

Дипломник: Мазалов Е. С.

Руководитель: Диденко Е. И.

Пакетная коммутация

«Системы коммутации региона»

Лабораторная работа №2

Руководство по использованию программы

Данная программа разработана для изучения и улучшения понимания принципов пакетной коммутации на основе технологии ATM.

Просмотр действия осуществляется одним щелчком **ЛЕВОЙ КЛАВИШИ МЫШИ**, или клавишей **ПРОБЕЛ**.

(Внимание!!! Перед тем как нажать клавишу дождитесь полного завершения действия).

Для возврата на предыдущее действие используйте клавишу «Page Up».

Для завершения просмотра презентации используйте клавишу «Esc».

Выделенные надписи типа **Технология ATM** являются гиперссылками и при одинарном клике левой кнопкой мыши осуществляют переход на соответствующий слайд.

Для перехода на последующий слайд используйте управляющую кнопку

-



Для возврата на последний просмотренный слайд используйте управляющую кнопку

-



Для перехода на первый слайд используйте управляющую кнопку

-



Для перехода на слайд содержания используйте кнопку

-

содержание

Для перехода к словарю терминов используйте кнопку

-



НАЧАЛО ОБУЧЕНИЯ



Содержание

1 Основы технологии пакетной коммутации

- 1.1 Основные понятия и определения
- 1.2 Принципы пакетной коммутации

2 Технология ATM

- 2.1 Функциональные возможности ATM
- 2.2 Компоненты заголовка

3 Структура стека протоколов ATM

- 3.1 Физический уровень
 - 3.1.1 Виды среды передачи
 - 3.1.2 TDM (схема)
- 3.2 Уровень ATM
 - 3.2.1 Управление трафиком
 - 3.2.2 Управление перегрузками
- 3.3 Уровень адаптации ATM
 - 3.3.1 Протоколы AAL
 - 3.3.2 Уровень AAL0
 - 3.3.3 Уровень AAL1
 - 3.3.4 Уровень AAL2
 - 3.3.5 Уровень AAL3/4
 - 3.3.6 Уровень AAL5

4 Сигнализация на сети ATM

- 4.1 Адрес DCC AESA
- 4.2 Адрес ICD AESA
- 4.3 Адрес E.164 AESA
- 4.4 Установление соединения

5 Структура коммутаторов ATM

- 5.1 Принципы построения коммутаторов
 - 5.1.1 Коммутатор с разделяемой памятью
 - 5.1.2 Коммутатор с общей средой
 - 5.1.3 Коммутатор с полностью связанной топологией
 - 5.1.4 Коммутатор с пространственным разделением
 - 5.1.5 Матричный ATM - коммутатор
 - 5.1.6 Баньяновидный ATM - коммутатор
 - 5.1.7 Организация буферов

6 Практическая часть

7 Словарь терминов



Основные понятия и определения технологии коммутации пакетов

Коммутация - это процесс *установления соединения* между определенными входом и выходом системы, *поддержания* его на время передачи информации пользователя и последующего *рассоединения*. Коммутация называется *цифровой*, если осуществляется с помощью операций с цифровыми сигналами, переносящими информацию пользователя, без их превращения в аналоговую форму. Различают два основных вида цифровой коммутации: *коммутация каналов* и *коммутация сообщений*. Если сообщения пользователя коммутируются *сегментами одинаковой длины*, то имеет место **коммутация пакетов**.

При цифровой коммутации каналов (channel switching) сначала создается сквозное соединение между входом и выходом системы, а затем по этому соединению в реальном времени происходит обмен информацией пользователей. Вызовы, поступающие при занятости всех путей соединения, как правило, теряются. Обмен в реальном времени определяет основную область использования коммутации каналов - передачу речи. Недостаток систем с этим видом коммутации - относительно плохое использование каналов.

Коммутация пакетов (packet switching) отличается от коммутации каналов тем, что выполняется не в реальном времени, не требует сквозного соединения между входом и выходом системы, а избыточные сообщения не теряются, а запоминаются и передаются с задержкой. Соответственно в системе образуются так называемые *виртуальные соединения*, могут быть значительными время использования каналов и длительность доставки сообщений. Передаваемые сообщения разделяются на пакеты (сегменты) одинаковой длины и каждый пакет передается независимо, как только освобождается доступный канал связи.

Коммутации пакетов свойственны *асинхронный способ передачи* и предоставление канала только при необходимости передачи пакета. *Быстрая коммутация пакетов* (Fast Packet Switching, FPS) пригодна для передачи любых сообщений, в частности и речевых, в реальном времени.



Повторитель (repeater) - устройство, которое функционирует на первом (физическом) уровне модели OSI. Его используют для физического соединения сегментов кабеля локальной сети с целью увеличения общей длины сети. Повторитель принимает сигнал с одного сегмента кабеля и побитно транслирует его в другой сегмент, восстанавливая при этом амплитуду и форму сигнала.

Концентратор (concentrator), или распределитель (hub) - это повторитель, который имеет несколько портов. Он позволяет объединять ряд сегментов сети, реализуя звездную топологию, которая упрощает диагностирование и присоединение PC к его портам.

Мост (bridge) - это интеллектуальное устройство, которое объединяет сегменты ЛВС и выполняет фильтрацию кадров между сегментами с целью уменьшения в них нагрузки. Мост работает на канальном уровне модели OSI и является прозрачным для протоколов высших уровней.

Коммутатор (switch) - это многопортовый мост, имеющий механизм коммутации, позволяющий сегментировать сети, а также выделять PC определенную пропускную способность в сети. Кроме того, коммутаторы позволяют создавать логические сети, т.е. поддерживать работу виртуальных сетей, использование которых в последнее время возрастает. Коммутаторы, как и мосты, работают на канальном уровне модели OSI.

Маршрутизатор (router) - это устройство, которое определяет оптимальный путь передачи пакетов на основании информации сетевого уровня. Маршрутизатор функционирует на сетевом уровне модели OSI. Маршрутизатор обрабатывает каждый пакет и посылает его получателю, поэтому его производительность должна быть значительной, чтобы функционирование сети было эффективным.

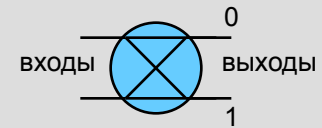
Принципы коммутации пакетов

Быстрая коммутация пакетов пригодна для обслуживания разноскоростных потоков, переносящих любую информацию - от данных до интерактивных телефонных разговоров и качественных программ телевидения. Это позволяет пользователю в каждом сеансе связи получать полосу частот, соответствующую текущей потребности, и оплачивать только реально использованные ресурсы сети. Определение «быстрая» означает, что обеспечиваются скорости передачи более 100 Мбит/с, задержки передачи (без учета времени пакетизирования информации) на уровне нескольких миллисекунд и соответственно быстрое установление виртуальных соединений.

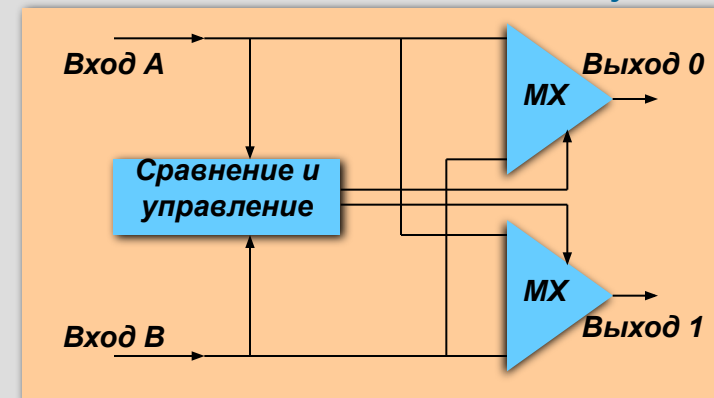
Особенности пакетной коммутации вызваны необходимостью самостоятельной маршрутизации каждого пакета с помощью адресной информации, имеющейся в его заголовке. Блок пакетной коммутации (БПК) можно построить из отдельных коммутационных элементов (КЭ) на n входов и столько же выходов ($n \times n$). В простейшем случае это может быть КЭ 2×2 .

Функции КЭ 2×2 заключаются в направлении пакета, поступившего по любому из двух входов на один из двух выходов. Для этого КЭ анализирует заголовок пакета и образует нужное соединение. В варианте 2×2 при этом достаточно информации одного двоичного разряда - в зависимости от его значения (0 или 1) пакет передается на выход 0 или выход 1.

Возможны и дополнительные функции КЭ: анализ нескольких разрядов, удаление проанализированного разряда, подсчет пакетов, буферизация пакетов, которые от разных входов направляются к одному выходу.



Основной элемент пакетной коммутации



Пример реализации КЭ 2×2



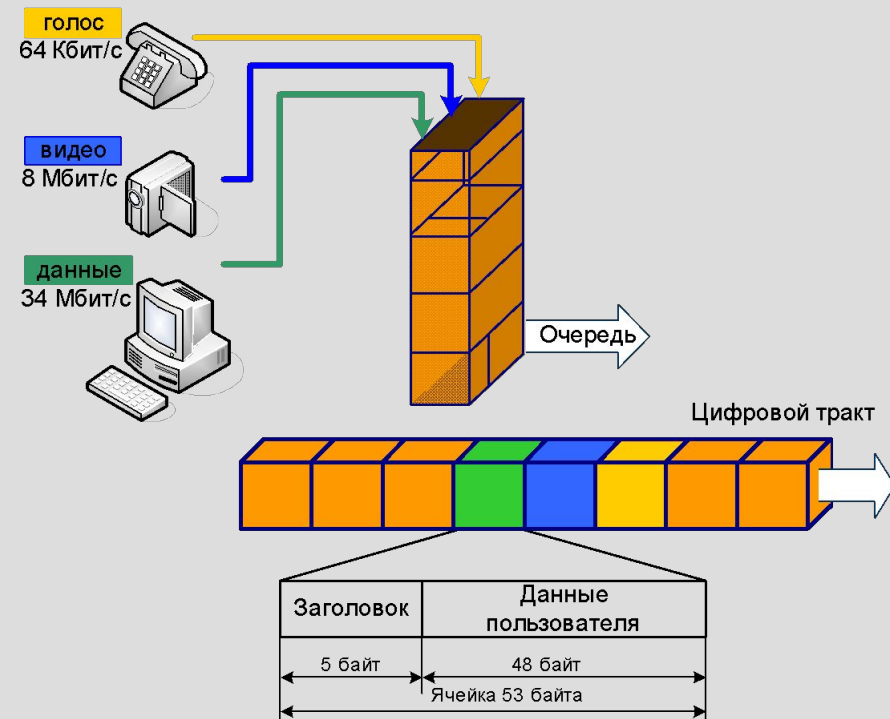
Технология ATM

Режим асинхронной передачи данных (Asynchronous Transfer Mode) – это технология коммутации ячеек, позволяющая с высокой скоростью коммутировать фреймы фиксированной длины в сетевой инфраструктуре при настройке последней с параметрами QoS (Quality of Service – качество обслуживания). Технология ATM позволяет передавать оцифрованный любым способом трафик с обеспечением приоритетов, что достигается использованием задержек при передаче ячеек и других специальных средств. Это технология передачи информации, при которой по сети одновременно передаются данные, аудио- и видеосигналы, а также соответствующие технические средства одноименной сети.

Корни ATM происходят из технологии B-ISDN (широкополосная цифровая сеть с комплексным обслуживанием).

Каждая ячейка ATM имеет заголовок и полезное поле. Заголовок (5 байт) содержит информацию для маршрутизации ячеек в коммутаторах сети ATM. Полезное поле (48 байт) включает пользовательские данные.

Передача данных с применением длинного полезного поля приводит к наибольшей пропускной способности сети и минимуму ячеек в сообщении (предполагается, что передаются достаточно длинные файлы). Чтобы достигнуть малого времени задержки речи, передача речевой информации должна производиться короткими ячейками. На телефонной сети время задержки при передаче между абонентами не должно превышать 250 – 300 мс.



Асинхронный метод переноса характерен следующими основными особенностями:

- отсутствием защиты от ошибок и управления потоком данных на уровне звена;
- ориентацией на соединение;
- ограниченным количеством функций, которые несет заголовок пакета ATM;
- относительно небольшой длиной информационной части ячейки.

Основные преимущества технологии ATM:

- динамическое управление полосой пропускания каналов связи;
- предоставление услуги качества QoS для разных типов трафика;
- возможности резервирования каналов связи и оборудования;
- возможность интегрирования самых различных типов трафика, включая голос, данные, видео;
- возможность экономии полосы пропускания за счет специальных технологий обработки голосового

трафика;

- возможность эмуляции «прозрачных» каналов связи;
- совместимость с технологией FR и предоставление сервисов пользователям FR;
- используя технологию MPLS (Tag Switching), сервис-провайдер, имеющий опорную сеть ATM, может

динамически коммутировать трафик IP по опорной сети ATM в реальном масштабе времени. При этом появляется возможность предоставлять необходимый уровень качества обслуживания QoS, соотнося уровни приоритезации IP и ATM.

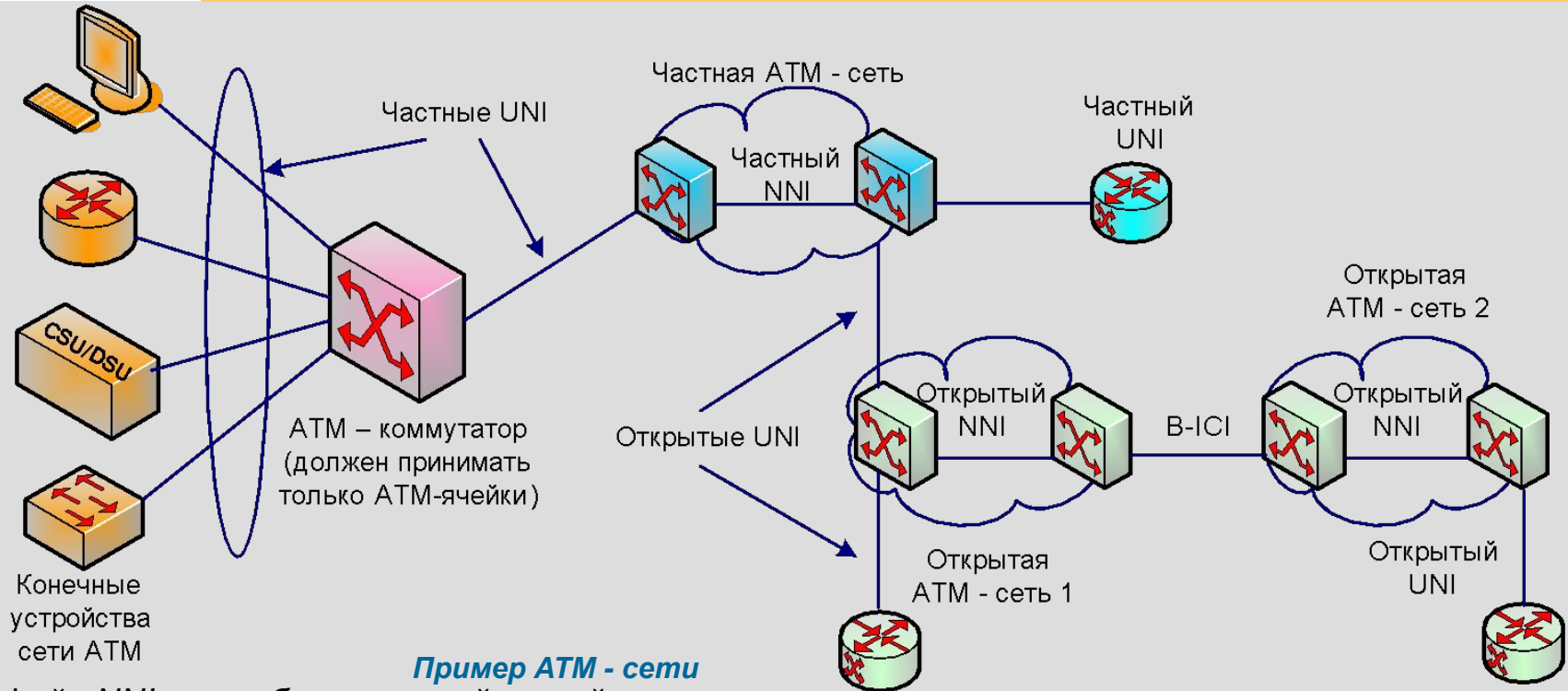
Основные недостатки технологии ATM:

- сложность технологии;
- относительно высокая цена оборудования;
- недостаточная совместимость оборудования от разных производителей;
- в специфических задачах (например, при частой передаче небольших объемов трафика)

применение технологии ATM может привести к неоправданно большим задержкам при установлении соединений и к довольно высокому проценту служебной информации, загружающей канал связи.



Функциональные возможности ATM



Пример ATM - сети

Интерфейс **NNI** может быть опорной точкой только между коммутаторами в пределах одного учреждения, а интерфейс **UNI** служит для подключения конечных ATM-устройств, таких, как узлы, маршрутизаторы и коммутаторы. В-ICL используется для соединения устройств, оставляя открытым ATM-коммутатором. ATM-интерфейсы имеют три основных типа: открытый NNI, частный NNI и частный UNI. Между двумя открытыми коммутаторами поддерживают три основных типа интерфейсов:

- Открытый интерфейс (Open Interface)** для соединения двух ATM-коммутаторов в пределах одного учреждения.
 - «пользователь - сеть» (User Network Interface - UNI);
- Частный интерфейс (Private Interface)** описывает соединение между двумя ATM-коммутаторами в пределах одного частного учреждения.
 - «межсетевой» (Network-to-Network Interface - NNI);
- Открытый интерфейс (Open Interface)** находится между конечным устройством и коммутатором в пределах одного частного учреждения.
 - широкополосный интерфейс для соединения устройств операторов связи (Broadband Inter-Carrier Interface - B-ICI).

Компоненты заголовка АТМ



Ячейка АТМ NNI



Ячейка АТМ UNI

Ячейки АТМ имеют длину 53 байта. Из них 5 – это заголовок, 48 – полезное поле.

Поля в заголовках имеют вид:

- общее управление потоками (*Generic Flow Control* - **GFC**) – содержит 4 бита в интерфейсе UNI, а в интерфейсе NNI отсутствует;
- идентификатор виртуального пути (*Virtual Path Identifier* - **VPI**) – содержит 8 бит в интерфейсе UNI и 12 бит в интерфейсе NNI;
- идентификатор виртуального канала (*Virtual Channel Identifier* - **VCI**) – содержит 16 бит;
- идентификатор типа полезной нагрузки (*Payload Type Identifier* - **PTI**) – содержит 3 бита;
- приоритет отбрасывания ячеек (*Cell Loss Priority* - **CLP**) – содержит 1 бит;
- алгоритм контроля за ошибками в заголовке (*Header Error Control* - **HEC**) – содержит 8 бит.



GFC – поле общего управления потоками

Поле GFC имеется только в формате заголовка UNI. Функция, выполняемая этим полем, не нашла широкого распространения и существенно зависит от механизма управления потоками. Поле GFC является механизмом различения многочисленных конечных точек в интерфейсе для указания приоритета трафика в заданном интерфейсе. Обычно поле GFC представляет собой набор из четырех двоичных нулей, но может быть использована альтернативная частная реализация: в таком случае оборудование на каждом конце канала передачи данных должно согласовываться друг с другом с целью возможности его использования.

Первоначально поле GFC предназначалось для контроля за общим трафиком, передаваемым по сети ATM. В заголовке каждой ячейки поля GFC передаются индикаторы контроля за величиной трафика. Информация GFC действует только на одном звене сети.

В отношении GFC на интерфейсе UNI определены два режима работы. Первый – *неконтролируемого доступа*. В нем во всех битах GFC записаны нули. Они игнорируются на принимающей стороне. Этот режим был определен для ранних применений ATM, когда не был введен механизм действия GFC. Второй режим – *контролируемого доступа*. В нем хосты сети ATM передают трафик в зависимости от информации, записанной в поле GFC принимаемой ячейки. В настоящее время поле GFC может не использоваться, так как появились более эффективные и надежные методы контроля величины трафика.

VPI – поле идентификатора виртуального пути

Поле VPI является одной из двух частей структуры данных, которое совместно с полем VCI создает локально важную пару значений, которую можно рассматривать как единое целое в пределах отдельной ячейки. Пара VPI/VCI является логическим представлением полного виртуального канала (Virtual Channel - VC), чем-то вроде маршрута на карте для указания всего пути к удаленной конечной точке.

В формате заголовка интерфейса NNI поле VPI имеет больший размер из-за отсутствия поля GFC. Увеличение поля VPI было сделано с целью увеличения количества виртуальных путей (Virtual Path -VP) в сети, в отличие от того числа, которое необходимо в локальном абонентском канале.

Независимо от того, используется ли формат NNI в пределах сети, входной коммутатор всегда терминирует интерфейс UNI, и устанавливает в поле GFC все нули. Такой подход делает поле GFC локально значимым инструментом при его использовании.

UNI-сегмент АТМ-сети может обеспечить $2^8 = 256$ уникальных виртуальных путей, а интерфейс NNI – $2^{12} = 4096$ уникальных путей.



VCI – поле идентификатора виртуального канала

Поле VCI является второй составной частью пары локально значимых идентификаторов.



Теоретически каждый интерфейс между конечной точкой и коммутатором или между двумя коммутаторами может обеспечить $2^{16} = 65536$ возможных виртуальных соединений на один виртуальный маршрут (Virtual Path -VP). Когда происходит распределение по возможным виртуальным маршрутам, теоретически в итоге может быть создано 16 777 216 виртуальных каналов (Virtual Channel - VC) UNI и 268 435 456 виртуальных каналов NNI.

Идентификаторы VCI с номерами от 0 до 31 включительно никогда не должны применяться для пользовательского трафика, особенно идентификатор VPI с номером 0, пользовательские данные должны иметь VCI со значениями, не меньшими 32 (зарезервированные значения VCI перечислены в [таблице 1](#)).

На [рисунке](#) проиллюстрирован процесс коммутации виртуального маршрута и виртуального канала.

Преимуществами такой схемы являются:

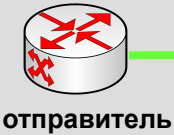
- увеличение эффективности при VP-коммутации по сравнению с коммутацией каналов, т.к. коммутатор не анализирует, не изменяет идентификаторы VCI, предпочитая сохранять эту часть заголовка ячейки в неизменном виде, и оставляя ячейку в том же виртуальном канале, даже если идентификация маршрута VP требует изменений;
- коммутатор будет нуждаться в хранении в своей таблице трансляции только VPI-информации, экономя т.о. память и, значит, ускорит процесс поиска нужной записи.

Таблица1 – зарезервированные значения VCI

VCI	Функция
0	Незанятые (не назначенные) ячейки
1	Метасигнализация (позволяет установить сигнальные каналы)
2	Общая широкополосная передача сигналов (фактически не используется)
3	Служба поддержки работоспособности, администрирования и обслуживания сегмента виртуальных маршрутов VP (Operation, Administration and Maintenance - OAM)
4	Служба OAM сквозных маршрутов VP
5	Передача сигналов от граничного устройства к его коммутатору доступа (UNI)
6	Управление ресурсами маршрута VP (Resource Management - RM)
15	Простой протокол для передачи сигналов по ATM-сети FORE (FORE's Simple Protocol for ATM Network Signaling - SPANS)
16	Промежуточный интерфейс локального управления (Interim Local Management Interface - ILMI) для обмена параметрами соединений
17	Эмуляция локальной сети (LAN-эмуляция - LANE)
18	Сигналы интерфейса между частными сетями (Private Network-Network Interface - PNNI) для ATM – маршрутизации в частных сетях

Термин метасигнализация (metasignaling) дословно означает «информация о передаче служебных сигналов», т.е. информация о самой сигнализации.

Коммутирующая таблица трансляции



VCI = 100

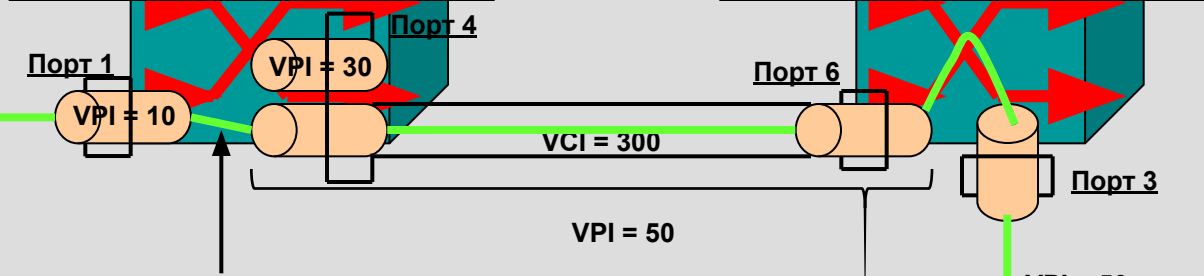


VCI = 200

Коммутация виртуальных путей и виртуальных каналов

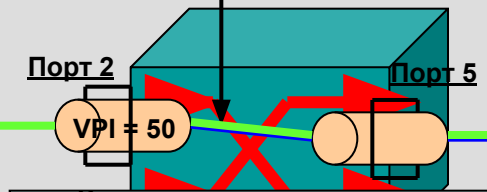
Коммутация виртуальных каналов	
В порт/VPI/VCI	Из порт/VPI/VCI
1/10/100	4/50/300

А



Внутренняя коммутация (и, возможно, изменяется) VPI и VCI в VC-коммутации

Может коммутироваться только VPI (и, возможно, изменяется) в VP-коммутации



Коммутация виртуальных каналов	
В порт/VPI/VCI	Из порт/VPI/VCI
5/20/100	2/50/200

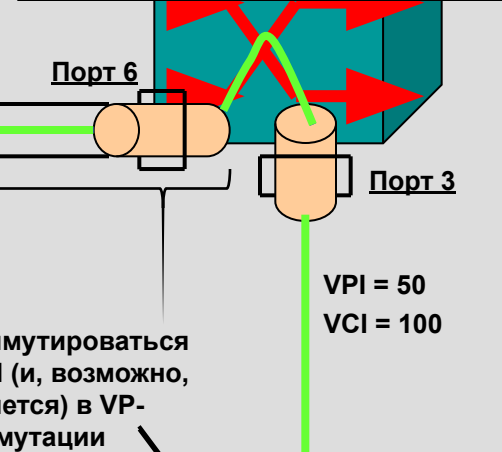
Г

VPI = 20
VCI = 100

VCI остается тем же самым

Коммутация виртуальных каналов	
В порт/VPI/VCI	Из порт/VPI/VCI
3/30/300	3/50/100

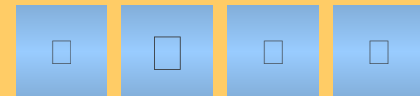
Б



Коммутация виртуальных путей	
В порт/VPI	Из порт/VPI
8/50	7/20

В

Каждая ячейка в виртуальном канале с идентификатором VPI 50, поступающая на физический порт 8. При коммутации виртуального маршрута, полный набор виртуальных каналов, присутствующих во входном виртуальном маршруте, должен быть скомутирован полностью, без каких-либо изменений идентификаторов VCI в исходящем виртуальном маршруте, не добавляя и не пропуская идентификаторы VCI в процессе коммутации. При этом идентификатор VCI должен оставаться неизменным (VCI 300) на выходе коммутатора.



PTI – поле идентификатора типа полезной нагрузки

Поле PTI (Payload Type Identifier) имеет размер 3 бита, которые действуют как индивидуальные сигналы. Основная цель распознавания типа полезного поля состоит в том, чтобы различать ячейки, несущие пользовательские и другие данные.

Значения поля PTI

VCI	Функция
000	Ячейка пользователя. Отсутствует заблаговременная индикация перегрузки (Explicit Forward Congestion Indication - EFCI), индикация уровня адаптации ATM (AAL-indicate) содержит значение false (ложь)
001	Ячейка пользователя. EFCI отсутствует, поле AAL-indicate содержит значение true (истина)
010	Ячейка пользователя. EFCI присутствует, поле AAL-indicate содержит значение false (ложь)
011	Ячейка пользователя. EFCI присутствует, поле AAL-indicate содержит значение true (истина)
100	Ячейка OAM: сегмент маршрута (Segment)
101	Ячейка OAM: сквозной маршрут (End-to-End)
110	Ячейка управления ресурсами (Resource Management Cell – RM-cell): контроль перегрузки для доступной скорости передачи (Available Bit Rate - ABR)
111	Зарезервировано для будущего применения

Первый бит отмечает, пользовательская ли это ячейка или нет. В случае передачи пользовательских ячеек второй бит используется как индикатор явной перегрузки в прямом направлении (EFCI). Он служит для фиксации возникшей перегрузки на пути от источника к получателю информации. Третий – бит индикатора типа блока данных услуги (SDU). Он может быть использован протоколом высокого уровня. Например, для индикации последней ячейки в пользовательском сообщении. Это необходимо для правильной сборки этого сообщения.

Если ячейки не пользовательские, то во втором бите указывается на наличие в ячейке информации (Административное и эксплуатационное обслуживание), связанной с виртуальным каналом (поток F5). При наличии отметки о принадлежности ячейки к потоку F5 OAM в третьем бите отмечается привязка информации системы **OAM** к сегменту сети или к соединению между конечными ее точками.



Формат ячейки OAM и его использование

Заголовок ячейки	Тип OAM	Тип функции	Поле зависящее от функции	Зарезервировано на будущее	Код для обнаружения ошибок (CRC-10)
5 байтов	4 бита	4 бита	45 байтов	6 битов	10 битов

Тип OAM-ячейки	Величина	Тип функции	Значение	Величина
Управление ошибками	0001	Отказ (AIS)	Индикация дефектов в прямом направлении	0000
		Дефект (RDI)	Индикация дефектов в обратном направлении	0001
		Контроль непрерывности	Непрерывный контроль соединений	0100
		Закольцовывание ячейки	Контрольное соединение перед запуском в систему	1000
Управление функционированием	0010	Контроль в прямом направлении	Оценка качества диалога	0000
		Контроль в обратном направлении	Оценка выполнения индикации в обратном направлении	0001
		Контроль и оповещение		0010
Активация/Деактивация	1000	Контроль производит-ти	Контроль выполнения активации и деактивации и непрерывный контроль	0000
		Контроль непрерывности		0001

Формат ячейки OAM и его использование



CLP – поле приоритета отбрасывания ячеек

Бит CLP используется для индикации приоритета потерь ячеек в оборудовании пользователя и выборочного их удаления сетевым оборудованием ATM. Если сетевое устройство определит, что отбрасывание ячейки необходимо для предотвращения катастрофического события, результатом которого будет перегрузка, ячейка со значением в поле CLP, равным единице, подлежит отбрасыванию раньше, чем ячейки с битом CLP, имеющим значение нуль. Происходит маркировка (tagging), т.е. присвоение CLP-биту значения 1 для указания ячеек, которые в большей степени подвержены отбрасыванию, чем ячейки с битом CLP, равным 0 (функция маркировки выполняется на уровне адаптации ATM (AAL)).



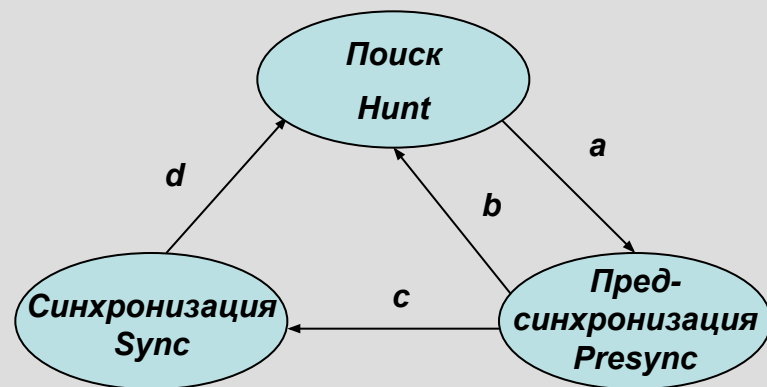
HEC – поле алгоритма контроля ошибок в заголовке

Поле HEC является многоцелевой структурой данных. Оно предназначено для обнаружения и исправления битовых ошибок заголовка, а также для использования в качестве ориентира для обнаружения границы ячеек и ее синхронизации в тех средах, которые не имеют своих собственных методов синхронизации. Поле HEC представляет собой механизм контроля с помощью циклического избыточного кода (CRC). Поле HEC вычисляется только на основании данных четырех октетов заголовка ячейки, но не какого-либо участка 48-октетной полезной нагрузки.

На принимающей стороне при выделении ячеек могут быть следующие состояния:

- **синхронизации (Sync State)**. Приемная сторона находится в этом состоянии пока информация в поле HEC указывает на отсутствие битовых ошибок. Если это значение за короткий период времени превышает некоторую переменную ALPHA (по умолчанию ALPHA = 7), нарушается синхронизация ячеек и принимающая сторона переходит в состояние поиска;

- **предсинхронизации (Presync State)**. Приемная сторона некоторое время увеличивает количество проверенных полей HEC. Если за период времени обнаружена битовая ошибка, принимающая сторона возвращается в состояние поиска. В противном случае, когда количество значащих полей HEC (используется переменная, названная DELTA, по умолчанию DELTA = 6) принимающая сторона переходит в состояние синхронизации.



Переходы состояний:

- (a) Первое обнаружение границы ячейки
- (b) Одно неправильное поле HEC
- (c) DELTA правильные поля HEC
- (d) ALPHA неправильные поля HEC

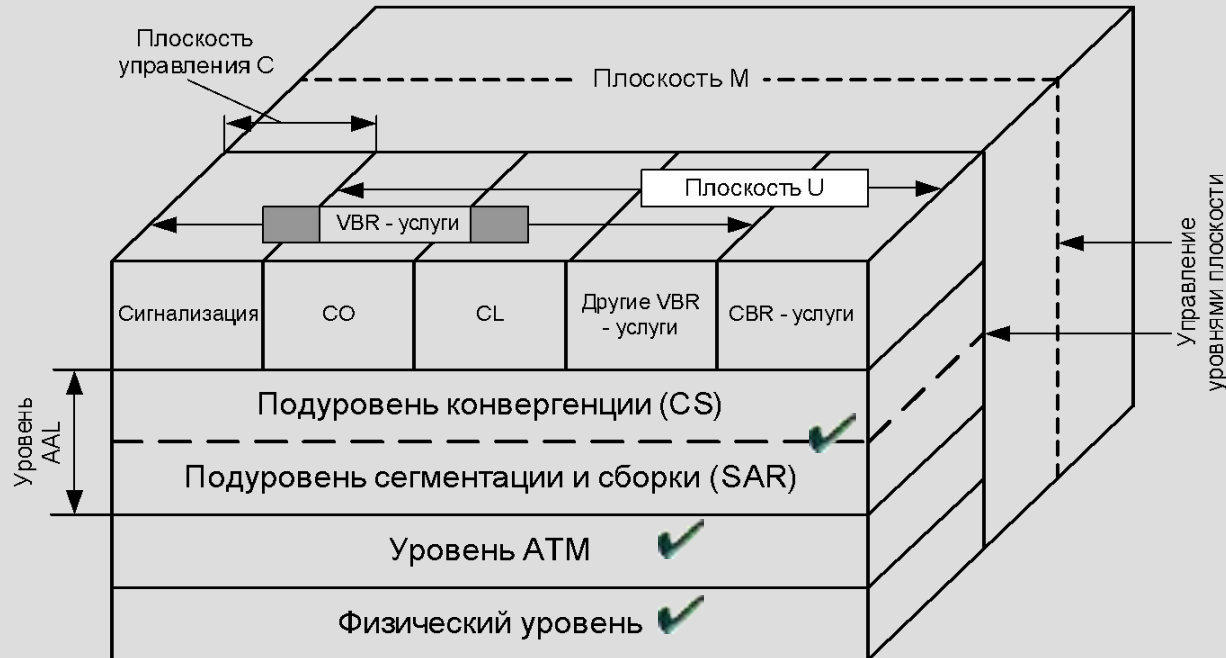
Процесс синхронизации при использовании поля HEC



Структура стека протоколов ATM

Технология ATM наилучшим образом подходит для построения широкополосных цифровых сетей с интеграцией служб (Broadband Integrated Services Digital Network, B-ISDN) и предоставления всевозможных услуг.

Стандартная модель протокола B-ISDN включает в себя несколько плоскостей. Плоскость U обеспечивает передачу пользовательской информации и связанную с ней информацию контроля потока, ошибок и восстановления данных. Плоскость управления C предназначена для передачи сигнальной информации, включающей функции установления и завершения соединения. Плоскость U и C имеют трехуровневую архитектуру протоколов ([таблица 2](#)). Перед изучением уровней ATM советуем рассмотреть более подробно !!! **системную архитектуру ATM**



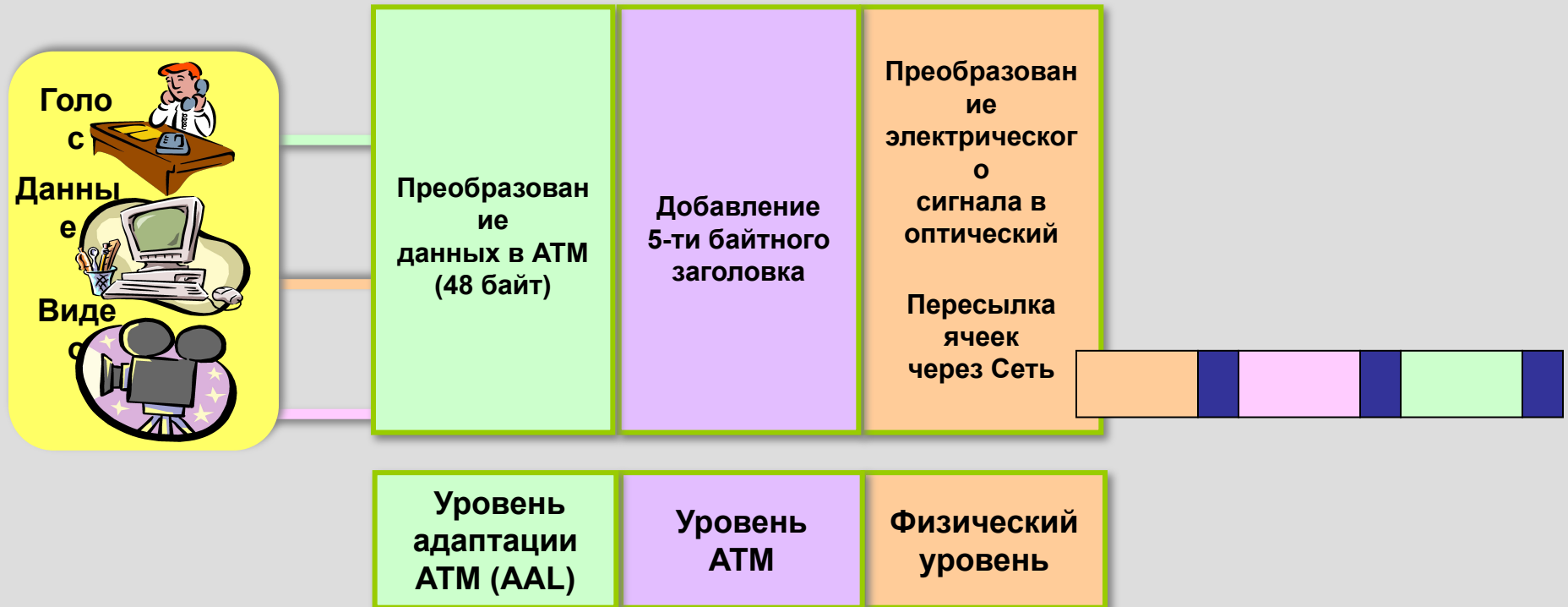
Стандартная модель протоколов B-ISDN

УРОВЕНЬ	ПОДУРОВЕНЬ	ФУНКЦИЯ
Адаптации АТМ	Конвергенции	Конвергенция с услугами
	Сегментации и сборки	Сегментация и сборка конвергированных блоков PDU
АТМ		Управление общим потоком
		Формирование заголовка ячейки
		Преобразование идентификаторов VPI/VCI ячеек
		Выделение ячеек
		Мультиплексирование ячеек
Физический	Конвергенции с системой передачи	Функции OAM
		Разделение ячеек по скорости
		Формирование поля контроля ошибок в заголовке ячейки
		Адаптация передаваемых кадров в системе передачи
		Формирование передаваемых кадров
	Физической среды передачи	Битовая синхронизация
		Передача двоичного сигнала по физической среде

Таблица 2 – Трехуровневая архитектура протоколов плоскостей U и C



Системная архитектура АТМ



Системная архитектура взаимодействия уровней АТМ при передаче разного рода трафика выглядит следующим образом:

- поступающие данные на уровне адаптации АТМ преобразовываются в данные АТМ, т.е. формируется поле полезной (пользовательской) информации 48 байт;
- затем, на уровне АТМ происходит добавление 5-ти байтового заголовка маршрута ячейки;
- после чего, на физическом уровне происходит преобразование сигналов в линии и осуществляется добавление ячеек АТМ к общему трафику, и пересылка ячеек через Сеть.

Физический уровень

Физический уровень отвечает за передачу бит между смежными сетевыми устройствами, такими как оборудование пользователей и сетевые узлы. На физическом уровне определяются среда передачи, разъемные соединители, правила преобразования ячеек из одной среды передачи в другую, битовая синхронизация.

Физический уровень имеет два подуровня:

- нижний подуровень – физической среды (*PM*). Определяет синхронизацию бит, физические характеристики среды и разъемные соединители стыков. На подуровне *PM* обеспечиваются следующие функции:

- согласование с физической средой (согласовываются линейные коды, электрические и/или оптические характеристики, параметры электрооптических преобразователей);

- битовая синхронизация (сигнальная информация формируется для обеспечения битовой синхронизации между передатчиком и приемником на одном звене сети АТМ);

- «верхний» подуровень конвергенции передачи (*ТС*). Связан с независимыми от среды показателями, такими как коррекция ошибок, выделение ячеек и формирование кадров передачи. В подуровень *ТС* входят следующие функции:

- формирование и восстановление кадра передачи;
- адаптация принимаемого кадра (структура ячеек задается внутри полезного кадра);
- контроль за ошибками в заголовке ячейки (*HEC*);
- согласование скорости потока ячеек со скоростью передачи кадра по транспортной сети;
- вставка/изъятие бит (позволяет повысить эффективность их выделения);
- функции административные и технического обслуживания (передаются специальные ячейки которые несут информацию физического уровня (*Operation, Administration and Maintenance, OAM*), функции *OAM* включают мониторинг передачи, обнаружение и предоставление сведений об ошибках передачи;
- выделение ячеек.

Первоначально ИТУ-Т определил для сети АТМ скорости потока 155,52 и 622 Мбит/с. Это соответствует стандартам синхронной цифровой иерархии STM-1 и STM-4. на скорости 155,52 Мбит/с при организации доступа на физическом уровне допускается использовать кабели – коаксиальный длиной до 200 м или волоконно-оптический до 2 км. По желанию пользователей в современных требованиях к АТМ предусмотрена поддержка различных скоростей. Это позволяет применять также симметричный кабель.

Форум АТМФ и другие организации определили следующие опции для физического уровня: АТМ – интерфейсы UNI сети общего пользования с передачей по цифровым каналам DS-1, E1, E3, DS-3 или E4, UNI или NNI сети общего пользования с передачей через систему STM-1 или STM-4, NNI с передачей через систему STM-16, UNI со скоростью 25,92 Мбит/с типа «STS-1/S» ведомственной сети с передачей по неэкранированным витым парам кабеля типа UTP-3, UNI со скоростью 100 Мбит/с ведомственной сети с использованием многомодового волоконно-оптического кабеля или кабельных пар UTP-5 и т.д.

На интерфейсе "пользователь-сеть" рекомендовано использование интерфейса со скоростью 155,52 Мбит/с и 622,080 Мбит/с, представляющего собой непрерывный поток ячеек, каждая из которых содержит 53 октета. Максимальное расстояние между смежными ячейками физического уровня составляет 26 ячеек уровня АТМ. После 26-ти ячеек уровня АТМ обязательно вставляется ячейка физического уровня, несущая служебную информацию эксплуатации и технического обслуживания аналогичную информации, которая переносится в SDH секционной служебной нагрузкой (SOH) и служебной нагрузкой тракта (POH).

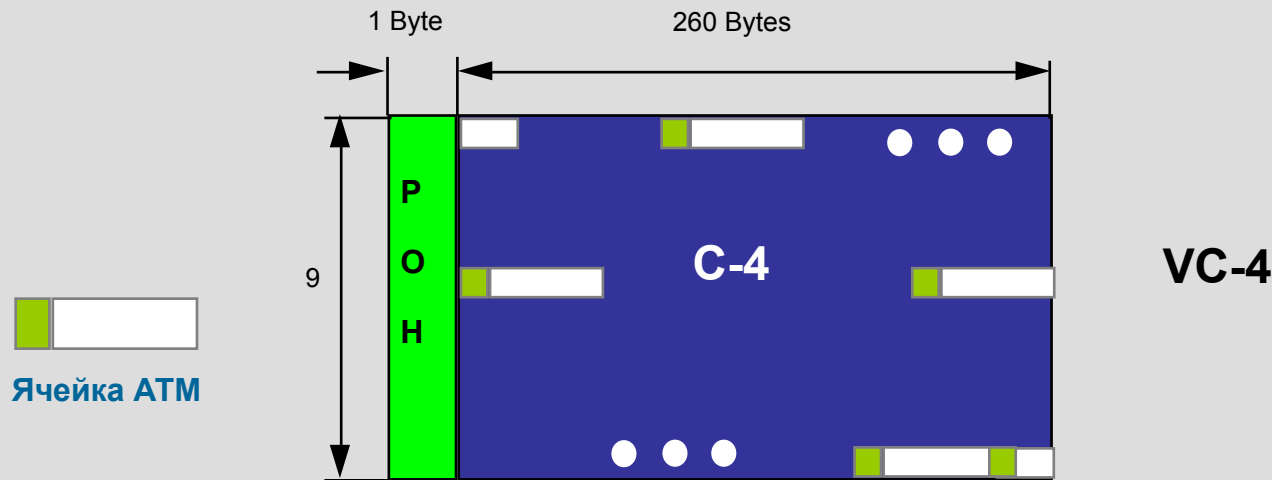
Формат передачи для SDH

В STM-1 передаются ячейки ATM, которые вставляются в виртуальный контейнер четвертого порядка (VC-4), переносящий в 9 рядах по 261 байту. Байты передаются слева направо, ряд за рядом.

Первый байт каждого ряда контейнера содержит служебную информацию (POH). Следовательно, один контейнер несет 2340 (260x9) байт ячеек ATM. Поскольку это число не кратно 53, в контейнере помещаются 44 ячейки и 8 байт 45-й ячейки.

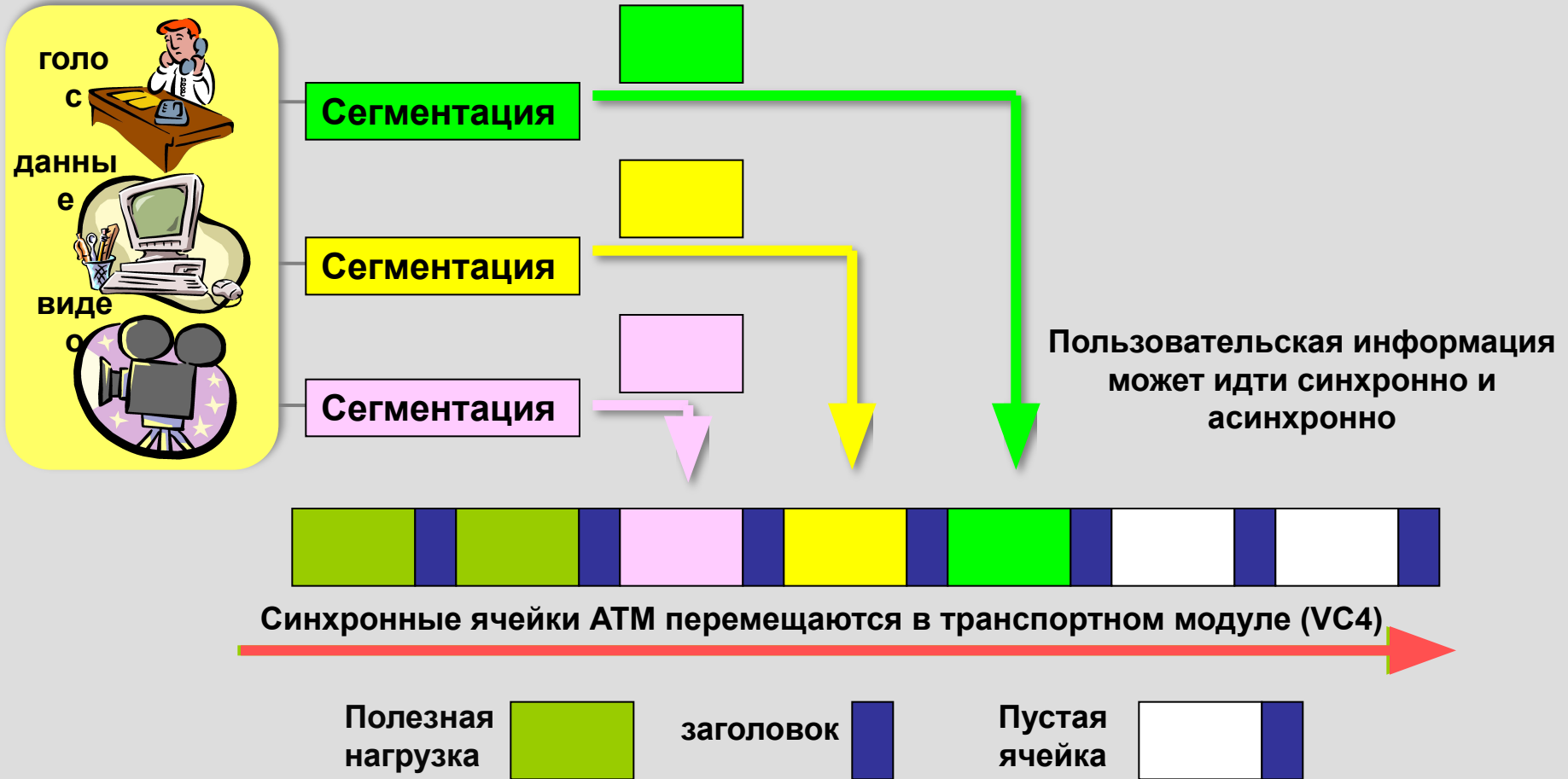
В следующем контейнере помещаются оставшиеся 45 байт 45-й ячейки и следующая новая партия ячеек. Контейнер вставляется в транспортный модуль. В нем первые 9 байт каждого ряда служат для передачи служебной информации. Она предназначена для работы регенераторной (RSOH) и мультиплексорной (MSOH) секции транспортной сети.

Административный указатель позволяет найти в виртуальном контейнере первый по счету байт. Транспортные модули собираются в кадры. Они передаются периодом 125 мкс и с учетом передачи служебной информации. Скорость передачи ячеек составляет 149,76 Мбит/с.



Структура STM - 1

Асинхронное временное мультиплексирование (TDM)



Код адреса метки в заголовке ячейки идентифицируется самой ячейкой

Сущность метода TDM

Уровень АТМ

Уровень АТМ отвечает, в первую очередь, за передачу ячеек от одного пункта сети к другому (например, от пользователя к коммутатору АТМ или между коммутаторами). На этом уровне происходят мультиплексирование и демultipлексирование ячеек, контроль за их потоком, а также их коммутация и маршрутизация. Формируется пятибайтовый заголовок. Он содержит информацию по маршрутизации ячеек, заданию приоритетов в их обслуживании и перегрузкам.

Уровень АТМ служит для передачи ячеек из узла в узел сети связи. На этом уровне коммутируются ячейки с установлением соединения. Уровень АТМ выполняет свои функции независимо от предоставляемых услуг на верхних уровнях, физической среды и скоростей потоков на интерфейсах UNI и NNI.

От узла, подключенного к сети, уровень АТМ принимает от соответствующего уровня ААL уже сегментированные 48-октетные элементы исходящих данных и присоединяет к ним пятиоктетные заголовки (добавляя и информацию в поля VCI и VPI). Именно уровень АТМ обеспечивает коммутацию (или ретрансляцию) ячеек в пределах сети и выполняет заключительную проверку, в результате чего конечная точка получает правильные ячейки (т.е. следит за правильностью внедрения ячеек в поток передаваемой информации). Мультиплексирование ячеек или одновременное их совместное использование несколькими виртуальными каналами физической полосы пропускания – это задача уровня АТМ.

На уровне АТМ формируется «полная» 53-байтная ячейка АТМ, у которой заголовок составляет 5 октет, а информационное поле – 48 байт. Передача ячеек происходит следующим образом:

- октеты передаются в порядке возрастания, начиная с первого;
- биты внутри октета передаются по убыванию, начиная с восьмого.

Для всех полей ячейки первый бит является наиболее значимым (MSB – Most Significant Bit).



Функции управления и перегрузками реализуются на уровне ATM. Уровень ATM определяет параметры управления трафиком, вне зависимости от особенностей подуровня AAL.

Параметры качества обслуживания QoS передаются в процессе установления соединений на сети ATM. Для этого используется сигнализация управления вызовом.

Параметры управления трафиком.

Пиковая скорость передачи ячейки (Peak Cell Rate, PCR).

Максимальный размер пачки данных (Burst Tolerance, BT) – измеряется числом ячеек.

Поддерживаемая скорость передачи ячейки (Sustained Cell Rate, SCR) – средняя, периодически измеряемая, скорость передачи ячейки.

Коэффициент потери ячеек (Cell Loss Ratio, CLR) – представляет собой процент потери ячеек на сети из-за битовых ошибок и перегрузок.

Задержка передачи ячеек (Cell Transfer Delay, CTD) – задержка ячейки при передаче через сеть из конца в конец.

Разброс задержки передачи ячейки (Cell Delay Variation, CDV) – указывает на пределы изменения величины CTD. Называется джиттером.

Минимальная скорость передачи ячейки (Minimum Cell Rate, MCR) – минимальная полоса пропускания, требуемая пользователем на сети ATM.

Постоянная битовая скорость (Constant Bit Rate, CBR). Соединения с услугой CBR используются для эмуляции коммутации каналов.

Переменная битовая скорость (Variable Bit Rate, VBR) – класс услуг, поддерживающий приложения VBR и обеспечивающий гарантии по потерям, задержке и полосе пропускания. Это услуги, предоставляемые в реальном масштабе времени – цифровое кабельное и интерактивное телевидение, теле- и видеоконференция.

Доступная битовая скорость (Available Bit Rate, ABR) – класс услуг, созданных для передачи данных (файлов и электронной почты).



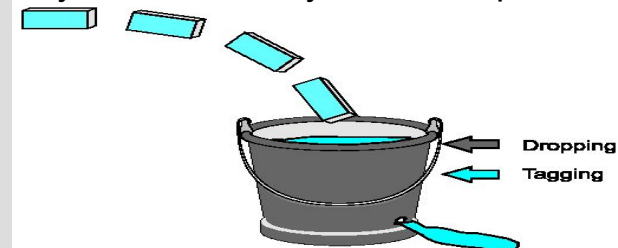
Управление перегрузками

Выделяются три способа управления перегрузками на сети ATM:

- первый способ избежать перегрузок на сети состоит в том, чтобы ее доступные ресурсы были бы адекватны требуемому качеству обслуживания QoS до установления новых соединений на сети ATM. Это - управление допуском соединения (Connection Admission Control, CAC). Оно возможно только при условии, когда установлены параметры QoS;

- второй способ управления перегрузками подразумевает применение алгоритма «дырявого ведра». Он также называется общим алгоритмом скорости передачи ячейки (Generic Cell Rate Algorithm, GCRA). Алгоритм преобразует поток входящих ячеек в виде пачек в более регулярный предсказуемый поток исходящих ячеек. Суть в том, что моделируется ситуация, в которой все поступающие ячейки попадают в «ведро» и выходят из «дыры» на дне с постоянной скоростью. Пока «ведро» не переполнено, указанный трафик считается комфортным, т.е. согласованным. Если ячейки поступают быстрее, чем выходят из «ведра», оно может переполниться. Ячейки переполнения – некомфортные и могут быть не приняты сетью, в результате чего, при дальнейшем возникновении перегрузки на сети, они могут быть удалены (используется отметка в поле CLP). Этот способ обеспечивает предсказуемость сети, с помощью чего мы сможем с большей вероятностью избежать перегрузки;

- третий способ управления перегрузками состоит в применении замкнутой петли обратной связи. В рассмотренном алгоритме GCRA предусматривается открытая петля обратной связи. Это позволяет обмениваться параметрами качества обслуживания QoS только при установлении соединения. Позже при появлении перегрузки параметры не могут динамично изменяться. В случае замкнутой петли обратной связи источники трафика извещаются динамично о перегрузках и могут соответствующим образом изменить передаваемый трафик.



Уровень адаптации АТМ

Уровень адаптации (AAL) предназначен для разделения протоколов верхних уровней и специфических средств сети АТМ. Этот уровень принимает конкретный пользовательский трафик от высших уровней и начинает процесс преобразования (адаптация) полученной информации в 48-октетные сегменты. Именно уровень AAL отвечает за отправку 48-октетных сегментов уровню АТМ, где затем происходит присоединение заголовка и дальнейшая обработка сегментов. AAL – это двусторонний интерфейс между АТМ и остальным миром, сосредоточенным на конечных точках (но не коммутаторах).

При этом каждый протокол уровня адаптации АТМ должен быть приспособлен к определенному классу трафика со своими специфическими характеристиками, определяющими уровень требований службы к временной и семантической прозрачности сети АТМ. Все функции уровня адаптации АТМ должны быть реализованы в терминальном оборудовании. Уровень адаптации АТМ принято в свою очередь делить на **два подуровня**:

- подуровень сегментации и сборки (SAR - Segmentation and Reassembly Sublayer);
- подуровень конвергенции или слияния (CS - Convergence Sublayer).

Основными функциями подуровня сегментации и сборки являются:

- на передающей стороне - сегментация протокольных блоков данных вышележащего уровня в 48 октетов информационного поля ячейки АТМ;
- на приемной стороне - сборка информационных полей ячеек в протокольный блок данных более высокого уровня.

В свою очередь подуровень конвергенции может делиться на две части:

- общую часть подуровня конвергенции (CPCS - Common Part Convergence Sublayer);
- служебно-ориентированный подуровень конвергенции (SSCS - Service-Specific-Convergence Sublayer).

Услуги, предоставляемые уровнем адаптации, подразделяются на четыре класса ([таблица](#)).

Существует несколько типов AAL



Класс услуг	А	В	С	Д
Режим соединения	С установлением соединения		Без установления соединения	
Битовая скорость	Постоянная		Переменная	
Временная зависимость между источником и получателем	Есть		Нет	
Тип уровня ААL	1	2	$\frac{3}{4}$; 5	$\frac{3}{4}$; 5

Класс А. Предоставляется услуга для пользователей, чувствительных к задержкам. Битовая скорость постоянна. Эта услуга с установлением соединения и эмуляцией услуг с коммутацией каналов. Сеть АТМ гарантирует требуемую полосу пропускания и минимальную задержку.

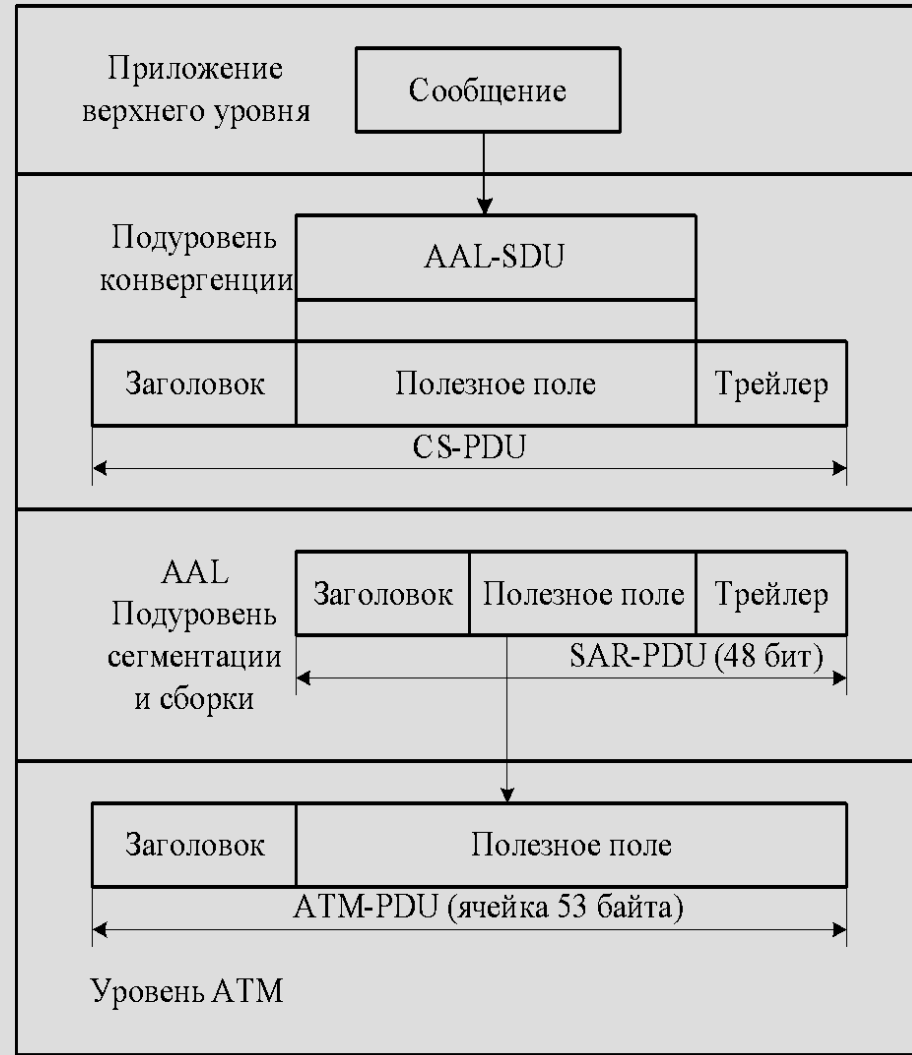
Класс В. Услуги предоставляются с установлением соединения для пользователей, чувствительных к задержкам. Разница между классами А и В в том, что источники класса В имеют переменную битовую скорость. Типичным примером этого могут быть услуги передачи аудио и видеoinформации с переменной скоростью. Например, услуга типа Video CODECs, при которой алгоритм кодирования меняется в зависимости от изменений видеосигнала.

Класс С. Предоставляются услуги с переменной скоростью передачи информации и установлением соединения для пользователей, не чувствительных к задержкам. Например, передача данных с установлением соединения и данных сигнализации.

Класс D. Отличается от класса С отсутствием установления соединения. Предоставляется услуга для пользователей, нечувствительных к задержкам. Примером может служить услуга передачи данных без установления соединения.

Протоколы AAL

На сети АТМ оба подуровня ААL обеспечивают внутреннее управление услугами и формирование блоков данных PDU (Protocol Data Unit), чтобы доставить их на соответствующий протокольный уровень. Приложение верхнего уровня создает сообщение или блок PDU. Формат или длина их зависит от используемого приложением протокола. Блок PDU появляется на подуровне CS в виде пользовательских данных. В зависимости от типа услуги, требуемой пользовательским приложением, подуровень CS для формирования блока CS-PDU может добавить к пользовательским данным заголовок и/или трейлер. Блок CS-PDU почти всегда длиннее, чем полезное поле ячейки. Именно поэтому на подуровне SAR блок CS-PDU сегментируется. Каждый сегмент представляет собой блок SAR-PDU длиной 48 байт. Он включает полезное поле блока, заголовок и/или трейлер. Блок SAR-PDU передается в полезном поле ячейки АТМ.



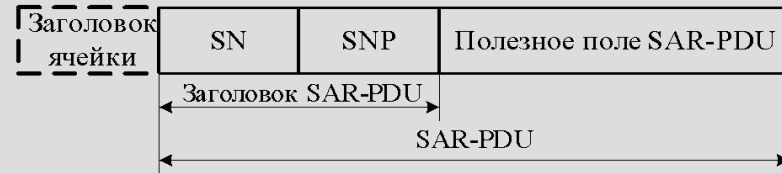
Структура блоков данных для протоколов разных уровней



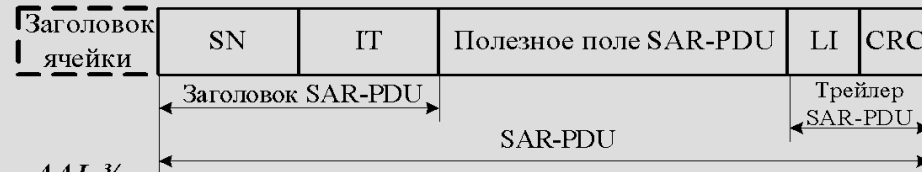
Перечень типов AAL:

- уровень AAL0
- уровень AAL1
- уровень AAL2
- уровень AAL3/4
- уровень AAL5

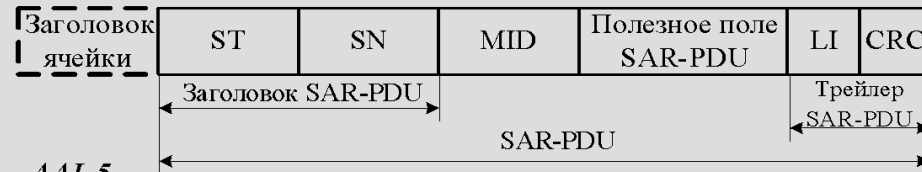
AAL 1



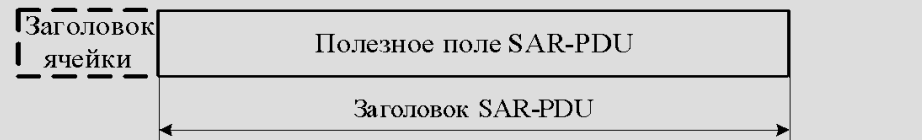
AAL 2



AAL 3/4



AAL 5



Форматы блоков SAR-PDU

Уровни AAL

	AAL1	AAL2	AAL3/4, AAL5	AAL5
<u>Сквозная синхронизация</u>	обязательная		необязательная	
<u>Битовая скорость</u>	постоянная	переменная		
<u>Режим соединения</u>	с установлением соединения		без установления соединения	

Характеристика типов AAL

Уровень AAL0

Уровень AAL0 не основывается на каких-либо стандартах или соглашениях по реализации. Он используется просто в качестве термина для обозначения процесса поступления в механизм AAL необработанных ячеек. Уровень AAL0 требует, чтобы оборудование пользователя обеспечивало собственные или основанные на стандартах AAL функциональные возможности по передаче из высших уровней полностью согласованных ATM-ячеек. Основными потребителями услуг этого типа AAL являются оборудование и приложения, требующие такой тип AAL, который не совместим со стандартными спецификациями, и поэтому предлагает в качестве входных в ATM-процесс такие структуры данных, которые не нуждаются в адаптации к ATM-сети.



Уровень AAL1

На уровне AAL1 предоставляются услуги по передаче информации от источника к получателю с постоянной битовой скоростью после того, как на сети ATM установлено виртуальное соединение. Для пользователей на уровне AAL1 выполняются следующие функции: передача и доставка блоков данных услуг (SOU) с постоянной битовой скоростью; синхронная передача информации между источником и получателем; передача информации о структуре данных; при необходимости индикация ошибок или потерь информации, которая не восстанавливается самим протоколом AAL.

Подуровень SAR принимает 47 байт данных от подуровня CS и затем добавляет заголовок SAR-PDU длиной 1 байт в каждый блок для формирования блока SAR-PDU.

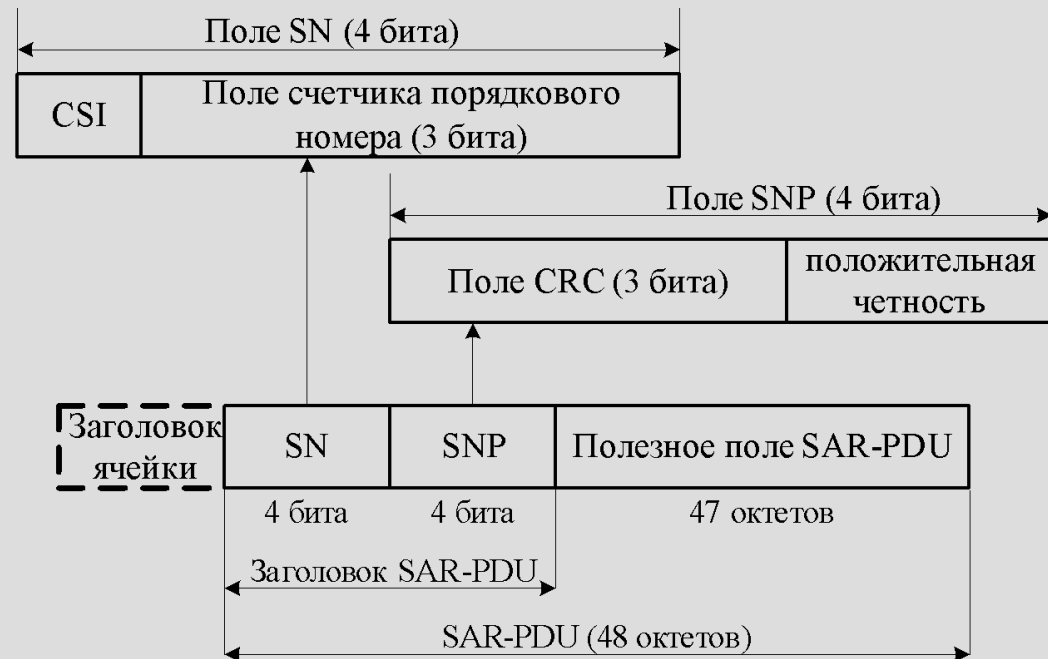
SN – порядковый номер;

SNP – защита порядкового номера;

CRC – проверка циклическим кодом;

CSI – индикация подуровня сходимости.

Поле SNP (4 бита) обеспечивает обнаружение ошибок и сходимости для 3 бит. Для уровня AAL1 имеет преимущества: сеть ATM вносит порядок битов, номер байта, подполе обеспечивает цифровую передачу с использованием надежного сетевого CRC (3 бита) по цифровым битовым потокам. Обнаруживает ошибки в поле SNP задающего генератора, для уменьшения избыточности старшие биты временных отметок сетевой синхронизации не передаются. Возможно исправление одних из инкрементов и потерь. Это битовые вычисления. Необходимо для отметки нечетных и четных ячеек и обнаружения потерянных.



Структура SAR-PDU уровня AAL1

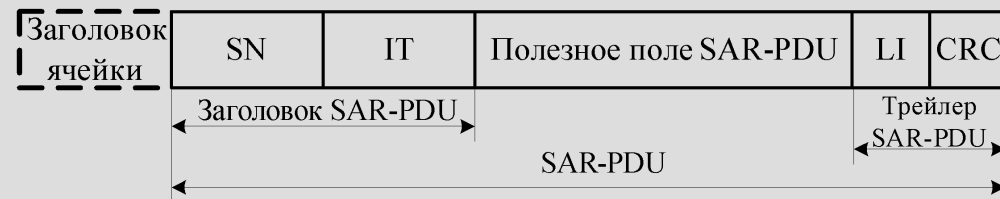
Уровень AAL2

Уровень AAL2 обеспечивает эффективное использование полосы пропускания при передаче низкоскоростных и коротких пакетов, а также пакетов с переменной длиной для приложений, восприимчивым к задержкам. С помощью уровня AAL2 короткие пакеты могут быть упакованы в одну или более ATM-ячеек. На уровне AAL2 информация должна передаваться с переменной битовой скоростью. Кроме этого, между источником и получателем передается информация синхронизации.

IT – тип информации;

LI – индикатор длины.

Поле SN содержит число последовательности для обеспечения восстановления потерянных или неправильно маршрутизированных ячеек.



Структура SAR-PDU уровня AAL2

Поле информации (IT) отмечает начало (BOM), продолжение (COM) и конец сообщения (EOM), или же то, что ячейки передают синхросигналы или другую информацию. Данные BOM, COM и EOM указывают на положение ячейки в передаваемом сообщении: первая, промежуточная или последняя. Следовательно, блок информации на уровне CS может иметь переменную длину.

Поле индикатора длины (LI) показывает число используемых байт в частично заполненных ячейках.

С помощью поля CRC на подуровне SAR можно обнаружить и откорректировать битовые ошибки в блоках SAR-PDU.

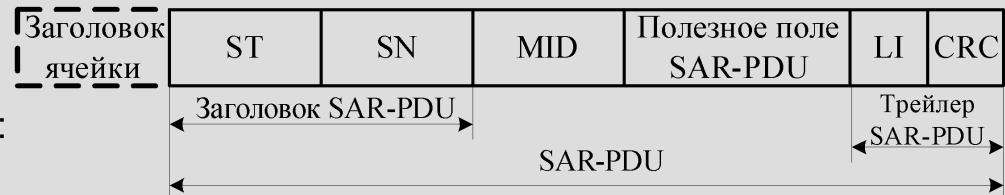
На подуровне CS должны выполняться следующие функции: восстановление синхросигналов путем вставок и изъятий временной информации (например, временной отметки); обработка ошибочно доставленных ячеек или ситуаций с потерянными ячейками; прямая (опережающая) коррекция ошибок (FEC) для аудио и видео услуг.

Уровень AAL3/4

Уровень AAL3/4 используется для передачи данных, которые чувствительны к потерям кадров, но не к временным задержкам. Этот уровень может быть использован для передачи данных как с установлением соединения, так и без него. Для обеспечения минимальной потери ячеек уровень AAL3/4 отвечает за обнаружение ошибок в каждой ячейке и использует усложненный механизм контроля ошибок, который совместно с двадцатью двумя дополнительными битами сигнальной информации заимствует 4 октета из каждой 48-октетной полезной нагрузки.

Блок SAR-PDU имеет 2 байта заголовка, 44 байта полезного поля и 2 байта трейлера.

Уровень SAR обеспечивает следующие функции:



Структура SAR-PDU уровня AAL3/4

- сегментирование и сборка сообщений высокого уровня (блоков CS-PDU разной длины). Блок SAR-PDU содержит для этой цели два поля. Поле типа сегмента (ST) – 2 бита. Оно указывает на то, какая часть блока CS-PDU переносится блоком SAR-PDU: начальная, средняя или конечная, а также то, что блок CS-PDU включает только один сегмент. Кодировка типа сегмента следующая: «10» - для BOM, «00» - для COM, «01» - для EOM и «11» - для сообщения с одним сегментом SSM. Поле индикатора длины (LI) – 6 бит. Поскольку последний или единственный сегмент блока SAR-PDU может иметь полезное поле меньше максимальной длины, то необходима индикация числа значащих байтов;
- обнаружение ошибок. Для обнаружения битовых ошибок в блоке SAR-PDU задано 10-битовое поле CRC. Кодировка поля основывается на полиноме $G(x)=1+x+x^4+x^5+x^9+x^{10}$;
- контроль за последовательностью принимаемых ячеек. С помощью поля SN (4 бита) могут быть обнаружены потерянные или неправильно вставленные ячейки;
- мультиплексирование множества блоков CS-PDU, передаваемых на уровне ATM по общему виртуальному каналу или пути. Мультиплексирование поддерживается 10-битовым идентификатором мультиплексирования (MID) (мультиплексирование с числом соединений на уровне AAL до 2^{10}).

Уровень AAL5

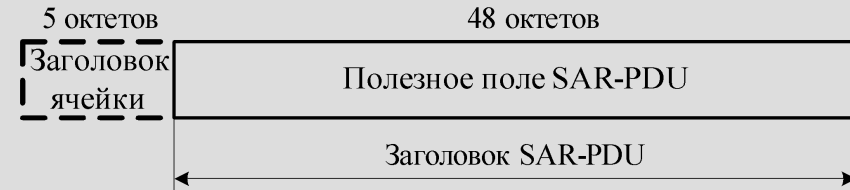
Уровень AAL5 обрабатывает пульсирующий трафик данных независимо от того, использует ли он процедуру установления соединения или нет, включая большинство потоков данных и вносит меньше издержек, чем уровень AAL3/4. AAL5 не поддерживает мультиплексирование. Уровень AAL5 обеспечивает более высокую эффективность, чем другие. Это достигается большим заполнением блоков SAR-PDU пользовательскими данными. Также обеспечивается лучшее обнаружение ошибок, хотя в каждой ячейке нет поля CRC. Функции, выполняемые подуровнем CS AAL5, такие же, как и подуровня CS AAL3/4, за исключением того, что не указывается принимающей стороне размер буфера записи. Кроме того, подуровень CS сам обеспечивает защиту от ошибок на уровне AAL5.

На подуровне SAR формируются блоки длиной по 48 байт. Они включают только полезное поле SAR-PDU.

Формат блока CS-PDU:

- **поле заполнения (LEN/PAD)** используется для выравнивания блока CS-PDU до размера, кратного блоку SAR-PDU. Если блок CS-PDU больше 48 байт, то для завершения блока передается 0, то есть будет содержаться 47 байт. Поле не несет какой-либо информации;

- **поле пользовательских данных (UU)** содержит один байт информации. Передается произвольным образом и может содержать любой байт. В AAL5 полезное поле блока CS-PDU больше 48 байт. Для определения структуры следующего блока (канализация) в его заголовке и хрейлере xv на стоящем в начале индикатор может принимать только значение «0». Это означает, что блок CS-PDU содержит пользовательские данные;



Структура SAR-PDU уровня AAL5

Пользовательские данные (User Data)	PAD	UU	CPI	LENGHT	CKC-32
Пользовательские данные				(0...65 535 байт)	
PAD		Заполнение		(0...47 байт)	
UU		Информация «пользователь - пользователь»		(1 байт)	
CPI		Индикатор для перспективного применения		(1 байт)	
LENGHT		Длина поля пользовательских данных		(2 байта)	
CKC-32		Проверка циклическим кодом		(4 байта)	

Формат блока CS-PDU уровня AAL5

Сигнализация на сети ATM

На сети ATM применяется сигнализация DSS2, в которой используется набор сообщений и адресов. Существуют четыре типа адресов ATM, каждый из которых имеет свое назначение. Для частных сетей ATM Форум определил три типа адресов конечных систем (ATM End System Addresses, AESA): **DCC AESA** и **ICD AESA**. Что касается сетей ATM общего пользования, здесь выбор состоит между исходным форматом адреса *E.164* и тремя адресами AESA, указанными выше. Кроме того, эти форматы могут использоваться совместно.

Первый – код данных страны (Data Country Code, DCC) – должен применяться в частных сетях ATM, прежде всего в сетях передачи данных.

Второй – международный указатель кода (International Code Designator, ICD) отличается от DCC тем, что должен служить для построения кодов, позволяющих распознавать адреса в международной сети (роль его аналогична роли штрихового кода продуктов); ICD не предназначен для идентификации коммутаторов.

Третий – точка доступа к сетевым службам (Network Service Access Point, E.164 NSAP) – бал создан для применения в сетях общего пользования, но затем постепенно превратился в альтернативу формата DCC. Кроме того, у формата E.164 NSAP есть аналог - E.164 Natural, предназначенный для тех же сетей.

Адреса AESA записываются в шестнадцатеричной форме, и каждый из них имеет длину в 20 байт. этот адрес имеет иерархическую структуру и состоит из двух основных частей: *IDP* (Initial Domain Part) и *DSP* (Domain Specific Part), каждая из которых имеет несколько полей.



Формат адреса DCC AESA



Сегмент **IDP** определяет тип адреса и тип уполномоченного, который отвечает за управление этим адресом. Он имеет два поля: однобайтовое поле *AFI* (Authority and Format Indicator) и двухбайтовое *IDI* (Initial Domain Identifier). Первое поле — для рассматриваемого формата адреса — всегда имеет фиксированное значение, равное 39, а поле IDI содержит код, идентифицирующий страну, в которой этот адрес зарегистрирован (Data Country Code, DCC). Вместе эти два поля идентифицируют данный адрес как DCC AESA и однозначно указывают страну, в которой он был зарегистрирован.

Сегмент **DSP** также иерархически разделяется на несколько полей: десятибайтовая часть *HO-DSP* (High Order Domain Specific Part), шестибайтовый идентификатор конечной системы (End System Identifier, ESI) и однобайтовое поле *SEL* (Selector).

Содержимое поля *HO-DSP* служит для идентификации сегмента адресного пространства, который выделен определенному пользователю или подсети.

ESI предназначен для идентификации конечной системы сети ATM в определенной подсети.

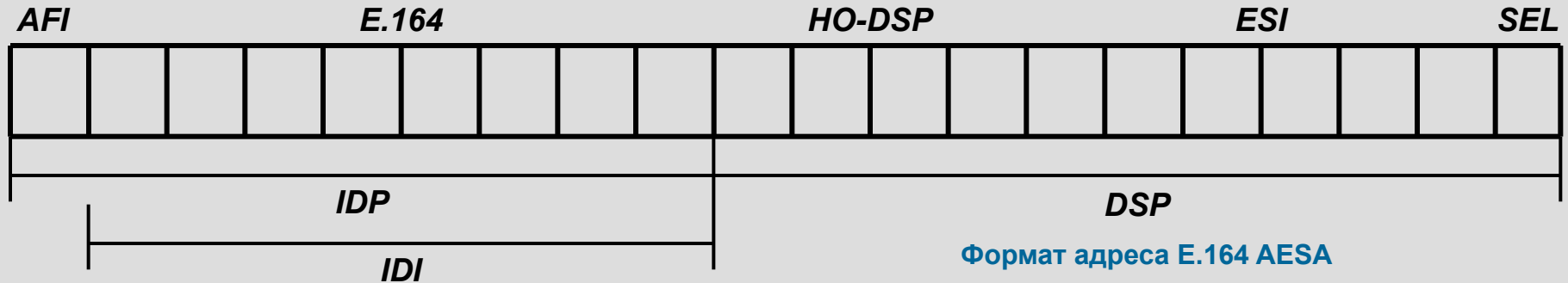
Поле *SEL* – *селектор*, не имеет существенного значения для выполнения маршрутизации запросов на установление коммутируемых виртуальных соединений в сети ATM, и оно доступно для использования конечной системой.

Формат адреса ICD AESA



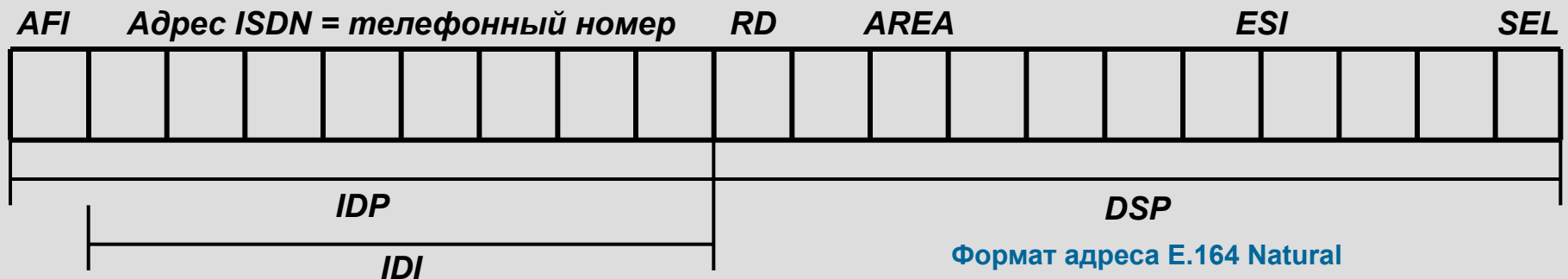
Формат адреса ICD AESA имеет сходную структуру с форматом DCC, за исключением следующих моментов: значение поля AFI равно 47, а не 39, а поле IDI содержит ICD (International Code Designator) — двухбайтовый идентификатор организации по стандартизации, значения которого устанавливаются Британской организацией по стандартизации (применяется для международной сети).

Формат адреса E.164 AESA



Адрес E.164 AESA также записывается в шестнадцатеричном виде. Сегмент *IDP* этого формата адреса разделяется на два поля: однобайтовое *AFI* и восьмибайтовое *IDI*. Первое имеет фиксированное значение 45, а второе содержит адрес в формате E.164.

В настоящее время назначение адресов E.164 AESA не контролируется ни одной организацией. Однако поскольку исходные адреса E.164 назначаются согласованным образом, то, как предполагается, владельцы таких адресов одновременно являются владельцами и координаторами адресов E.164 AESA. Присвоение исходных адресов E.164 выполняется на распределительной основе государственными организациями в каждой стране (например, в США эта функция в настоящее время выполняется Bellcore). Кроме этого, у формата E.164 AESA есть аналог – E.164 Natural, для применения в сетях общего пользования (ТФОП). Этот формат содержит поля: *RD* (Routing Domain) – домен маршрутизации, для разделения маршрутов внутри сети; *AREA* (Area Identifier) – идентификатор зоны.



Процесс установления соединения на сети АТМ

До того как конечные системы в сети АТМ смогут начать взаимодействовать друг с другом, они должны установить между собой коммутируемое виртуальное соединение. Процессу установления соединения предшествует отправка запроса. Подача запроса позволяет известить сеть АТМ о том, что она должна организовать соединение по указанному адресу. Все операции по установлению соединения, его поддержке и завершению определяются протоколом сигнализации для общего и частного интерфейсов UNI. Таким образом, сфера действия протокола сигнализации ограничена участком "конечная станция — коммутатор АТМ", между которыми обмен сообщениями и происходит.

В первый момент взаимодействия отправитель передает запрос коммутатору АТМ, к которому он подключен напрямую. Коммутатор идентифицирует этот запрос, так как сигнал помечается отправителем определенными значениями полей VCI и VPI. Комбинация VCI/VPI, которая ассоциируется с запросом на установление соединения, определена в спецификации АТМ значениями VCI = 5, VPI = 0. Получив запрос с такими значениями идентификаторов, коммутатор АТМ определяет его как запрос на установление соединения

Запрос на установление соединения от отправителя — это содержащаяся в ячейках комбинация извещений, адресной и другой аналогичной информации. Он включает адрес отправителя и параметры качества услуг, необходимые для установления соединения с нужным адресатом. Если исходить из аналогии с телефонным вызовом, то запрос на установление соединения можно представить как комбинацию из следующих действий: подъем трубки и набор телефонного номера желаемого абонента.

После того как коммутатор АТМ идентифицировал данный запрос, он возвращает сообщение о начале обработки запроса, и с этого момента и начинается собственно процесс разрешения адресов.

Всю совокупность сигнальных сообщений в сети АТМ можно разделить по принадлежности к функциональным группам: **установление соединения**, **отслеживание статуса соединения**, **завершение соединения**. Более подробно рассмотрим этапы установления и завершения соединения.



Установка соединения

- **SETUP.** Сообщение посылается отправителем ближайшему коммутатору ATM через интерфейс UNI и получателю ближайшим к нему коммутатором. Оно служит для инициирования процедуры установления соединения и содержит необходимую для этого информацию (адрес получателя, параметры качества обслуживания и т. д.);
- **CALL PROCEEDING.** Сообщение посылается получателем в сеть и из сети отправителю для информирования о том, что процесс установления соединения инициирован;
- **CONNECT.** Сообщение посылается получателем в сеть и из сети отправителю для извещения о том, что получатель принимает запрос на установление соединения;
- **CONNECT ACKNOWLEDGE.** Сообщение посылается отправителем в сеть и из сети получателю для извещения о том, что запрос согласован.

УСТАНОВЛЕНИЕ СОЕДИНЕНИЯ

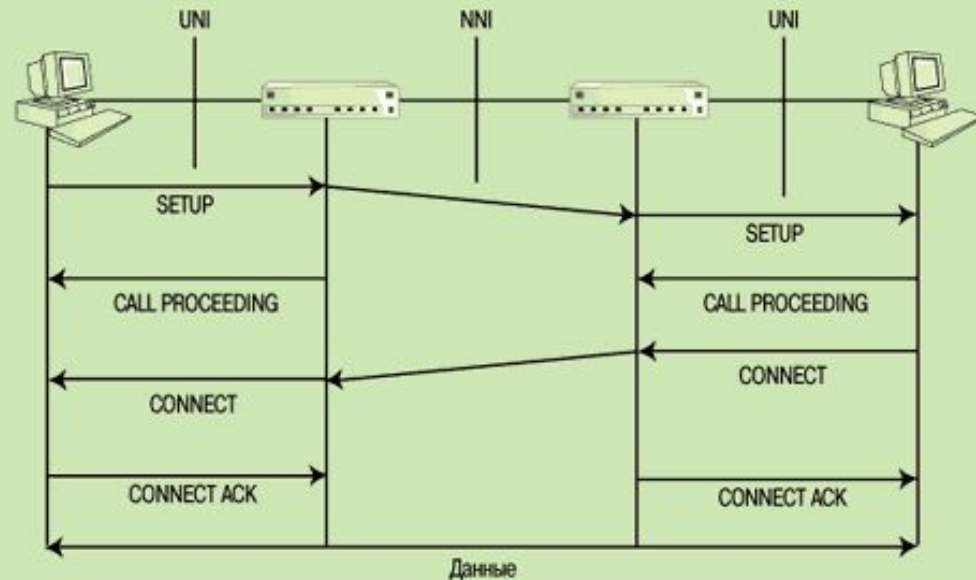


Схема обмена сообщениями при установлении соединения

Завершение соединения

- **RELEASE**. Сообщение посылается одним из абонентов с просьбой к сети завершить соединение или сетью для извещения о том, что соединение должно быть завершено и получатель сообщения должен освободить виртуальный канал после посылки сообщения **RELEASE COMPLETE**;
- **RELEASE COMPLETE**. Сообщение посылается одним из абонентов или сетью для информирования о том, что виртуальный канал должен быть освобожден.

ЗАВЕРШЕНИЕ СОЕДИНЕНИЯ

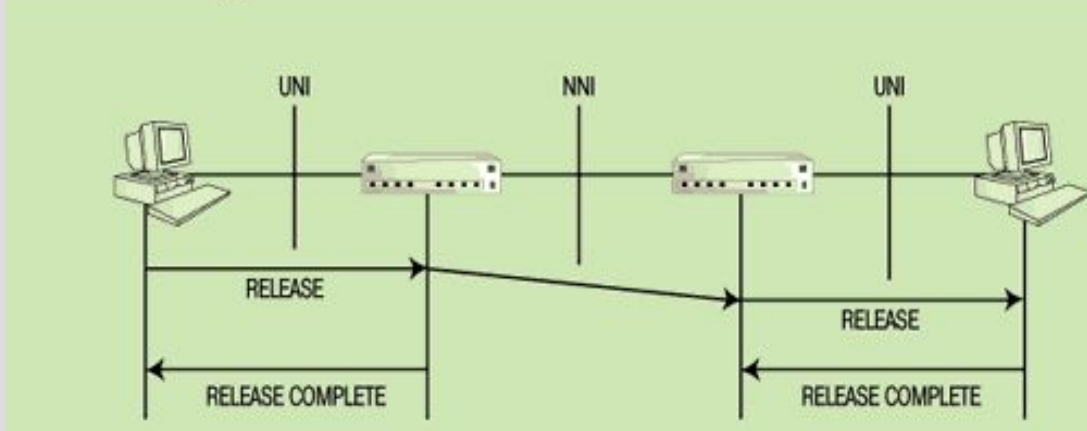


Схема обмена сообщениями при завершении соединения

Если конечной станции в сети ATM необходимо завершить соединение с другим абонентом, то она инициирует процесс закрытия такого соединения. Этот процесс, в основном, обратен процессу установления соединения. Задействованные для поддержания соединения ресурсы коммутаторов освобождаются и могут быть использованы для открытия новых соединений. В целом процедура аналогична завершению телефонного разговора при опускании телефонной трубки.

Завершающая соединение конечная станция передает специальное сообщение коммутатору, к которому она подключена, который преобразует формат ячейки UNI в формат PNNI, а она затем передается следующему коммутатору в соответствии с парой идентификаторов VCI/VPI, ассоциирующихся с соединением. После того как коммутатор послал завершающее сообщение, он удаляет эту пару идентификаторов из своей таблицы и обновляет текущую информацию о доступных ресурсах.

При передаче завершающего сообщения от коммутатора к коммутатору, они последовательно закрывают это соединение. После того как соединение было завершено, его ресурсы становятся доступными для последующих соединений.

Структура коммутаторов ATM

Коммутатор ATM включает следующие основные компоненты:

- корпус;
- источник питания;
- электронные платы и буферную память;
- электронные платы управляющего процессора;
- электронные и электронно-оптические платы линейных интерфейсных модулей;
- вспомогательные (дополнительные) платы.

Коммутатор для рабочих групп может представлять собой настольное устройство; для соединения нескольких рабочих групп – размещаться в вертикальной или горизонтальной секции; магистральный коммутатор может быть оформлен в виде больших стоек.

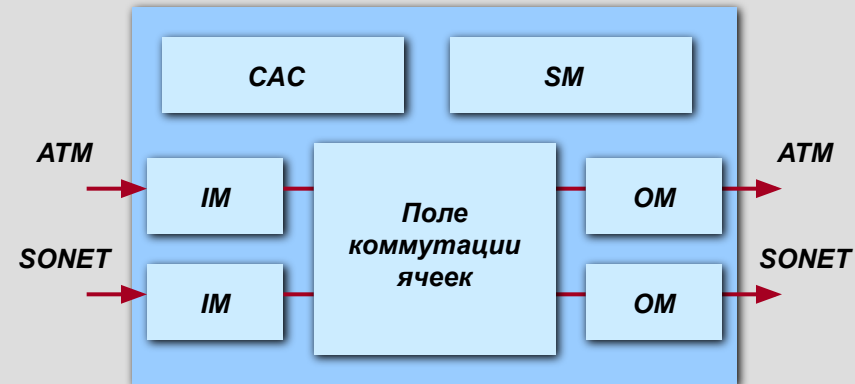
Общая структура коммутатора:

- входные модули (Input Module, IM);
- выходные модули (Output Module, OM);
- поле коммутации ячеек;
- модули контроля над установлением соединения (Connection Admission Control, CAC);
- модули управления коммутаторами (Switch Management, SM).

OM подготавливает потоки ATM для физической передачи:

Модуль **SM** выполняет следующие функции:

- управление конфигурацией компонентов коммутатора и защитой его баз данных;
- управление трафиком, процедурами администрирования и интерфейсом UNI;
- проверка заголовка на наличие ошибок, подтверждение правильности значений идентификаторов виртуального пути (поле VPI) и виртуального канала;
- управление соединениями по протоколам систем сигнализации;
- управление выходом в сеть на нагрузку сети SDH и генерирует соответствующие заголовки;
- преобразование цифровых ячеек в аналоговый вид для передачи по оптическим линиям.



Архитектура коммутатора ATM

Принципы построения коммутаторов АТМ

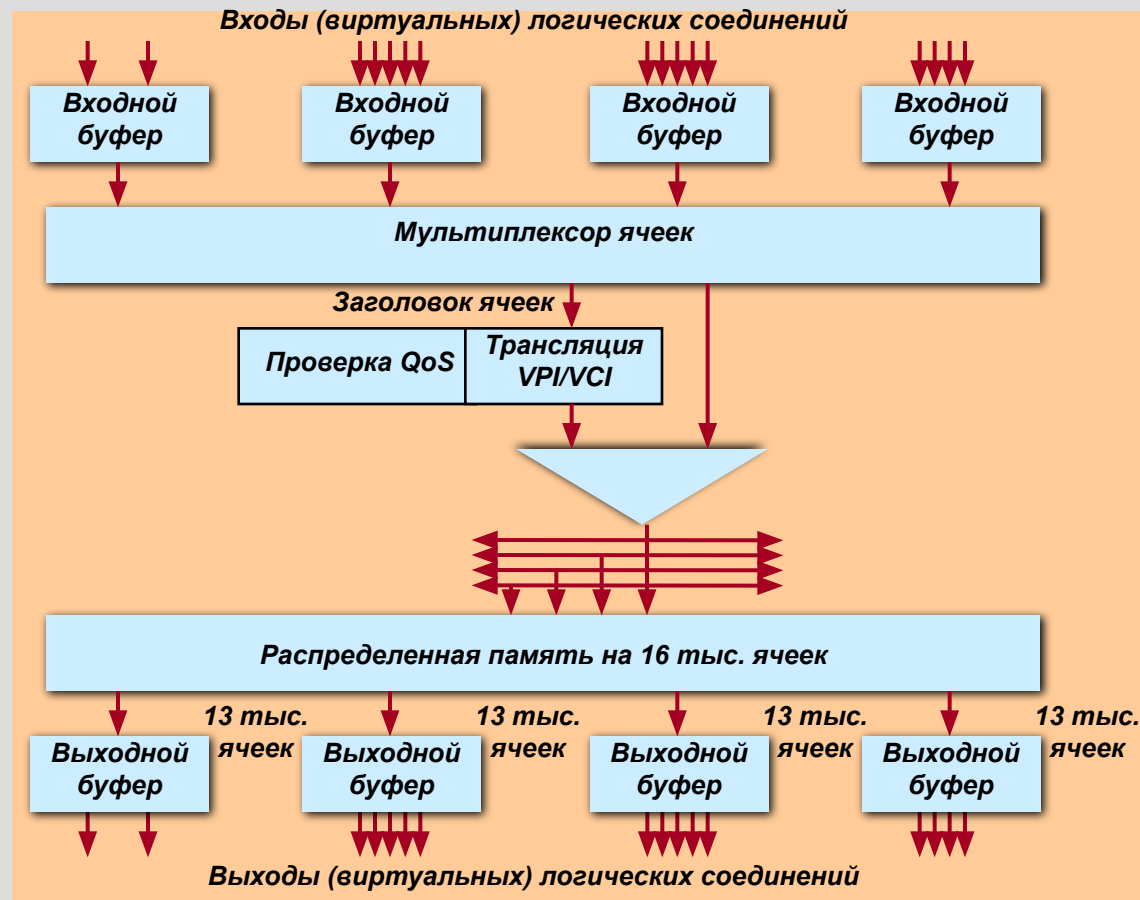
Построение коммутаторов основывается на различных методах коммутации:

- *с разделяемой памятью*
- *с общей средой*
- *с полностью связной топологией*
- *с пространственным разделением:*
 - *матричные коммутаторы*
 - *баньяновидные (каскадные) коммутаторы*



Коммутатор с разделяемой памятью

Коммутатор имеет память, общую для входных и выходных блоков. В таком коммутаторе все управляется централизованно. Входящие ячейки преобразуются из последовательного формата и записываются в порт ОЗУ. Используя заголовки ячеек с тэгами маршрутизации, контроллер памяти решает, в каком порядке ячейки будут считываться из нее. Выходящие ячейки демультиплексируются при передаче на выходные порты и преобразуются из параллельного формата в последовательный. Данный метод коммутации подразумевает организацию очередей на выходных портах, где все буферы формируют единое пространство памяти. Он привлекателен тем, что дает возможность приблизиться к теоретическому пределу производительности. Совместный доступ к буферной памяти минимизирует ее емкость при заданной доле потерянных ячеек. Недостатком такого метода построения являются высокие требования к быстродействию памяти и контроллера.



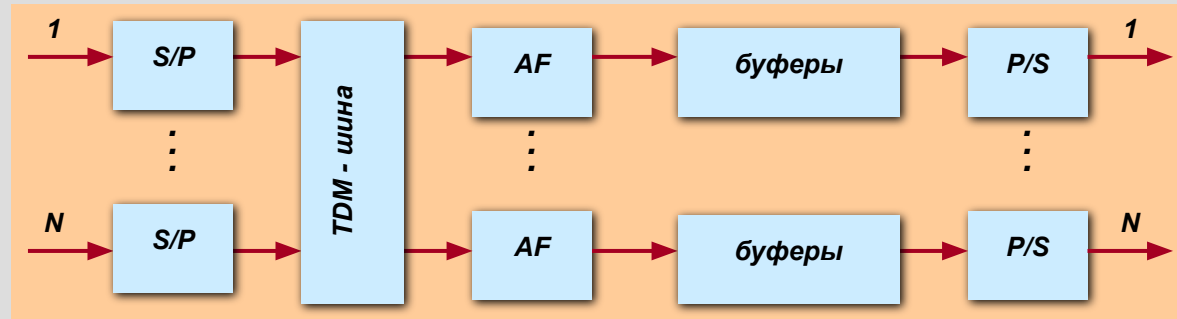
Структура коммутатора с разделяемой памятью

Коммутатор с общей средой

AF — адресный фильтр;

S/P — последовательно-параллельное преобразование;

P/S — параллельно-последовательное преобразование



Структура коммутатора с общей средой

В данном методе ячейки передаются через общую среду — кольцо, шину или двойную шину, в данном случае, примером является шина с временным разделением (**TDM**). Входящие ячейки передаются на шину циклически. На каждом выходе адресные фильтры (*Address Filter*, **AF**) в соответствии с тэгами маршрутизации считывают и пересылают свои ячейки в выходные буферные устройства. Скорость шины должна быть равной, по крайней мере, NV ячейкам в секунду, V ячеек/с – скорость работы одного порта. Выходные буферы здесь не являются общими, поэтому для сохранения прежней вероятности потерь требуется большая суммарная емкость буферов, чем при методе с разделяемой памятью. Выходные очереди здесь строго закреплены и организованы по принципу «первым пришел – первым обслужен». Модуляция выходных каналов упрощает работу адресных фильтров, а широкополосная передача с селекцией — функционирование всей системы. Этот метод построения коммутаторов относится к типу «без самомаршрутизации» и основан на мультиплексировании входящего трафика в единый поток.

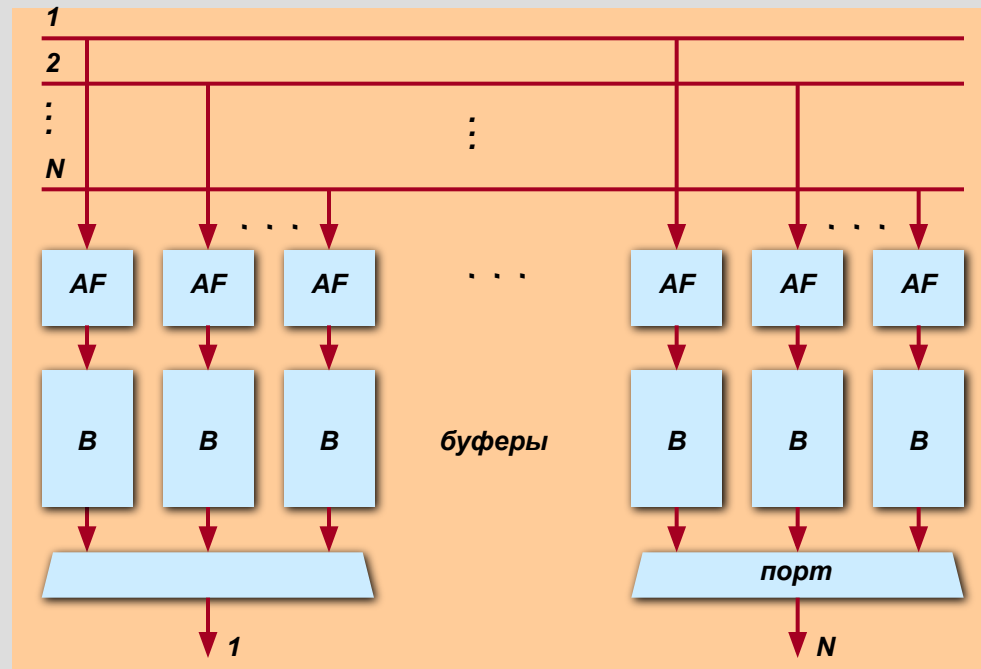
Коммутатор с полносвязной топологией

AF — адресные фильтры; **B** — буферы

Отличительная особенность данного метода — существование независимого пути для каждой из N^2 возможных пар входов и выходов. Таким образом, входящие ячейки транслируются на отдельные шины выходных каналов, а адресные фильтры пропускают эти ячейки в выходные очереди. Преимущества рассматриваемого типа коммутации заключаются в том, что буферизация ячеек происходит на выходных портах и (как в методе с общей средой) отсутствуют ограничения на групповую и широковещательную передачу. Реализация адресных фильтров и выходных буферов достаточно проста: нужно лишь обеспечить требуемую скорость обмена через порт. Метод полносвязной топологии допускает

простое масштабирование в широких пределах и позволяет достичь высокой скорости функционирования коммутатора, поскольку все его аппаратные модули работают с одной и той же скоростью.

К сожалению, квадратичный рост числа буферов ограничивает количество выходных портов, хотя скорость обмена через порт лимитируется только физическим быстродействием адресных фильтров и выходных буферов.



Структура коммутатора с полносвязной топологией

Коммутатор с пространственным разделением

DM – демультиплексор; **M** - мультиплексор

Каждый вход коммутатора связан с демультиплексором (DM). Элементы ATM, поступающие на вход, записываются в тот из N буферов, который связан с требуемым выходом. Мультиплексор (M), связанный с конкретным выходом, объединяет в единый поток элементы из буферов, связанных с N входами.

Коммутатор с пространственным разделением сразу устанавливает несколько соединений от входных портов к выходным. Управление таким коммутатором осуществляется по портам. Существенный недостаток коммутаторов такого типа – невозможность создания соединения при занятости всех внутренних ресурсов.

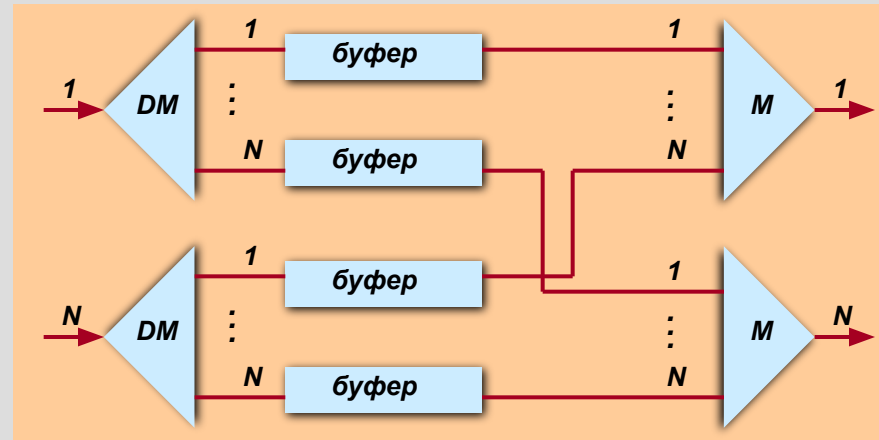
Такая ситуация называется *внутренней блокировкой*. Основное отличие коммутаторов данного типа – невозможность выходной буферизации данных.

Управление в таком коммутаторе может быть как централизованным, так и распределенным.

Коммутаторы с пространственным разделением делятся:

- на *матричные*;

- *баньяновидные (каскадные)*.



Структура коммутатора с пространственным разделением

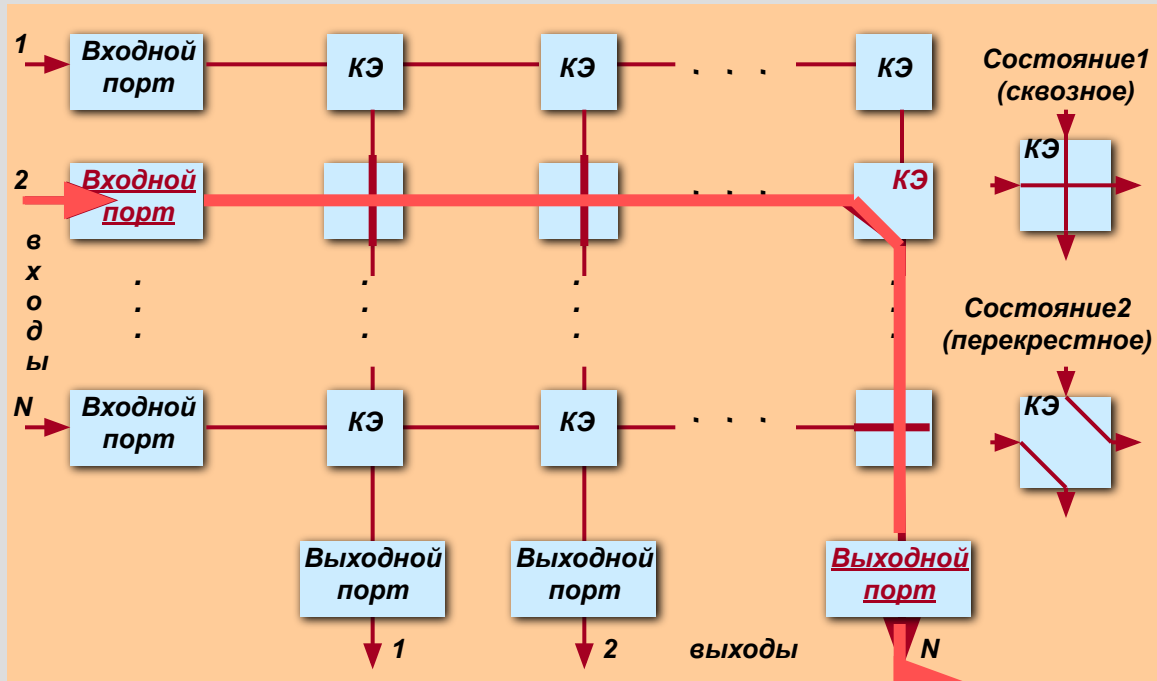
Матричный ATM - коммутатор

Матричные коммутаторы основаны на матрице, в узлах которой расположены ключи (коммутационные элементы, **КЭ**), соединяющие горизонтальные и вертикальные входы и выходы.

Коммутационный элемент матрицы может находиться в одном из двух состояний: сквозном (состояние 1) и перекрестным (состояние 2). Если, например, необходимо установить соединение входа 2 с выходом N, то **КЭ_{2,N}** должен быть установлен в состояние 2, а все КЭ_{2,k} (где k = 1, 2, ..., N-1) и все КЭ_{i,N} (где i = 3, 4, ..., N) – в состояние 1. Состояния остальных КЭ не имеет значения для данного соединения.

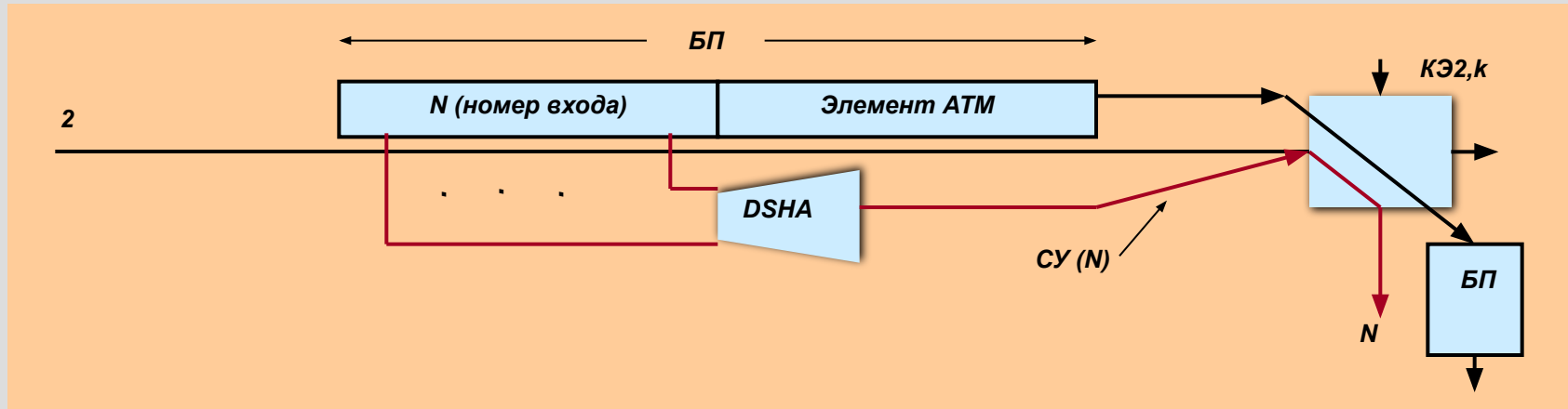
В процессе маршрутирования быстрый пакет на входе 2 снабжается адресом выхода N.

При пересылке его по горизонтали 2 он может самостоятельно переключить требуемый **КЭ_{2,N}** в перекрестное состояние. Это свойство называется **самомаршрутизацией**. Для этого КЭ должен иметь дешифратор адреса (**DSHA**). При этом реализуется распределенное управление коммутацией, в нем может принимать участие любой КЭ. Если номер выхода N в заголовке совпадает с номером вертикали данного КЭ, то на выходе **DSHA** формируется сигнал управления СУ (N), переводящий КЭ в состояние 2.



Структура матричного ATM - коммутатора

Коммутация при самомаршрутизации



Управление коммутацией при самомаршрутизации

Использование самомаршрутизации дает существенное увеличение скорости коммутации. Одно из ограничений такой КС состоит в том, что одновременно коммутировать два и более БП на один выход нельзя. Поэтому при отсутствии буферов на выходах все конкурирующие БП, кроме одного, теряются.

Баньяновидный (каскадный) ATM - коммутатор

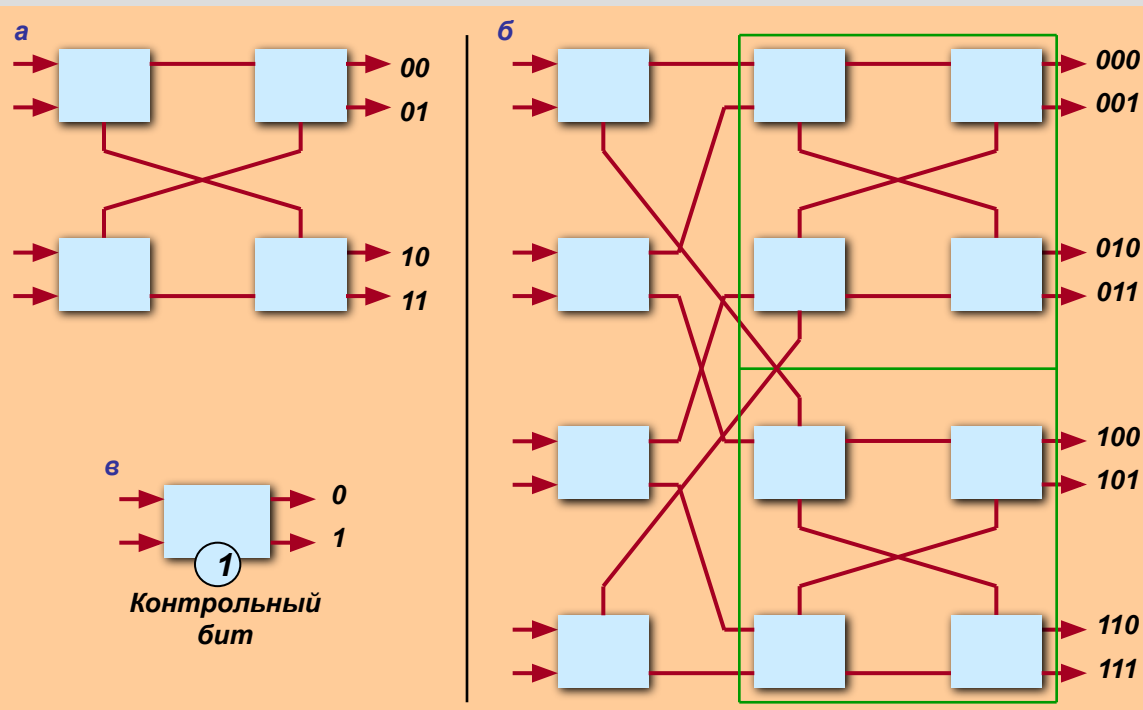
содержание

Баньяновидные сети (свое название они получили потому, что схожи по форме с одноименным тропическим деревом), строятся путем формирования каскадов коммутационных элементов. Основной коммутационный элемент 2x2 обрабатывает входящую ячейку в соответствии с управляющим битом выходного адреса. Если этот бит равен нулю, то ячейка направляется на верхний выходной порт кросса, в противном случае — на нижний.

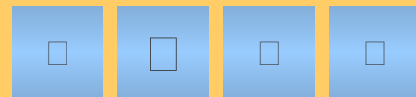
На рисунке показан коммутационный элемент 2x2 (в), и последовательное соединение коммутационных элементов, формирующих

баньяновидные сети 4x4 (а) и 8x8 (б). При построении сети 4x4 два каскада коммутационных элементов 2x2 могут быть соединены с использованием первого бита выходного адреса (для определения коммутационного элемента, на который направляется ячейка), а затем второго бита (для определения порта коммутационного элемента второго каскада). Сеть 8x8 формируется рекурсивно, при этом первый бит применяется для транспортировки ячейки через первый каскад, а последние два бита — для маршрутизации ячейки через сеть 4x4 на соответствующий выходной порт.

Для построения коммутатора типа «Баньян» с N входами и N выходами потребуется $M = N \times \log_2 N / 2$ коммутационных элементов.



ATM – коммутатор с баньяновидной структурой



Положительные свойства структур типа Баньян таковы:

- соединение входа с выходом реализуется аппаратными средствами децентрализованно по способу самомаршрутизации за время существенно меньшее, чем при программном управлении соединением; получаемая структура КС является регулярной, что позволяет удешевить СБИС коммутационных модулей узла с БКП;
- отказ от программного управления коммутацией позволяет просто наращивать емкость узла с БКП путем добавления новых модулей без изменения существующей структуры и алгоритмов коммутации.

Негативными свойствами рассматриваемых структур являются:

- единственный путь между одним из входов и конкретными выходом;
- одновременно может быть установлено не более чем N соединений;
- внутренние блокировки снижают пропускную способность до неприемлемой для практики величины.

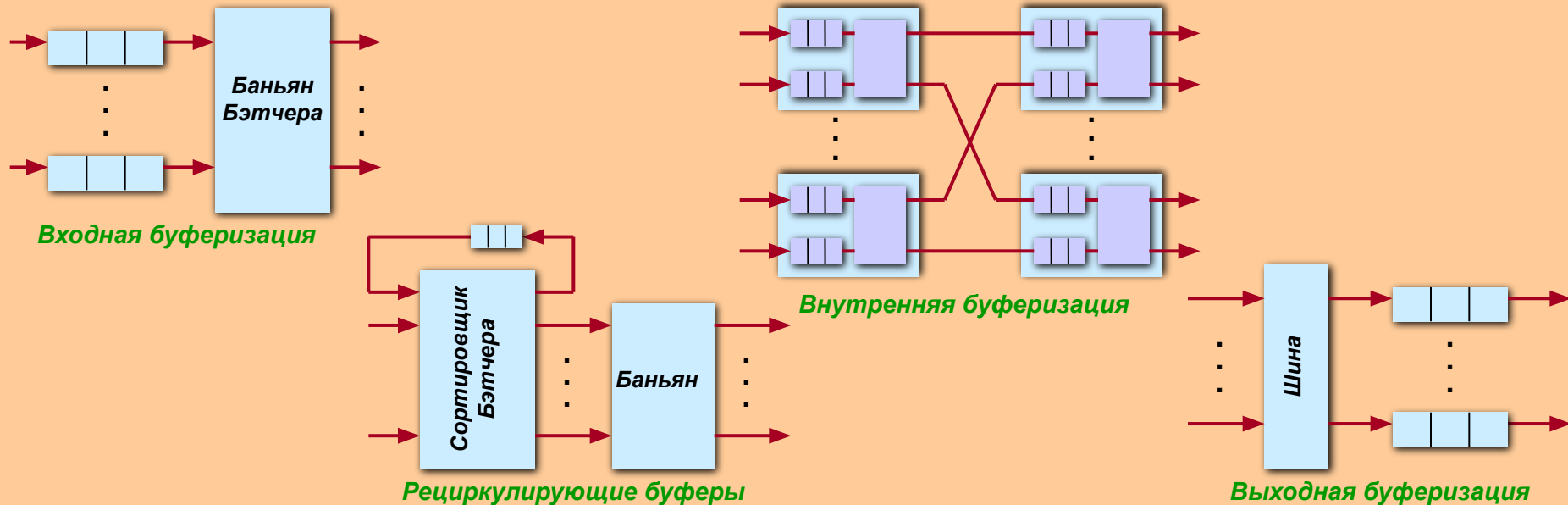
Основным недостатком схем типа Баньян – возможность внутренних блокировок. Блокировка возникает при столкновении двух БП на одном выходе КЭ. **Решением проблемы внутренней блокировки** может быть **организация буферов** перед коммутационными элементами и использование специального устройства для предварительной сортировки ячеек, называемого сортировщиком Бэтчера. Сортировщик Бэтчера распределяет входной поток по разным входам основной коммутационной структуры. Сортировщик позволяет избежать блокировок при адресации ячеек на различные входные порты, но если они одновременно адресуются на один и тот же выходной порт, единственным решением становится буферизация.

Организация буферов

Буферизация ячеек необходима при любой архитектуре коммутационного поля. Существует три метода буферизации: организация буферов, разделение буферов и управление буферами. Более подробно остановимся на организации буферов.

Существует четыре варианта размещения буферов в коммутаторах ATM:

- *входная буферизация* – буферы организуются на входных портах неблокирующей структуры с пространственным разделением;
- *выходная буферизация* – буферы организуются в выходных портах структуры с разделяемой шиной;
- *внутренняя буферизация* – для структуры с пространственным разделением буферы устанавливаются внутри коммутационных элементов;
- *рециркуляционная буферизация* – используется рециркуляционный буфер, внешний к коммутационному полю, ячейки могут повторно проходить по полю с пространственным разделением, если несколько ячеек одновременно адресуются на один и тот же выходной порт.



Практическая часть

Практическая часть содержит две части.

В первой части практической работы необходимо по *заданию* преподавателя построить схему коммутатора «баньяновидного» типа и сформировать по построенной схеме маршрутное поле быстрого пакета (*БП*), описание прохождения БП необходимо представить в отчете лабораторной работы.

Вторая часть практической работы содержит контрольное тестирование по теме «Пакетная коммутация». Для прохождения тестирования необходимо закрыть программу презентации и открыть приложение **TEST**, находящееся в папке, совместно с обучающей программой **Пакетная коммутация**.

Перед выполнением теста советую еще раз повторить [процесс передачи информации в сети ATM!](#)



Практическая часть (варианты задания)

Задания для первой части практической работы

Вариант 1

Сеть 16x16 КЭ 2x2
4вх – 8вых

Вариант 4

Сеть 8x8 КЭ 2x2
6вх, адрес пакета 010

Вариант 7

Сеть 8x8 КЭ 2x2
1вх – 6вых

Вариант 10

Сеть 16x16 КЭ 2x2
2вх, адрес пакета 1101

Вариант 2

Сеть 8x8 КЭ 2x2
2вх, адрес пакета 101

Вариант 5

Сеть 16x16 КЭ 2x2
14вх, адрес пакета 1000

Вариант 8

Сеть 16x16 КЭ 2x2
10вх – 5вых

Вариант 11

Сеть 8x8 КЭ 2x2
3вх, адрес пакета 110

Вариант 3

Сеть 16x16 КЭ 2x2
9вх, адрес пакета 0011

Вариант 6

Сеть 16x16 КЭ 2x2
15вх – 11вых

Вариант 9

Сеть 8x8 КЭ 2x2
7вх – 2вых

Вариант 12

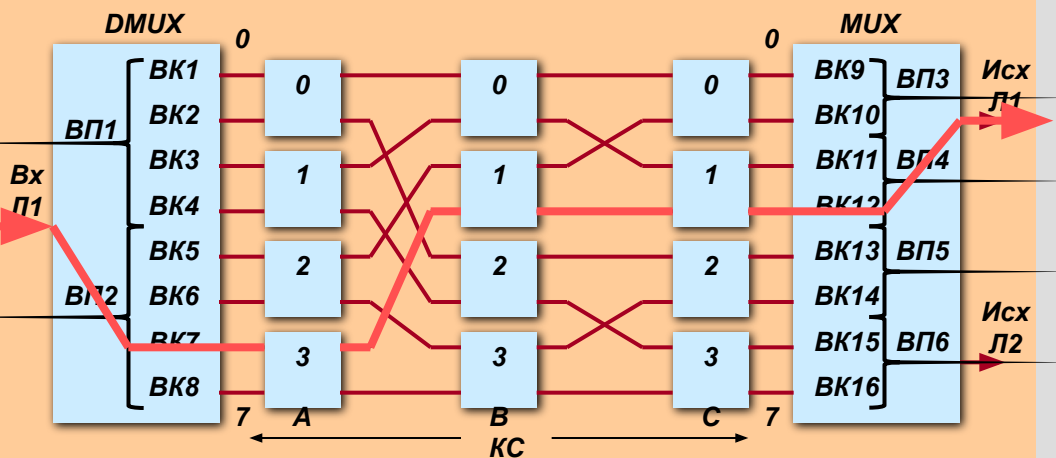
Сеть 16x16 КЭ 2x2
7вх – 13вых

Пример выполнения задания



Практическая часть (пример выполнения задания)

Рассмотрим пример формирования маршрутного поля (МП) БП для реализации способа самомаршрутизации в КС, построенной по трехкаскадной схеме баньян. Для формирования МП необходимо иметь данные о направлении передачи трафика от входного порта до выходного (в рассматриваемом случае коммутация осуществляется по пути: 6й вход – 3й выход). Закрепление ВК за входами и выходами КС приведено в **таблицах**.



Закрепление ВК входящей линии за входами КС

Номер ВП	Номер ВК	Номер входа КС
1	1	0
	2	1
	3	2
	4	3
2	5	4
	6	5
	7	6
	8	7

Закрепление ВК исходящей линии за выходами КС

Номер исх. лин	Номер ВП	Номер ВК	Номер вых. КС
1	3	9	0
		10	1
	4	11	2
2	5	12	3
		13	4
	6	14	5
		15	6
		16	7

Элемент АТМ приходит с первой входящей линии и попадает на демультиплексор: проходит через ВП2 - ВК7, затем попадает на 6й вход 3го элемента каскада А. Далее попадает на 1й элемент каскада В, выйдя через второй выход, попадет на 1й элемент каскада С. Затем элемент АТМ заходит в мультиплексор через требуемый 3й выход КС, далее ВК12 – ВП4 и выходит через первую исходящую линию. В итоге мы видим, что на вход поступает управляющая информация – 011.

Словарь терминов

<u>ATM</u>	<i>Asynchronous Transfer Mode</i>	<i>Асинхронный метод передачи</i>
<u>AAL</u>	<i>ATM Adaptation Layer</i>	<i>Уровень адаптации ATM</i>
<u>AESA</u>	<i>ATM End System Addresses</i>	<i>Адреса конечных систем ATM</i>
<u>AFI</u>	<i>Authority and Format Identifier</i>	<i>Идентификатор формата</i>
<u>B-ISDN</u>	<i>Broadband ISDN</i> Широкополосная	<i>Цифровая Сеть Интегрального Обслуживания (Ш-ЦСИО)</i>
<u>B-NT</u>	<i>Broadband Network Termination</i>	<i>Широкополосные устройства сетевого окончания</i>
<u>B-TE</u>	<i>Broadband Terminal Equipment</i>	<i>Широкополосное терминальное оборудование</i>
<u>CLR</u>	<i>Cell Loss Ratio</i>	<i>Доля потерянных ячеек</i>
<u>CBR</u>	<i>Constant Bit Rate</i>	<i>Постоянная битовая скорость</i>
<u>CS</u>	<i>Convergence Sublayer</i>	<i>Подуровень конвергенции</i>
<u>DCC</u>	<i>Data Country Code</i>	<i>Цифровой код страны</i>
	<i>Data link</i>	<i>Канал передачи данных</i>
	<i>Data link layer</i>	<i>Канальный уровень</i>
<u>DSP</u>	<i>Domain Specific Part</i>	<i>Определяющая часть адресного пространства</i>
<u>ESI</u>	<i>End System Identifier</i>	<i>Идентификатор конечной системы</i>
<u>FPS</u>	<i>Fast Packet Switching</i>	<i>Быстрая Коммутация Пакетов (БКП)</i>
	<i>Frame</i>	<i>Кадр</i>
<u>GFC</u>	<i>Generic Flow Control</i>	<i>Общее управление потоком</i>
<u>HO-DSP</u>	<i>High Order DSP</i>	<i>Старшая часть</i>
<u>ID</u>	<i>Identifier</i>	<i>Идентификатор</i>
<u>IDI</u>	<i>Initial Domain Identifier</i> Идентификатор	<i>Начальной области адресного пространства</i>
<u>IDP</u>	<i>Initial Domain Part</i>	<i>Начальная часть адресного пространства</i>
<u>ICD</u>	<i>International Code Designator</i>	<i>Код международной организации</i>
<u>ITU-T</u>	<i>International Telecommunications Union - Telecommunications Sector</i>	<i>Сектор по стандартизации Телекоммуникаций Международного Союза Электросвязи (МСЭ - Т)</i>

<u>IP</u>	<i>Internet Protocol</i>	<i>Межсетевой протокол</i>
<u>LAN</u>	<i>Local-Area Network</i>	<i>Локальная сеть</i>
<u>MCR</u>	<i>Minimum Cell Rate</i>	<i>Минимальная скорость передачи</i>
	<i>Network</i>	<i>Сеть</i>
<u>NNI</u>	<i>Network to Network Interface</i>	<i>Интерфейс "сеть-сеть"</i>
<u>OSI</u>	<i>Open System Interconnection</i>	<i>Взаимодействие Открытых Систем (ВОС)</i>
<u>PS</u>	<i>Packet Switching</i>	<i>Коммутация Пакетов (КП)</i>
<u>PL</u>	<i>Physical Layer</i>	<i>Физический уровень</i>
	<i>Point-to-multipoint</i>	<i>"Один ко многим"</i>
	<i>Point-to-point</i>	<i>"Один к одному"</i>
<u>QoS</u>	<i>Quality of Service</i>	<i>Гарантируемое качество обслуживания</i>
<u>RTP</u>	<i>Real-time Transport Protocol</i>	<i>Протокол передачи в реальном времени</i>
	<i>Router</i>	<i>Маршрутизатор</i>
<u>RIF</u>	<i>Routing Information Field</i>	<i>Поле маршрутной информации</i>
<u>SEL</u>	<i>Selector</i>	<i>Селектор</i>
<u>SC</u>	<i>Subscriber channel</i>	<i>Абонентский канал</i>
	<i>Switching node</i>	<i>Узел Коммутации (УК)</i>
<u>TCP/IP</u>	<i>Transmission Control Protocol/Internet Protocol</i>	<i>Протокол управления передачей</i>
<u>UNI</u>	<i>User to Network Interface</i>	<i>Интерфейс пользователь-сеть</i>
<u>VC</u>	<i>Virtual Channel</i>	<i>Виртуальный Канал (ВК)</i>
<u>VCI</u>	<i>Virtual Channel Identifier</i>	<i>Идентификатор виртуального канала</i>
<u>VP</u>	<i>Virtual Path</i>	<i>Виртуальный Тракт (ВТ)</i>
<u>VPI</u>	<i>Virtual Path Identifier</i>	<i>Идентификатор Виртуального Тракта (ИВТ)</i>
<u>VCC</u>	<i>Virtual Channel Connection</i>	<i>Соединение по Виртуальным Каналам (СВК)</i>
<u>VPC</u>	<i>Virtual Path Connection</i>	<i>Соединение по Виртуальным Трактам (СВТ)</i>

Рассмотрим пример передачи данных в режиме без установления соединения с использованием индикатора MID.

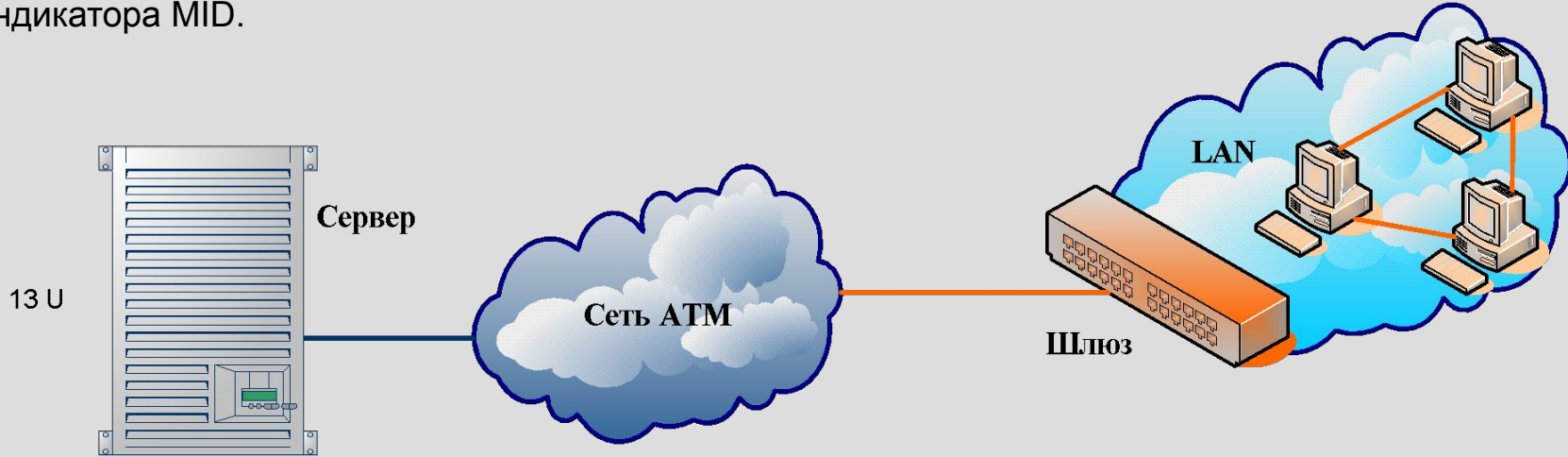


Схема связи сети LAN с сервером через сеть ATM

Множество терминалов объединены сетью LAN, которая через один шлюз на сети ATM связана с сервером. Информация передается от всех терминалов через одно виртуальное соединение на сети ATM к серверу. Сервер работает с сетью LAN в режиме без установления соединения и, основываясь на идентификаторе MID, разделять принимает трафик от каждого из терминалов. Затем направляет его к требуемому получателю на сети ATM. Заметим, что маршрутизация соединений осуществляется только на уровне ATM. Сервер, работающий в режиме без установления соединения, организует взаимосвязь между полем идентификатора MID и сетевой информацией маршрутизации, содержащейся в первом сегменте (BOM) уровня AAL блока CS-PDU.

Поскольку все блоки CS-PDU относятся к одному соединению сети ATM, то она обеспечивает для этих блоков одинаковое качество обслуживания (QoS).

