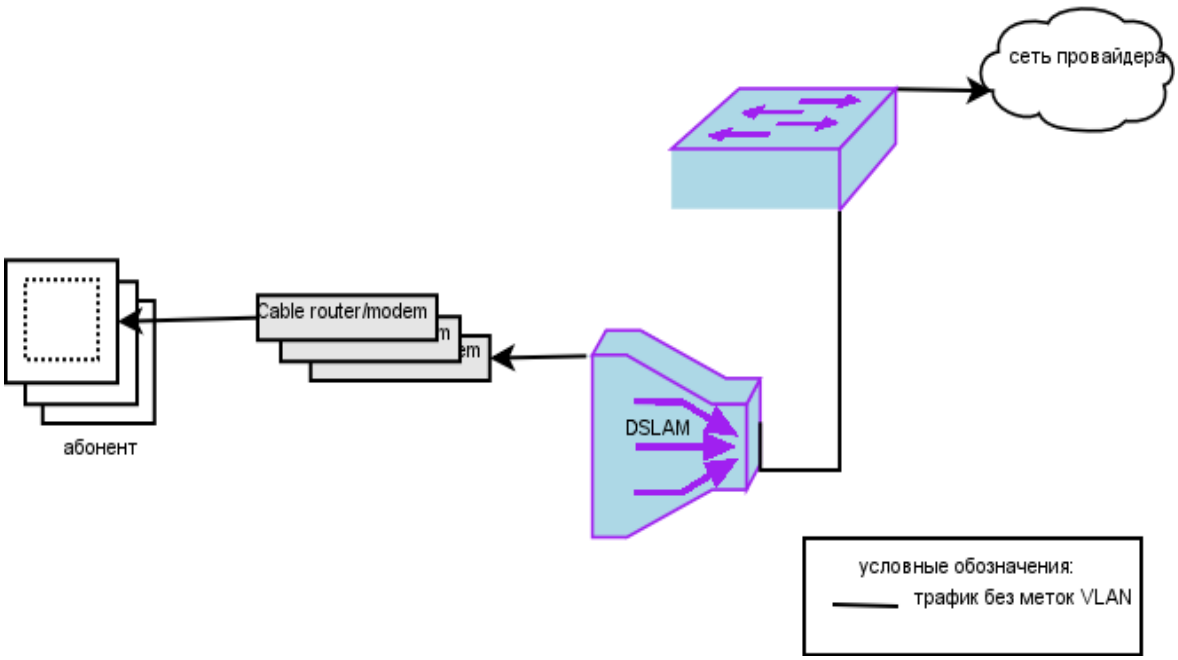

Иллюстрация 11.: Схема простого способа подключения DSLAM

Для этого надо выполнить следующую последовательность:

Таблица 7.Последовательность действий для назначения IP адреса на устройство

Шаг	Действие	Описание действия
1.	**als(aal5)[encap ethernet]# context ip router**	Данная команда переводит пользователя в режим редактирования настроек маршрутизатора IP.
2.	**als(cntx-ip)[router]# ifconfig hbr0 172.16.32.67 netmask 255.255.0.0 up**	Назначает на интерфейс hbr0 IP адрес с маской и поднимает его.
3.		

Вторая команда требует объяснения. При создании интерфейса dslam_bridge br0 автоматически создается хост интерфейс, который выбирает все пакеты, предназначенные данному хосту (процессору) из всех проходящих пакетов. В принципе этот интерфейс нужен для того, чтобы отделить статистику по аппаратному интерфейсу, который пропускает все пакеты, проходящие с одного порт на другой и пакеты, проходящие на процессор устройства. К тому же это позволяет отключив хост интерфейс оставить поток пакетов на его нижнем уровне. Таким образом отключается управление.

Такие интерфейсы автоматически порождаются всеми ETHERNET совместимыми интерфейсами.

Таблица 8 Примеры автоматического генерирования имен хост интерфейсов

Интерфейс	Хост-интерфейс
interface dslam_bridge br0	hbr0
port uplink uplink0	huplink0
encapsulation ethernet	heoa0

Как видно из таблицы – имя хост интерфейса всегда начинается на «h» (от слова host). Кроме того, UPLINK порты и EoA интерфейсы имеют схожие команды с синтаксисом:

```
[no] listen [bridge].
```

Если у порта или интерфейса в настройках написано

```
no listen
```

то интерфейс не перенаправляет мосту, к которому он подключен выловленные для устройства пакеты. Наоборот, если написано

```
listen bridge
```

то интерфейс отправляет и получает пакеты на/с моста, к которому он подключен.

Таким образом, для того, чтобы отключить управление со стороны какого-либо порта, необходимо в его конфигурации прописать

```
no listen
```

и не назначать на его хост интерфейс IP адрес.

Конфигурация с навешиванием VLAN меток на SHDSL порт (private VLAN) и управлением по другому VLAN.

Данная конфигурация довольно сильно отличается по безопасности от предыдущей, но изменений потребует немного. Отобразим конфигурацию на схеме.

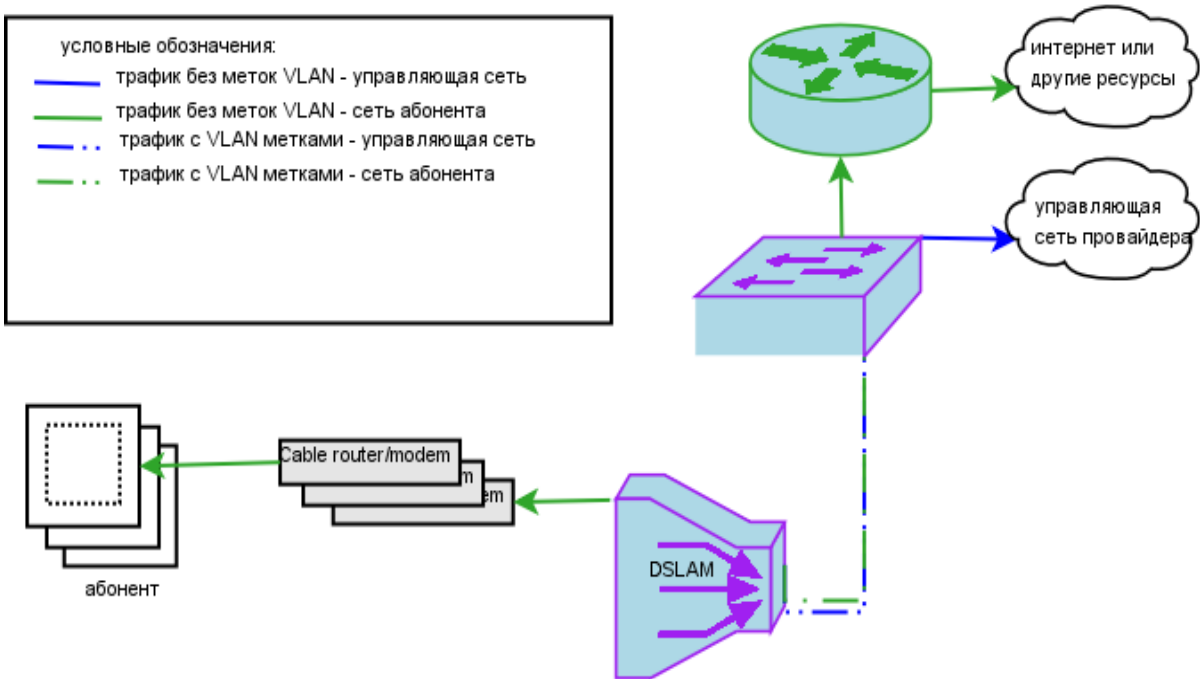

Иллюстрация 12.:

Для того, чтобы реализовать изображенную схему необходимо в предыдущей конфигурации (полученной в прошлом разделе) создать профиль VLAN для абонентского трафика. Для этого достаточно выполнить следующую команду:

```
als$> profile vlan a_inet id 333
```

где a_inet -имя создаваемого профиля, а 333 – идентификатор тега. После этого необходимо все нужные интерфейсы encapsulation ethernet перевести в режим навешивания тега командой:

```
als$> context dslam
als(cntx-dslam)# interface aal5 aal50 encapsulation ethernet use vmod tagged a_inet
```

можно было и по-другому:

```
als$> context dslam
als(cntx-dslam)# interface aal5 aal50
als(interface)[aal5 aal50]# encapsulation ethernet use vmod tagged a_inet
```

В любом случае это надо сделать на всех портах. После этого получается, что абонентский трафик помечается метками VLAN. Трафик управления на dslam_bridge остается неизменным. Для того, чтобы он тоже начал тагироваться, необходимо в контексте IP ROUTER

выполнить следующие команды:

Таблица 9.Последовательность действий для назначения VLAN управления на устройство

Шаг	Действие	Описание действия
1.	**als(aal5)[encap ethernet]# context ip router**	Данная команда переводит пользователя в режим редактирования настроек маршрутизатора IP.
2.	**als(cntx-ip)[router]# ifconfig hbr0 0.0.0.0** **als(cntx-ip)[router]# ifconfig hbr0 mtu 1504 up**	Сброс IP адреса хост интерфейса. Назначение mtu на четыре байта больше (VLAN заголовок занимает 4 байта). Это надо для того, чтобы интерфейс hbr0.444 работал с mtu 1500.
3.	**als(cntx-ip)[router]# vconfig add hbr0 444**	Создание нового виртуального интерфейса, работающего со VLAN 444. В результате должен появиться новый интерфейс hbr0.444.
4.	**als(cntx-ip)[router]# ifconfig hbr0.444 172.16.32.67 netmask 255.255.0.0 mtu 1500 up**	Назначает на интерфейс hbr0, vlan 444 IP адрес с маской и поднимает его.
5.		

Если на портах будет включен *listen bridge*, все равно абоненты не смогут управлять устройством. Полученная конфигурация не будет давать абонентам взаимодействовать друг с другом несмотря на то, что они находятся в одном VLAN(это называется *private vlan*). Если этого не требуется (т.е. абоненты должны иметь возможность получить доступ к друг другу), то необходимо выполнить следующую команду:

```
als(cntx-dslam)# interface dslam_bridge br0 promisc_us
```

В таком режиме абоненты имеют доступ к компьютерам друг друга, если находятся в одном VLAN.

Внимание!

Следует отметить, что весь обмен между пользователями выполняется программно, поэтому это может замедлить передачу данных, т.е. не следует ожидать полной скорости по ADSL порту при таком обмене.

Если необходимо предоставить такую услугу только части абонентам, то это можно сделать только выполнив на их портах команду типа:

```
als(interface)[aal5 aal50]# encapsulation ethernet no listen
```

Настройка IGMP snooping.

IGMP snooping (IGMP v2) механизм для маршрутизаторов описан в RFC 2236. Для мостов этот механизм несколько проще. Он позволяет не расходовать пропускную способность канала на ненужные абоненту IGMP трафик. Для этого DSLAM перехватывает все IGMP запросы,

идущие от абонента и анализирует, IGMP report, join и leave пакеты. В данной версии поддерживаются IGMPv1, IGMPv2. Версия IGMPv3 тоже реализована, но тестировалась на реальных сетях.

По умолчанию IGMP snooping включен. Поэтому в конфигурации, которая получилась в предыдущем разделе он будет работать автоматически. Snooping таймер задается глобально для всего контекста и по-умолчанию равен 3 минутам. Этого значения обычно достаточно для большинства систем. Иногда некоторым абонентам требуется отключить группу. В зависимости от того, сколько групп необходимо отключить, а сколько оставить включенными, можно действовать разными методами.

Настройки по доступным MULTICAST группам отдельные у каждого encapsulation ethernet. Кроме того имеется и глобальная настройка, которая находится в контексте DSLAM и профилях MULTICAST.

Все MULTICAST группы делятся на три вида:

- Статическая группа – добавленный вручную в конфигурации в encapsulation ethernet профиль MULTICAST;
- Профильная группа – добавленный автоматически по запросу со стороны абонента в encapsulation ethernet профиль MULTICAST;
- Динамическая группа – добавленный автоматически по запросу со стороны абонента в encapsulation ethernet MULTICAST адрес.

Каждую из этих групп можно либо включить, либо выключить либо заблокировать для конкретного encapsulation ethernet.

Таким образом, если необходимо выключить несколько групп, а остальные оставить включенные, то их надо прописать в профилях MULTICAST. После этого следует отключить или заблокировать профильные группы у encapsulation ethernet у тех абонентов, которым их надо выключить. Если необходимо выключить у всех абонентов сразу, то можно просто выключить саму группу вызвав команду *shutdown*.

Поясним примером (здесь показан фрагмент конфигурации, отвечающий за IGMP).

```
...
#----------multicast------------------------------#
profile multicast default
    type multicast
    no shutdown
    ipaddr 224.0.0.1
    no use vlan

profile multicast brdcst
    type broadcast
    no shutdown
...
```

```
        mode llc
        priority 0
        no fcs
        no accounting
        use pvc default
        bind atm0
        description "aal5 interface"
        no shutdown
#----------interface eoa--------------------------#
        encapsulation ethernet
            learning
            mapping
            authentification
            no usFiltering
            no preservePriority
            no configPriority
            use vmod tagged a_inet
            listen bridge
            igmp
                static
                profiled
                dynamic
                group static brdcst
                    no profileVlan
                    manual
                    shutdown
            bind com0 300 128
            description "eoa interface"
            no shutdown
```

Итак, для того, чтобы заблокировать группу *default* на приведенном EoA интерфейсе необходимо перевести его *igmp* параметр *profiled* следующим образом:

```
als(encap eth)[igmp]# no profiled
```

После этого, если абонент запросит IGMP группу, то группа не будет включена, и абонент не получит данные, приходящие из сети провайдера.

Для того, чтобы разрешить только выбранные группы, создать профили для каждой из выбранной группы, и добавить их в encapsulation ethernet статически. После этого необходимо заблокировать или выключить все типы групп кроме статических. Например, нам необходимо запретить абоненту все группы кроме 224.1.2.3. Для этого необходимо создать группу и включить ее.

```
als(encap eth)[igmp]# profile multicast grp123 ipaddr 224.1.2.3
als(encap eth)[igmp]# profile multicast grp123 no shutdown
```

Соединяем ее с интересующим абонентом

```
als(aal5)[encap ethernet]# igmp group static grp123
als(igmp)[group static grp123]# no shutdown
```

Запрещаем остальные типы

```
als(igmp)[group static grp123]# exit
als(encap eth)[igmp]# no dynamic
als(encap eth)[igmp]# no profiled
```

После этого абонент не сможет получить трафик по другим группам.

Для просмотра состояния

Настройка DHCP relay (option 82).

DHCP Relay и его опции описаны в RFC3046. Для динамического выделения IP адресов абонентам можно использовать DHCP сервер, находящийся в сети провайдера. Алгоритм выделения можно прочитать на http://ru.wikipedia.org/wiki/DHCP. Основная проблема для провайдера в этом случае, как определить, сколько и какие IP адреса были запрошены с данного порта. Для этой цели в SHDSL-16.EFM в каждом encapsulation ethernet можно включить DHCP Relay Agent, чья задача: дополнять проходящие через него DHCP запросы информацией о том, на какой порт и с какими параметрами AAL5 был принят данный запрос, а также физический адрес самого SHDSL-16.EFM . Вся эта информация находится в части DHCP запроса, называемой option 82.

В случае применения DHCP relay отпадает необходимость использования нескольких DHCP серверов, т.к. можно четко определить, откуда был запрос.

Формат поля DHCP option 82 для SHDSL-16.EFM.

Опция 82 может включать в свой состав Circuit-ID и Remote-ID. В Circuit-ID обычно указывается информация о схеме (номере порта, VPI/VCI), через которую был получен запрос. Формат опции может быть настроен глобально.

```
Tarhova32_0_0$> context dslam
Tarhova32_0_0(cntx-dslam)# dhcp
Tarhova32_0_0(cntx-dslam)[dhcp]#
```

Содержимое Circuit-ID может быть в бинарном и в текстовом виде. Изначально поддерживался только бинарный вид и только со включенными Circuit-ID и Remote-ID. В последствии работы с протоколом PPPoE+ выяснилось, что иногда более удобно ориентироваться не на MAC-адрес, включаемый в RemoteID, а на hostname платы. Для простоты вся информация заполняется в Circuit-ID, а Remote-Id можно игнорировать и не передавать. Таким образом, если меняется плата (в случае например выхода из строя), а конфигурация остается, то на RADIUS-сервере и на DHCP-сервере не нужно менять настроек, связанных с изменением MAC адреса.

Чтобы включить бинарный формат, необходимо выполнить команду:

```
Tarhova32_0_0(cntx-dslam)[dhcp]# circuit-id binFormat
```

Бинарный формат поля опции с Circuit-ID (в нем указывается номер ADSL порта коммутатора и PVC, в котором проходил запрос):

1	2	3	4	5	6	7	8	9	10
0x01	0x08	0x08	0x06	номер порта		VPI		VCI	

1. Тип поля опции;2. длина 3. тип Curcuit Id (8 =EoA) 4. Длина данного типа

5, 6 Номер ADSL порта, с которого получен DHCP пакет

7,8,9,10 VPI/VCI с которых получен DHCP пакет.

Чтобы включить текстовый формат, необходимо выполнить команду:

Tarhova32_0_0(cntx-dslam)[dhcp]# circuit-id textFormat "%h atm %p.%vpi:%vci"

В качестве аргумента к этой команде используется строка, в которой задается шаблон, куда будут заноситься конкретные значения следующих переменных:

%h – имя хоста;

%p – номер порта;

%vpi – VPI;

%vci – VCI.

Имеется возможность вообще не включать Circuit-Id в опцию. Для этого можно выполнить команду

Tarhova32_0_0(cntx-dslam)[dhcp]# no circuit-id

Аналогично Circuit-ID можно настраивать включение Remote-ID в пакет:

для выключения выполнить

Tarhova32_0_0(cntx-dslam)[dhcp]# no remote-id

для включения выполнить

Tarhova32_0_0(cntx-dslam)[dhcp]# remote-id

Формат поля опции с Remote-ID только бинарный, в нем указывается MAC-адрес устройства, являющегося агентом DHCP Relay:

1	2	3	4	5	6	7	8	9	10
0x02	0x08	0x00	0x06	MAC адрес коммутатора					

1. Тип поля опции;2. длина 3. тип Remote Id (0 = MAC address) 4. Длина данного типа

5,6,7,8,9,10. MAC адрес коммутатора, добавившего DHCP option 82.

Включение, выключение и конфигурирование параметров.

После создания объекта encapsulation ethernet DHCP relay по умолчанию выключен. Для его включения необходимо выполнить следующее:

Таблица 10.Включение DHCP Relay

Шаг	Действие	Описание действия
1.	**als(aal5)[encap ethernet]# dhcpRelay**	Данная команда включает DHCP Relay на интерфейсе и переводит пользователя в режим редактирования его настроек.
2.		

По умолчанию Relay создается со следующими параметрами:

- untrusted
- mtu 1524

DHCP Relay может работать в двух режимах: Untrusted – режим, когда Relay не доверяет данным, пришедшим в DHCP пакете. Согласно RFC3046, если в пакете уже есть option 82, такой пакет должен просто отбрасываться. Trusted режим отличается тем, что если DHCP пакет уже содержит option 82, то пакет будет передан DHCP серверу в неизменном виде.

Параметр MTU контролирует максимальный размер пакета для DHCP запроса. Если получаемый после добавления DHCP option 82 пакет, будет больше чем MTU, то такой пакет будет передаваться без изменений.

Для отключения DHCP relay необходимо сделать следующее:

Таблица 11.Выключение DHCP Relay

Шаг	Действие	Описание действия
1.	**als(aal5)[encap ethernet]# no dhcpRelay**	Данная команда выключает DHCP Relay на интерфейсе.
2.		

Просмотр конфигурации и статистики.

Для того, чтобы просмотреть конфигурацию DHCP Relay, необходимо находясь в encapsulation ethernet или в encapsulation ethernet->dhcpRelay набрать команду:

show config

Например,

```
als(aal5)[encap ethernet]# dhcpRelay
als(encap eth)[dhcp]# show config
            dhcpRelay
                untrusted
                mtu 1524
```

Для просмотра статистики можно использовать следующие варианты:

Пример 1:

```
als(encap eth)[dhcp]# show
        DHCP statistic:
        fwd: 0 inserted: 0
        discarded: 0 oversize: 0
```

Пример 2:

```
als(aal5)[encap ethernet]# show dhcp
        DHCP statistic:
        fwd: 0 inserted: 0
        discarded: 0 oversize: 0
```

Оба варианта показывают одно и то же. Если DHCP Relay выключен, то выдастся соответствующее предупреждение.

Таблица 12: Специфичные параметры статистики для UPLINK порта

Название поля	Описание
fwd	Количество распознанных как DHCP и пересланных в сторону провайдера
inserted	Количество пакетов, в которые была включена option 82
discarded	Количество отклоненных пакетов (по причине untrusted)
oversize	Количество пакетов с размером, превышающим MTU

Настройка функции PPPoE+ (RFC4679).

Данная функция аналогична функции DHCP option 82, но применительно к протоколу PPPoE (см. RFC2516). На этапе Discovery PPPoE пакеты посылаются клиентом с EtherType 0x8863. При включенной функции PPPoE+ эти пакеты принимаются и обрабатываются SHDSL-16.EFM программно. Эта возможность включается глобально, однако параметры преобразования включается/выключается на отдельном encapsulation ethernet. В пакета PADI и PADR добавляется информация специального вида о том, через какой порт и плату он прошел.

Формат поля PPPoE+ для SHDSL-16.EFM.

В PPPoE приложениях информация о физическом положении подписчика может быть включена в PPPoE discovery пакеты. За это отвечает Vendor-Specific 0x0105 тэг (см. RFC2516). Содержимое данного пакета показано ниже:

0	1	2	3
0x0105 (Vendor-Specific)		Tag-Length	
0x00000de9 (3561, т.е. «ADSL Forum IANA entry»)			
Circuit-ID (опционально)			
Remote-ID(опционально)			

Содержимое Circuit-ID может быть в бинарном и в текстовом виде. Изначально

поддерживался только бинарный вид и только со включенными Circuit-ID и Remote-ID. В последствии работы с протоколом PPPoE+ выяснилось, что иногда более удобно ориентироваться не на MAC-адрес, включаемый в RemoteID, а на hostname платы. Для простоты вся информация заполняется в Circuit-ID, а Remote-Id можно игнорировать и не передавать. Таким образом, если меняется плата (в случае например выхода из строя), а конфигурация остается, то на RADIUS-сервере и на DHCP-сервере не нужно менять настроек, связанных с изменением MAC адреса.

Чтобы включить бинарный формат, необходимо выполнить команду:

Tarhova32_0_0(cntx-dslam)[pppoe_plus]# circuit-id binFormat

Бинарный формат поля опции с Circuit-ID (в нем указывается номер ADSL порта коммутатора и PVC, в котором проходил запрос):

1	2	3	4	5 6	7 8	9 10
0x01	0x08	0x08	0x06	номер порта	VPI	VCI

1. Тип поля опции;2. длина 3. тип Curcuit Id (8 =EoA) 4. Длина данного типа

5, 6 Номер ADSL порта, с которого получен PPPoE пакет

7,8,9,10 VPI/VCI с которых получен PPPoE пакет.

Чтобы включить текстовый формат, необходимо выполнить команду:

Tarhova32_0_0(cntx-dslam)[pppoe_plus]# circuit-id textFormat "%h atm %p.%vpi:%vci"

В качестве аргумента к этой команде используется строка, в которой задается шаблон, куда будут заноситься конкретные значения следующих переменных:

%h – имя хоста;

%p – номер порта;

%vpi – VPI;

%vci – VCI.

Имеется возможность вообще не включать Circuit-Id в опцию. Для этого можно выполнить команду

Tarhova32_0_0(cntx-dslam)[pppoe_plus]# no circuit-id

Аналогично Circuit-ID можно настраивать включение Remote-ID в пакет:

для выключения выполнить

Tarhova32_0_0(cntx-dslam)[pppoe_plus]# no remote-id

для включения выполнить

Tarhova32_0_0(cntx-dslam)[pppoe_plus]# remote-id

Формат поля опции с Remote-ID только бинарный, в нем указывается MAC-адрес устройства, являющегося агентом PPPoE+ Relay:

1	2	3	4	5	6	7	8	9	10
0x02	0x08	0x00	0x06	MAC адрес коммутатора					

1. Тип поля опции;2. длина 3. тип Remote Id (0 = MAC address) 4. Длина данного типа 5,6,7,8,9,10. MAC адрес коммутатора, добавившего PPPoE+ информацию.

Включение, выключение и конфигурирование параметров.

Для того, чтобы SHDSL-16.EFM начал принимать PPPoE Discovery пакеты, необходимо выполнить следующее:

Таблица 13.Глобальное включение/выключение PPPoE+

Шаг	Действие	Описание действия
1.	**als(cntx-dslam)# pppoe_plus**	Данная команда включает прием PPPoE Discovery пакетов на всех интерфейсах.
2.	**als(cntx-dslam)# no pppoe_plus**	Команда выключения приема PPPoE Discovery.

После включения приема PPPoE Discovery пакетов можно настроить каждый интерфейс при помощи encapsulation ethernet параметров.

После создания объекта encapsulation ethernet PPPoE relay по умолчанию выключен. Для его включения необходимо выполнить следующее:

Таблица 14.Включение PPPoE Relay

Шаг	Действие	Описание действия
1.	**als(aal5)[encap ethernet]# pppoeRelay**	Данная команда включает PPPoE Relay на интерфейсе и переводит пользователя в режим редактирования его настроек.
2.		

По умолчанию Relay создается со следующими параметрами:

- untrusted
- mtu 1484

PPPoE Relay может работать в двух режимах: Untrusted/Trusted режимах. В отличие от DHCP Relay PPPoE Relay всегда пытается добавить в пакет инофрмацию. Однако, если дилна после добавления будет превышать MTU, то в режиме Trusted пакет будет передан дальше в неизменном виде, а в Untrusted отброшен.

Для отключения PPPoE relay необходимо сделать следующее:

Таблица 15.Выключение PPPoE Relay

Шаг	Действие	Описание действия
1.	**als(aal5)[encap ethernet]# no pppoeRelay**	Данная команда выключает PPPoE Relay на интерфейсе.
2.		

Просмотр конфигурации и статистики.

Для того, чтобы просмотреть конфигурацию DHCP Relay, необходимо находясь в encapsulation ethernet или в encapsulation ethernet->pppoeRelay набрать команду:

show config

Например,

```
als(aal5)[encap ethernet]# dhcpRelay
als(encap eth)[pppoe]# show config
            pppoeRelay
                untrusted
                mtu 1484
```

Для просмотра статистики можно использовать следующие варианты:

Пример 1:

```
als(encap eth)[pppoe]# show
        DHCP statistic:
        fwd: 0 inserted: 0
        discarded: 0 oversize: 0
```

Пример 2:

```
als(aal5)[encap ethernet]# show pppoe
        PPPoE statistic:
        fwd: 0 inserted: 0
        discarded: 0 oversize: 0
```

Оба варианта показывают одно и то же. Если PPPoE Relay выключен, то выдается соответствующее предупреждение.

Таблица 16: Специфичные параметры статистики для UPLINK порта

Название поля	Описание
fwd	Количество распознанных как PPPoE и пересланных в сторону провайдера
inserted	Количество пакетов, в которые была включена информация
discarded	Количество отклоненных пакетов (по причине untrusted)
oversize	Количество пакетов с размером, превышающим MTU

Настройка и использование ACL.

ACL (Access control list) – средство, которое позволяет провайдеру ограничить список

абонент устройств для заданного порта, задав MAC их адреса. Если абонент подключит устройство с неизвестным MAC адресом, то оно не будет получать ответы из сети провайдера. Кроме того, с помощью ACL можно жестко привязать определенный MAC адрес за определенным портом. Если устройство будет подключено через другой порт, то работать оно там не будет.

Поскольку за доступ отвечает MAC адрес, то и сама настройка находиться в encapsulation ethernet. Для того, чтобы запретить остальным MAC адресам работать на данном encapsulation ethernet, необходимо отключить на нем LEARNING. В случае привязки устройства к порту без ограничения доступа остальных устройств этого делать не надо. Для того чтобы отключить LEARNING необходимо выполнить следующее:

```
als(aal5)[encap ethernet]# no learning
```

После этого можно начать добавлять MAC адреса. Для этого каждому MAC адресу необходимо создать соответствующий MAC профиль.

Например, создадим профиль для STB с его MAC:

```
als(aal5)[encap ethernet]# profile mac stb_224498 addr 00:11:22:33:22:33
```

Затем добавим его в ACL в нужный encapsulation ethernet

```
als(aal5)[encap ethernet]# acl stb_224498
```

Если encapsulation ethernet использует VLAN, то в необходимо ACL необходимо указать идентификатор VLAN в сети провайдера. Т.е. если encapsulation ethernet настроено как use vmod tagged vlan_name, то в ACL тоже нужно использовать VLAN vlan_name. Если здесь указать другой VLAN или принудительно отключить его использование, то в *downstream* направлении будут приходить пакеты из указанного VLAN или вообще без меток.

Примечание:
Нельзя использовать один и тот же MAC на ACL более одного раза. При такой конфигурации будет работать первый ACL, а остальные не будут. Это особенность аппаратной реализации.

Конфигурация с использованием multiple PVC.

Итак, мы рассмотрели основные конфигурации и их расширения для одной линии. Теперь можно попробовать создать схему, при которой можно будет использовать несколько PVC. Отобразим ее на схеме.

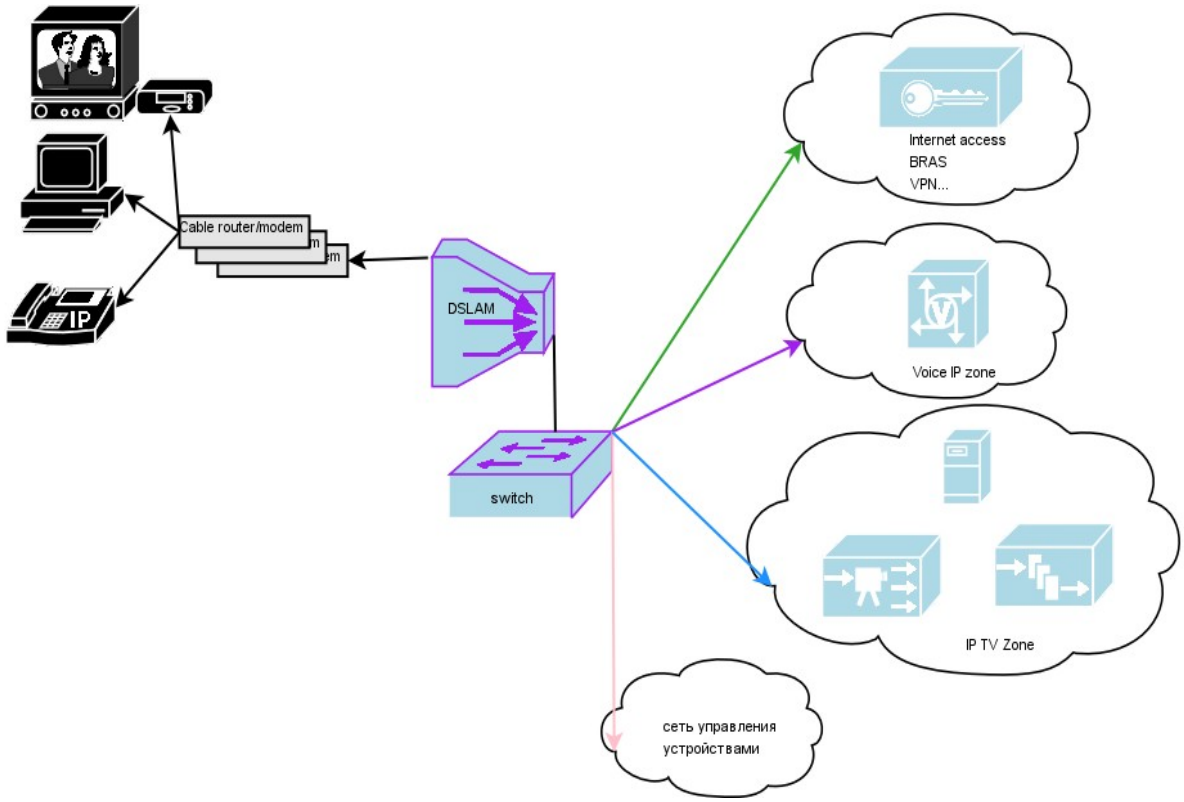

Иллюстрация 13.: Схема с использованием multiple PVC

На данной схеме изображено примерное использование нескольких PVC. Каждому PVC соответствует свой VLAN. У абонента стоит модем с поддержкой multiple PVC и настроенным routing`ом. В зависимости от destanation IP сети оборудование попадает через routing. Таким образом пользователь может иметь доступ сразу к нескольким сетям. Для провайдера это может быть удобным, т.к. посредством multiple PVC можно открывать и закрывать доступ к отдельным сетям. Кроме того, для каждого PVC можно задать разный приоритет, на уровне ATM.

Эта конфигурация получается при помощи создания нескольких профилей PVC и нескольким профилей VLAN. Типовая конфигурация на восемь портов находится в приложении.

1. ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ

Изм.	Стр.	Номер документа	Подпись