

Технология SDH

Недостатки и ограничения технологии PDH были учтены и преодолены разработчиками технологии **синхронной цифровой иерархии SDH**. Эта технология направлена на создание высокоскоростных магистральных каналов, способных соединять сети PDH как американского, так и международного вариантов.

В качестве среды передачи данных оборудование SDH использует *одномодовый волоконно-оптический кабель*, передавая по нему модулированные световые сигналы с длиной волны 1310 нм или 1550 нм.

Мультиплексоры SDH выполняют операции мультиплексирования и коммутации над *электрическими* сигналами — для этого они преобразуют *оптические* сигналы, пришедшие от оптических портов, в электрические, а после выполнения необходимых операций выполняют обратное преобразование сигналов. Такой тип работы называется работой по схеме **ОЕО**, от английского термина Optical-Electrical-Optical.

Функциональные уровни SDH

Функции оборудования SDH могут быть разделены на четыре уровня (рис. 8.9, *a*). Подчеркнем, эти уровни никак не соотносятся с уровнями модели OSI, для которой вся сеть SDH представляется как оборудование физического уровня. Перечислим их:

- **Уровень тракта** является самым высоким уровнем — он отвечает за доставку данных между двумя конечными пользователями сети. **Тракт (Path)** — это составное виртуальное соединение между пользователями. На этом уровне выполняется прием данных, поступающих в пользовательском формате, например формате E1, и отображение их в блоки данных SDH, соответствующие данному уровню. В результате этих действий к данным пользователя добавляется *заголовок тракта (POH)*, который несет информацию о типе и структуре полученного блока данных SDH, а также информацию, предназначенную для механизма контроля по четности.
- **Уровень линии** отвечает за передачу данных по линии *между двумя мультиплексорами* сети, поэтому линию также часто называют **мультиплексной секцией**. В функции этого уровня входит выполнение мультиплексирования и демультимплексирования, ввода-вывода пользовательских данных, а также реконфигурирование в случае отказа какого-либо элемента мультиплексной секции — оптического волокна, порта или соседнего мультиплексора.
- **Уровень секции** поддерживает физическую целостность сети. **Регенераторной секцией** в технологии SDH называется каждый непрерывный отрезок волоконно-оптического кабеля, который соединяет между собой такие, например, пары устройств SDH, как мультиплексор и регенератор, регенератор и регенератор, *но не два мультиплексора*. На этом уровне компоненты регенераторной секции выполняют тестирование и администрирование секции, контролируют ошибки.

- Фотонный уровень SDH** представляет собой модулированный световой сигнал одной волны из диапазона 1310 нм или 1550 нм. В сетях SDH для передачи данных на скоростях до 10 Гбит/с включительно используется *модуляция с двумя состояниями света* — «свет включен/свет выключен» (OnOff Keying, **OOK**). При этом используется метод кодирования NRZ «без возвращения к нулю», когда последовательность единиц передается непрерывным световым лучом. Для обеспечения самосинхронизации приемника исходный код *скремблируется*, так что длинные последовательности единиц в нем не встречаются. На скорости 40 Гбит/с код OOK приводит к слишком широкому спектру сигнала, поэтому применяются более сложные коды, использующие амплитудную и фазовую модуляцию световой волны.

На рис. 8.9, б показано распределение функций SDH по типам оборудования SDH: регенераторы поддерживают только два нижних уровня, мультиплексоры вводавывода — три, а терминальные мультиплексоры ответственны за решение всего комплекса задач по доставке данных между двумя конечными пользователями.

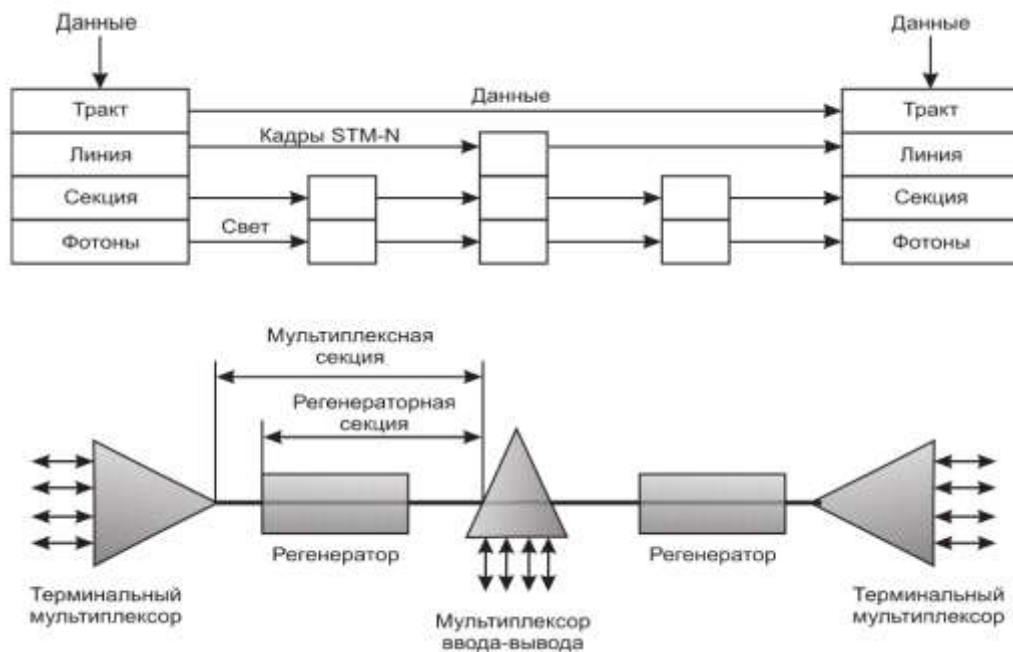


Рис. 8.9. Функциональные уровни технологии

Топологии сетей SDH

В сетях SDH наиболее часто используются кольца и линейные цепи мультиплексоров, но все большее применение также находит ячеистая топология, близкая к полносвязной.

Кольцо SDH строится из мультиплексоров вводавывода, имеющих, по крайней мере, по два агрегатных порта (рис. 8. 10, а). Пользовательские потоки вводятся в кольцо и выводятся из кольца через трибутарные порты, образуя двухточечные соединения (на рисунке показаны в качестве примера два таких соединения). Кольцо является классической регулярной топологией, обладающей потенциальной отказоустойчивостью — при однократном обрыве

кабеля или выходе из строя мультиплексора соединение сохранится, если его направить по кольцу в противоположном направлении. Кольцо обычно строится на основе кабеля с двумя оптическими волокнами, но иногда для повышения надежности и пропускной способности применяют четыре волокна.

Линейная цепь (рис. 8.10, б) — это последовательность мультиплексоров ввода-вывода с двумя терминальными мультиплексорами на окончаниях линии. Обычно сеть с топологией цепи применяется в тех случаях, когда узлы имеют соответствующее географическое расположение, например, вдоль магистрали железной дороги или трубопровода. Правда, в таких случаях может применяться и **плоское кольцо** (рис. 8.10, в), обеспечивающее более высокий уровень отказоустойчивости за счет двух дополнительных волокон в магистральном кабеле и по одному дополнительному агрегатному порту у терминальных мультиплексоров.

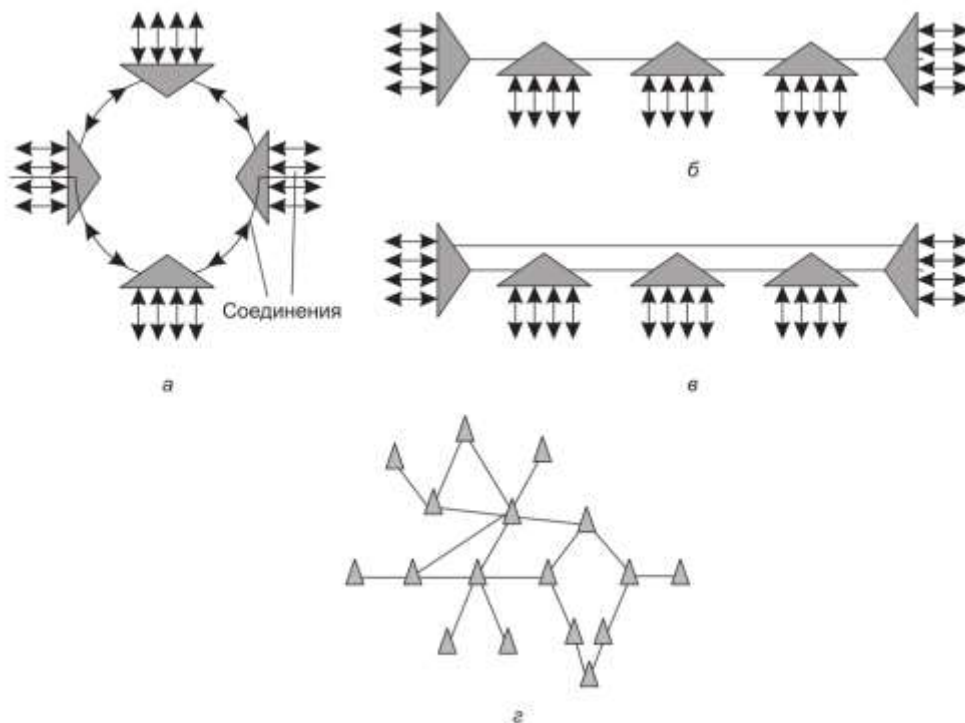


Рис. 8.10. Типовые топологии SDH

Наиболее общим случаем является **ячеистая топология** (рис. 8.10, г), при которой мультиплексоры соединяются друг с другом большим количеством агрегатных связей, за счет чего сеть может достичь очень высокой степени производительности и надежности.

В примере на рис. 8.11 линейная цепь образована четырьмя мультиплексорами MUX1 — MUX4, расположенными в географически разнесенных точках А, В, С и D. В линейной топологии схема мультиплексирования SDH позволяет устанавливать любые соединения между пользовательскими потоками. На рисунке показаны три таких соединения: соединение 1 между пользователями в терминальных точках А и D, соединение 2 между пользователями промежуточных точек В и С, соединение 3 между пользователем промежуточной точки С и терминальной точки D.

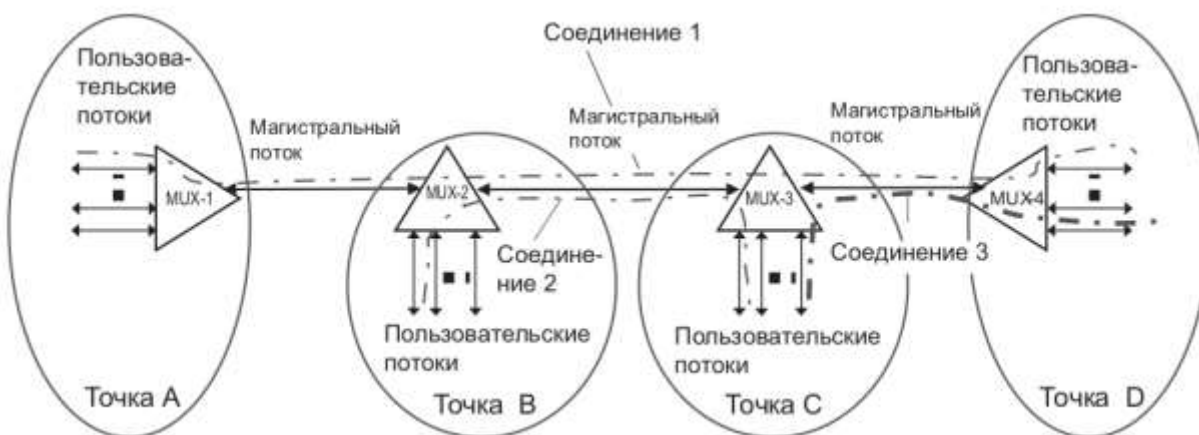


Рис. 8.11. Линейная цепь SDH

Иерархия скоростей

Так как сети SDH строились как магистральные сети для объединения сетей PDH, то иерархия скоростей PDH и стала иерархией скоростей, которую SDH поддерживает для соединений со своими клиентами, а именно: E1 (2,048), E2 (8,488), E3 (34,368) и E4 (139,264) Мбит/с. Поэтому для таких клиентов, например, коммутаторов телефонных сетей, для которых интерфейсы E1E4 являются «родными», сеть SDH является эффективным средством передачи данных. Клиенты же других типов, например, маршрутизаторы компьютерных сетей, должны подстраиваться и либо обзаводиться подобными интерфейсами или же использовать специальные адаптеры, которые преобразуют один или несколько интерфейсов Ethernet в интерфейсы PDH.

Иерархия скоростей магистральных соединений SDH (или собственно иерархия скоростей SDH) начинается там, где заканчивается иерархия скоростей ее клиента. То есть начальная скорость технологии SDH была выбрана так, чтобы она могла передавать один поток *наивысшей скорости* PDH E4 (139,264 Мбит/с). Эта начальная скорость и соответствующий ей кадр SDH получили название **STM1** (Synchronous Transport Module level 1) — *синхронный транспортный модуль уровня 1*. Этот модуль является основой системы SDH.

Однако некоторые клиенты могут иметь оборудование с менее скоростными интерфейсами, такими как E1, E2 или E3. Поэтому было решено сделать кадр STM1 способным передавать данные не только потока E4, но и всех остальных менее скоростных потоков PDH, то есть E1, E2 и E3. Стремление как можно эффективнее передавать потоки PDH и в *разнообразных их комбинациях* (то есть по несколько потоков каждого типа) привело к достаточно сложной структуре кадра STM1. Сложные методы интегрирования пользовательских потоков в кадр STM1 потребовали включения в кадр большего количества *дополнительной* служебной информации, необходимой для понимания его структуры при демультиплексировании.

Организация всех уровней иерархии скоростей выше первого, носящих общее название **STMN**, значительно проще, чем **STM1**, так как каждый последующий уровень допускает только один вариант

мультиплексирования: мультиплексирование четырех кадров предыдущего уровня в один кадр, и никаких других комбинаций потоков, как в случае уровня STM1, у него нет.

Всего было стандартизовано пять уровней скоростей SDH (табл. 8.1).

Таблица 8.1. Стандартизация уровней скоростей SDH

Обозначение	STM1	STM4 (STM1 × 4)	STM16 (STM1 × 16)	STM64 (STM1 × 64)	STM256 (STM1 × 256)
Скорость (Мб/с)	155,520	622,080	2488	9953	39 810

Заметим, что скорость передачи данных в канале STM1 (155,520 Мбит/с) *выше*, чем скорость E4 (139,264), это объясняется тем, что при неизменной величине цикла объем данных в кадре увеличился из-за включения в него дополнительной служебной информации (заголовков). Уровень STM256 (около 40 Гбит/с) является наивысшим уровнем стандарта SDH. Дальнейшее повышение скорости первичных сетей стало происходить в рамках технологии OTN.

Трафик компьютерных сетей не считался в 80е годы чемто приоритетным, поэтому иерархия скоростей SDH никак не соотносится с иерархией скоростей Ethernet, а это означает, что пропускная способность сетей SDH используется неэффективно при передаче компьютерного трафика. Например, в кадр STM1 можно поместить только один поток Ethernet 100 Мбит/с, это означает, что пропускная способность канала связи SDH используется компьютерными данными только на 65%. Аналогичная ситуация возникает и при передаче компьютерных потоков 1 Гбит/с по каналу STM16, когда из 2,5 Гбит/с используются только 2 Гбит/с.

Формат кадра SDH

Технология SDH выполняет мультиплексирование своих кадров на основе техники TDM — разделения магистрального канала по времени. Каждый кадр вышележащего уровня получается путем побайтного мультиплексирования кадров нижележащего уровня. Например, если на каждый вход мультиплексора приходят кадры STM1, которые мультиплексируются в один кадр следующего уровня скорости (STM4) побайтно, то в кадре STM4 последовательно чередуются байты из первого, второго, третьего и четвертого кадров STM1. Учитывая циклический характер кадра, в SDH принято изображать кадр в виде матрицы.



Рис. 8.12. Формат кадра STM1

Для кадра STM1 (рис. 8.12) такая матрица состоит из 9 строк, каждая из которых состоит из 270 байт. Строка включает 9 байт заголовка, 260 байт пользовательских данных и один служебный байт, принадлежащий *заголовку тракта POH*, о котором речь ниже.

Мультиплексирование в STMN

Как мы знаем, мультиплексирование модулей уровней *выше первого* выполняется в соответствии с простым правилом: каждый последующий уровень STMN получается мультиплексированием *четырёх блоков предыдущего уровня* $STM(N - 1)$.

Рассмотрим, например, как блок STM4 получается в результате мультиплексирования *четырёх* кадров STM1 (рис. 8.13). Для упрощения рисунка на нем не показан заголовок кадра POH. Мультиплексор принимает первый байт первого кадра STM1 и копирует его значение в первый байт кадра STM4. Затем он копирует значение первого байта второго кадра STM1 во второй байт кадра STM4, значение первого байта третьего кадра STM1 — в третий байт кадра STM4, а значение первого байта четвертого кадра STM1 — в четвертый байт кадра STM4. Далее этот цикл повторяется уже со вторыми байтами кадров STM1.

В результате содержимое кадра STM4 представляет собой чередующиеся байты кадров STM1, причем позиция байтов каждого из кадров STM1 известна и фиксирована, как это и должно быть при *TDM мультиплексировании*. Демультимплексирование может быть выполнено «на лету», например, для вывода из кадра STM4 третьего кадра STM1 достаточно копировать в выходной порт получателя байты, находящиеся в позициях (таймслотах), кратных трем.

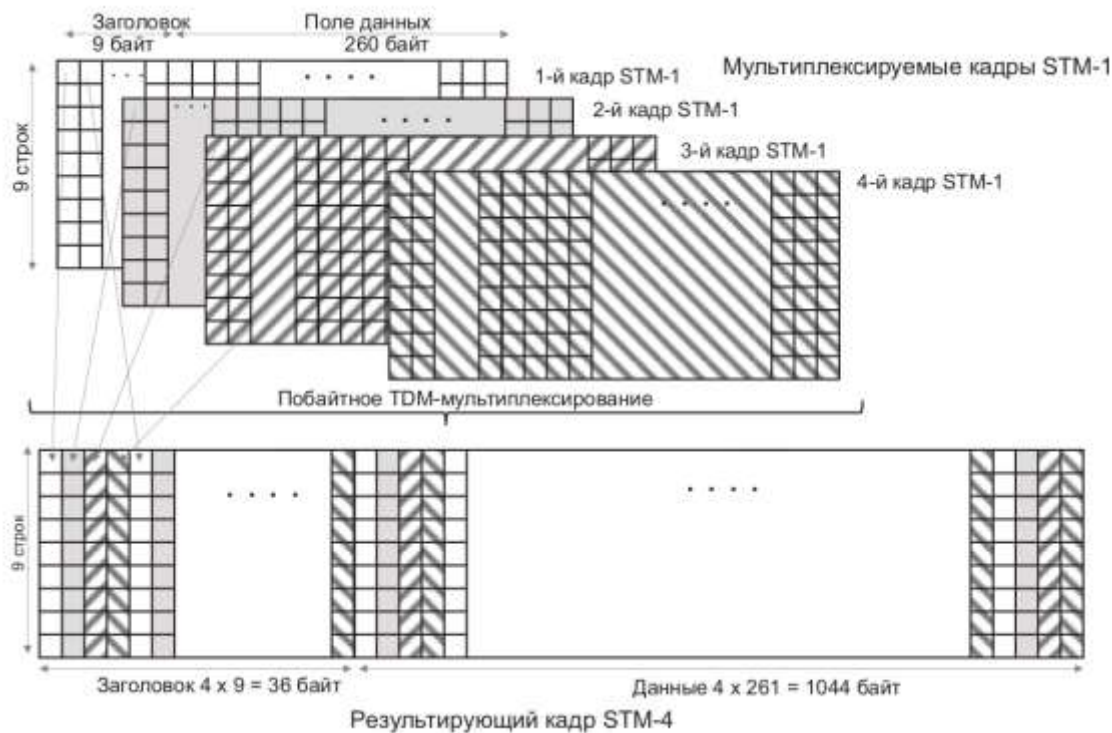


Рис. 8.13. Мультиплексирование четырех кадров STM1 в кадр STM4

Синхронность работы мультиплексоров SDH проявляется в том, что за каждый такт его работы на каждом его входе принимается один бит, а на выходе — передается один бит. Так как здесь байты являются неделимой единицей информации, то очередной пришедший байт может быть вынужден ожидать своего таймслота для передачи на выходной порт мультиплексора. Но эта задержка очень мала, например, для выходного кадра STM4 время передачи одного байта составляет 1,6 нс. Максимальное время ожидания таймслота равно времени передачи трех байтов, то есть 4,8 нс, что пренебрежимо мало по сравнению с временем цикла генерации оцифрованных замеров голоса 125 мкс.

Кадры старших уровней STM16, STM64 и STM256 формируются аналогично, так что размер поля заголовка кадра в строке кадра STMN равен $9 \times N$ байт, а размер поля данных — $260 \times N$ байт. Количество строк кадров любого уровня всегда равно 9.

Время цикла, или, что одно и то же, время приема данных от всех мультиплексируемых потоков и формирования из них кадра, — всегда равно 125 мкс^1 , независимо от уровня скорости STMN магистрального канала. Следовательно, размер кадра (включая заголовок) связан со скоростью следующим соотношением: $(STMN) \times 125/8$ байт (табл. 8.2).

Таблица 8.2. Соотношения размера кадра и скорости

Обозначение	STM1	STM4	STM16	STM64	STM256
-------------	------	------	-------	-------	--------

¹ Выбор времени цикла равным интервалу между замерами оцифрованного с частотой 8 кГц голоса показывает ориентацию технологии SDH прежде всего на качественную передачу телефонного трафика.

Скорость (Мб/с)	155,520	622,080	2488	9953	39 810
Размер кадра (байт)	2430	9720	38 880	155 520	622 080

Мультиплексирование в STM1

Мультиплексирование в STM1 занимает особое место. Кадр STM1 по сравнению с другими типами кадров STMN имеет более сложную структуру, позволяющую агрегировать в общий магистральный поток потоки PDH *различных* скоростей. STM1 является начальной, самой низкой скоростью, которая, тем не менее, соответствует *наивысшей скорости* PDH E4. Задача разработчиков технологии SDH состояла в том, чтобы сделать возможной передачу в кадре STM1 данные потоков с более *низкими скоростями*, например, несколько потоков E1, несколько потоков E2 и несколько потоков E3. В результате была создана схема (рис. 8.14), позволяющая гибко компоновать кадр STM1.

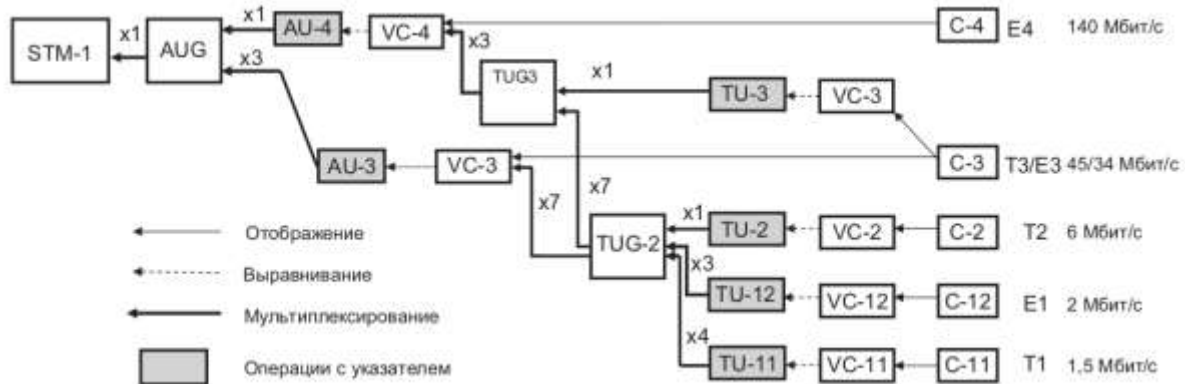


Рис. 8.14. Схема компоновки кадра STM1 из потоков PDH разной скорости

В схеме представлен каждый из потоков-клиентов PDH, включая как международную (E1, E3, E4), так и американскую (T1, T2, T3) версии. Чтобы попасть в кадр STM1, данные из потока клиента должны прежде всего быть представлены в виде пользовательского контейнера.

Контейнер пользовательских данных (Container, C) — это набор байтов пользователя, которые поступают в мультиплексор за время одного цикла 125 мкс и которыми мультиплексор оперирует как единым целым.

Схема показывает, что существуют различные пути попадания данных пользовательского контейнера каждого типа в кадр STM1 и что на этом пути данные контейнера претерпевают определенные преобразования в результате выполнения над ними операций *отображения, выравнивания и мультиплексирования*, получая при этом последовательно различные названия, такие как VC (виртуальный контейнер), TU, TUG, AU и AUG.

Первой операцией на пути данных пользователя в кадр STM1 является *отображение* пользовательских данных из контейнера пользователя в виртуальный контейнер.

Виртуальный контейнер (Virtual Container, **VC**) является единицей коммутации сети SDN, он передается между конечными точками соединения пользователей без изменения вместе со своим заголовком.

Как видно из схемы на рис. 8.14, одни виртуальные контейнеры, а именно VC3 и VC4, являются *структурированными*, то есть могут включать несколько мультиплексированных виртуальных контейнеров VC11, VC12 или VC2, а другие — VC11, VC12 и VC2 — всегда содержат только один пользовательский контейнер. Структура контейнеров VC4 и VC3 описывается в их заголовках, и мультиплексор использует эти данные при их демультимплексировании. Виртуальный контейнер вместе с полем указателя образует блок данных другого типа, называемый *трибутарным* или *трибным блоком TU* для контейнеров VC11, VC12 и VC2 и *административным блоком AU* для контейнеров VC3 и VC4. При побайтном мультиплексировании нескольких блоков TU образуется *группа блоков TUG*, а при мультиплексировании блоков AU — *группа административных блоков AUG* (блоки TU, TUG, AU и AUG имеют обозначения, соответствующие уровню виртуальных контейнеров, которые в них входят, например, блок TU12 или блок TUG3, блок AUG индекса не имеет, так как он может быть только одного уровня — верхнего). Блоки TUG и AUG своих заголовков не имеют.

В результате отображения пользовательских данных в данные виртуального контейнера к ним добавляется служебная информация в виде *заголовка тракта* (Path Overhead, *POH*), называемого также *маршрутным заголовком*, и байтов грубого выравнивания скоростей пользовательских данных и виртуального контейнера. *Заголовок POH* включает информацию, позволяющую выполнять различные полезные функции, такие как контроль по четности данных виртуального контейнера, индикацию в направлении передатчика об обнаружении ошибки по четности, а также информацию о типе и структуре виртуального контейнера. Заголовок POH располагается в первых байтах каждой строки виртуального контейнера. На рис. 8.15, *a* показана структура виртуального контейнера VC4.

Мультиплексирование в STM1, так же как и мультиплексирование кадров в STMN, выполняется побайтно. *Коэффициент мультиплексирования* ($\times 1$, $\times 3$ и др.) на схеме показывает, сколько блоков определенного типа мультиплексируется в блок следующего типа.

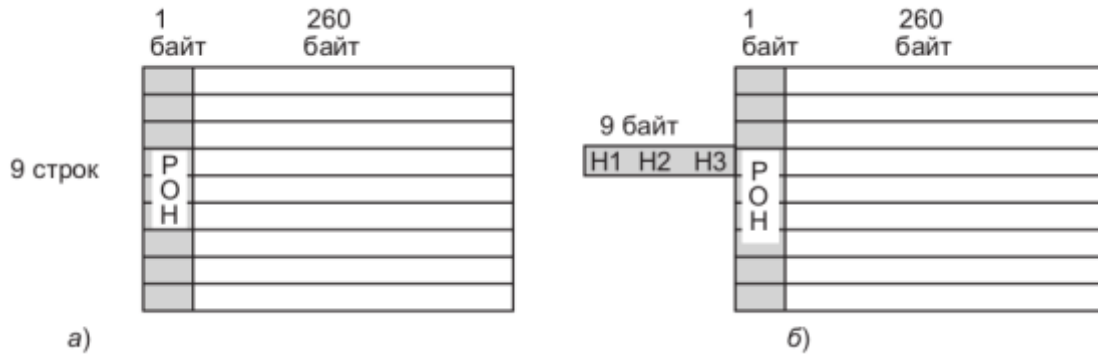


Рис. 8.15. Виртуальный контейнер VC4 (а) и его блок AU4 (б)

В любой блок, допускающий мультиплексирование, могут быть помещены блоки либо одного, либо другого типа, но *смешение типов не допускается*. Например, в блок TUG2 может быть мультиплексировано или 4 блока TU11, или 3 блока TU12, или 1 блок TU2. При этом два блока TUG2, один из которых построен из блоков TU11, а другой из блоков TU12, попрежнему считаются блоками одного типа.

Несмотря на это ограничение, предложенная схема мультиплексирования дает возможность переносить в кадре различные сочетания пользовательской нагрузки. Так, если нам необходимо передавать по несколько потоков E1 и T2, то для потоков E1 можно выбрать один или несколько из семи имеющихся блоков TUG2, помещая в них по 4 контейнера C11, содержащих потоки E1, а для потоков T2 выбрать оставшиеся блоки TUG2. Для кадра STM1, например, могут быть предложены следующие варианты: (а) 1 поток E4; (б) 63 потока E1; (в) 1 поток E3 и 42 потока E1 (возможны и другие варианты).

Нужно подчеркнуть, что все мультиплексоры сети SDH после их конфигурирования администратором настроены на какойто *один определенный* вариант мультиплексирования, который может быть получен из данной схемы.

Выравнивание

Выравнивание скоростей выполняется во время генерации каждого кадра STM1 и во время мультиплексирования. Существует грубое и тонкое выравнивание. В первом случае согласуют номинальную скорость пользовательских данных и номинальную скорость виртуального контейнера, которая всегда выше по двум причинам: во-первых, изза необходимости переносить дополнительные байты заголовка тракта POH, во-вторых, она должна быть такой, чтобы при мультиплексировании в блоки более высокого уровня помещалось целое число виртуальных контейнеров данного типа. Для этого случая используются *байты грубого выравнивания* из заголовка POH.

Тонкое выравнивание скоростей устраняет рассогласование скорости пользовательских данных (кадров PDH) и скорости передачи кадров STM1, которое является следствием несинхронности пользовательского оборудования PDH и оборудования SDH. Для тонкого выравнивания скоростей применяются указатели.

Концепция указателей — *ключевая* в технологии SDH, она заменяет принятое в PDH выравнивание скоростей асинхронных источников посредством битстаффинга. **Указатель** определяет текущее положение виртуального контейнера в кадре. С помощью указателя виртуальный контейнер может «смещаться» в определенных пределах внутри кадра, если скорость пользовательского потока несколько отличается от скорости кадра SDH, в который этот поток мультиплексируется.

Именно благодаря системе указателей мультиплексор находит положение пользовательских данных в синхронном потоке байтов кадров STMN и «на лету» извлекает их оттуда, чего механизм мультиплексирования, применяемый в PDH, делать не позволяет.

Поле указателя состоит из нескольких байтов (их количество зависит от уровня виртуального контейнера, на который он указывает). Значение указателя должно быть достаточно большим, чтобы указать смещение начала виртуального контейнера в кадре, в то же время в силу циклического характера передачи виртуальных контейнеров значение указателя не превышает размера виртуального контейнера.

Рассмотрим *принцип работы* указателя на примере выравнивая скорости виртуального контейнера VC4, который «плавает» относительно указателя блока AU4. На рис. 8.15, б) показан блок AU4, его *заголовок* состоит из 9 байт, которые делятся на *три подполя*, H1, H2 и H3, каждое по три байта, при этом они всегда предшествуют первому байту четвертой строки матрицы VC4. Структура блока AU4 выглядит несколько странно, с четвертой строкой, имеющей больший размер, чем остальные, но эта странность объясняется тем, что «лишние» байты четвертой строки в действительности размещаются в поле заголовка кадра STM1 (рис. 8.16).

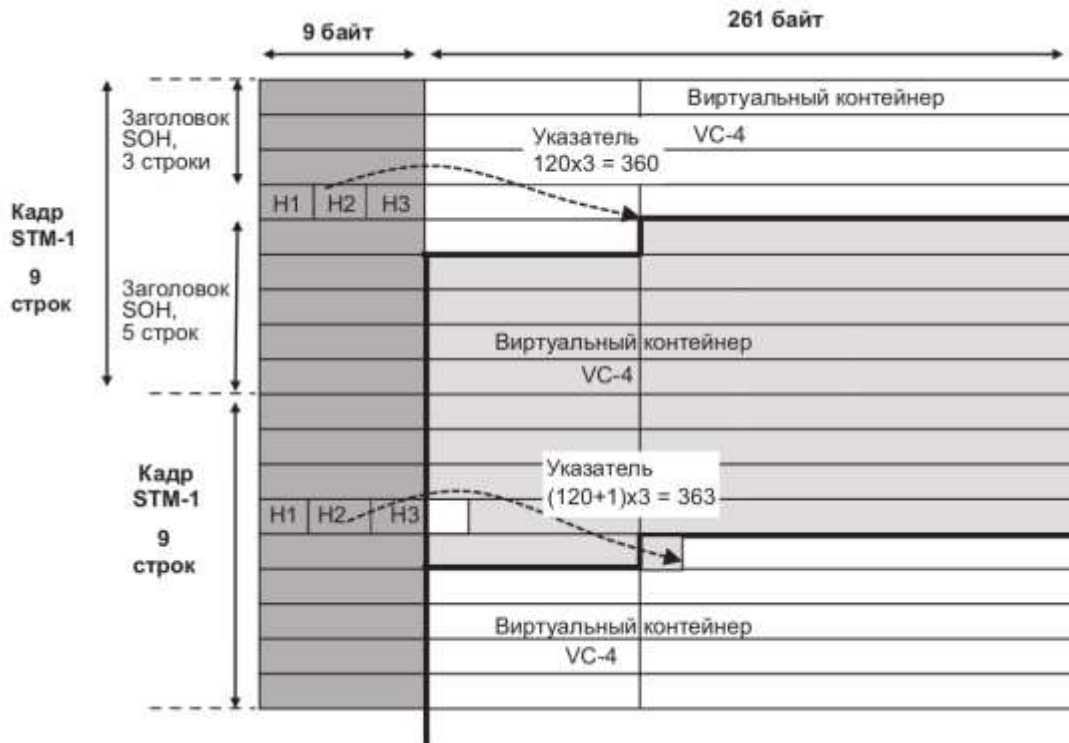


Рис. 8.16. Компенсация несинхронности пользовательского потока с помощью указателя

Кадр STM1 состоит из поля данных и заголовка. В поле данных помещается *виртуальный контейнер VC4* (формально, в соответствии со схемой мультиплексирования, поле данных кадра STM1 занимает блок AUG, но в случае размещения в нем блока AU4 блок AUG совпадает с блоком AU4, так как этот блок представлен в единственном экземпляре и мультиплексировать блоку AUG нечего).

Заголовок кадра STM1, называемый *транспортным заголовком*, или заголовком секции (Section Overhead, *SOH*), занимает 9 начальных байтов каждой из 9 строк кадра, кроме строки 4, которую он уступает полю *указателя блока AU4*. Заголовок содержит данные, необходимые для демultipлексирования содержащихся в кадре блоков данных пользователей, обнаружения ошибок и уведомления передатчика о них, мониторинга качества работы сети, а также синхробайты, необходимые для распознавания начала кадра и его синхронизации. С помощью байтов синхронизации мультиплексоры распространяют по сети сигналы от точных источников синхронизации, то есть участвуют в создании сети синхронизации.

Заголовки кадров STMN более высокого уровня представляют собой побайтно мультиплексированный набор заголовков входящих в них кадров более низкого уровня, и в конечном счете заголовков кадров STM1. Собственных полей кадры уровня выше 1го не имеют.

Под значение указателя отведены 10 бит полей H1 и H2, значение этих 10 бит рассматривается как целое положительное число. *Значение указателя, умноженное на 3, дает смещение в байтах начала виртуального контейнера, отсчитываемое от поля H3*. То есть когда указатель

равен 0, виртуальный контейнер VC4 начинается сразу же за полем НЗ. На рисунке показан другой случай, когда значение указателя равно 120 и виртуальный контейнер начинается с 360го байта после поля НЗ.

Коэффициент 3 при вычислении смещения нужен для того, чтобы смещение могло покрыть любые значения в пределах длины контейнера в 2340 байт, а десятиразрядное двоичное число для этого имеет недостаточную длину — его максимальное значение равно 1023. Умножение на 3 дает нам 3069, что избыточно, поэтому разрешенными значениями указателя являются числа от 0 до 782. Заметим, что отсчет начала виртуального контейнера от поля НЗ приводит к тому, что виртуальный контейнер VC4 всегда сдвинут относительно начала кадра STM1 на четыре строки, то есть располагается в двух последовательных кадрах.

Теперь посмотрим, как указатель работает в случае рассогласования скорости пользовательских данных и мультиплексора. Предположим, что скорость пользовательских данных ниже, чем скорость мультиплексора. Это неизбежно приведет к ситуации, что в какойто момент времени приема данных у мультиплексора просто не будет хватать их очередной порции для заполнения байтов виртуального контейнера. В таком случае мультиплексор вставляет три «холостых» (незначащих) байта в данные виртуального контейнера, после чего продолжает заполнение VC4 «подоспевшими» за время паузы пользовательскими данными. Указатель наращивается на единицу, что отражает запаздывание начала очередного контейнера VC4 на три байта, а факт вставки холостых трех байтов отмечается в поле Н1. Эта операция над указателем называется *положительным выравниванием*. В итоге средняя скорость отправляемых пользовательских данных становится равной скорости их поступления, причем без вставки дополнительных битов (битстафинга) в стиле PDH. В нашем примере значение указателя стало равным 121, что соответствует смещению начала виртуального контейнера на 363 байта.

Если же скорость поступления данных контейнера VC4 выше, чем скорость отправки кадра STM1, то у мультиплексора периодически возникает проблема размещения «лишних», то есть преждевременно пришедших байтов, для которых в поле VC4 нет места. В качестве дополнительного пространства используется трехбайтовое поле НЗ. Указатель при этом уменьшается на единицу, поэтому такая операция носит название *отрицательного выравнивания*.

Для выравнивания других типов виртуальных контейнеров используются аналогичные механизмы на основе указателей, хранящихся в заголовках соответствующих блоков.

Мы уже говорили о том, что мультиплексоры SDH в основном не буферизуют данные пользователей, передавая их побайтно и потактно на выходные порты. В то же время значения указателей и данные о структуре виртуальных контейнеров буферизуются, чтобы мультиплексор мог «на лету» выполнять операции мультиплексирования, демультимплексирования и коммутации. При изменении значения какоголибо указателя его новое значение заносится в буфер и применяется для нахождения начала очередного виртуального контейнера. Для контейнеров верхнего уровня VC4 и VC3 (в блоке AU3) начало контейнера находится непосредственно по значению указателя в 4й строке заголовка кадра STM1, а для контейнеров нижнего уровня VC11, VC12, VC2 (и VC3 с блоком TU2) — за две операции: сначала нужно

найти начало контейнера VC4 по его указателю, затем прочитать значение указателя контейнера нижнего уровня и найти положение этого контейнера относительно указателя в последовательности байтов контейнера верхнего уровня.

Коммутация в SDH

Теперь рассмотрим, каким образом сеть SDH *коммутирует* пользовательские соединения STM-N. Напомним, что, в отличие от компьютерных сетей с коммутацией пакетов, сети SDH являются сетями с *коммутацией каналов*, причем соединения между пользователями сети с помощью этих каналов носят полупостоянный характер, то есть не изменяются в течение длительного промежутка времени между операциями по переконфигурации сети, когда создаются новые соединения. Кроме того, каждое соединение канала характеризуется определенной и известной пропускной способностью.

Единицей коммутации в технологии SDH являются виртуальные контейнеры — именно они, а не кадры STM1, фигурируют в таблицах коммутации мультиплексоров сети. Следовательно, контейнеры должны быть однозначно *идентифицированы* в структуре кадра STM1. Не существует единого стандартного способа идентификации виртуальных контейнеров SDH, но на основе *имеющейся у администратора сети схемы* отображения и мультиплексирования это достаточно просто сделать, присвоив номера таймслотам контейнеров при их мультиплексировании в блоки более высокого уровня.

Пусть, например, пользовательские данные в нашей сети обрабатываются согласно схеме, фрагмент которой приведен на рис. 8.17, а номера таймслотов, выделенных каждому из мультиплексируемых блоков, соответствуют указанным на рисунке. Тогда набор чисел (2, 5, 3) однозначно идентифицирует виртуальный контейнер VC12.

Заметим, что при локализации байтов виртуального контейнера VC12 в кадре STM1 по его составному идентификатору мы использовали знание *конкретного* варианта отображения и мультиплексирования, назначенного данному мультиплексору администратором при конфигурировании. Без этого мы бы не смогли однозначно определить, какой тип контейнера задает идентификатор 2, 5, 3 и какой конкретно виртуальный контейнер из множества контейнеров этого типа имеется в виду.

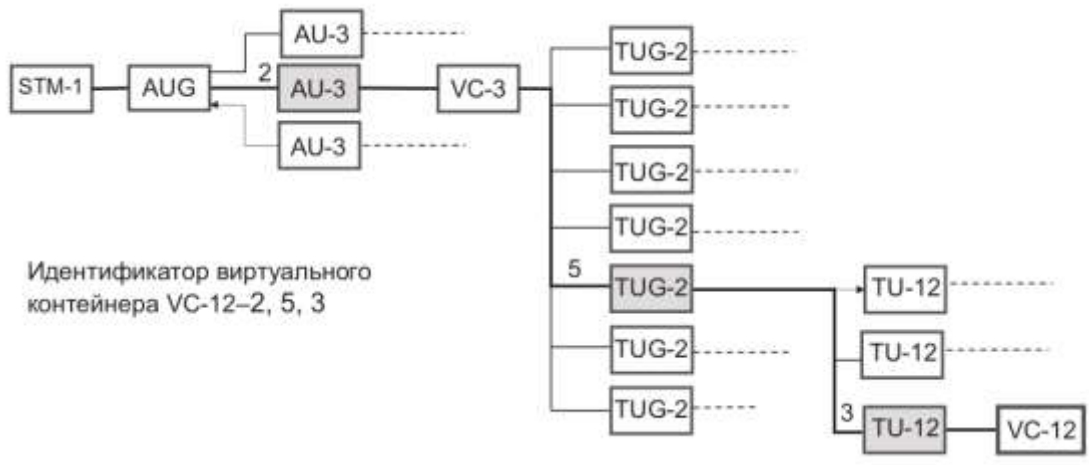


Рис. 8.17. Пример идентификации виртуального контейнера

Если добавить к идентификатору 2, 5, 3 виртуального контейнера номер порта, на который поступают или передаются его данные, например P3, то мы получим полный идентификатор P3, 2, 5, 3, который может использоваться в таблицах коммутации мультиплексора STMN.

Проиллюстрируем технику коммутации виртуальных контейнеров на примере сети (рис. 8.18).

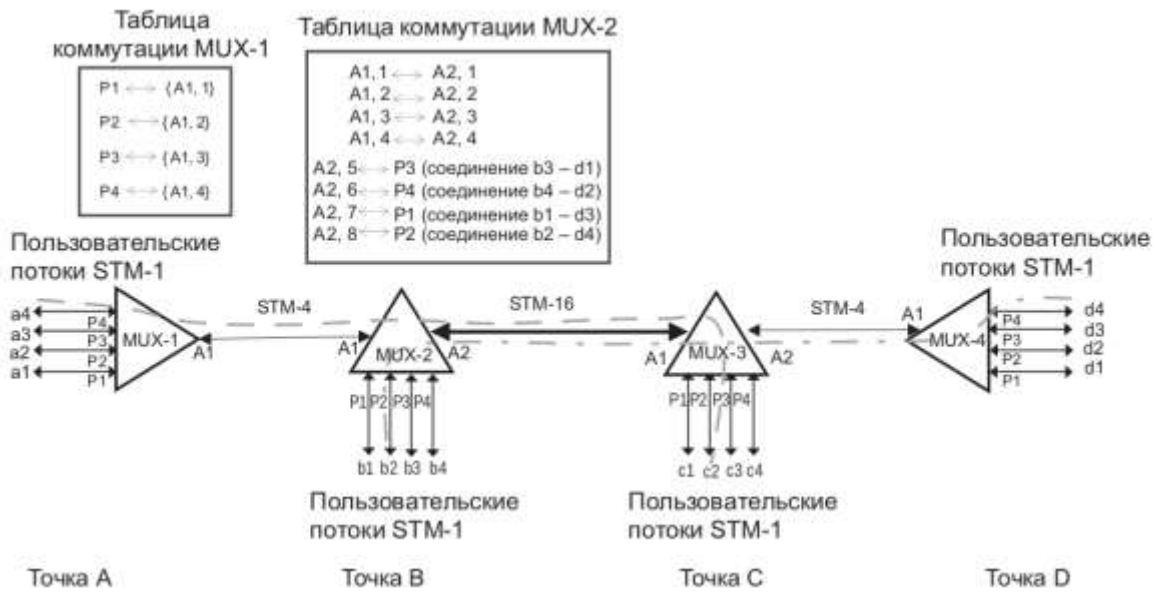


Рис. 8.18. Коммутация пользователей STM1 в сети STM4/STM16

Пусть у нас имеются следующие исходные данные для проектирования сети:

- пользователи сети a1—a4, b1—b4, c1—c4 и d1—d4 находятся в четырех географических точках А, В, С и D соответственно;

в каждой точке имеется по четыре пользователя и каждый из них имеет оборудование с интерфейсом STM1 для подключения к нашей сети;

пользователям необходимы следующие постоянные соединения скорости STM1: a1—c2, a2—c1, a3—c4, a4—c3, b1—d3, b2—d4, b3—d1, b4—d2.

На рисунке, с тем чтобы его не загромождать, показаны только два требуемых соединения: a4 — b2 и c2 — d4.

Исходя из скоростей пользователей понятно, что в каждой точке нужно установить мультиплексоры с четырьмя трибутарными портами STM1. Предположим, география расположения точек такова, что эффективным решением является топология сети типа «цепь» с терминальными мультиплексорами MUX1 и MUX4 в точках А и D и промежуточными мультиплексорами вводавывода в точках В и С.

Топология соединений пользователей такова, что на участках АВ и CD необходимо передавать четыре потока STM1 через агрегатные порты мультиплексоров, а на участке BC — восемь потоков STM1. В соответствии с таким распределением соединений мы можем выбрать для участков АВ и CD агрегатные порты STM4, а для участка BC — агрегатные порты STM16 как порты минимального уровня, которые способны передавать данные восьми потоков STM1.

Возможно, это не самое дальновидное решение, потому что на участках АВ и CD не остается свободной пропускной способности, изза чего наша сеть не сможет обслуживать новых пользователей. Тем не менее текущие потребности пользователей удовлетворить мы можем, если правильно составим и загрузим в мультиплексоры соответствующие таблицы коммутации.

Мы выбрали достаточно простой случай, требующий коммутации контейнеров VC4, поэтому идентификатор контейнера будет состоять:

- для портов STM1 — только из номера порта, так как в кадре STM1 имеется только один контейнер VC4, не требующий идентификации;
- для портов STM4 — из двух компонент: номера порта и номера STM1 в кадре STM4. Дальнейшая идентификация не нужна, поскольку виртуальный контейнер VC4 имеется в кадре STM1 в единственном экземпляре.

Если считать, что номер пользователя соответствует номеру порта P_n мультиплексора, а агрегатный порт обозначается как A1, то *таблица коммутации мультиплексора MUX1* может иметь следующий вид:

P1 ↔ {A1, 1},

P2 ↔ {A1, 2},

P3 ↔ {A1, 3},

P4 ↔ {A1, 4}.

Каждая запись говорит о том, что байты контейнера VC4 из кадра STM1 трибутарного порта Pk копируются в байты виртуального контейнера VC4 kго кадра STM1 составного кадра STM4 агрегатного порта A1.

Мультиплексор вводавывода MUX2 должен передавать кадры STM1, поступающие на его агрегатный порт A1, без изменения на агрегатный порт A2. Так как порт A2 — это порт STM16, то у него имеется возможность передавать данные четырех пользовательских потоков STM1 в любых четырех из 16 кадров STM1, входящих в состав кадра STM16. Пусть мультиплексор MUX2 выбрал из 16 кадров первые четыре. Тогда в *его таблице коммутации MUX2* могут быть сформированы следующие записи:

$$\{A1, 1\} \longleftrightarrow \{A2, 1\},$$
$$\{A1, 2\} \longleftrightarrow \{A2, 2\},$$
$$\{A1, 3\} \longleftrightarrow \{A2, 3\},$$
$$\{A1, 4\} \longleftrightarrow \{A2, 4\}.$$

Кроме того, мультиплексор MUX2 должен вывести из кадра STM16 четыре кадра STM1 от пользователей d1–d4, которым нужны соединения с пользователями b1–b4. Если предположить, что мультиплексор MUX3 сконфигурирован так, что он передает данные пользователя d1 в пятом кадре STM1 кадра STM16, пользователя d2 — в шестом, пользователя d3 — в седьмом и пользователя d4 — в восьмом, то дополнительно в его таблице коммутации должны появиться такие строки:

$$\{A2, 5\} \longleftrightarrow P3 \text{ (соединение } b3 \text{ — } d1\text{);}$$
$$\{A2, 6\} \longleftrightarrow P4 \text{ (соединение } b4 \text{ — } d2\text{);}$$
$$\{A2, 7\} \longleftrightarrow P1 \text{ (соединение } b1 \text{ — } d3\text{);}$$
$$\{A2, 8\} \longleftrightarrow P2 \text{ (соединение } b2 \text{ — } d4\text{).}$$

Читатель может сам завершить конфигурирование мультиплексоров сети, предложив таблицы коммутации мультиплексоров MUX3 и MUX4.

Технология SDH не предусматривает автоматическое нахождение маршрута, как это делают протоколы маршрутизации в пакетных сетях. В связи с этим таблица коммутации формируется оператором сети либо вручную, либо с помощью программного обеспечения системы управления сетью.

Отказоустойчивость сетей SDH

Одним из главных требований к магистральным сетям, наряду с высокой скоростью, является высокая надежность. *Надежность является одной из сильных сторон сетей SDH* и обеспечивается за счет ряда средств отказоустойчивости, которые позволяют сети быстро (менее чем за 50 миллисекунд) восстановить работоспособность в случае отказа какоголибо элемента сети — линии связи, порта мультиплексора или мультиплексора в целом. Такое быстрое переключение на резервный путь очень важно при передаче голосового трафика,

поскольку обеспечивает приемлемое для абонентов телефонной сети качество связи даже в условиях отказов элементов сети.

В SDH в качестве общего названия механизмов отказоустойчивости используется термин «**автоматическое защитное переключение**» (Automatic Protection Switching), который означает переход на резервный путь или резервный элемент мультиплексора при отказе основного. В сетях SDH применяются три схемы защиты:

- **Защита 1+1.** Это означает, что резервный элемент выполняет ту же работу, что и основной. Например, при защите трибутарного порта по схеме 1+1 трафик проходит как через рабочий порт (резервируемый), так и через защитный порт (резервный). Это наиболее качественный способ защиты, так как он обеспечивает минимальное время переключения на резервный элемент, который постоянно передает данные. Таким образом, приемной стороне нужно только начать использовать его данные, а не данные рабочего элемента. Это основная схема защиты сетей SDH.
- **Защита 1:1.** Эта схема подразумевает, что защитный элемент в нормальном режиме не выполняет функции защищаемого элемента, а переключается на них только в случае отказа, то есть этот процесс потенциально более медленный, чем 1+1, но более экономичный.
- **Защита 1:N.** Такая схема предусматривает выделение одного защитного элемента на N защищаемых. При отказе одного из защищаемых элементов его функции начинает выполнять защитный элемент, при этом остальные элементы остаются без защиты — до тех пор, пока отказавший элемент не будет заменен. Отметим, это самый экономичный, но вместе с тем и наименее качественный механизм защиты.

На основе этих схем строятся различные механизмы защиты сетей SDH.

В топологии линейной цепи SDH чаще всего применяется **защита мультиплексной секции** (Multiplex Section Protection). Эта защита работает между двумя смежными мультиплексорами. Она включает две пары портов и две линии связи: для рабочего канала (верхняя пара соединенных кабелем портов на рис. 8.19) конфигурируется защитный канал (нижняя пара портов, канал показан пунктиром).

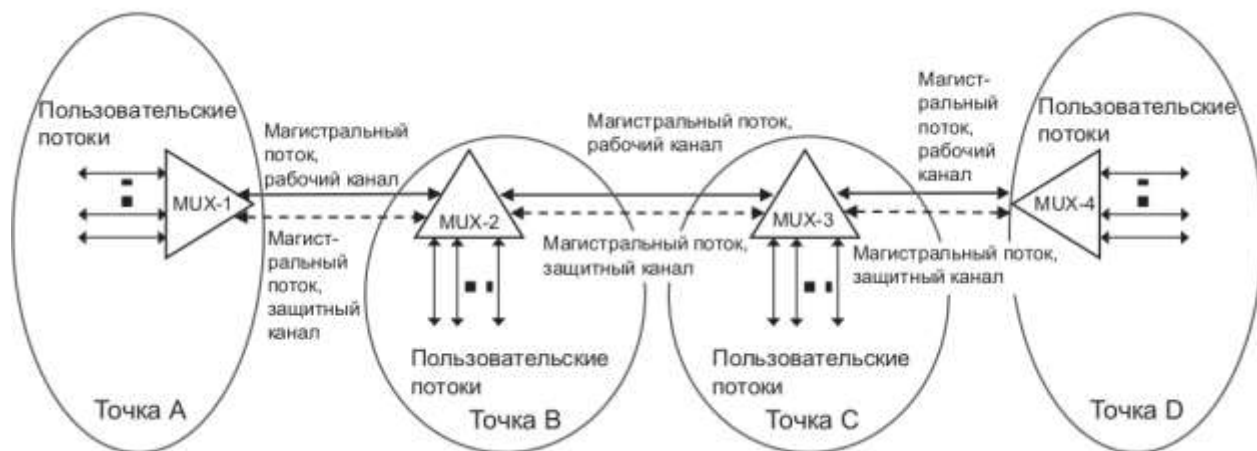


Рис. 8.19. Защита мультиплексной секции

В исходном состоянии весь трафик передается по обоим каналам (как по рабочему, так и по защитному), то есть применяется защита по схеме 1+1. В случае отсутствия или искажения принимаемых данных по рабочему каналу (изза отказа порта, ошибки сигнала, деградации сигнала и т. п.) *принимающий* мультиплексор фиксирует отказ и переходит на прием данных по защитному каналу.

Топология кольца является хорошей основой отказоустойчивой сети, так как она не требует дополнительных портов и кабелей для организации защитного пути: любые две точки на кольце всегда могут быть связаны двумя путями — по и против часовой стрелки. В кольце мультиплексоров SDN применяется механизм отказоустойчивости, называемый **разделяемой защитой мультиплексной секции в кольцевой топологии** (Multiplex Section Shared Protection Ring). Этот механизм основан на резервировании некоторой части пропускной способности кольца для *пользовательских соединений*. В этом его отличие от вышеописанного механизма защиты мультиплексной секции, который резервирует целиком все физическое соединение между соседними мультиплексорами, то есть образующие его порты и кабели.

Зарезервированная часть пропускной способности магистрали кольца не закреплена за определенными соединениями, а выделяется динамически в зависимости от ситуации, то есть это — защита по схеме 1:N.

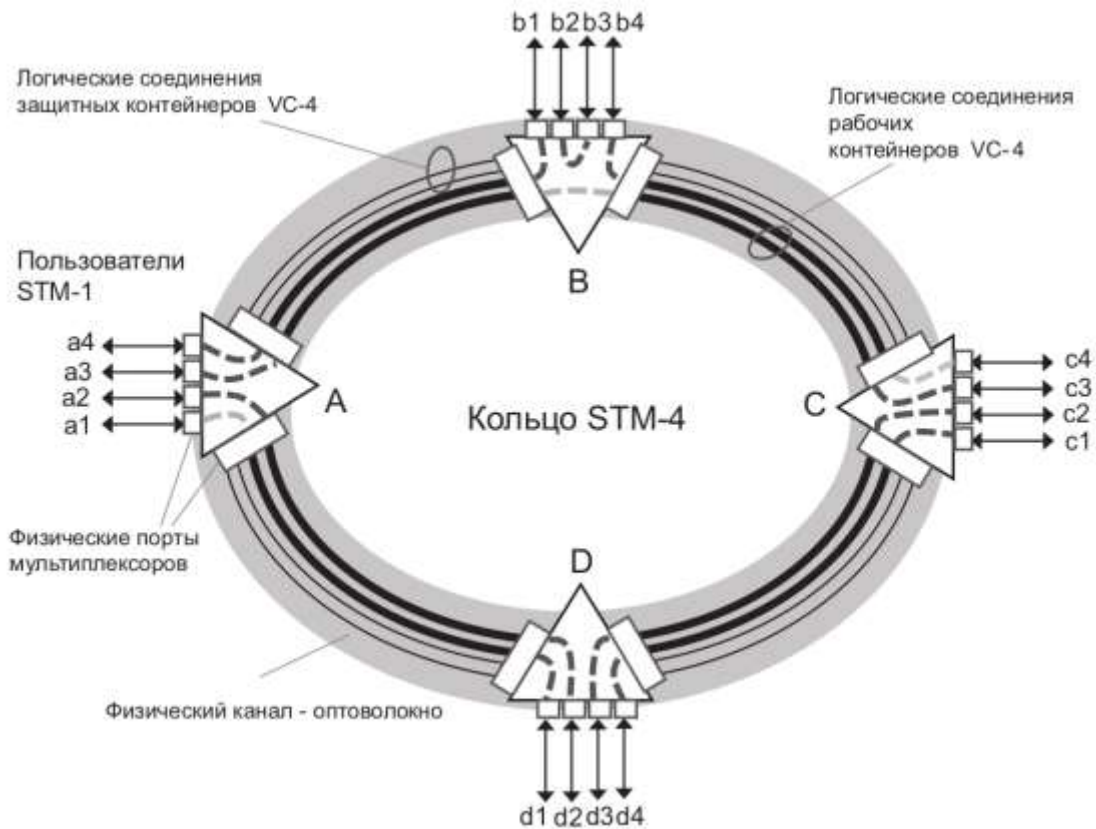


Рис. 8.20. Кольцо STM1 с пользователями STM1

Проиллюстрируем работу механизма *разделяемой защиты мультиплексной секции* на примере сети, изображенной на рис. 8.20. Эта сеть состоит из четырех соединенных в магистральное кольцо мультиплексоров вводавывода, A, B, C и D, к каждому из которых подключено оборудование четырех пользователей: a1—a4, b1—b4, c1—c4 и d1—d4. Оборудование пользователей работает на скорости STM1 и пользовательские данные инкапсулированы в контейнер VC4. Магистральные порты мультиплексоров работают на скорости STM4. Магистральный поток состоит из четырех контейнеров VC4, что позволяет выполнять различные варианты коммутации пользовательских контейнеров через магистраль. Половина пропускной способности магистрали (два контейнера VC4 кадра STM4, изображены толстыми линиями) отведена соединениям пользователей, а вторая половина (оставшиеся два контейнера VC4, изображены тонкими линиями) зарезервирована для защиты этих соединений. На рисунке показаны соединения между пользовательскими контейнерами, выполненными в сети: a3—c3, a4—b1 и др. Контейнеры магистрали VC4, которые используются в этих соединениях, являются рабочими контейнерами соединений, а неиспользованные — защитными.

На рис. 8.21 показана реакция *разделяемой защиты мультиплексной секции* на повреждение кабеля между мультиплексорами A и B.

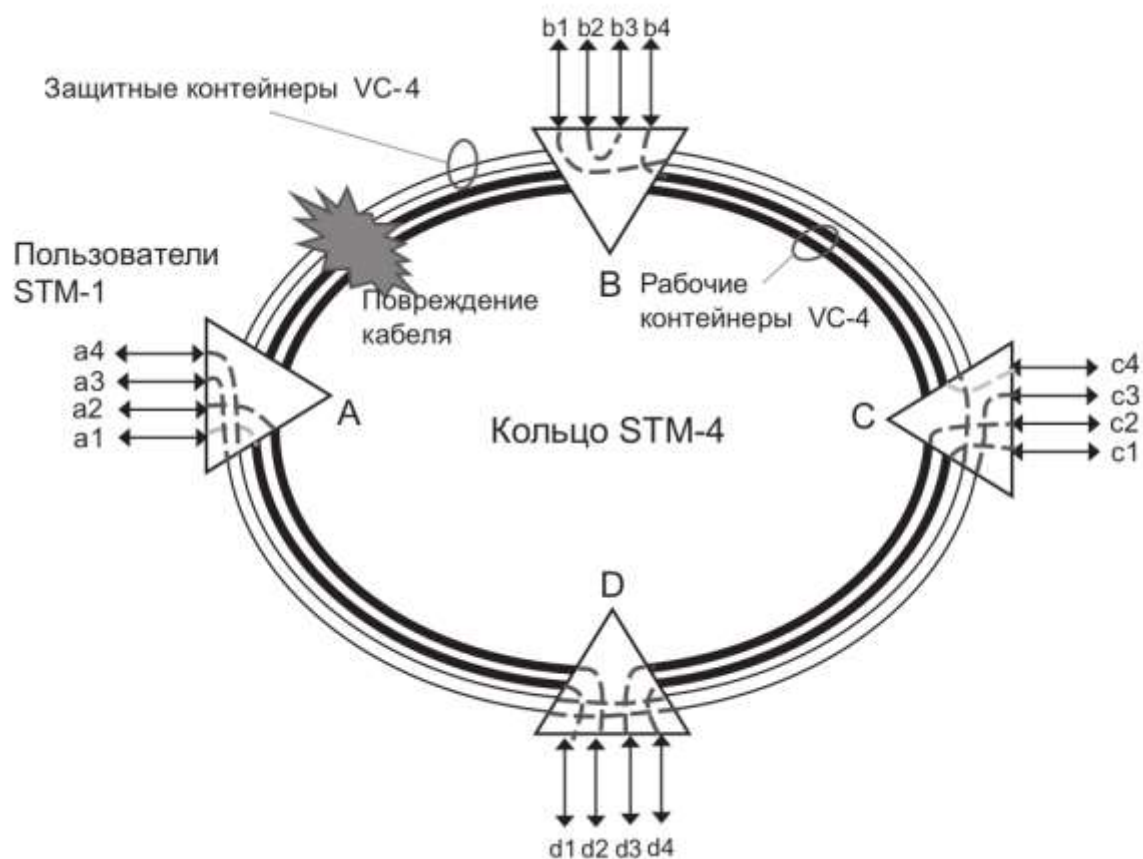


Рис. 8.21. Реконфигурация кольца при отказе

Мультимплексоры A и B, заметив прекращение получения сигналов по рабочим контейнерам соединений a3 — c3 и a4 — b1, реконфигурируют таблицы этих соединений, используя защитные контейнеры и неповрежденные связи кольца между мультимплексорами в направлении против часовой стрелки. В результате соединение a3 — c3 теперь проходит через промежуточный мультимплексор D, а не B, как это было до отказа. Соединение a4 — b1 стало проходить через промежуточные мультимплексоры D и C. В нашем примере защита организована по схеме 1:1, так как резервирование половины пропускной способности гарантирует, что для каждого соединения всегда найдется резервный контейнер в каждом мультимплексоре. Возможна реализации защиты по схеме 1:N, если в кольце выделить только Nю часть пропускной способности для резервирования.