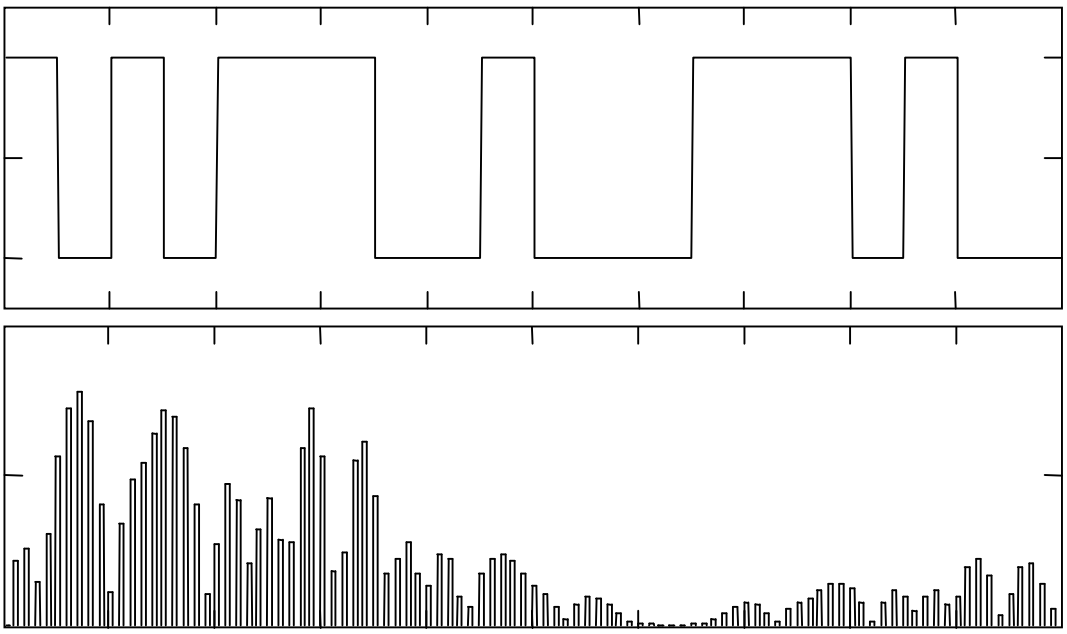
ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ
УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ (ТУСУР)

С.Г. Михальченко, Е.Ю. Агеев

ЭКСПЛУАТАЦИЯ И РАЗВИТИЕ КОМПЬЮТЕРНЫХ СИСТЕМ И СЕТЕЙ

Раздел 2

Учебное пособие



ТОМСК — 2007

Федеральное агентство по образованию

ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ (ТУСУР)

Кафедра промышленной электроники

С.Г. Михальченко, Е.Ю. Агеев

ЭКСПЛУАТАЦИЯ И РАЗВИТИЕ КОМПЬЮТЕРНЫХ СИСТЕМ И СЕТЕЙ

Раздел 2

Учебное пособие

2007

Михальченко С.Г., Агеев Е.Ю.
Эксплуатация и развитие компьютерных систем и сетей: Учебное пособие. В 2-х разделах. — Томск: Томский государственный университет систем управления и радиоэлектроники, 2007. — Раздел 2. — 213 с.

Рассмотрены вопросы эксплуатации и развития компьютерных систем и сетей, вопросы доступа к среде передачи информации, способы коммутации и мультиплексирования, методы кодирования и адресации сетевых устройств. Изучаются вопросы установки, настройки и обслуживания аппаратного и программного обеспечения компьютерных информационных сетей.

Предназначено для студентов вузов, обучающихся по специальности «Промышленная электроника».

ОГЛАВЛЕНИЕ

5 ПРОТОКОЛЫ И СТАНДАРТЫ ЛОКАЛЬНЫХ СЕТЕЙ

При организации взаимодействия узлов в локальных сетях основная роль отводится классическим технологиям Ethernet, Token Ring, FDDI основанным на использовании разделяемых сред. Разделяемые среды поддерживаются не только классическими технологиями локальных сетей, но и новыми — Fast Ethernet, 100VG-AnyLAN, Gigabit Ethernet.

Современной тенденцией является частичный или полный отказ от разделяемых сред: соединение узлов индивидуальными связями (например, в технологии ATM), широкое использование коммутируемых связей и микросегментации. Еще одна важная тенденция — применение полнодуплексного режима работы практически для всех технологий локальных сетей.

5.1 Структура стандартов IEEE 802x

Комитет IEEE 802.x разрабатывает стандарты, которые содержат рекомендации для проектирования нижних уровней локальных сетей — физического и канального.

Стандарты подкомитета 802.1 носят общий для всех технологий характер и постоянно пополняются. Наряду с определением локальных сетей и их свойств, стандартами межсетевого взаимодействия, описанием логики работы моста/коммутатора к результатам работы комитета относится и стандартизация виртуальных локальных сетей VLAN.

Подкомитет 802.2 разрабатывает и поддерживает стандарт LLC. Стандарты 802.3, 802.4, 802.5 описывают технологии локальных сетей, которые появились в результате улучшений фирменных технологий, легших в их основу, соответственно Ethernet, ArcNet, Token Ring.

Более поздние стандарты изначально разрабатывались не одной компанией, а группой заинтересованных компаний, а потом передавались в соответствующий подкомитет IEEE 802 для утверждения.

Сегодня комитет **IEEE 802** включает следующий ряд подкомитетов, в который входят как уже упомянутые, так и некоторые другие:

- **802.1** — Internetworking — объединение сетей;
- **802.2** — Logical Link Control — управление логической передачей данных;
- **802.3** — Ethernet с методом доступа CSMA/CD;
- **802.4** — Token Bus LAN — локальные сети с методом доступа Token Bus;
- **802.5** — Token Ring — локальные сети с методом доступа Token Ring;
- **802.6** — Metropolitan Area Network, MAN — сети мегаполисов;
- **802.7** — Broadband Technical Advisory Group — техническая консультационная группа по широкополосной передаче;
- **802.8** — Fiber Optic Technical Advisory Group — техническая консультационная группа по волоконно-оптическим сетям;
- **802.9** — Integrated Voice and data Networks — интегрированные сети передачи голоса и данных;
- **802.10** — Network Security — сетевая безопасность;
- **802.11** — Wireless Networks — беспроводные сети;
- **802.12** — Demand Priority Access LAN, 100VG-AnyLAN — локальные сети с методом доступа по требованию с приоритетами.

5.2 Канальный уровень

Канальный уровень (***Data Link Layer***) делится в локальных сетях на два подуровня:

- логической передачи данных (***Logical Link Control, LLC***);
- управления доступом к среде (***Media Access Control, MAC***).

Уровень MAC появился из-за существования в локальных сетях разделяемой среды передачи данных. Именно этот уровень обеспечивает корректное совместное использование общей среды, предоставляя ее в соответствии с определенным алгоритмом в распоряжение той или иной станции сети. После того как доступ к среде получен, ею может пользоваться более высокий уровень — уровень LLC, организующий передачу логических единиц данных — кадров информации, с различным уровнем качества транспортных услуг. В современных локальных сетях получи-

ли распространение несколько протоколов уровня MAC, реализующих различные алгоритмы доступа к разделяемой среде. Эти протоколы полностью определяют специфику таких технологий, как Ethernet, Fast Ethernet, Gigabit Ethernet, Token Ring, FDDI, 100VG-AnyLAN.

Уровень LLC отвечает за передачу кадров данных между узлами с различной степенью надежности, а также реализует функции интерфейса с прилегающим к нему сетевым уровнем. Именно через уровень LLC сетевой протокол запрашивает у канального уровня нужную ему транспортную операцию с нужным качеством. На уровне LLC существует несколько режимов работы, отличающихся наличием или отсутствием на этом уровне процедур восстановления кадров в случае их потери или искажения, то есть отличающихся качеством транспортных услуг этого уровня.

Протоколы уровней MAC и LLC взаимно независимы — каждый протокол уровня MAC может применяться с любым протоколом уровня LLC, и наоборот.

Стандарт IEEE 802.2. Уровень LLC

Стандарт *IEEE 802.2* описывает подуровень *LLC — Logical Link Control* — подуровень управления логической передачей данных.

Сервисы уровня LLC. Протокол LLC описывает для технологий локальных сетей нужное качество транспортной службы, передавая свои кадры либо дейтаграммным способом, либо с помощью процедур с установлением соединения и восстановлением кадров (см. рис. 5.1).

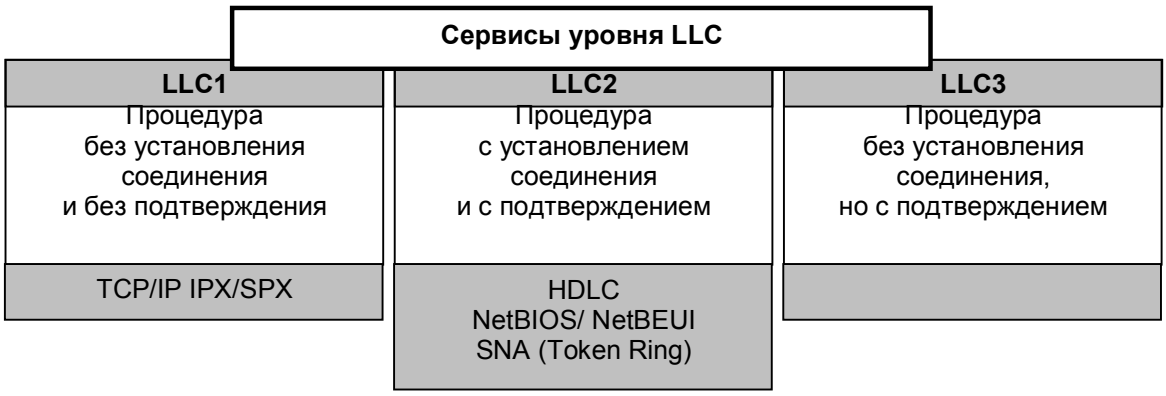


Рис. 5.1 — Уровень LLC. Стандарт IEEE 802.2

LLC предоставляет верхним уровням три типа сервисов: процедуру без установления соединения и без подтверждения (*LLC1*); процедуру с установлением соединения и подтверждением (*LLC2*); процедуру без установления соединения, но с подтверждением (*LLC3*). Этот набор процедур является общим для всех методов доступа к среде, определенных стандартами 802.3—802.5, а также стандартом FDDI и стандартом 802.12 на технологию 100VG-AnyLAN.

- *Процедура без установления соединения и без подтверждения LLC1* дает пользователю средства для передачи данных с минимумом издержек. Это дейтаграммный режим работы. Обычно этот вид процедуры используется, когда такие функции, как восстановление данных после ошибок и упорядочивание данных, выполняются протоколами вышележащих уровней, поэтому нет нужды дублировать их на уровне LLC. В стеке TCP/IP уровень LLC всегда работает в режиме LLC1, аналогично используется уровень LLC стеком IPX/SPX.
- *Процедура с установлением соединений и подтверждением LLC2* дает пользователю возможность установить логическое соединение перед началом передачи любого блока данных и, если это требуется, выполнить процедуры восстановления после ошибок и упорядочивание потока этих блоков в рамках установленного соединения. Протокол LLC2 во многом аналогичен протоколам семейства HDLC (LAP-B, LAP-D, LAP-M), которые применяются в глобальных сетях для обеспечения надежной передачи кадров на зашумленных линиях. Логический канал протокола LLC2 является дуплексным, так что данные могут передаваться в обоих направлениях. Протокол LLC в режиме с установлением соединения использует алгоритм скользящего окна.

Стек Microsoft/IBM, основанный на протоколе NetBIOS/NetBEUI в режиме с восстановлением потерянных и искаженных данных использует режим LLC2. Если же протокол NetBIOS/NetBEUI работает в дейтаграммном режиме, то протокол LLC работает в режиме LLC1. Режим LLC2 используется также стеком протоколов SNA с технологией Token Ring.

- В некоторых случаях (например, при использовании сетей в системах реального времени, управляющих промышленными объектами), когда временные издержки установления логическо-

го соединения перед отправкой данных неприемлемы, а подтверждение о корректности приема переданных данных необходимо, базовая процедура без установления соединения и без подтверждения не подходит. Для таких случаев предусмотрена дополнительная процедура, называемая *процедурой без установления соединения, но с подтверждением LLC3*.

Кадры уровня LLC. По своему назначению все кадры уровня LLC подразделяются на три типа — информационные, управляющие и ненумерованные.

- *Информационные кадры* (***Information***) предназначены для передачи информации в процедурах с установлением логического соединения LLC2 и должны обязательно содержать поле информации. В процессе передачи информационных блоков осуществляется их нумерация в режиме скользящего окна.
- *Управляющие кадры* (***Supervisory***) предназначены для передачи команд и ответов в процедурах с установлением логического соединения LLC2, в том числе запросов на повторную передачу искаженных информационных блоков.
- *Ненумерованные кадры* (***Unnumbered***) предназначены для передачи ненумерованных команд и ответов, выполняющих в процедурах без установления логического соединения передачу информации, идентификацию и тестирование LLC-уровня, а в процедурах с установлением логического соединения LLC2 — установление и разъединение логического соединения, а также информирование об ошибках. Все типы кадров уровня LLC имеют единый формат (рис. 5.2).

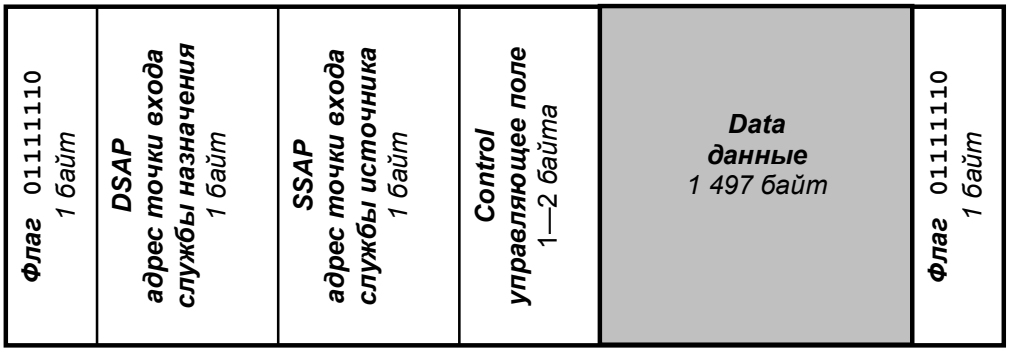


Рис. 5.2 — Формат кадра LLC

Кадр LLC вкладывается в кадр уровня MAC (кадр Ethernet, Token Ring, FDDI и т.д.), кадр LLC содержит поле данных и заголовок, который состоит из трех полей:

- адрес точки входа службы назначения (***Destination Service Access Point, DSAP***);
- адрес точки входа службы источника (***Source Service Access Point, SSAP***);
- управляющее поле (***Control***).

Поле данных кадра LLC предназначено для передачи по сети пакетов вышележащих протоколов. Поле данных может отсутствовать в управляющих кадрах и некоторых ненумерованных кадрах.

Адресные поля DSAP и SSAP занимают по 1 байту. Они позволяют указать, какая служба верхнего уровня пересылает данные с помощью этого кадра.

Поле управления имеет различную структуру для различных режимов LLC1 (1 байт) и LLC2, LLC3 — (2 байта), оно описывает параметры установления соединения, передает положительные и/или отрицательные квитанции, размер и границы скользящего окна и информацию о скорости передачи и о готовности приемника/передатчика.

Используя поле управления, протокол LLC с помощью управляющих кадров имеет возможность регулировать поток данных, поступающих от узлов сети. Это особенно важно для коммутируемых сетей, в которых нет разделяемой среды, автоматически тормозящей работу передатчика при высокой загрузке сети.

5.3 Доступ к среде передачи. Уровень MAC

Локальная вычислительная сеть включает единицы-десятки, реже сотни компьютеров, объединяемых средой передачи данных, общей для всех узлов. Одна из типичных сред передачи данных в ЛВС — отрезок (сегмент) коаксиального кабеля. К нему через аппаратуру окончания канала данных подключаются узлы (станции данных), которыми могут быть компьютеры и разделяемое узлами периферийное оборудование. Поскольку среда передачи данных общая, а запросы на сетевые обмены у узлов по-

являются асинхронно, то возникает проблема обеспечения доступа к сети. Эти вопросы решаются на канальном уровне OSI, вернее, в его подуровне MAC.

Доступом к сети называют взаимодействие узла сети со средой передачи данных для обмена информацией. Управление доступом к среде — это установление последовательности, в которой узлы получают полномочия по доступу к среде передачи данных. Главной особенностью той или иной реализации канального уровня (MAC) является метод доступа к среде передачи. Методы доступа могут быть случайными или детерминированными.

Метод множественного доступа с контролем несущей и обнаружением конфликтов CSMA/CD

Основным используемым методом случайного доступа является метод множественного доступа с контролем несущей и обнаружением конфликтов (столкновений, коллизий) — ***Carrier Sense Multiple Access/Collision Detection*** (***CSMA/CD***).

Этот метод основан на контроле (прослушивании) несущей и устранении конфликтов, возникающих из-за попыток одновременного начала передачи двумя или более станциями. Устранение столкновений осуществляется посредством прекращения передачи конфликтующими узлами и повторением попыток захвата линии каждым из этих узлов через некоторый случайный отрезок времени (см. рис. 5.3).

Данный метод доступа используется только в сетях с общей средой передачи данных — это шинная или звездообразная топология в кабельных линиях, либо общий частотный диапазон радиоэфира. Все компьютеры такой сети имеют возможность передавать и/или принимать данные в общей среде передачи. Говорят, что станции работают в режиме *коллективного доступа* ***(multiply-access, MA)***.

Все данные, передаваемые по сети, помещаются в кадры определенной структуры и снабжаются уникальным адресом станции назначения (MAC-адрес, зашитый в ПЗУ каждого сетевого адаптера). Затем кадр передается по кабелю. Все станции, подключенные к кабелю, могут распознать факт передачи кадра, и та станция, которая узнает собственный адрес в заголовках кадра,

записывает его содержимое в свой внутренний буфер и обрабатывает полученные данные.

Все узлы, имеющие данные для передачи по сети, контролируют состояние линии передачи данных. Прежде чем начать передачу станция некоторое короткое время прослушивает среду (***CS — Carrier Sense,*** *прослушивание несущей*). Если линия свободна, то в ней отсутствуют электрические колебания (для *Ethernet* — частоты 10 МГц) и станция, начинает передачу сигнала. Любая другая станция при обнаружении электрических колебаний в линии откладывает посылку своего пакета до момента окончания ведущейся передачи.

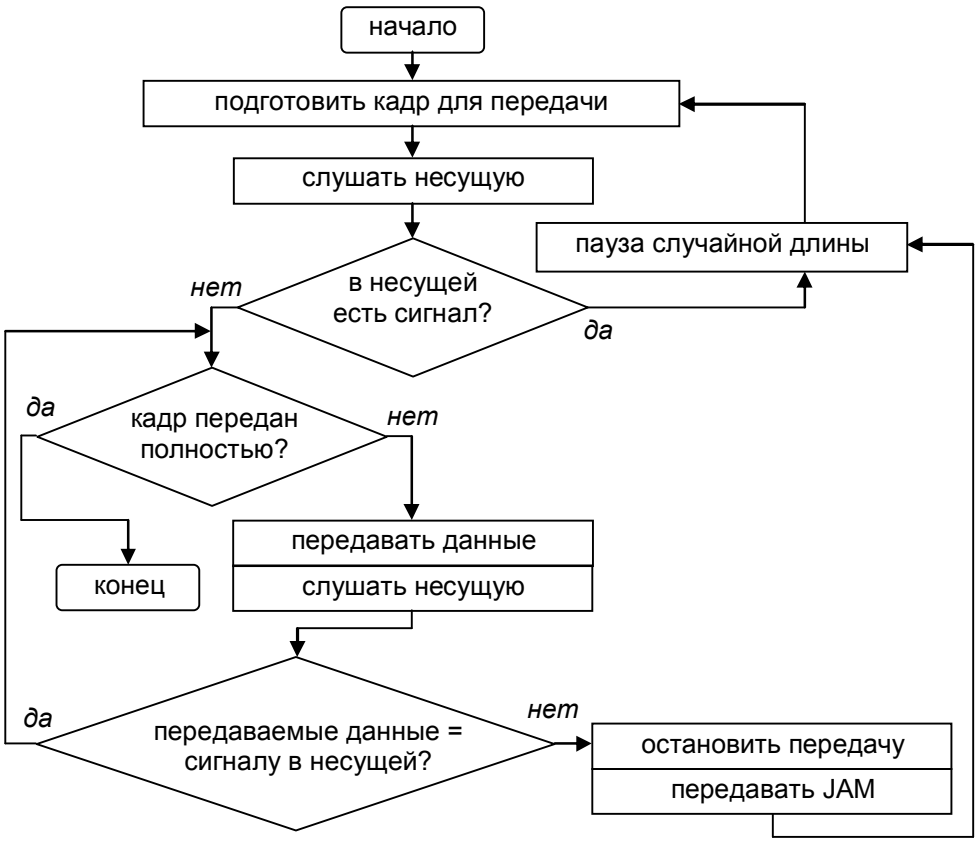


Рис. 5.3 — MAC уровень. Упрощенный алгоритм доступа к среде

После окончания передачи по кабелю станция должна выждать небольшую дополнительную паузу, называемую *межкадровым интервалом* (***interframe gap***), что позволяет узлу назначения принять и обработать передаваемый кадр, и после этого начать передачу следующего кадра. Все станции, при обнаружении колебаний в линии воспринимают их как передачу пакета и пытаются

принимать и распознавать информацию. Приняв заголовок передаваемого кадра, все активные устройства сети определяют MAC-адрес станции назначения и, в зависимости от этого, загружают пакет целиком в свой буфер или прекращают прием пакета.

Но даже при таком алгоритме две станции одновременно могут решить, что по шине в данный момент времени нет передачи, и начать одновременно передавать свои кадры. Говорят, что при этом происходит ***коллизия***, так как содержимое обоих кадров смешивается в общей среде передачи, что приводит к искажению информации. *Конфликты* (*столкновения,* ***коллизии)*** возникают, когда два или большее число узлов одновременно пытаются захватить линию. Понятие *одновременность событий* в связи с конечной скоростью распространения сигналов по линии конкретизируется как распределение событий по времени не более чем на величину *2d*, называемую окном столкновений, где *d* — время прохождения сигналов между конфликтующими станциями. Если какие-либо узлы начали передачу в окне столкновений, то наложение сигналов этих узлов друг на друга приводит к распространению по сети искаженных данных, это искажение и используется для *обнаружения конфликта* (***CD — Collision detection***).

Обнаружив коллизию, передающая станция (только она может определить, что пакет передается неверно, и информация перемешалась) вместо остатка пакета начинает передавать случайную помеху (***JAM***) продолжительностью 32—48 бит. Это проделывается для того, чтобы коллизия продолжалась достаточно долго и все передающие (и принимающие) станции ее обнаружили и прекратили передачу (прием). После завершения передачи *JAM*-помехи станция прекращает передачу на время случайное время t_d — задержку. Инициализация генератора случайных чисел происходит при запуске сетевой карты, и в ней участвует уникальный сетевой адрес адаптера.

В случае повторных коллизий существует максимально возможное *число попыток повторной передачи кадра* (***attempt limit***), которое равно 16, при достижении этого предела фиксируется ошибка передачи кадра, сообщение о которой передается протоколу верхнего уровня. В случае нескольких коллизий подряд происходит перезапуск с перезагрузкой генератора случайных чисел. Время паузы после *n*-ой коллизии полагается равным

случайному целому числу из диапазона [0, 2n]. Так величина диапазона растет до 10 попытки (напомним, что их не может быть больше 16), а далее диапазон остается равным [0, 1024]. Такое временное расписание длительности паузы называется *усеченным двоичным экспоненциальным алгоритмом отсрочки* (***truncated binary exponential back off***). Пауза всегда составляет целое число интервалов отсрочки.

Из описания метода доступа видно, что он носит вероятностный характер, и вероятность успешного получения в свое распоряжение общей среды зависит от загруженности сети, то есть от интенсивности возникновения в станциях потребности передачи кадров.

Таким образом, первый источник потерь времени a, следовательно, и снижение пропускной способности сети — это время на обнаружение несущей. В случае столкновения, второй источник — это передача помехи и вынужденный простой на случайный период времени, а третий — это время, затраченное на передачу испорченного пакета. Поскольку количество столкновений растет с увеличением нагрузки, то наступает момент, когда все полезное время уходит на вышеперечисленные потери. В результате пропускная способность падает до нуля (см. рис. 5.4). При больших входных загрузках из-за коллизий сначала наступает насыщение, а затем и резкий спад пропускной способности (*Ethernet collapse*). Это свойство сетей с CSMA/CD дает определенные преимущества сетям с маркерным доступом: Token Ring, FDDI и др., где пропускная способность не бывает нулевой.

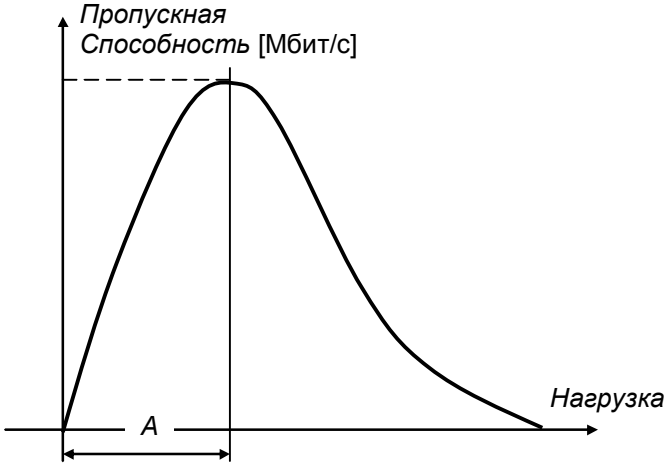


Рис. 5.4 — Зависимость пропускной способности сети со схемой доступа CSMA/CD от суммарной загрузки

Реально максимум пропускной способности, которую может обеспечить сеть с соревновательным методом доступа, составляет 40—60 % от номинальной, т.е. 4—6 Мбит/с для 10Base или 40—60 Мбит/с для 100Base соответственно.

Появление стандарта Full Duplex, разработанного в рамках проекта 100Base-Tx, улучшило ситуацию, позволив повысить пропускную способность сетей Ethernet практически до 100 %. При этом коллизии отсутствуют, а регулирование трафика происходит за счет передачи коротких служебных пакетов *Pause* (пауза) от ведущего участника соединения (обычно от коммутатора) к ведомому (серверу или рабочей станции). Этот пакет имеет зарезервированные значения величин «длина/тип и адрес приемника», он обрабатывается только на канальном уровне и не передается протоколам верхнего уровня.

Метод множественного доступа с контролем несущей и предотвращением конфликтов CSMA/CA

Метод ***CSMA/CA*** — (*carrier sense multiple access with collision avoid*) — множественный доступ с контролем несущей и предотвращением конфликтов. При этом способе станция прежде, чем передать данные, оповещает о своих намерениях другие станции. При этом методе увеличивается объем передаваемых служебных сообщений. Этот метод используется в RadioEthernet, в архитектуре сетей корпорации Apple (AppleTalk, Local Talk). Программное обеспечение сетей Apple встроено в операционную систему Macintosh.

Маркерное кольцо. IEEE 802.5

Среди *детерминированных методов доступа* к сети преобладают маркерные методы доступа (IEEE 802.4 — Token Bus — локальные сети с методом доступа маркерная шина, IEEE 802.5 — Token Ring — метод доступа маркерное кольцо). ***Маркерные методы*** основаны на передаче полномочий на доступ к сети одной из станций при помощи специального информационного пакета, называемого *маркером* (*token*). Это может происходить централизовано (требуется топология типа *звезда* и активное центральное устройство) или децентрализовано (станции, подключенные в кольцо, просто передают маркер друг другу). Наиболее известные стандарты, использующие доступ с передачей маркера — это 100VG AnyLan, ARCNet, Token Ring, FDDI.

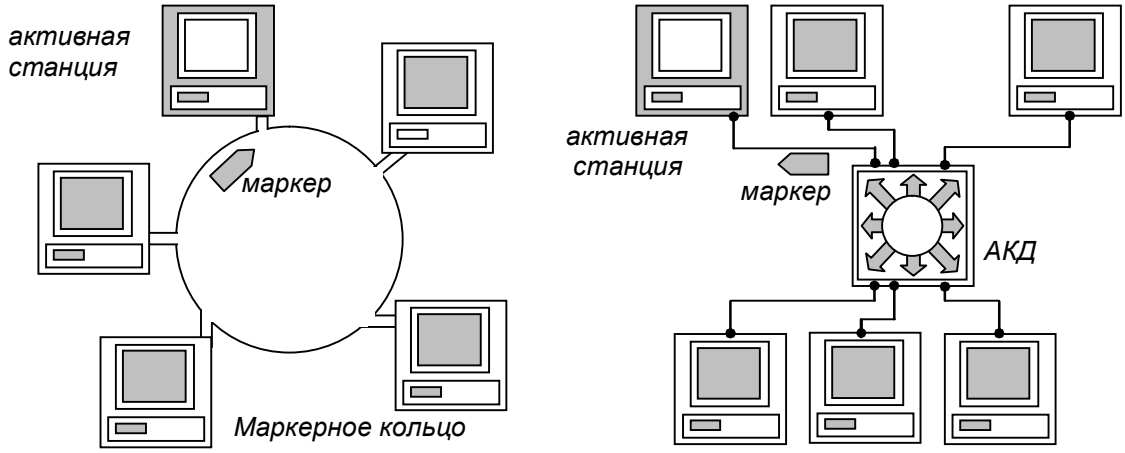


Рис. 5.6 — Передача маркера

Применяется ряд разновидностей маркерных методов доступа. Например, в эстафетном методе передача маркера выполняется в порядке очередности; в способе селекторного опроса (квантированной передачи) сервер (или коммутатор) опрашивает станции и передает полномочие одной из тех, которые готовы к передаче (100VG). В кольцевых одноранговых сетях широко применяется тактируемый маркерный доступ, при котором маркер циркулирует по кольцу и используется узлами для передачи своих данных (Token Ring, FDDI).

Маркерная шина. IEEE 802.4

Среда ***ArcNet*** (*Attached resource computer Network*) была разработана Datapoint Corporation в 1977 году. Первые платы ArcNet были выпущены в 1983году. Это простая, гибкая, недорогая сетевая архитектура для сетей масштаба группы. Организует логическое кольцо, используя общую среду передачи. По логическому кольцу передается маркер. Устройство, получившее маркер, имеет право на передачу порции данных в канал. Стандартный кадр ArcNet может содержать до 508 байт данных. В ArcNet Plus эта величина увеличена до 4 096 байт. Принимает данные то устройство, чей адрес указан в блоке данных. Каждому подключенному устройству присваивается номер. Последовательность обхода маркера определяется номерами устройств. Первые сети ArcNet использовали скорость передачи 2.5 Мбит/с. Скорость передачи ArcNet Plus доведена до 20 Мбит/с. Внешне может выглядеть как

звезда и как общая шина. В первом случае общая среда передачи реализуется внутри концентратора.

Приоритетный доступ. IEEE 802.12

При этом способе концентратор, получив одновременно два запроса, отдает предпочтение тому, который имеет более высокий приоритет.

100VG (*Voice Grade*) *AnyLan*. Эта технология объединяет в себе стандарты и идеи Ethernet и Token Ring, разрабатывается в рамках стандарта IEEE 802.12. Использует общую среду передачи. Технология доступа реализуется в виде системы с опросом. Интеллектуальный концентратор (*hub*) опрашивает подключенные к нему узлы. Узлу, выставившему запрос на передачу, разрешается передача данных. При наличии запросов от нескольких узлов очередность передачи определяется в соответствии с их приоритетами. Технология 100VG AnyLan имеет следующие возможности:

- скорость передачи данных более 100Мбит/с;
- поддержка структурированной кабельной системы на основе витой пары категории 3, 4, 5 и оптоволоконного кабеля;
- метод доступа — по приоритету запроса (различают только два уровня приоритета);
- поддержка концентратором средств фильтрации персонально адресованных кадров (для повышения степени конфиденциальности);
- поддержка передачи кадров Ethernet и Token Ring.

5.4 Технология Ethernet

Ethernet — это самый распространенный на сегодняшний день стандарт локальных сетей. На основе стандарта Ethernet был разработан стандарт ***IEEE 802.3***. В зависимости от типа физической среды стандарт IEEE 802.3 имеет различные модификации — 10Base-5, 10Base-2, 10Base-T, 10Base-F.

Для передачи двоичной информации по кабелю для всех вариантов физического уровня технологии Ethernet используется модификация манчестерского кода. Все виды стандартов Ethernet используют один и тот же метод разделения среды — метод

CSMA/CD — метод общего доступа к среде с контролем несущей и обнаружением коллизий (рис. 5.5).

Параметры протокола Ethernet подбирались таким образом, чтобы при нормальной работе узлов сети коллизии всегда четко распознавались (эти параметры приведены в табл. 5.1). Именно из этих соображений минимальная длина поля данных кадра должна быть не менее 46 байт (что вместе со служебными полями дает минимальную длину кадра в 72 байта или 576 бит).

Длина кабельной системы выбирается таким образом, чтобы за время передачи кадра минимальной длины сигнал коллизии успел бы распространиться до самого дальнего узла сети. Поэтому для скорости передачи данных 10 Мб/с, используемой в стандартах Ethernet, максимальное расстояние между двумя любыми узлами сети не должно превышать 2 500 м. Так же на технологию Ethernet накладывается ограничение по количеству узлов в сети — оно не должно быть более 1 024 узлов.

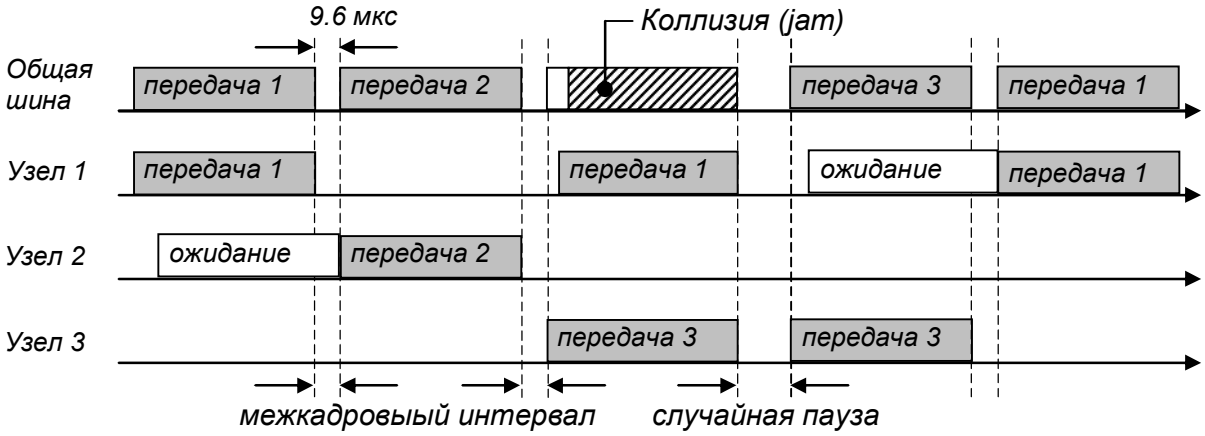


Рис. 5.5 — Метод случайного доступа CSMA/CD

В стандартах 802.3 большинство временных интервалов измеряется в количестве *битовых интервалов* [*bt*], которые для битовой скорости 10 Мб/с составляют 0.1 мкс и равны времени передачи одного бита.

Интервал отсрочки (***slot time***) — это время, в течение которого станция гарантированно может узнать, что в сети нет коллизии. Это время тесно связано с другим важным временным параметром сети — *окном коллизий* (***collision window***). Окно коллизий равно времени двукратного прохождения сигнала между самыми

удаленными узлами сети — наихудшему случаю задержки, при которой станция еще может обнаружить, что произошла коллизия. Интервал отсрочки выбирается равным величине окна коллизий плюс некоторая дополнительная величина задержки для гарантии:

интервал отсрочки = окно коллизий + дополнительная задержка

Величина интервала отсрочки в стандарте 802.3 равняется 512 bt, и эта величина рассчитана для максимальной длины коаксиального кабеля в 2.5 км. Величина 512 bt определяет и минимальную длину кадра в 64 байта, так как при кадрах меньшей длины станция может передать кадр и не успеть заметить факт возникновения коллизии из-за того, что искаженные коллизией сигналы дойдут до станции в наихудшем случае после завершения передачи. Такой кадр будет просто потерян.

Значения основных параметров процедуры, обеспечивающей передачу кадра стандарта 802.3 приведено в табл. 5.1.

Таблица 5.1

Битовая скорость	10 Мб/с
Интервал отсрочки	512 bt
Межкадровый интервал	9.6 мкс
Максимальное число попыток передачи	16
Максимальное число возрастания диапазона паузы	10
Длина jam-последовательности	32 бита
Максимальная длина кадра (без преамбулы)	1518 байтов
Минимальная длина кадра (без преамбулы)	64 байта (512 бит)
Длина преамбулы	64 бита
Мин. длина случайной паузы после коллизии	0 bt
Макс. длина случайной паузы после коллизии	524 000 bt
Макс. Расстояние между станциями сети	2 500 м
Макс. Число станций в сети	1 024

Учитывая приведенные параметры, нетрудно рассчитать максимальную производительность сегмента Ethernet в таких единицах, как число *переданных пакетов минимальной длины в секунду — **packets-per-second*** [*pps*].

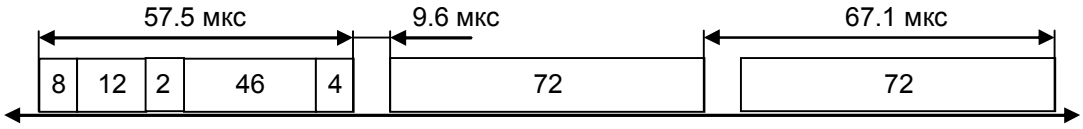


Рис. 5.6 — Ethernet. Кадры минимальной длины

Так как размер пакета минимальной длины вместе с преамбулой составляет 64+8 = 72 байта или 576 битов, то на его передачу затрачивается 57.6 мкс. Прибавив межкадровый интервал в 9.6 мкс, получаем, что период следования минимальных пакетов равен 67.2 мкс. Это соответствует максимально возможной пропускной способности сегмента Ethernet:

$$C_П^{min} = 14\,880 \text{ pps.} \qquad (5.1)$$

Кадры максимальной длины технологии Ethernet имеют поле длины 1 500 байт, что вместе со служебной информацией дает 1 518 байт, а с преамбулой составляет 1 526 байт или 12 208 бит. Максимально возможная пропускная способность сегмента Ethernet для кадров максимальной длины составляет:

$$C_П^{max} = 813 \text{ pps.} \qquad (5.2)$$

Очевидно, что при работе с большими кадрами нагрузка на мосты, коммутаторы и маршрутизаторы довольно ощутимо снижается.

Теперь рассчитаем, какой максимальной полезной пропускной способностью в бит в секунду обладают сегменты Ethernet при использовании кадров разного размера. Под ***полезной пропускной способностью*** протокола понимается скорость передачи пользовательских данных, которые переносятся полем данных кадра. Эта пропускная способность всегда меньше номинальной битовой скорости протокола Ethernet за счет нескольких факторов:

- служебной информации кадра;
- межкадровых интервалов;
- ожидания доступа к среде.

Для кадров минимальной длины полезная пропускная способность равна:

$$C_П^{min} = 14880 * 46 * 8 = 5.48 \text{ Мбит/с.} \qquad (5.3)$$

Это намного меньше 10 Мбит/с, но следует учесть, что кадры минимальной длины используются в основном для передачи

квитанций, так что к передаче собственно данных файлов эта скорость отношения не имеет. Для кадров максимальной длины полезная пропускная способность равна:

$$C_П^{max} = 813 * 1500 * 8 = 9.76 \text{ Мбит/с}, \quad (5.4)$$

что весьма близко к номинальной скорости протокола. К сожалению, это возможно только в том случае, когда двум взаимодействующим узлам в сети Ethernet другие узлы не мешают, что бывает крайне редко.

При использовании кадров среднего размера с полем данных в 512 байт пропускная способность сети составит

$$C_П = 9.29 \text{ Мбит/с}, \quad (5.5)$$

что тоже достаточно близко к предельной пропускной способности.

Отношение текущей пропускной способности сети к ее максимальной пропускной способности называется коэффициентом использования сети (***network utilization***). При отсутствии коллизий и ожидания доступа коэффициент использования сети зависит от размера поля данных кадра и имеет максимальное значение 0,976 при передаче кадров максимальной длины. При значениях этого коэффициента свыше 0.5 полезная пропускная способность сети резко падает: из-за роста интенсивности коллизий, а также увеличения времени ожидания.

Форматы кадров технологии Ethernet

Стандарт на технологию Ethernet, описанный в документе 802.3, дает описание единственного формата кадра MAC-уровня, но на практике в сетях Ethernet на канальном уровне используются заголовки 4-х типов. Ниже приводится описание всех четырех модификаций заголовков кадров Ethernet (причем под заголовком кадра понимается весь набор полей, которые относятся к канальному уровню):

- Кадр 802.3/LLC (или кадр Novell 802.2).
- Кадр Raw 802.3 (или кадр Novell 802.3).
- Кадр Ethernet DIX (или кадр Ethernet II).
- Кадр Ethernet SNAP.

Стандарт 802.3 определяет восемь полей заголовка (см. рис. 5.7):

Рис. 5.7 — Кадр Ethernet MAC-уровня

- ***Поле преамбулы*** состоит из семи байтов синхронизирующих данных. Каждый байт содержит одну и ту же последовательность битов — **10101010**. При манчестерском кодировании эта комбинация представляется в физической среде периодическим волновым сигналом. Преамбула используется для того, чтобы дать время и возможность схемам приемопередатчиков (transceiver) прийти в устойчивый синхронизм с принимаемыми тактовыми сигналами.
- ***Начальный ограничитель*** кадра состоит из одного байта с набором битов **10101011**. Появление этой комбинации является указанием на предстоящий прием кадра.
- ***Адрес получателя*** — может быть длиной 2 или 6 байтов (MAC-адрес получателя). Первый бит адреса получателя — это признак того, является адрес индивидуальным или групповым: если **0**, то адрес указывает на определенную станцию, если **1**, то это групповой адрес нескольких (возможно всех) станций сети. При широковещательной адресации все биты поля адреса устанавливаются в **1**. Общепринятым является использование 6-байтовых адресов.
- ***Адрес отправителя*** — 2-х или 6-ти байтовое поле, содержащее адрес станции отправителя. Первый бит — всегда имеет значение 0.
- Двухбайтовое ***поле длины*** определяет длину поля данных в кадре.
- ***Поле данных*** может содержать от 0 до 1500 байт. Но если длина поля меньше 46 байт, то используется следующее поле — поле заполнения, чтобы дополнить кадр до минимально допустимой длины.

- ***Поле заполнения*** состоит из такого количества байтов заполнителей, которое обеспечивает определенную минимальную длину поля данных (46 байт). Это обеспечивает корректную работу механизма обнаружения коллизий. Если длина поля данных достаточна, то поле заполнения в кадре не появляется.
- ***Поле контрольной суммы*** — 4 байта, содержащие значение, которое вычисляется по циклическому методу с полиномом CRC-32. После получения кадра рабочая станция выполняет собственное вычисление контрольной суммы для этого кадра, сравнивает полученное значение со значением поля контрольной суммы и, таким образом, определяет, не искажен ли полученный кадр. При вычислении CRC используется образующий полином:

$$g(x) = x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x + 1.$$

Кадр 802.3 является кадром MAC-подуровня, в соответствии со стандартом 802.2 в его поле данных вкладывается кадр подуровня LLC с удаленными флагами начала и конца кадра. Формат кадра LLC был описан выше. Так как кадр LLC имеет заголовок длиной 3 байта, то максимальный размер поля данных уменьшается до 1497 байт.

Кадр Raw 802.3 (то есть грубый вариант 802.3) или кадр Novell 802.3 это кадр MAC-подуровня стандарта 802.3, но без вложенного кадра подуровня LLC. Компания Novell долгое время не использовала служебные поля кадра LLC в своей операционной системе NetWare из-за отсутствия необходимости идентифицировать тип информации, вложенной в поле данных — там всегда находился пакет протокола IPX, долгое время бывшего единственным протоколом сетевого уровня в ОС NetWare. Теперь, когда необходимость идентификации протокола верхнего уровня появилась, компания Novell стала использовать возможность инкапсуляции в кадр MAC-подуровня кадра LLC, то есть использовать стандартные кадры 802.3/LLC.

Кадр стандарта Ethernet DIX, называемый также кадром Ethernet II, похож на кадр Raw 802.3 тем, что он также не использует заголовки подуровня LLC, но отличается тем, что на месте поля длины в нем определено поле типа протокола (поле Type). Это поле предназначено для тех же целей, что и поля DSAP и SSAP кадра LLC — для указания типа протокола верхнего уров-

ня, вложившего свой пакет в поле данных этого кадра. Для кодирования типа протокола используются значения, превышающие значение максимальной длины поля данных, равное 1500, поэтому кадры Ethernet II и 802.3 легко различимы.

Еще одним популярным форматом кадра является кадр Ethernet SNAP (SNAP — SubNetwork Access Protocol, протокол доступа к подсетям). Кадр Ethernet SNAP определен в стандарте 802.2H и представляет собой расширение кадра 802.3 путем введения дополнительного поля идентификатора организации, которое может использоваться для ограничения доступа к сети компьютеров других организаций.

В табл. 5.2 приведены данные о том, какие типы кадров Ethernet обычно поддерживают реализации популярных протоколов сетевого уровня.

Таблица 5.2

Тип кадра	***Сетевые протоколы***
Ethernet II	IPX, IP, AppleTalk Phase I
Ethernet 802.3	IPX
Ethernet 802.2	IPX, FTAM
Ethernet SNAP	IPX, IP, AppleTalk Phase II

Спецификации физической среды Ethernet

Физические спецификации технологии Ethernet на сегодняшний день включают следующие среды передачи данных:

10Base-5 — коаксиальный кабель диаметром 0.5 дюйма, называемый *толстым* коаксиалом. Имеет волновое сопротивление 50 Ом. Максимальная длина сегмента — 500 м (без повторителей).

10Base-2 — коаксиальный кабель диаметром 0.25 дюйма, называемый *тонким* коаксиалом. Имеет волновое сопротивление 50 Ом. Максимальная длина сегмента — 185 метров (без повторителей).

10Base-T — кабель на основе неэкранированной витой пары. Образует звездообразную топологию с концентратором. Расстояние между концентратором и конечным узлом — не более 100 м.

10Base-F — оптоволоконный кабель. Топология аналогична стандарту на витой паре. Имеется несколько вариантов этой спецификации — *FOIRL*, *10Base-FL*, *10Base-FB*.

Число 10 обозначает битовую скорость передачи данных этих стандартов — 10 Мб/с, а слово Base — метод передачи на одной базовой частоте 10 МГц (в отличие от стандартов, использующих несколько несущих частот, которые называются *broadband* — широкополосными).

Схема взаимодействия различных подуровней при реализации протокола IEEE 802.3 показана на рис. 5.8. Выше уровня логического канала (LLC) размещаются верхние уровни OSI, включая прикладной. Через интерфейс AUI данные передаются с использованием манчестерского кода.

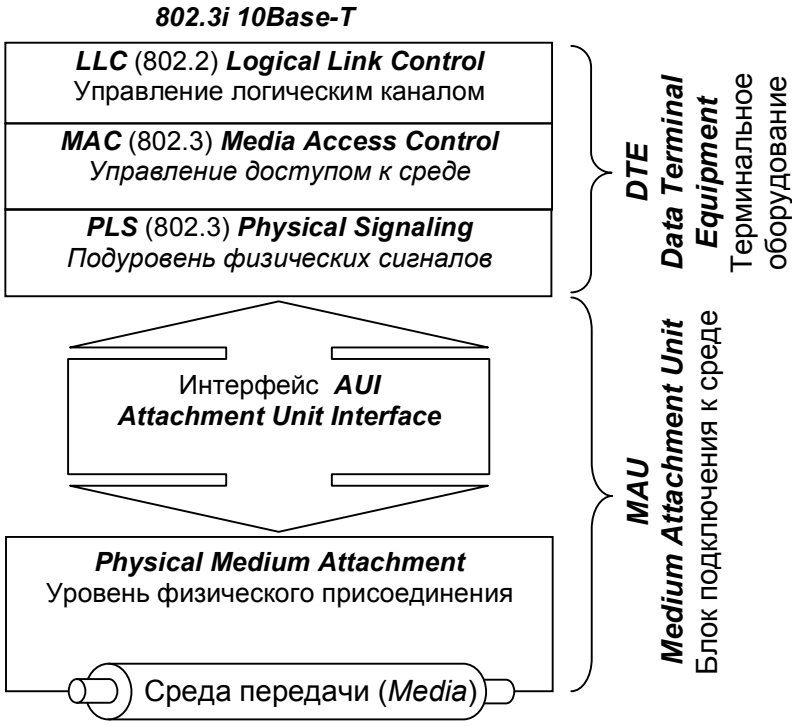


Рис. 5.8 — Схема взаимодействия подуровней 802.3 (CSMA/CD)

Манчестерский код объединяет в битовом сигнале передачу данных и синхронизацию. Каждый бит-символ делится на две части, причем вторая часть всегда является инверсной по отношению первой. В первой половине кодируемый сигнал представлен в логически дополнительном виде, а во второй — в обычном.

Таким образом, сигнал логического **0** характеризуется в первой половине высоким уровнем, а во второй — низким. Соответственно сигнал единицы **1** характеризуется в первой половине бит-символа низким уровнем, а во второй — высоким. Примеры форм сигналов при манчестерском кодировании представлены на рис. 5.9.

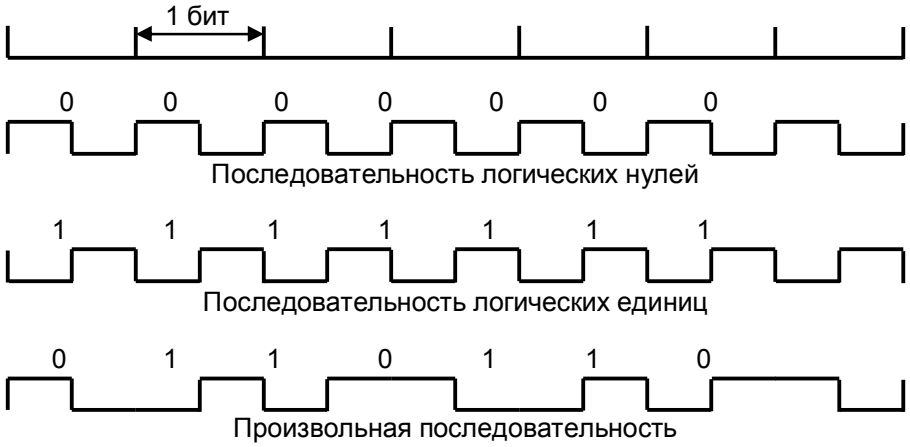


Рис. 5.9 — Примеры кодировки с использованием манчестерского кода

Исторически первые сети технологии Ethernet были созданы на коаксиальном кабеле диаметром 0.5 дюйма. В дальнейшем были определены и другие спецификации физического уровня для стандарта Ethernet, позволяющие использовать различные среды передачи данных в качестве общей шины. Метод доступа CSMA/CD и все временные параметры Ethernet остаются одними и теми же для любой спецификации физической среды.

Стандарт 10Base-5

Стандарт 10Base-5 соответствует экспериментальной сети Ethernet фирмы Xerox и может считаться классическим. Он использует в качестве среды передачи данных коаксиальный кабель с диаметром центрального медного провода 2,17 мм и внешним диаметром около 10 мм (толстый Ethernet).

Кабель используется как моноканал для всех станций. Сегмент кабеля имеет максимальную длину 500 м (без повторителей) и должен иметь на концах согласующие терминаторы сопротив-

лением 50 Ом, поглощающие распространяющиеся по кабелю сигналы и препятствующие возникновению отраженных сигналов.

Станция должна подключаться к кабелю при помощи приемопередатчика — трансивера. Трансивер устанавливается непосредственно на кабеле и питается от сетевого адаптера компьютера (рис. 5.10). Трансивер может подсоединяться к кабелю как методом прокалывания, обеспечивающим непосредственный физический контакт, так и бесконтактным методом.

Трансивер соединяется с сетевым адаптером интерфейсным кабелем ***AUI*** (*Attachment Unit Interface*) длиной до 50 м, состоящим из 4 витых пар (адаптер должен иметь разъем AUI). Допускается подключение к одному сегменту не более 100 трансиверов, причем расстояние между подключениями трансиверов не должно быть меньше 2.5 м.

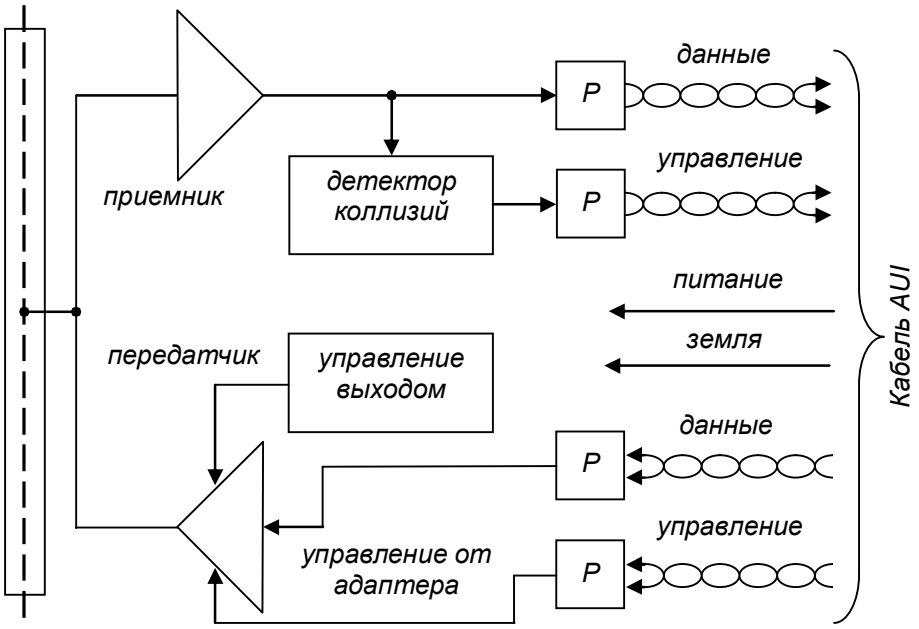


Рис. 5.10 — Структурная трансивера стандарта 10Base-5

Детектор коллизий определяет наличие коллизии в коаксиальном кабеле по повышенному уровню постоянной составляющей сигналов. Если постоянная составляющая превышает определенный порог, то это означает, что на кабель работает более чем один передатчик.

Трансивер выполняет следующие функции:

- прием и передача данных с кабеля 10Base-5 на кабель AUI и обратно;
- определение коллизий на кабеле 10Base-5;
- обеспечение электрической развязки (*P*) между кабелем и остальной частью адаптера;
- защита кабеля от некорректной работы адаптера.

Последнюю функцию часто называют *контролем болтливости* (***jabber control***). При возникновении неисправностей в адаптере может возникнуть ситуация, когда на кабель будет непрерывно выдаваться последовательность случайных сигналов. Так как кабель — это общая среда для всех станций, то работа сети будет заблокирована одним неисправным адаптером. Чтобы этого не случилось, на выходе передатчика ставится схема, которая проверяет количество битов, переданных в пакете. Если максимальная длина пакета превышается, то эта схема просто отсоединяет выход передатчика от кабеля.

К *достоинствам* стандарта 10Base-5 относятся:

- хорошая защищенность кабеля от внешних воздействий,
- сравнительно большое расстояние между узлами,
- возможность простого перемещения рабочей станции в пределах длины кабеля AUI.

К *недостаткам* следует отнести:

- высокую стоимость кабеля,
- сложность его прокладки из-за большой жесткости,
- наличие специального инструмента для заделки кабеля,
- при повреждении кабеля или плохом соединении происходит останов работы всей сети,
- необходимо заранее предусмотреть подводку кабеля ко всем возможным местам установки компьютеров.

Стандарт 10Base-2

Стандарт 10Base-2 использует в качестве передающей среды *коаксиальный кабель* с диаметром центрального медного провода 0,89 мм и внешним диаметром около 5 мм. Максимальная длина сегмента без повторителей составляет 185 м, сегмент должен иметь на концах согласующие терминаторы 50 Ом.

Станции подключаются к кабелю с помощью ***Т-коннектора***, который представляет из себя тройник, один отвод ко-

торого соединяется с сетевым адаптером, а два других — с двумя концами разрыва кабеля. Максимальное количество станций, подключаемых к одному сегменту — 30 шт. Минимальное расстояние между станциями — 1 м.

Этот стандарт очень близок к стандарту 10Base-5. Но трансиверы в нем объединены с сетевыми адаптерами за счет того, что более гибкий тонкий коаксиальный кабель может быть подведен непосредственно к выходному разъему платы сетевого адаптера, установленной в шасси компьютера. Кабель в данном случае висит на сетевом адаптере, что затрудняет физическое перемещение компьютеров.

Реализация этого стандарта на практике приводит к наиболее простому решению для кабельной сети, так как для соединения компьютеров требуются только сетевые адаптеры и Т-коннекторы. Однако этот вид кабельных соединений наиболее сильно подвержен авариям и сбоям: кабель восприимчив к помехам, в моноканале имеется большое количество механических соединений (каждый Т-коннектор дает три механических соединения, два из которых имеют жизненно важное значение для всей сети), пользователи имеют доступ к разъемам и могут нарушить целостность моноканала. Кроме того, эстетика и эргономичность этого решения оставляют желать лучшего, так как от каждой станции через Т-коннектор отходят два довольно заметных провода, которые под столом часто образуют моток кабеля — запас, необходимый на случай даже небольшого перемещения рабочего места.

Общим недостатком стандартов 10Base-5 и 10Base-2 является отсутствие оперативной информации о состоянии моноканала. Повреждение кабеля обнаруживается сразу же (сеть престает работать), но для поиска отказавшего отрезка кабеля необходим специальный прибор — кабельный тестер.

Стандарт 10Base-T

В качестве среды передачи используется двойная неэкранированная витая пара. Соединения станций осуществляются по топологии *точка — точка* со специальным устройством — *многопортовым повторителем* (***hub***) с помощью двух витых пар. Одна витая пара используется для передачи данных от станции к

повторителю (выход *Tx* сетевого адаптера), а другая — для передачи данных от повторителя станции (вход *Rx* сетевого адаптера). На рис. 5.11 показан пример трехпортового повторителя.

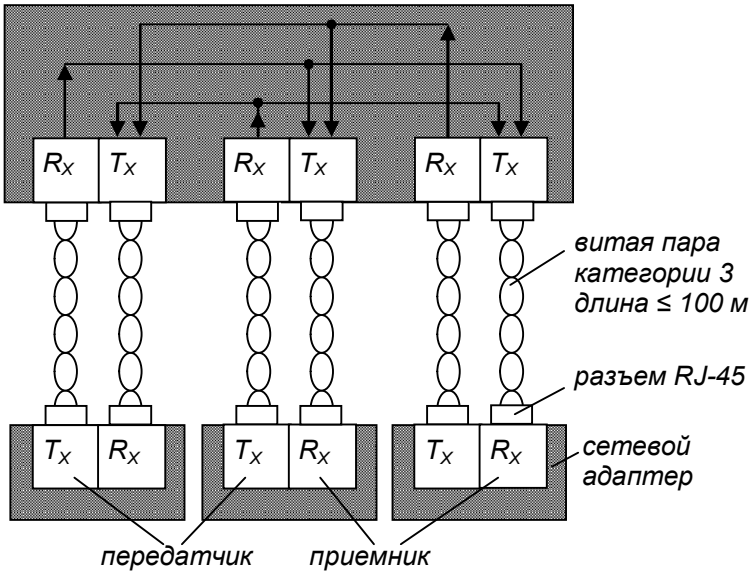


Рис. 5.11 — Концентратор 10Base-T

Концентратор осуществляет функции повторителя сигналов на всех отрезках витых пар, подключенных к его портам, так что образуется единая среда передачи данных — моноканал (шина). Повторитель создает коллизию в сегменте в случае одновременной передачи сигналов по нескольким своим *Rx* входам и посылает jam-последовательность на все свои *Tx* выходы. Стандарт определяет битовую скорость передачи данных 10 Мб/с и максимальное расстояние отрезка витой пары между двумя непосредственно связанными узлами (станциями и концентраторами) не более 100 м при использовании витой пары качества не ниже категории 3.

Возможно иерархическое соединение концентраторов в дерево. Для обеспечения синхронизации станций при реализации процедур доступа CSMA/CD и надежного распознавания станциями коллизий в стандарте определено максимально число концентраторов между любыми двумя станциями сети. Общее количество станций в сети 10Base-T не должно превышать 1024.

Сети, построенные на основе стандарта 10Base-T, обладают по сравнению с коаксиальными вариантами многими преимуще-

ствами. Эти преимущества связаны с разделением общего физического кабеля на отдельные кабельные отрезки, подключенные к центральному коммуникационному устройству. И хотя логически эти отрезки по-прежнему образуют общий домен коллизий, их физическое разделение позволяет контролировать их состояние и отключать в случае обрыва, короткого замыкания или неисправности сетевого адаптера на индивидуальной основе. Это обстоятельство существенно облегчает эксплуатацию больших сетей *Ethernet*, так как концентратор обычно автоматически выполняет такие функции, уведомляя при этом администратора сети о возникшей проблеме.

Стандарт 10Base-F

Стандарт 10Base-F использует в качестве среды передачи данных оптоволокно. Функционально сеть стандарта 10Base-F состоит из тех же элементов, что и сеть стандарта 10Base-T — сетевых адаптеров, многопортового повторителя и отрезков кабеля, соединяющих адаптер с портом повторителя. Как и при использовании витой пары, для соединения адаптера с повторителем используется два оптоволокна — одно соединяет выход *Tx* адаптера со входом *Rx* повторителя, а другое — вход *Rx* адаптера с выходом *Tx* повторителя.

Стандарт ***FOIRL*** (Fiber Optic Inter-Repeater Link) — это первый стандарт комитета 802.3 для использования оптоволокна в сетях Ethernet. Он гарантирует длину оптоволоконной связи между повторителями до 1 км при общей длине сети не более 2 500 м. Максимальное число повторителей — 4.

Стандарт ***10Base-FL*** предназначен для соединения конечных узлов с концентратором и работает с сегментами оптоволокна длиной не более 2 000 м при общей длине сети не более 2 500 м. Максимальное число повторителей — 4.

Стандарт ***10Base-FB*** предназначен для магистрального соединения повторителей. Он позволяет иметь в сети до 5 повторителей при максимальной длине одного сегмента 2 000 м и максимальной длине сети 2 740 м. Повторители, соединенные по стандарту 10Base-FB постоянно обмениваются специальными последовательностями сигналов, отличающимися от сигналов кадров данных, для обнаружения отказов своих портов. Поэтому, кон-

центраторы стандарта 10Base-FB могут поддерживать резервные связи, переходя на резервный порт при обнаружении отказа основного с помощью тестовых специальных сигналов. Концентраторы этого стандарта передают как данные, так и сигналы простоя линии синхронно, поэтому биты синхронизации кадра не нужны и не передаются. Стандарт 10Base-FB поэтому называют также *синхронный Ethernet*. Стандарты 10Base-FL и 10Base-FB не совместимы между собой.

Методика расчета конфигурации сети Ethernet

В технологии Ethernet, независимо от применяемого стандарта физического уровня, существует понятие домена коллизий.

Домен коллизий (*collision domain*) — это часть сети Ethernet, все узлы которой распознают коллизию независимо от того, в какой части этой сети коллизия возникла. Сеть Ethernet, построенная на повторителях, всегда образует один домен коллизий. Домен коллизий соответствует одной разделяемой среде. Мосты, коммутаторы и маршрутизаторы делят сеть Ethernet на несколько доменов коллизий.

Узлы, образующие один домен коллизий, работают синхронно, как единая распределенная электронная схема.

Правило 4-х повторителей. При описании топологии сети стандарта 10Base-5 приводились ограничения на длину одного непрерывного отрезка коаксиального кабеля, используемого в качестве общей шины передачи данных для всех станций сети. Отрезок кабеля, завершающийся на обоих концах терминаторами и имеющий общую длину не более 500 м называется физическим сегментом сети. Однако при расчете окна коллизий общая максимальная длина сети 10Base-5 считалась равной 2500 м. Противоречия здесь нет, так как стандарт 10Base-5 (впрочем как и остальные стандарты физического уровня Ethernet) допускает соединение нескольких сегментов коаксиального кабеля с помощью повторителей, которые обеспечивают увеличение общей длины сети. Повторитель соединяет два сегмента коаксиального кабеля и выполняет функции регенерации электрической формы сигналов и их *синхронизации* (*retiming*). Повторитель прозрачен для станций, он обязан передавать кадры без искажений, модификации, потери или дублирования. Имеются ограничения на

максимально допустимые величины дополнительных задержек распространения битов нормального кадра через повторитель, а также битов jam-последовательности, которую повторитель обязан передать на все подключенные к нему сегменты при обнаружении коллизии на одном из них. Воспроизведение коллизии на всех подключенных к повторителю сегментах — одна из его основных функций.

В общем случае стандарт 10Base-5 допускает использование *до 4-х повторителей*, соединяющих в этом случае *5 сегментов* длиной до 500 м каждый, если используемые повторители удовлетворяют ограничениям на допустимые величины задержек сигналов. При этом общая длина сети будет составлять 2500 м, и такая конфигурация гарантирует правильное обнаружение коллизии крайними станциями сети. Только 3 сегмента из 5 могут быть нагруженными, то есть сегментами с подключенными к ним трансиверами конечных станций.

Правила 4-х повторителей и максимальной длины каждого из сегментов легко использовать на практике для определения корректности конфигурации сети. Однако эти правила применимы только тогда, когда все соединяемые сегменты представляют собой одну физическую среду, то есть в нашем случае толстый коаксиальный кабель, а все повторители также удовлетворяют требованиям физического стандарта 10Base-5. Аналогичные простые правила существуют и для сетей, все сегменты которых удовлетворяют требованиям другого физического стандарта, например, 10Base-T или 10Base-F. Однако для смешанных случаев, когда в одной сети Ethernet присутствуют сегменты различных физических стандартов, правила, основанные только на количестве повторителей и максимальных длинных сегментов становятся более запутанными.

Поэтому более надежно рассчитывать время полного оборота сигнала (PDV) по смешанной сети с учетом задержек в каждом типе сегментов и в каждом типе повторителей.

Расчет PDV и PVV. Для того чтобы сеть *Ethernet*, состоящая из сегментов различной физической природы, работала корректно, необходимо, чтобы выполнялись три основных условия:

- ***Количество станций*** в сети не превышает ***1024*** (с учетом ограничений для коаксиальных сегментов).

- Максимальная длина каждого физического сегмента не более величины, определенной в соответствующем стандарте физического уровня;
- Удвоенная задержка распространения сигнала (***Path Delay Value***, ***PDV***) между двумя самыми удаленными друг от друга станциями сети не превышает ***575 bt*** (количество битовых интервалов в пакете минимальной длины с учетом преамбулы).
- Сокращение межкадрового расстояния (***Path Variability Value***, ***PVV***) при прохождении последовательности кадров через все повторители не более чем на ***49 bt*** (напомним, что при отправке кадров станция обеспечивает начальное межкадровое расстояние в 96 битовых интервалов, значит после прохождения сегмента оно должно быть не меньше, чем 96 – 49 = 47 bt).

Соблюдение этих требований обеспечивает корректность работы сети даже в случаях, когда нарушаются простые правила конфигурирования, определяющие максимальное количество повторителей и максимальную длину сегментов каждого типа.

Физический смысл ограничения задержки распространения сигнала по сети (*PDV*) уже пояснялся — соблюдение этого требования обеспечивает своевременное обнаружение коллизий.

Требование на минимальное межкадровое расстояние связано с тем, что при прохождении кадра через повторитель это расстояние уменьшается. Каждый пакет, принимаемый повторителем, ресинхронизируется для исключения дрожания сигналов, накопленного при прохождении последовательности импульсов по кабелю и через интерфейсные схемы. Процесс ресинхронизации обычно увеличивает длину преамбулы, что уменьшает межкадровый интервал. При прохождении кадров через несколько повторителей межкадровый интервал может уменьшиться настолько, что сетевым адаптерам в последнем сегменте не хватит времени на обработку предыдущего кадра, в результате чего кадр будет просто потерян. Поэтому не допускается суммарное уменьшение межкадрового интервала более чем на 49 битовых интервалов. Величину уменьшения межкадрового расстояния при переходе между соседними сегментами обычно называют в англоязычной литературе *Segment Variability Value*, ***SVV***, а суммарную величину уменьшения межкадрового интервала при прохождении всех повторителей — *Path Variability Value,* ***PVV***. Очевид-

но, что величина *PVV* равна сумме *SVV* всех сегментов, кроме последнего.

Расчет PDV. Для упрощения расчетов обычно используются справочные данные, содержащие значения задержек распространения сигналов в повторителях, приемопередатчиках и в различных физических средах. В табл. 5.3 приведены данные, необходимые для расчета значения PDV для всех физических стандартов сетей Ethernet.

Таблица 5.3

Тип сегмента	***База левого сегмента***	***База промежуточного сегмента***	***База правого сегмента***	***Задержка среды на 1 м***	***max длина сегмента***
10Base-5	11.8	46.5	169.5	0.0866	500
10Base-2	11.8	46.5	169.5	0.1026	185
10Base-T	15.3	42.0	165.0	0.113	100
10Base-FB	–	24.0	–	0.1	2000
10Base-FL	12.3	33.5	156.5	0.1	2000
FOIRL	7.8	29.0	152.0	0.1	1000
AUI (>2 м)	0	0	0	0.1026	2+48

Поясним терминологию, использованную в этой таблице, на примере сети, изображенной на рис. 5.12.

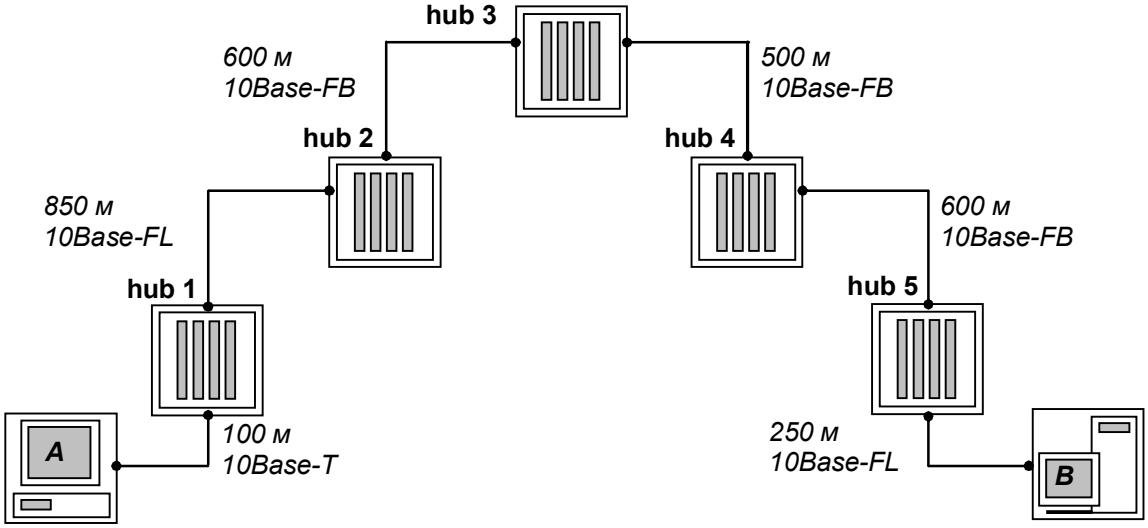


Рис. 5.12 — Пример сети *Ethernet*, состоящей из сегментов различных физических стандартов

Левым сегментом называется сегмент, в котором начинается путь сигнала от выхода передатчика (выход *Tx*) конечного узла. Затем сигнал проходит через промежуточные сегменты и доходит до приемника (вход *Rx*) наиболее удаленного узла наиболее удаленного сегмента, который называется *правым*. С каждым сегментом связана постоянная задержка $PDV_i^{БАЗА}$, названная базой, которая зависит только от типа сегмента и от положения сегмента на пути сигнала (левый, промежуточный или правый).

Кроме этого, с каждым сегментом связана задержка распространения сигнала вдоль кабеля сегмента, которая зависит от длины сегмента L_i и вычисляется путем умножения времени распространения сигнала по одному метру кабеля $t_i^{1\,м}$ (в битовых интервалах) на длину кабеля в метрах:

$$PDV_i = PDV_i^{БАЗА} + L_i \times t_i^{1м}. \qquad (5.6)$$

Общее значение *PDV* равно сумме базовых и переменных задержек всех сегментов сети:

$$PDV = \sum_i PDV_i. \qquad (5.7)$$

Значения констант в табл. 5.3 даны с учетом удвоения величины задержки при круговом обходе сети сигналом, поэтому удваивать полученную сумму не нужно.

Так как левый и правый сегмент имеют различные величины базовой задержки, то в случае различных типов сегментов на удаленных краях сети необходимо выполнить расчеты дважды: один раз принять в качестве левого сегмента сегмент одного типа (*A–B*), а во второй раз — сегмент другого типа (*B–A*), а результатом считать максимальное значение *PDV*.

В табл. 5.4 приведены расчеты PDV для каждого сегмента (рис. 5.12) в обоих направлениях, в результате задержка распространения сигнала получилась равной 574.1 bt. Так как значение PDV меньше максимально допустимой величины 575, то эта сеть проходит по величине максимально возможной задержки оборота сигнала. Несмотря на то, что ее общая длина существенно больше 2 500 м и используется целых 5 повторителей.

Таблица 5.4

A–B	***Ethernet***	***Длина, м***	***Расчет***	***PDV***
A–1 (левый)	10Base-T	100	15.3 + 100 × 0.113	26.6
1–2	10Base-FL	850	33.5 + 850 × 0.1	118.5
2–3	10Base-FB	600	24 + 600 × 0.1	84
3–4	10Base-FB	500	24 + 500 × 0.1	74
4–5	10Base-FB	600	24 + 600 × 0.1	84
5–B (правый)	10Base-FL	250	156 + 250 × 0.1	181
		2900		**568.1**
B–A	***Ethernet***	***Длина, м***	***Расчет***	***PDV***
B–5 (левый)	10Base-FL	250	12.3 + 250 × 0.1	37.3
5–4	10Base-FB	600	24 + 600 × 0.1	84
4–3	10Base-FB	500	24 + 500 × 0.1	74
3–2	10Base-FB	600	24 + 600 × 0.1	84
2–1	10Base-FL	850	33.5 + 850 × 0.1	118.5
1–A (правый)	10Base-T	100	165 + 100 × 0.113	176.3
		2900		**574.1**

Расчет PVV. Для расчета PVV также можно воспользоваться табличными значениями максимальных величин уменьшения межкадрового интервала при прохождении повторителей различных физических сред (табл. 5.5).

Таблица 5.5

Тип сегмента	***Передающий сегмент***	***Промежуточный сегмент***
10Base-5 или 10Base-2	16	11
10Base-FB	–	2
10Base-FL	10.5	8
10Base-T	10.5	8

В соответствии с этими данными рассчитаем значение PVV для нашего примера.

Таблица 5.6

A–B	*Ethernet*	*PVV*
A–1 (левый)	10Base-T	10.5
1–2	10Base-FL	8
2–3	10Base-FB	2
3–4	10Base-FB	2
4–5	10Base-FB	2
5–B (правый)	10Base-FL	8
		32.5

B–A	*Ethernet*	*PVV*
B–5 (левый)	10Base-FL	10.5
5–4	10Base-FB	2
4–3	10Base-FB	2
3–2	10Base-FB	2
2–1	10Base-FL	8
1–A (правый)	10Base-T	8
		32.5

Сумма этих величин дает значение PVV, равное 32.5 bt, что меньше предельного значения в 49 битовых интервалов.

В результате, приведенная в примере сеть по всем параметрам соответствует стандартам Ethernet.

5.5 Основные характеристики стандарта Token Ring

Сети стандарта Token Ring, также как и сети Ethernet, используют разделяемую среду передачи данных, которая состоит из отрезков кабеля, соединяющих все станции сети в кольцо. Кольцо рассматривается как общий разделяемый ресурс, и для доступа к нему используется не случайный алгоритм, как в сетях Ethernet, а детерминированный, основанный на передаче станциями права на использование кольца в определенном порядке. Право на использование кольца передается с помощью кадра специального формата, называемого ***маркером*** или токеном (***token***).

Сети Token Ring работают с двумя битовыми скоростями — 4 Мб/с и 16 Мб/с. Смешение станций, работающих на различных скоростях, в одном кольце не допускается. Сети Token Ring, работающие со скоростью 16 Мб/с, имеют и некоторые усовершенствования в алгоритме доступа по сравнению со стандартом 4 Мб/с.

Маркерный метод доступа к разделяемой среде

В сетях с ***маркерным методом*** доступа право на доступ к среде передается циклически от станции к станции по логическому кольцу. Кольцо образуется отрезками кабеля, соединяю-

щими соседние станции (рис. 5.13). Таким образом, каждая станция связана со своей предшествующей и последующей станцией и может непосредственно обмениваться данными только с ними. Для обеспечения доступа станций к физической среде по кольцу циркулирует кадр специального формата и назначения — маркер.

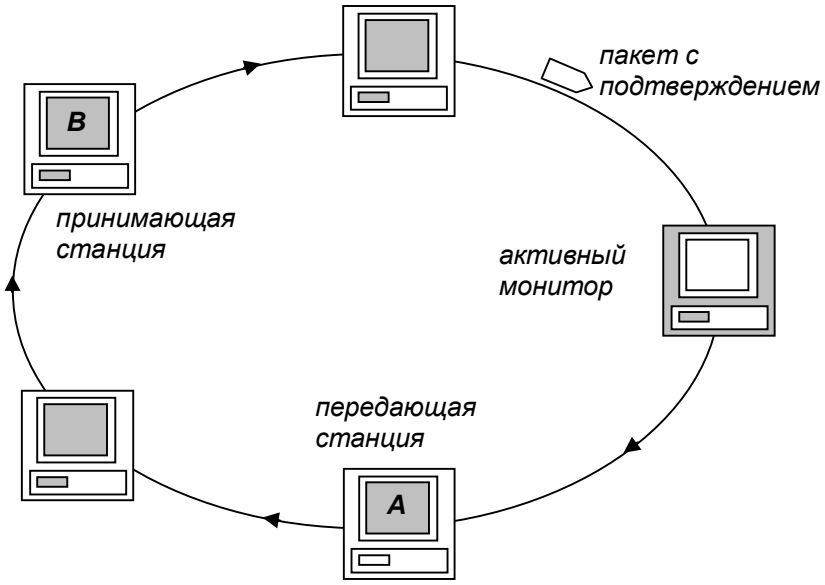


Рис. 5.13 — Принцип маркерного доступа

Получив маркер, станция анализирует его, при необходимости модифицирует и при отсутствии у нее данных для передачи обеспечивает его продвижение к следующей станции. Станция, которая имеет данные для передачи, при получении маркера изымает его из кольца, что дает ей право доступа к физической среде и передачи своих данных. Затем эта станция выдает в кольцо кадр данных установленного формата последовательно по битам. Переданные данные проходят по кольцу всегда в одном направлении от одной станции к другой.

При поступлении кадра данных к одной или нескольким станциям, эти станции копируют для себя этот кадр и вставляют в этот кадр подтверждение приема. Станция, выдавшая кадр данных в кольцо, при обратном его получении с подтверждением приема изымает этот кадр из кольца и выдает новый маркер для обеспечения возможности другим станциям сети передавать данные.

Время удержания одной станцией маркера ограничивается *тайм-аутом удержания маркера,* после истечения которого, станция обязана передать маркер далее по кольцу.

В сетях Token Ring 16 Мб/с используется также несколько другой алгоритм доступа к кольцу, называемый алгоритмом *раннего освобождения маркера* (***Early Token Release***). В соответствии с ним станция передает маркер доступа следующей станции сразу же после окончания передачи последнего бита кадра, не дожидаясь возвращения по кольцу этого кадра с битом подтверждения приема. В этом случае пропускная способность кольца используется более эффективно и приближается к 80% от номинальной.

Каждая станция имеет механизмы обнаружения и устранения неисправностей сети, возникающих в результате ошибок передачи или переходных явлений (например, при подключении и отключении станции). Одна из станций обозначается как ***активный монитор***, что означает дополнительную ответственность по управлению кольцом. Активный монитор осуществляет управление тайм-аутом в кольце, порождает новые маркеры (если необходимо), чтобы сохранить рабочее состояние, и генерирует диагностические кадры при определенных обстоятельствах. Активный монитор выбирается, когда кольцо инициализируется, и в этом качестве может выступить любая станция сети. Если монитор отказал по какой-либо причине, существует механизм, с помощью которого другие станции (резервные мониторы) могут договориться, какая из них будет новым активным монитором.

Для различных видов сообщений передаваемым данным могут назначаться различные ***приоритеты***. Каждый кадр или маркер имеет приоритет (значение от 0 до 7, где 7 — наивысший приоритет). Станция может воспользоваться маркером, если только она получила маркер с приоритетом, меньшим или равным, чем ее собственный. Сетевой адаптер станции, если ему не удалось захватить маркер, помещает свой приоритет в ***резервные биты маркера***, но только в том случае, если записанный в резервных битах приоритет ниже его собственного. Эта станция будет иметь преимущественный доступ при последующем поступлении к ней маркера.

Сначала монитор помещает в поле текущего приоритета максимальное значение (7), а поле резервного приоритета обнуляется (0). Маркер проходит по кольцу, в котором станции имеют свои текущие приоритеты. Как правило, эти значения меньше чем 7, следовательно, захватить маркер станции не могут, но они записывают свое значение приоритета в поле резервного приоритета, если их приоритет выше его текущего значения. В результате маркер возвращается к монитору со значением максимального резервного приоритета. Монитор переписывает это значение в поле основного приоритета, а значение резервного приоритета обнуляет, и снова отправляет маркер по кольцу. При этом обороте его захватывает станция с наивысшим приоритетом в кольце в данный момент времени.

Форматы кадров Token Ring

В Token Ring существует три различных формата кадров:

- *маркер;*
- *кадр данных;*
- *прерывающая последовательность.*

Маркер

Кадр маркера состоит из трех полей, каждое длиной в один байт.

- *Поле начального ограничителя* появляется в начале маркера, а также в начале любого кадра, проходящего по сети. Поле состоит из уникальной серии электрических импульсов, которые отличаются от тех импульсов, которыми кодируются единицы и нули в байтах данных. Поэтому начальный ограничитель нельзя спутать ни с какой битовой последовательностью.
- *Поле контроля доступа*: Приоритет (3 бита), признак маркера (1 бит), признак монитора (1 бит), резервные биты (3 бита).

Бит маркера имеет значение **0** для маркера и **1** для кадра. Бит монитора устанавливается в **1** активным монитором и в **0** любой другой станцией, передающей маркер или кадр. Если активный монитор видит маркер или кадр, содержащий бит монитора в **1**, то активный монитор знает, что этот кадр или маркер уже однажды обошел кольцо и не был обработан станциями. Если это кадр, то он удаляется из кольца. Если это маркер, то ак-

тивный монитор переписывает приоритет из резервных битов полученного маркера в поле приоритета. Поэтому при следующем проходе маркера по кольцу его захватит станция, имеющая наивысший приоритет.

- *Поле конечного ограничителя* — последнее поле маркера. Так же, как и поле начального ограничителя, это поле содержит уникальную серию электрических импульсов, которые нельзя спутать с данными. Кроме отметки конца маркера это поле также содержит два подполя: бит промежуточного кадра и бит ошибки.

Кадр данных

Каждый кадр данных состоит из нескольких групп полей:

- *последовательность начала кадра;*
- *адрес получателя;*
- *адрес отправителя;*
- *данные;*
- *последовательность контроля кадра;*
- *последовательность конца кадра.*

Кадр данных может переносить данные либо для управления кольцом (данные MAC-уровня), либо пользовательские данные (LLC-уровня). Стандарт Token Ring определяет 6 типов управляющих кадров MAC-уровня. *Поле последовательность начала кадра* определяет тип кадра (MAC или LLC) и, если он определен как MAC, то поле также указывает, какой из шести типов кадров представлен данным кадром.

Назначение этих шести типов кадров следующее.

- Чтобы удостовериться, что ее адрес уникальный, станция посылает MAC-кадр *«тест дублирования адреса»*, когда впервые присоединяется к кольцу.
- Чтобы сообщить другим станциям, что он еще функционирует, активный монитор запускает MAC-кадр «*активный монитор существует»* так часто, как только может.
- MAC-кадр *«существует резервный монитор»* отправляется любой станцией, не являющейся активным монитором.
- Резервный монитор отправляет MAC-кадр *«маркеры заявки»*, когда подозревает, что активный монитор отказал. Резервные мониторы затем договариваются между собой, какой из них станет новым активным монитором.

- Станция отправляет MAC-кадр *«сигнал»* в случае возникновения серьезных сетевых проблем, таких как оборванный кабель, или при обнаружении станции, передающей кадры без ожидания маркера. Определяя, какая станция отправляет кадр сигнала, диагностирующая программа может локализовать проблему.
- MAC-кадр *«очистка»* отправляется после того, как произошла инициализация кольца, и новый активный монитор заявляет о себе.

Каждый кадр (MAC или LLC) начинается с ***последовательности начала кадра***, которая содержит три поля:

- Начальный ограничитель, такой же, как и для маркера.
- Управление доступом, также совпадает для кадров и для маркеров.
- Контроль кадра — это однобайтовое поле, содержащее два подполя — *тип кадра* и *идентификатор управления MAC*. Если это LLC-кадр, то тип кадра равен **01**, а остальные биты не используются; для кадров MAC уровня тип кадра равен **00**, а биты идентификатора управления MAC определяют тип кадра управления кольцом из приведенного выше списка шести управляющих кадров MAC.

Адрес получателя (либо 2, либо 6 байтов). Первый бит определяет групповой или индивидуальный адрес как для двухбайтовых, так и для шестибайтовых адресов. Второй бит в шестибайтовых адресах говорит, назначен адрес локально или глобально. ***Адрес отправителя*** имеет тот же размер и формат, что и адрес получателя.

Поле данных кадра может содержать данные одного из описанных управляющих кадров MAC или запись пользовательских данных, предназначенных для (или получаемых от) протокола более высокого уровня, такого как IPX или NetBIOS. Это поле не имеет определенной максимальной длины, хотя существуют практические ограничения на его размер, основанные на временных требованиях к тому, как долго некоторая станция может управлять кольцом.

Последовательность контроля кадра используется для обнаружения ошибок, она состоит из четырех байтов остатка циклически избыточной контрольной суммы, вычисляемой по ал-

горитму CRC-32, осуществляющему циклическое суммирование по модулю 32.

Последовательность конца кадра состоит из двух полей: конечный ограничитель и статус кадра. *Конечный ограничитель* в кадре данных имеет дополнительное значение по сравнению с маркером. Кроме уникальной последовательности электрических импульсов он содержит два однобитовых поля: бит промежуточного кадра и бит обнаружения ошибки. Бит промежуточного кадра устанавливается в **1**, если этот кадр является частью многокадровой передачи, или в **0** для последнего или единственного кадра. Бит обнаружения ошибки первоначально установлен в **0**; каждая станция, через которую передается кадр, проверяет его на ошибки (по коду CRC) и устанавливает бит обнаружения ошибки в **1**, если она выявлена. Очередная станция, которая видит уже установленный бит обнаружения ошибки, должна просто передать кадр. Исходная станция заметит, что возникла ошибка, и повторит передачу кадра.

Статус кадра имеет длину 1 байт и содержит 4 резервных бита и два подполя: бит распознавания адреса и бит копирования кадра. Используется для корректного распознавания кадра и контроля ошибки.

Прерывающая последовательность

Состоит из двух байтов, содержащих начальный ограничитель и конечный ограничитель. Прерывающая последовательность может появиться в любом месте потока битов и сигнализирует о том, что текущая передача кадра или маркера отменяется.

Как видно из описания процедур обмена данными, в сети Token Ring на уровнях MAC и LLC применяются процедуры без установления связи, но с подтверждением получения кадров.

Физическая реализация сетей Token Ring

Стандарт Token Ring фирмы IBM предусматривает построение связей в сети как с помощью непосредственного соединения станций друг с другом, так и образование кольца с помощью концентраторов (называемых ***MAU — Media Attachment Unit*** или ***MSAU — Multi-Station Access Unit***). На рис. 5.14 показаны ос-

новные аппаратные элементы сети Token Ring и способы их соединения.

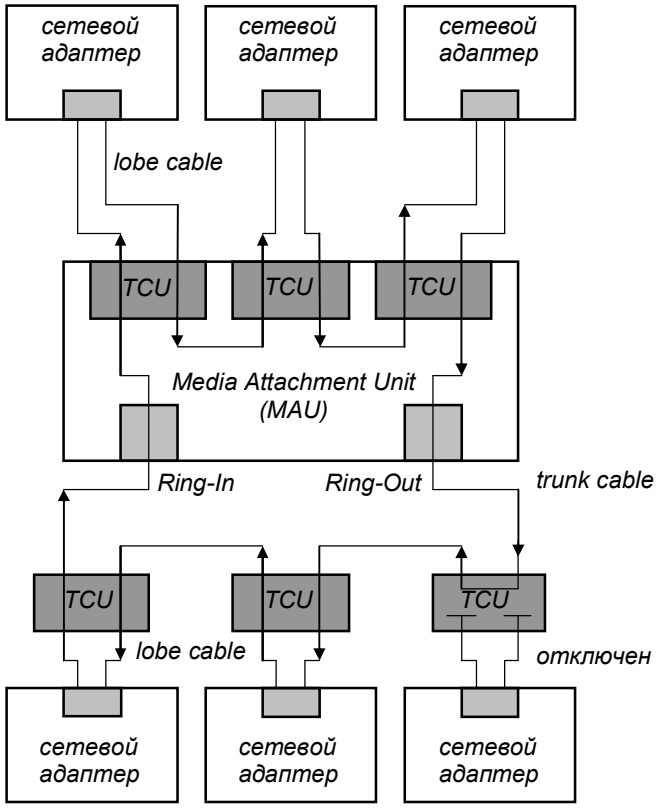


Рис. 5.14 — Конфигурация кольца Token Ring

В приведенной конфигурации показаны станции двух типов. Станции, подключаемые к кольцу через концентратор. Обычно такими станциями являются компьютеры с установленными в них сетевыми адаптерами.

Станции этого типа соединяются с концентратором ответвительным кабелем (***lobe cable***), который обычно является экранированной витой парой (STP), соответствующей стандартному типу кабеля из кабельной системы IBM (Type 1, 2, 6, 8, 9). Максимальная длина ответвительного кабеля зависит от типа концентратора, типа кабеля и скорости передачи данных. Обычно для скорости 16 Мб/с максимальная длина кабеля Type 1 может достигать 200 м, а для скорости 4 Мб/с — 600 м. Концентраторы Token Ring делятся на активные и пассивные. Пассивные концентраторы обеспечивают только соединения портов внутри концентратора в кольцо, активные выполняют и функции повторителя,

обеспечивая ресинхронизацию сигналов и исправление их амплитуды и формы. Активные концентраторы поддерживают большие расстояния, чем пассивные.

Остальные станции сети соединены в кольцо непосредственными связями. Такие связи называются магистральными (***trunk cable***). Обычно связи такого рода используются для соединения концентраторов друг с другом для образования общего кольца. Порты концентраторов, предназначенные для такого соединения, называются портами ***Ring-In*** и ***Ring-Out***.

Для предотвращения влияния отказавшей или отключенной станции на работу кольца станции подключаются к магистрали кольца через специальные устройства, называемые устройствами подключения к магистрали (***Trunk Coupling Unit, TCU***). В функции такого устройства входит образование обходного пути, исключающего заход магистрали в MAC-узел станции при ее отключении или отказе. Обычно для этих целей в TCU используются реле, которые подпитываются постоянным током во время нормальной работы. При пропадании тока подпитки контакты реле переключаются и образуют обходной путь, исключая станцию.

При подключении станции в кольцо через концентратор, устройства TCU встраивают в порты концентратора.

Максимальное количество станций в одном кольце — 250.

Кроме экранированной витой пары существуют сетевые адаптеры и концентраторы Token Ring, поддерживающие неэкранированную витую пару и оптоволокно.

5.6 Fast Ethernet

Технология ***Fast Ethernet*** является эволюционным развитием классической технологии Ethernet. Структура физического уровня технологии Fast Ethernet более сложная. Это вызвано тем, что в ней используется три варианта кабельных систем — оптоволокно, 2-х парная витая пара категории 5 и 4-х парная витая пара категории 3. Причем, по сравнению с вариантами физической реализации Ethernet (а их насчитывается шесть), здесь отличия каждого варианта от других глубже — меняется и количество проводников, и методы кодирования.

Официальный стандарт 100Base-T (***IEEE 802.3u***) установил три различных спецификации для физического уровня (в терминах семиуровневой модели OSI) для поддержки следующих типов кабельных систем:

- ***100Base-TX*** для двухпарного кабеля на неэкранированной витой паре UTP категории 5, или экранированной витой паре STP Type 1;
- ***100Base-T4*** для четырехпарного кабеля на неэкранированной витой паре UTP категории 3, 4 или 5;
- ***100Base-FX*** для многомодового оптоволоконного кабеля.

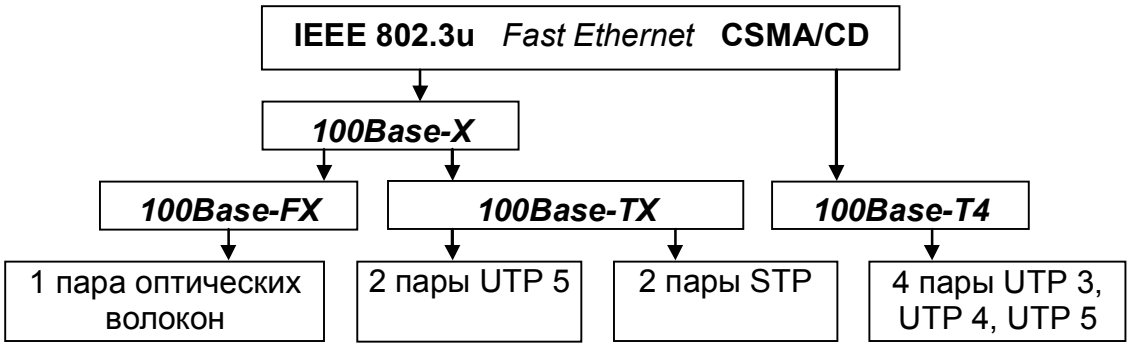


Рис. 5.15 — Физические интерфейсы стандарта Fast Ethernet

Физические интерфейсы стандарта Fast Ethernet (IEEE 802.3u) и их основные характеристики приведены в таблице 5.7.

Таблица 5.7

Физический интерфейс	100Base-FX	100Base-TX	100Base-T4
Порт устройства	Duplex SC	RJ-45	RJ-45
Среда передачи	Оптическое волокно	Витая пара UTP 5	Витая пара UTP 3,4,5
Сигнальная схема	4B/5B	4B/5B	8B/6T
Битовое Кодирование	NRZI	MLT-3	NRZI
Число витых пар/ волокон	2 волокна	2 витых пары	4 витых пары
Протяженность сегмента	до 412 м (mmf) до 2 км (mmf) до 100 км (smf)	до 100 м	до 100 м

Основными достоинствами технологии Fast Ethernet являются:

- увеличение пропускной способности сегментов сети до 100 Мб/с;
- сохранение метода случайного доступа CSMA/CD;
- сохранение звездообразной топологии сетей и поддержка традиционных сред передачи данных — витой пары и оптоволоконного кабеля.

Указанные свойства позволяют осуществлять постепенный переход от сетей 10Base-T — наиболее популярного еще вчера варианта Ethernet — к скоростным сетям, сохраняя значительную преемственность.

Fast Ethernet. Метод доступа к среде CSMA/CD

По сравнению с технологией Ethernet, подуровни LLC и MAC в стандарте Fast Ethernet не претерпели изменений.

Подуровень LLC обеспечивает интерфейс протокола Ethernet с протоколами вышележащих уровней, например, с IP или IPX. Кадр LLC вкладывается в кадр MAC и позволяет за счет полей DSAP и SSAP идентифицировать адрес сервисов назначения и источника соответственно. Поле управления кадра LLC позволяет реализовать процедуры обмена данными трех типов:

- *Процедура LLC1* определяет обмен данными без предварительного установления соединения и без повторной передачи кадров в случае обнаружения ошибочной ситуации, то есть является процедурой дейтаграммного типа. Поле управления для этого типа процедур имеет значение 03, что определяет все кадры как ненумерованные.
- *Процедура LLC2* определяет режим обмена с установлением соединений, нумерацией кадров, управлением потоком кадров и повторной передачей ошибочных кадров.
- *Процедура LLC3* определяет режим передачи данных без установления соединения, но с получением подтверждения о доставке информационного кадра адресату.

Существует расширение формата кадра LLC, называемое *SNAP* (*Subnetwork Access Protocol*). В случае использования расширения SNAP в поля DSAP и SSAP записывается значение *AA*, тип кадра по-прежнему равен 03. Заголовки LLC или LLC/SNAP

используются мостами и коммутаторами для трансляции протоколов канального уровня по стандарту ***IEEE 802.2h***.

Подуровень MAC ответственен за формирование кадра Ethernet, получение доступа к разделяемой среде передачи данных и за отправку с помощью физического уровня кадра по физической среде узлу назначения.

MAC-подуровень каждого узла сети получает от физического уровня информацию о состоянии разделяемой среды. Если она свободна, и у MAC-подуровня имеется кадр для передачи, то он передает его через физический уровень в сеть. Физический уровень одновременно с побитной передачей кадра следит за состоянием среды. Если за время передачи кадра коллизия не возникла, то кадр считается переданным. Если же за это время коллизия была зафиксирована, то передача кадра прекращается, и в сеть выдается 32-битная JAM-последовательность, которая должна помочь однозначно распознать коллизию всеми узлами сети.

После фиксации коллизии MAC-подуровень делает случайную паузу, а затем вновь пытается передать данный кадр. Случайный характер паузы уменьшает вероятность одновременной попытки захвата разделяемой среды несколькими узлами при следующей попытке.

MAC-подуровень узла приемника, который получает биты кадра от своего физического уровня, проверяет поле адреса кадра, и если адрес совпадает с его собственным, то он копирует кадр в свой буфер. Затем он проверяет, не содержит ли кадр специфические ошибки: по контрольной сумме (*FCS error*), по максимально допустимому размеру кадра (*jabber error*), по минимально допустимому размеру кадра (*runts*), по неверно найденным границам байт (*alignment error*). Если кадр корректен, то его поле данных передается на LLC-подуровень, если нет — то отбрасывается.

Спецификации физического уровня Fast Ethernet

Форматы кадров технологии Fast Ethernet не отличаются от форматов кадров технологий 10-Мегабитного Ethernet'а. На рис. 5.16 приведен формат MAC-кадра Ethernet, а также временные параметры его передачи по сети для скорости 100 Мб/с.

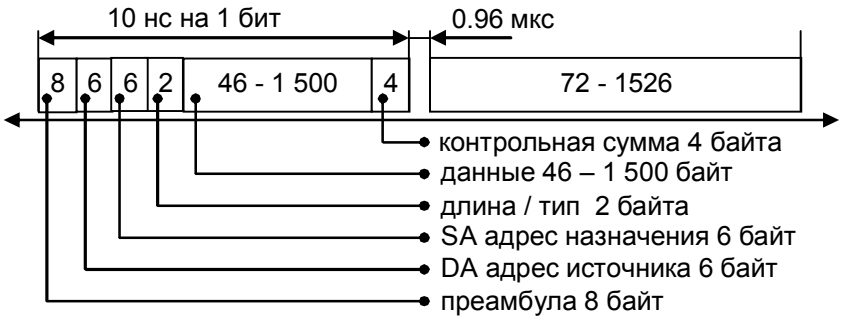


Рис. 5.16 — Формат MAC-кадра и временные характеристики его передачи

Для технологии Fast Ethernet разработаны различные варианты физического уровня, отличающиеся не только типом кабеля и электрическими параметрами импульсов, но и способом кодирования сигналов, и количеством используемых в кабеле проводников. Поэтому физический уровень Fast Ethernet имеет более сложную структуру, чем классический Ethernet. Эта структура представлена на рис. 5.17.

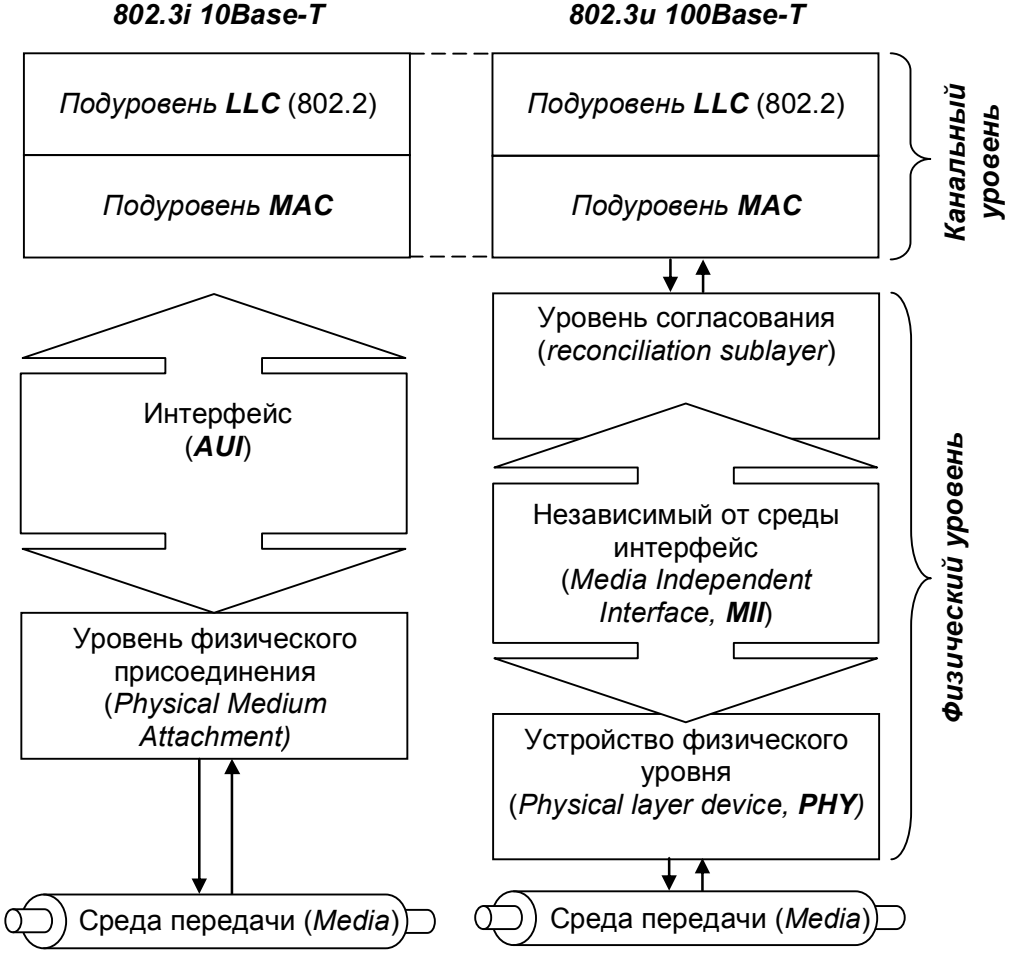


Рис. 5.17 — Структура физического уровня Fast Ethernet

Физический уровень состоит из трех подуровней:

- Уровень согласования (***reconciliation sublayer***).
- Независимый от среды интерфейс (***Media Independent Interface, MII***).
- Устройство физического уровня (***Physical layer device, PHY).***

Устройство физического уровня (***PHY***) обеспечивает кодирование данных, поступающих от MAC-подуровня для передачи их по кабелю определенного типа, синхронизацию передаваемых по кабелю данных, а также прием и декодирование данных в узле-приемнике.

Media Independent Interface MII

Интерфейс ***MII*** поддерживает независимый от используемой физической среды способ обмена данными между MAC-подуровнем и подуровнем PHY. Этот интерфейс аналогичен по назначению интерфейсу AUI классического Ethernet'a за исключением того, что интерфейс AUI располагался между подуровнем физического кодирования сигнала (для любых вариантов кабеля использовался одинаковый метод физического кодирования — манчестерский код) и подуровнем физического присоединения к среде, а интерфейс MII располагается между MAC-подуровнем и подуровнями кодирования сигнала, которых в стандарте Fast Ethernet три — FX, TX и T4. Подуровень согласования нужен для того, чтобы согласовать работу подуровня MAC с интерфейсом MII.

Существует два варианта реализации интерфейса MII: внутренний и внешний.

При внутреннем варианте микросхема, реализующая подуровни MAC и согласования, с помощью интереса MII соединяется с микросхемой трансивера внутри одного и того же конструктива, например, платы сетевого адаптера или модуля маршрутизатора. Микросхема трансивера реализует все функции устройства PHY.

Внешний вариант соответствует случаю, когда трансивер вынесен в отдельное устройство и соединен кабелем MII через разъем MII с микросхемой MAC-подуровня. Разъем MII в отличие от разъема AUI имеет 40 контактов, максимальная длина ка-

беля MII составляет 1 метр. Сигналы, передаваемые по интерфейсу MII, имеют амплитуду 5 В.

Интерфейс MII может использоваться не только для связи PHY с MAC, но и для соединения устройств PHY с микросхемой повторения сигналов в многопортовом повторителе-концентраторе.

Интерфейс MII использует 4-битные порции данных для параллельной передачи их между MAC и PHY. Канал передачи данных от MAC к PHY образован *4-битной шиной данных*, которая синхронизируется тактовым сигналом, генерируемым PHY, а также сигналом *«Передача»,* генерируемым MAC-подуровнем. Аналогично, канал передачи данных от PHY к MAC образован другой 4-битной шиной данных, которая синхронизируется тактовым сигналом и сигналом *«Прием»,* которые генерируются PHY.

Если устройство PHY обнаружило ошибку в состоянии физической среды, то оно может передать сообщение об этом на подуровень MAC в виде сигнала *«Ошибка приема»*. Подуровень MAC (или повторитель) сообщают об ошибке устройству PHY с помощью сигнала *«Ошибка передачи»*. Обычно, повторитель, получив от PHY какого-либо порта сигнал «Ошибка приема», передает на все устройства PHY остальных портов сигнал «Ошибка передачи».

В MII определена *двухпроводная шина управления* для обмена между MAC и PHY управляющей информацией. MAC-подуровень использует эту шину для передачи PHY данных о режиме его работы. PHY передает по этой шине информацию по запросу о статусе порта и линии. Данные о конфигурации, а также о состоянии порта и линии хранятся соответственно в двух регистрах: регистре управления (*Control Register*) и регистре статуса (*Status Register*).

Регистр управления используется для установки скорости работы порта, для указания, будет ли порт принимать участие в процессе автопереговоров о скорости линии, для задания режима работы порта — полудуплексный или полнодуплексный, и т.п. Функция автопереговоров (*Auto-negotiation*) позволяет двум устройствам, соединенным одной линией связи, автоматически, без вмешательства оператора, выбрать наиболее высокоскоростной режим работы, который будет поддерживается обоими устройствами.

Регистр статуса содержит информацию о действительном текущем режиме работы порта, в том числе и в том случае, когда режим выбран в результате проведения автопереговоров. Регистр статуса может содержать данные об одном из следующих режимов:

- 100Base-T4;
- 100Base-TX full-duplex;
- 100Base-TX half-duplex;
- 10 Mb/s full-duplex;
- 10 Mb/s half-duplex;
- ошибка на дальнем конце линии.

Физический уровень 100Base-FX — многомодовое оптоволокно

Физический уровень PHY ответственен за прием данных в параллельной форме от MAC-подуровня, трансляцию их в один (TX или FX) или три последовательных потока бит с возможностью побитной синхронизации и передачу их через разъем на кабель. Аналогично, на приемном узле уровень PHY должен принимать сигналы по кабелю, определять моменты синхронизации бит, извлекать биты из физических сигналов, преобразовывать их в параллельную форму и передавать подуровню MAC. Структура физического уровня 100Base-FX представлена на рис. 5.18.

Рис. 5.18 — Структурная схема физического уровня 100Base-FX

Эта спецификация определяет работу протокола Fast Ethernet по многомодовому оптоволокну в полудуплексном и полнодуплексном режимах на основе хорошо проверенной схемы кодирования и передачи оптических сигналов, использующейся уже на протяжении ряда лет в стандарте FDDI. Как и в стандарте FDDI, каждый узел соединяется с сетью двумя оптическими волокнами, идущими от приемника (Rx) и от передатчика (Tx).

Метод кодирования 4B/5B. Метод кодирования 4B/5B определен в стандарте FDDI, и он без изменений перенесен в спецификацию PHY FX/TX. При этом методе каждые 4 бита данных MAC-подуровня (называемых символами) представляются 5 битами кодовой последовательности. Использование избыточного бита позволяет применить потенциальные коды при представлении каждого из пяти бит в виде электрических или оптических импульсов. Из-за того, что прямое использование потенциальных кодов для передачи исходных данных без избыточного бита невозможно из-за плохой самосинхронизации приемника и источника данных: при передаче длинной последовательности единиц или нулей в течение долгого времени сигнал не изменяется и приемник не может определить момент чтения очередного бита. При использовании пяти бит для кодирования шестнадцати исходных 4-х битовых комбинаций, можно построить такую таблицу кодирования, в которой любой исходный 4-х битовый код представляется 5-ти битовым кодом с чередующимися нулями и единицами. Тем самым обеспечивается синхронизация приемника с передатчиком. Так как исходные биты MAC-подуровня должны передаваться со скоростью 100 Мб/с, то наличие одного избыточного бита вынуждает передавать биты результирующего кода 4B/5B со скоростью 125 Мб/с, то есть межбитовое расстояние в устройстве PHY составляет 8 наносекунд. Так как из 32 возможных комбинаций 5-битовых порций для кодирования порций исходных данных нужно только 16, то остальные 16 комбинаций в коде 4B/5B используются в служебных целях.

Наличие служебных символов позволило использовать в спецификациях FX/TX схему непрерывного обмена сигналами между передатчиком и приемником и при свободном состоянии среды, что отличает их от спецификации 10Base-T, когда незанятое состояние среды обозначается полным отсутствием на ней

импульсов информации. Для обозначения незанятого состояния среды используется служебный символ Idle (**11111**), который постоянно циркулирует между передатчиком и приемником, поддерживая их синхронизм и в периодах между передачами информации, а также позволяя контролировать физическое состояние линии. Существование запрещенных комбинаций символов позволяет отбраковывать ошибочные символы, что повышает устойчивость работы сетей с PHY FX/TX.

Передача 5-битовых кодов по линии методом NRZI. После преобразования 4-битовых порций MAC-кодов в 5-битовые порции PHY их необходимо представить в виде оптических или электрических сигналов в кабеле, соединяющем узлы сети. Спецификации PHY FX и PHY TX используют для этого различные методы физического кодирования — NRZI и MLT-3 соответственно. Для обеспечения частых изменений сигнала метода NRZI, а значит и для поддержания самосинхронизации приемника, нужно исключить из кодов слишком длинные последовательности нулей. Коды 4B/5B построены так, что гарантируют не более трех нулей подряд при любом сочетании бит в исходной информации. Основное преимущество NRZI кодирования по сравнению с NRZ кодированием в более надежном распознавании передаваемых **1** и **0** на линии в условиях помех.

Физический уровень 100Base-TX — двухпроводная витая пара

Структура физического уровня спецификации PHY TX представлена на рис. 5.19. Основные отличия от спецификации PHY FX — использование метода MLT-3 для передачи сигналов 5-битовых порций кода 4B/5B по витой паре, а также наличие функции *автопереговоров* (***auto-negotiation***) для выбора режима работы порта.

Метод MLT-3 использует потенциальные сигналы двух полярностей для представления 5-битовых порций информации.

Кроме использования метода MLT-3, спецификация PHY TX отличается от спецификации PHY FX тем, что в ней используется пара шифратор-дешифратор (*scrambler-descrambler*), как это определено в спецификации ANSI TP-PMD. Шифратор принимает 5-битовые порции данных от подуровня PCS, выпол-

няющего кодирование 4B/5B, и зашифровывает сигналы перед передачей на подуровень MLT-3 таким образом, чтобы равномерно распределить энергию сигнала по всему частотному спектру — это уменьшает электромагнитное излучение кабеля.

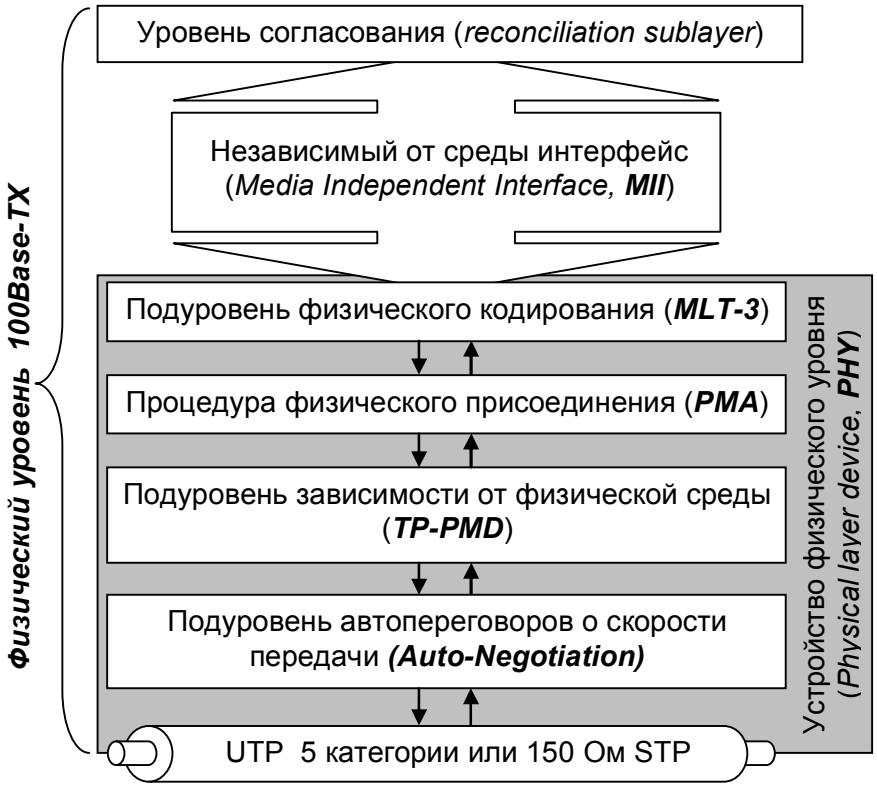


Рис. 5.19 — Структурная схема физического уровня 100Base-TX

Автопереговорный процесс. Спецификации PHY TX и PHY T4 поддерживают функцию *Auto-negotiation*, с помощью которой два взаимодействующих устройства PHY могут автоматически выбрать наиболее эффективный режим работы. Всего в настоящее время определено 5 различных режимов работы, которые могут поддерживать устройства PHY TX или PHY T4 на витых парах:

10Base-T	две пары UTP-3
10Base-T full-duplex	две пары UTP-3
100Base-TX	две пары UTP-5 (или Type 1A STP)
100Base-TX full-duplex	две пары UTP-5 (или Type 1A STP
100Base-T4	четыре пары UTP-3

Режим 10Base-T имеет самый низкий приоритет при переговорном процессе, а режим 100Base-T4 — самый высокий. Переговорный процесс происходит при включении питания устройства, а также может быть инициирован и в любой момент модулем управления.

Для организации переговорного процесса используются служебные сигналы проверки целостности линии технологии 10Base-T — link test pulses, если узел-партнер поддерживает только стандарт 10Base-T. Узлы, поддерживающие функцию Auto-negotiation, также используют существующую технологию сигналов проверки целостности линии, при этом они посылают пачки таких импульсов, инкапсулирующие информацию переговорного процесса Auto-negotiation. Такие пачки носят название *Fast Link Pulse burst (FLP)*.

Устройство, начавшее процесс auto-negotiation, посылает своему партнеру пачку импульсов FLP, в котором содержится 8-битное слово, кодирующее предлагаемый режим взаимодействия, начиная с самого приоритетного, поддерживаемого данным узлом. Если узел-партнер поддерживает функцию Auto-negotuiation и также может поддерживать предложенный режим, то он отвечает пачкой импульсов FLP, в которой подтверждает данный режим, и на этом переговоры заканчиваются. Если же узел-партнер может поддерживать менее приоритетный режим, то он указывает его в ответе и этот режим выбирается в качестве рабочего. Таким образом, всегда выбирается наиболее приоритетный общий режим узлов.

Узел, который поддерживает только технологию 10Base-T, каждые 16 миллисекунд посылает импульсы для проверки целостности линии, связывающей его с соседним узлом. Такой узел не понимает запрос FLP, который делает ему узел с функцией Auto-negotiation, и продолжает посылать свои импульсы. Узел, получивший в ответ на запрос FLP только импульсы проверки целостности линии, понимает, что его партнер может работать только по стандарту 10Base-T и устанавливает этот режим работы и для себя.

Полнодуплексный режим работы. Узлы, поддерживающие спецификации PHY FX и PHY TX, могут работать в полнодуплексном режиме (***full-duplex mode***). В этом режиме не использу-

ется метод доступа к среде CSMA/CD и отсутствует понятие коллизий — каждый узел одновременно передает и принимает кадры данных по каналам *Tx* и *Rx*.

Полнодуплексная работа возможна только при соединения сетевого адаптера с ***коммутатором*** или же при непосредственном соединении коммутаторов друг с другом. При полнодуплексной работе стандарты 100Base-TX и 100Base-FX обеспечивают скорость обмена данными между узлами 200 Мб/с. В полнодуплексном режиме необходимо определить процедуры управления потоком кадров, так как без этого механизма возможны ситуации, когда буферы коммутатора переполнятся и он начнет терять кадры Ethernet, что всегда крайне нежелательно, так как восстановление информации будет осуществляться более медленными протоколами транспортного или прикладного уровней.

Ввиду отсутствия стандартов на полнодуплексные варианты Ethernet каждый производитель сам определяет способы управления потоком кадров в коммутаторах и сетевых адаптерах. Обычно, при заполнении буфера устройства до определенного предела, это устройство посылает передающему устройству сообщение о временном прекращении передачи (*XOFF*). При освобождении буфера посылается сообщение о возможности возобновить передачу (*XON*).

Физический уровень 100Base-T4 — четырехпроводная витая пара

Спецификация PHY T4 была разработана для того, чтобы можно было использовать для высокоскоростного *Ethernet*'а имеющуюся проводку на UTP-3. Эта спецификация использует все 4 пары кабеля для того, чтобы можно было повысить общую пропускную способность за счет одновременной передачи потоков бит по нескольким витым парам.

Кодирование 8B/6T. Вместо кодирования 4B/5B в этом методе используется кодирование 8B/6T. Каждые 8 бит информации MAC-уровня кодируются 6-ю троичными цифрами (*ternary symbols*), то есть цифрами, имеющими три состояния (см. рис. 5.20). Каждая троичная цифра имеет длительность 40 наносекунд.

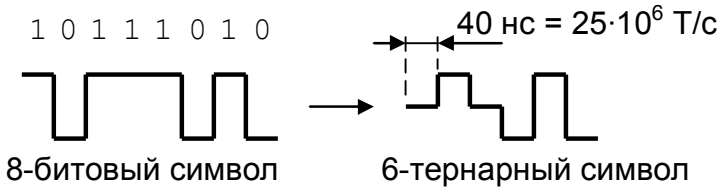


Рис. 5.20 — Кодирование 8B/6T

Группа из 6-ти троичных цифр затем передается на одну из трех передающих витых пар, независимо и последовательно. Четвертая пара всегда используется для прослушивания несущей частоты в целях обнаружения коллизии. Скорость передачи данных по каждой из трех передающих пар равна 33.3 Мб/с, поэтому общая скорость протокола 100Base-T4 составляет 100 Мб/с. В то же время из-за принятого способа кодирования скорость изменения сигнала на каждой паре равна всего 25 Мбод, что и позволяет использовать UTP-3. На рис. 5.21 показано соединение порта MDI сетевого адаптера 100Base-T4 с портом MDI-X повторителя.

Рис. 5.21 — Соединение узлов по спецификации PHY T4

Из рисунка видно, пара (1–2) всегда используется для передачи данных от порта MDI к порту MDI-X, пара (3–6) всегда используется для приема данных портом MDI от порта MDI-X, а пары (4–5) и (7–8) являются двунаправленными и используются и для приема, и для передачи, в зависимости от потребности.

Интерфейс 100Base-T4 имеет один существенный недостаток — принципиальную невозможность поддержки дуплексного режима передачи. И если при строительстве небольших сетей Fast Ethernet с использованием повторителей, 100Base-TX не

имеет преимуществ перед 100Base-T4 (существует коллизионный домен, полоса пропускания которого не больше 100 Мбит/с), то при строительстве сетей с использованием коммутаторов недостаток интерфейса 100Base-T4 становится очевидным и очень серьезным. Поэтому данный интерфейс не получил столь большого распространения, как 100Base-TX и 100Base-FX.

Типы устройств Fast Ethernet

Технология *Fast Ethernet* рассчитана на подключение конечных узлов — компьютеров с соответствующими сетевыми адаптерами — к многопортовым концентраторам-повторителям или коммутаторам.

Правила корректного построения сегментов сетей Fast Ethernet включают:

- ограничения на максимальные длины сегментов, соединяющих DTE с DTE;
- ограничения на максимальные длины сегментов, соединяющих DTE с портом повторителя;
- ограничения на максимальный диаметр сети;
- ограничения на максимальное число повторителей и максимальную длину сегмента, соединяющего повторители.

В качестве *DTE (Data Terminal Equipment)* может выступать любой источник кадров данных для сети: сетевой адаптер, порт моста, порт маршрутизатора, модуль управления сетью и другие подобные устройства. Порт повторителя не является DTE. В типичной конфигурации сети Fast Ethernet несколько DTE подключается к портам повторителя, образуя сеть звездообразной топологии. Спецификация IEEE 802.3u определяет следующие максимальные значения сегментов DTE-DTE:

Таблица 5.8

Стандарт	***Тип кабеля***	***Максимальная длина сегмента***
100Base-TX	UTP-5, UTP-5e	100 м
100Base-FX	MultiMode Fibber 62.5/125 мкм	412 м (полудуплекс) 2 км (полный дуплекс)
100Base-T4	UTP-3, UTP-4, UTP-5	100 метров

Трансивер это двухпортовое устройство, охватывающее подуровни PCS, PMA, PMD и AUTONEG, и имеющее с одной стороны MII интерфейс, с другой — один из зависимых от среды физических интерфейсов (100Base-FX, 100Base-TX или 100Base-T4). Трансиверы используются сравнительно редко, как и редко используются сетевые карты, повторители, коммутаторы с интерфейсом MII.

Конвертер (*media converter*) — это двухпортовое устройство, оба порта которого представляют зависимые от среды интерфейсы. Конвертеры в отличии от повторителей могут работать в дуплексном режиме, за исключение случая, когда имеется порт 100Base-T4. Распространены конвертеры 100Base-TX/100Base-FX. Конвертерные шасси, объединяющие несколько отдельных модулей 100Base-TX/100Base-FX позволяют подключать множество сходящих в центральном узле волоконно-оптических сегментов к коммутатору оснащенному дуплексными портами RJ-45 (100Base-TX).

Коммутатор — одно из наиболее важных устройств при построении корпоративных сетей. Большинство современных коммутаторов Fast Ethernet, либо допускают работу в режиме автоопредления 100/10 Мбит/с по портам RJ-45, либо могут работать исключительно в этом режиме. Естественно, в таких коммутаторах возможна дуплексная передача (за исключением 100Base-T4). Коммутаторы могут иметь специальные дополнительные слоты для установления *uplink*-модуля. В качестве интерфейсов у таких модулей могут выступать оптический порты типа Fast Ethernet 100Base-FX, FDDI, ATM (155 Мбит/с), Gigabit Ethernet и др.

Повторители класса I и класса II. По параметру максимальных временных задержек при ретрансляции кадров, повторители Fast Ethernet подразделяются на два класса:

Класс I. Задержка на двойном пробеге PDV не должна превышать 130 bt. В силу менее жестких требований, повторители этого класса могут иметь порты T4 и TX/FX, а также объединяться в стек. Поддерживают все типы систем кодирования физического уровня: 100Base-TX/FX и 100Base-T4.

Класс II. Повторители класса II поддерживают только один тип системы кодирования физического уровня — 100Base-TX/FX

или 100Base-T4. К повторителям этого класса предъявляются более жесткие требования по задержке на двойном пробеге: PDV < 92 bt, если порты типа TX/FX; и PDV < 67 bt, если все порты типа T4. В силу значительных отличий в организации физических уровней, возникает большая задержка кадра при ретрансляции между портами интерфейсов T4 и TX/FX, поэтому повторители, совмещающие в пределах одного устройства порты T4 с портами TX/FX, отнесены по стандарту к классу I.

В одном домене коллизий допускается наличие только одного повторителя класса I. Это связано с тем, что такой повторитель вносит большую задержку при распространении сигналов из-за необходимости трансляции различных систем сигнализации. Максимальное число повторителей класса II в домене коллизий — 2, причем они должны быть соединены между собой кабелем не длиннее 5 м.

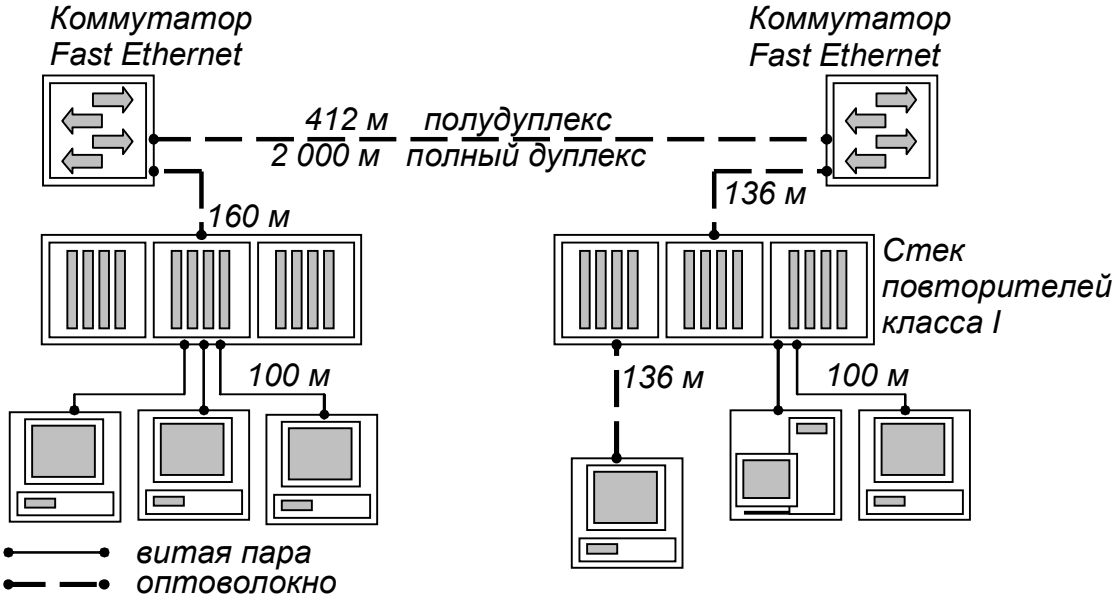


Рис. 5.22 — Пример построения сети с помощью повторителей класса I

Небольшое количество повторителей Fast Ethernet не является серьезным препятствием при построении сетей. Во-первых, наличие стековых повторителей снимает проблемы ограниченного числа портов — все каскадируемые повторители представляют собой один повторитель с достаточным числом портов — до нескольких сотен. Во-вторых, применение коммутаторов и мар-

шрутизаторов делит сеть на несколько доменов коллизий, в каждом из которых обычно имеется не очень большое число станций. Характеристики построения сети на основе повторителей класса I приведены в таблице 5.9.

Таблица 5.9

Тип кабелей	*Максимальный диаметр сети*	*Максимальная длина сегмента*
Только витая пара (TX)	200 м	100 м
Только оптоволокно (FX)	272 м	136 м
Несколько сегментов на витой паре и один на оптоволокне 260 м	100 м (TX)	160 м (FX)
Несколько сегментов на витой паре и несколько сегментов на оптоволокне 272 м	100 м (TX)	136 м (FX)

5.7 Протокол Gigabit Ethernet

В марте 1996 г. комитет IEEE 802.3 одобрил проект стандартизации Gigabit Ethernet 802.3z и создал *Gigabit Ethernet Alliance*, который к началу 1998 г. насчитывал уже более 100 компаний. Через Альянс 29 июня 1998 г. обеспечивается одобрение спецификаций стандартов Gigabit Ethernet IEEE 802.3z, которые регламентируют использование одномодового, многомодового волокна, а также витой пары UTP-5 на короткие расстояния — до 25 м. Позднее, 28 июня 1999 г. была разработана спецификация IEEE 802.3ab — передача по неэкранированной витой паре на расстояния до 100 м со специальным помехоустойчивым кодом.

Архитектура стандарта Gigabit Ethernet

На рис. 5.23 показана структура уровней Gigabit Ethernet. Как и в стандарте Fast Ethernet, в Gigabit Ethernet не существует универсальной схемы кодирования сигнала, которая была бы идеальной для всех физических интерфейсов — для стандартов 1000Base-LX/SX/CX используется кодирование 8B/10B, а для стандарта 1000Base-T используется специальный расширенный линейный код TX/T2. Функцию кодирования выполняет подуро-

вень кодирования PCS, размещенный ниже среданезависимого интерфейса GMII

GMII интерфейс. Среданезависимый интерфейс ***GMII*** (***Gigabit Media Independent Interface***) обеспечивает взаимодействие между уровнем MAC и физическим уровнем. GMII интерфейс является расширением интерфейса MII и может поддерживать скорости 10, 100 и 1000 Мбит/с. Он имеет отдельные 8 битные приемник и передатчик, и может поддерживать как полудуплексный, так и дуплексный режимы. Кроме этого, GMII интерфейс несет один сигнал, обеспечивающий синхронизацию (clock signal), и два сигнала состояния линии — первый (в состоянии ON) указывает наличие несущей, а второй (в состоянии ON) говорит об отсутствии коллизий — и еще несколько других сигнальных каналов и питание. Трансиверный модуль, охватывающий физический уровень и обеспечивающий один из физических средазависимых интерфейсов, может подключать например к коммутатору Gigabit Ethernet посредством GMII интерфейса.

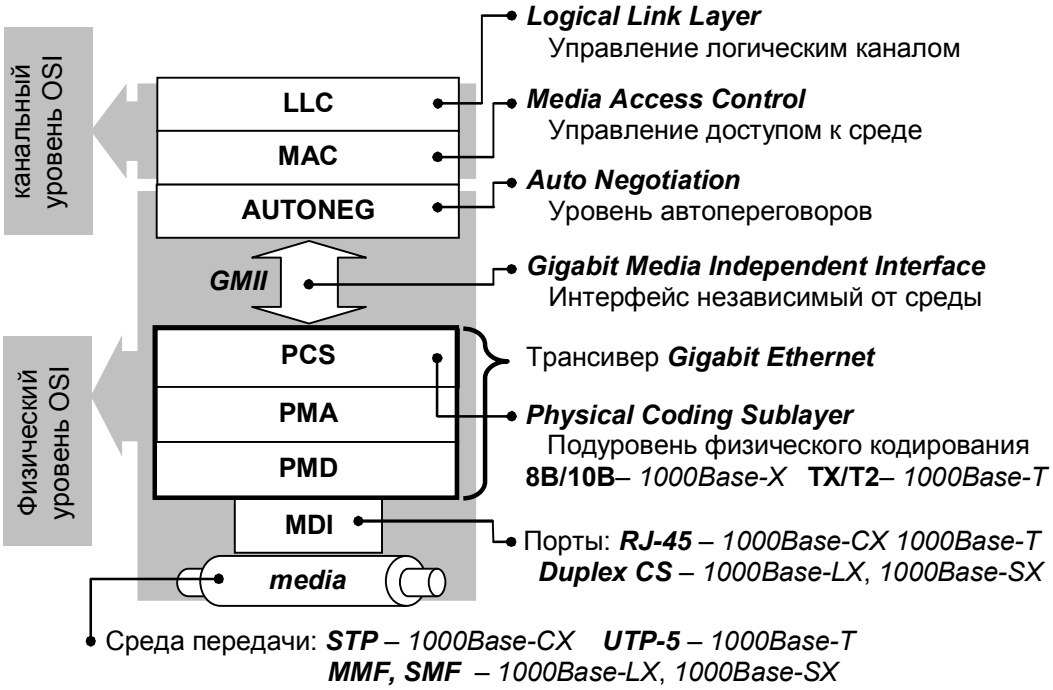


Рис. 5.23 — Структура уровней стандарта Gigabit Ethernet

Подуровень физического кодирования PCS. При подключении интерфейсов группы 1000Base-X, подуровень PCS использует блочное избыточное кодирование 8B/10B, заимствованное из

стандарта ANSI X3T11 Fibre Channel. Аналогичного рассмотренному стандарту FDDI, только на основе более сложной кодовой таблицы каждые 8 входных битов, предназначенных для передачи на удаленный узел, преобразовываются в 10 битные символы (code groups). Кроме этого в выходном последовательном потоке присутствуют специальные контрольные 10 битные символы. Примером контрольных символов могут служить символы, используемые для расширения носителя (дополняют кадр Gigabit Ethernet до его минимально размера 512 байт). При подключении интерфейса 1000Base-T, подуровень PCS осуществляет специальное помехоустойчивое кодирование, для обеспечения передачи по витой паре UTP 5 на расстояние до 100 метров — линейный код TX/T2. Два сигнала состояния линии — сигнал наличие несущей и сигнал отсутствие коллизий — генерируются этим подуровнем.

Подуровни PMA и PMD. Физический уровень Gigabit Ethernet использует несколько интерфейсов, включая традиционную витую пару категории 5, а также многомодовое и одномодовое волокно. Подуровень PMA преобразует параллельный поток символов от PCS в последовательный поток, а также выполняет обратное преобразование (распараллеливание) входящего последовательного потока от PMD. Подуровень PMD определяет оптические/электрические характеристики физических сигналов для разных сред. Всего определяются 4 различный типа физических интерфейса среды, которые отражены в спецификация стандарта 802.3z (1000Base-X) и 802.3ab (1000Base-T), (рис. 5.24).

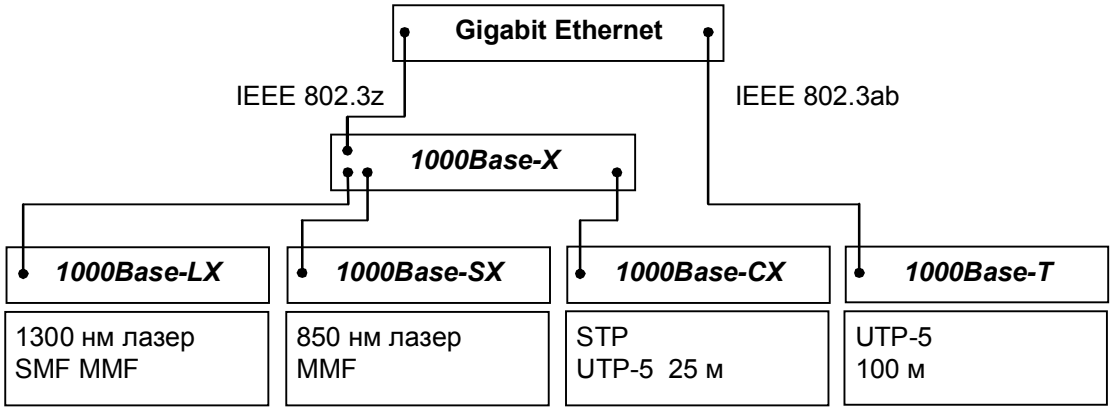


Рис. 5.24 — Физические интерфейсы стандарта Gigabit Ethernet

Интерфейс 1000Base-X

Интерфейс 1000Base-X основывается на стандарте физического уровня Fibre Channel. Fibre Channel — это технология взаимодействия рабочих станций, суперкомпьютеров, устройств хранения и периферийных узлов. Fibre Channel имеет 4-х уровневую архитектуру. Два нижних уровня FC-0 (интерфейсы и среда) и FC-1 (кодирование/декодирование) перенесены в Gigabit Ethernet. Поскольку Fibre Channel является одобренной технологией, то такое перенесение сильно сократило время на разработку оригинального стандарта Gigabit Ethernet.

Блочный код 8B/10B аналогичен коду 4B/5B, принятому в стандарте FDDI. Однако код 4B/5B был отвергнут в Fibre Channel, потому что этот код не обеспечивает баланса по постоянному току. Отсутствие баланса потенциально может привести к зависящему от передаваемых данных нагреванию лазерных диодов, поскольку передатчик может передавать больше битов «1» (излучение есть), чем «0» (излучения нет), что может быть причиной дополнительных ошибок при высоких скоростях передачи.

1000Base-X подразделяется на три физических интерфейса, основные характеристики которых приведены ниже:

Интерфейс 1000Base-SX определяет лазеры с допустимой длиной излучения в пределах диапазона 770—860 нм, мощность излучения передатчика в пределах от –10 до 0 дБм, при отношении ON/OFF (сигнал / нет сигнала) не меньше 9 дБ. Чувствительность приемника –17 дБм, насыщение приемника 0 дБм;

Интерфейс 1000Base-LX определяет лазеры с допустимой длиной излучения в пределах диапазона 1270—1355 нм, мощность излучения передатчика в пределах от –13,5 до –3 дБм, при отношении ON/OFF (есть сигнал / нет сигнала) не меньше 9 дБ. Чувствительность приемника –19 дБм, насыщение приемника –3 дБм;

1000Base-CX экранированная витая пара (STP »twinax») на короткие расстояния.

При кодировании 8B/10B битовая скорость в оптической линии составляет 1250 бит/с. Это означает, что полоса пропускания участка кабеля допустимой длины должна превышать 625 МГц.

Стандарт	Тип волокна/ медного кабеля	Полоса пропускания (не хуже), МГц*км	Максимальное расстояние, м
1000Base-LX (лазерный диод 1300 нм)	Одномодовое волокно (9 мкм)	–	5000
	Многомодовое волокно (50 мкм)	500	550
	Многомодовое волокно (62,5 мкм)	320	400
1000Base-SX (лазерный диод 850 нм)	Многомодовое волокно (50 мкм)	400	500
	Многомодовое волокно (62,5 мкм)	200	275
	Многомодовое волокно (62,5 мкм)	160	220
1000Base-CX	Экранированная витая пара: STP 150 Ом	–	25

Особенности использования многомодового волокна

В мире существует огромное количество корпоративных сетей на основе многомодового волоконно-оптического кабеля, с волокнами 62,5/125 и 50/125. По этому естественно, что еще на этапе формирования стандарта Gigabit Ethernet возникла задача адаптации этой технологии для использования в существующих многомодовых кабельных системах. В ходе исследований по разработке спецификаций 1000Base-SX и 1000Base-LX была выявлена одна очень интересная аномалия, связанная с использованием лазерных передатчиков совместно с многомодовым волокном.

Многомодовое волокно конструировалось для совместного использования со светоизлучающими диодами (спектр излучения 30—50 нс). Некогерентное излучение от таких светодиодов попадает в волокно по всей площади светонесущей сердцевины. В результате в волокне возбуждается огромное число модовых групп. Распространяющийся сигнал хорошо поддается описанию на

языке межмодовой дисперсии. Эффективность использования таких светодиодов в качестве передатчиков в стандарте Gigabit Ethernet низкая, в силу очень высокой частоты модуляции — скорость битового потока в оптической линии равна 1250 Мбод, а длительность одно импульса — 0,8 нс. Максимальная скорость, когда еще используются светодиоды для передачи сигнала по многомодовому волокну, составляет 622,08 Мбит/с (STM-4, с учетом избыточности кода 8B/10B битовая скорость в оптической линии 777,6 Мбод).

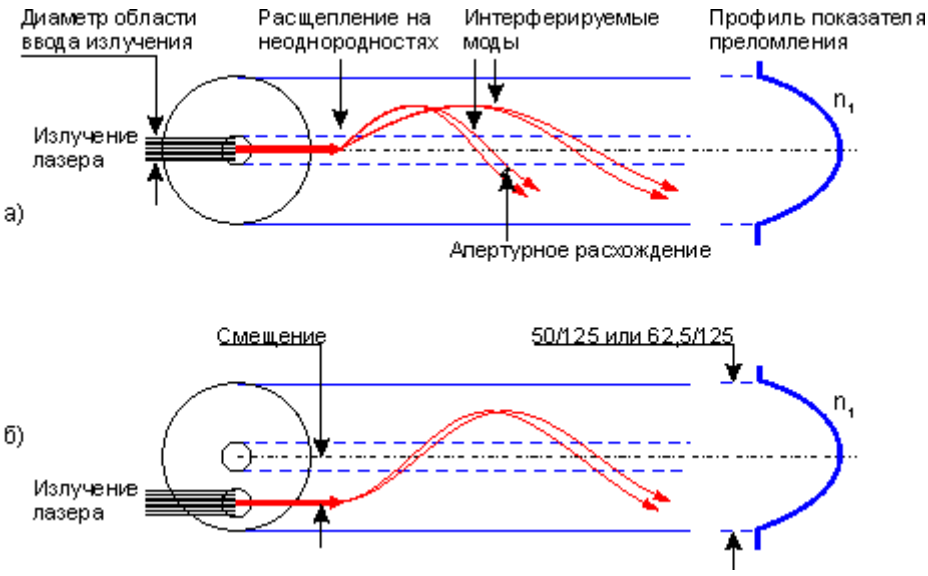


Рис. 5.25 — Распространение когерентного излучения в многомодовом волокне:
а — проявление эффекта дифференциальной модовой задержки (DMD) при осевом вводе излучения;
б — неосевой ввод когерентного излучения в многомодовое волокно

Gigabit Ethernet стал первым стандартом, регламентирующим использование лазерных оптических передатчиков совместно с многомодовым волокном. Площадь ввода излучения в волокно от лазера значительно меньше, чем размер сердцевины многомодового волокна. Этот факт сам по себе еще не приводит к проблеме. В то же время, в технологическом процессе производства стандартных коммерческих многомодовых волокон допускается наличие некоторых некритичных при традиционном использовании волокна дефектов (отклонений в пределах допус-

тимого), в наибольшей степени сосредоточенных вблизи оси сердцевины волокна. Хотя такое многомодовое волокно полностью удовлетворяет требованиям стандарта, когерентный свет от лазера, введенный по центру такого волокна, проходя через области неоднородности показателя преломления, способен расщепиться на небольшое число мод, которые затем распространяются по волокну разными оптическими путями и с разной скоростью.

Это явление известно как ***дифференциальная модовая задержка DMD***. В результате появляется фазовый сдвиг между модами, приводящий к нежелательной интерференции на приемной стороне и к значительному росту числа ошибок (рис. 5.25, *а*). Замети, что эффект проявляется только при одновременном стечении ряда обстоятельств: менее удачное волокно, менее удачный лазерный передатчик (разумеется удовлетворяющие стандарту) и менее удачный ввод излучения в волокно. С физической стороны, эффект DMD связан с тем, что энергия от когерентного источника распределяется внутри небольшого числа мод, в то время как некогерентный источник равномерно возбуждает огромное число мод. Исследования показывают, что эффект проявляется сильней при использовании длинноволновых лазеров (окно прозрачности 1300 нм).

Указанная аномалия в худшем случае может вести к уменьшению максимальной длины сегмента на основе многомодового ВОК. Поскольку стандарт должен обеспечивать 100-процентную гарантию работы, максимальна длина должна сегмента регламентироваться с учетом возможного проявления эффекта DMD.

Интерфейс 1000Base-LX. Для того, чтобы сохранить большее расстояние и избежать непредсказуемости поведения канала Gigabit Ethernet из-за аномалии, предложено вводить излучение в нецентральную часть сердцевины многомодового волокна. Излучение из-за апертурного расхождения успевает равномерно распределиться по всей сердцевине волокна, сильно ослабляя проявление эффекта, хотя максимальная длина сегмента и после этого остается ограниченной. Специально разработаны переходные одномодовые оптические шнуры ***MCP*** (***mode conditioning patch-cords***), у которых один из соединителей (а именно тот, который планируется сопрягать с многомодовым волокном) имеет не-

большое смещение от оси сердцевины волокна. Оптический шнур, у которого один соединитель — Duplex SC со смещенной сердцевиной, а другой — обычный Duplex SC, может называться так: MCP Duplex SC — Duplex SC. Разумеется такой шнур не подходит для использования в традиционных сетях, например в Fast Ethernet, из-за больших вносимых потерь на стыке с MCP Duplex SC. Переходной MCP может быть комбинированным на основе одномодового и многомодового волокна и содержать элемент смещения между волокнами внутри себя. Тогда одномодовым концом он подключается к лазерному передатчику. Что же касается приемника, то к нему может подключаться стандартный многомодовый соединительный шнур. Использование переходных MCP шнуров позволяет заводить излучение в многомодовое волокно через область, смещенную на 10—15 мкм от оси (рис. 5.25, *б*). Таким образом, сохраняется возможность использования интерфейсных портов 1000Base-LX и с одномодовыми ВОК, поскольку там ввод излучения будет осуществляться строго по центру сердцевины волокна.

Интерфейс 1000Base-SX. Так как интерфейс 1000Base-SX стандартизован только для использования с многомодовым волокном, то смещение области ввода излучения от центральной оси волокна можно реализовать внутри самого устройства, тем самым снять необходимость использования согласующего оптического шнура.

Интерфейс 1000Base-T

1000Base-T — это стандартный интерфейс Gigabit Ethernet передачи по неэкранированной витой паре категории 5 и выше на расстояния до 100 метров. Для передачи используются все четыре пары медного кабеля, скорость передачи по одной паре 250 Мбит/с. Предполагается, что стандарт будет обеспечивать дуплексную передачу, причем данные по каждой паре будут передаваться одновременно сразу в двух направлениях — двойной дуплекс (dual duplex). 1000Base-T. Технически реализовать дуплексную передачу 1 Гбит/с по витой паре UTPcat.5 оказалось довольно сложно, значительно сложней чем в стандарте 100Base-TX. Влияние ближних и дальних переходных помех от трех соседних витых пар на данную пару в четырехпарном кабеле требу-

ет разработки специальной скремблированной помехоустойчивой передачи, и интеллектуального узла распознавания и восстановления сигнала на приеме. Несколько методов кодирования первоначально рассматривались в качестве кандидатов на утверждение в стандарте 1000Base-T, среди которых: 5-уровневое импульсно-амплитудное кодирование PAM-5; квадратурная амплитудная модуляция QAM-25, и др. Ниже приведены кратко идеи PAM-5, окончально утвержденного в качестве стандарта.

Почему 5-уровневое кодирование. Распространенное четырехуровневое кодирование обрабатывает входящие биты парами. Всего существует 4 различных комбинации — 00, 01, 10, 11. Передатчик может каждой паре бит установить свой уровень напряжения передаваемого сигнал, что уменьшает в 2 раза частоту модуляции четырехуровневого сигнала, 125 МГц вместо 250 МГц, (рис. 5.26), и следовательно частоту излучения. Пятый уровень добавлен для создания избыточности кода. В результате чего становится возможной коррекция ошибок на приеме. Это дает дополнительный резерв 6 дБ в соотношении сигнал/шум.

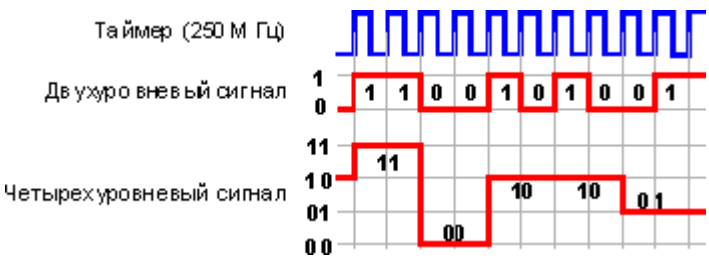


Рис. 5.26 — Схема 4-х уровневого кодирования PAM-4

Уровень MAC

Уровень MAC стандарта Gigabit Ethernet использует тот же самый протокол передачи CSMA/CD что и его предки Ethernet и Fast Ethernet. Основные ограничения на максимальную длину сегмента (или коллизионного домена) определяются этим протоколом.

В стандарте Ethernet IEEE 802.3 принят минимальный размер кадра 64 байта. Именно значение минимального размера кадра определяет максимальное допустимое расстояние между станциями (диаметр коллизионного домена). Время, которого станция

передает такой кадр — время канала — равно 512 BT или 51,2 мкс. Максимальная длина сети Ethernet определяется из условия разрешения коллизий, а именно время, за которое сигнал доходит до удаленного узла и возвращается обратно RDT не должно превышать 512 BT (без учета преамбулы).

При переходе от Ethernet к Fast Ethernet скорость передачи возрастает, а время трансляции кадра длины 64 байта соответственно сокращается — оно равно 512 BT или 5,12 мкс (в Fast Ethernet 1 BT = 0,01 мкс). Для того, чтобы можно было обнаруживать все коллизии до конца передачи кадра, как и раньше необходимо удовлетворить одному из условий:

- сохранить прежнюю максимальную длину сегмента, но увеличить время канала (и следовательно, увеличить минимальную длину кадра), или
- сохранить время канала, (сохранить прежний размер кадра), но уменьшить максимальную длину сегмента.

В Fast Ethernet был оставлен такой же минимальный размер кадра, как в Ethernet. Это сохранило совместимость, но привело к значительному уменьшению диаметра коллизионного домена.

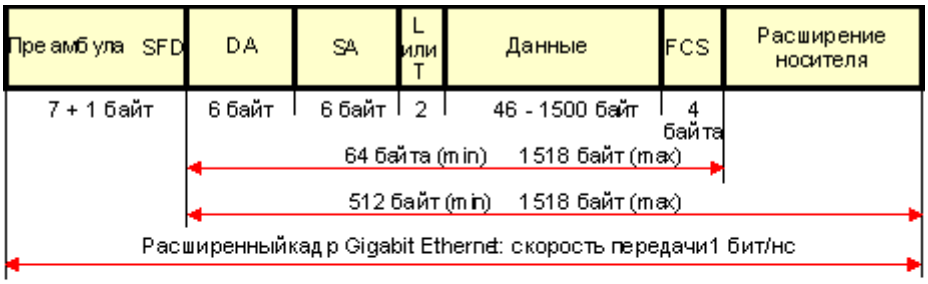
Опять же в силу преемственности стандарт Gigabit Ethernet должен поддерживать те же минимальный и максимальный размеры кадра, которые приняты в Ethernet и Fast Ethernet. Но поскольку скорость передачи возрастает, то соответственно уменьшается и время передачи пакета аналогичной длины. При сохранении прежней минимальной длины кадра это привело бы к уменьшению диаметра сети, который не превышал бы 20 метров, что могло быть мало полезным. Поэтому, при разработке стандарта Gigabit Ethernet было принято решение увеличить время канала. В Gigabit Ethernet оно составляет 4096 BT и в 8 раз превосходит время канала Ethernet и Fast Ethernet. Но, чтобы поддержать совместимость со стандартами Ethernet и Fast Ethernet, минимальный размер кадра не был увеличен, а было добавлено к кадру дополнительное поле, получившее название «расширение носителя».

Расширение носителя (carrier extension)

Символы в дополнительном поле обычно не несут служебной информации, но они заполняют канал и увеличивают «колли-

зионное окно». В результате, коллизия будет регистрироваться всеми станциями при большем диаметре коллизионного домена.

Если станция желает передать короткий (меньше 512 байт) кадр, до при передаче добавляется это поле — расширение носителя, дополняющее кадр до 512 байт. Поле контрольной суммы вычисляется только для оригинального кадра и не распространяется на поле расширения. При приеме кадра поле расширения отбрасывается. Поэтому уровень LLC даже и не знает о наличии поля расширения. Если размер кадра равен или превосходит 512 байт, то поле расширения носителя отсутствует. На рис. 5.27 показан формат кадра Gigabit Ethernet при использовании расширения носителя.



SFD : Start of frame Delimiter - ограничитель начала кадра
DA : Destination Address - адрес назначения
SA : Source Address - адрес источника
L : длина поля данных (для кадра 802.3)
T : тип поля данных (для кадра Ethernet_II)
FCS : FrameCheck Sequence - контрольная последовательность кадра

Рис. 5.27 — Кадр Gigabit Ethernet с полем расширения носителя

Типы устройств

В настоящее время поставляется полный перечень сетевых продуктов Gigabit Ethernet: сетевые карты, повторителе, коммутаторы, а также маршрутизаторы. Предпочтение отдается устройствам с оптическим интерфейсами (1000Base-FL, 1000Base-SX) Duplex SC. Так как стандартизация оптических интерфейсов произошла примерно на 1 год раньше, чем интерфейса на витую пару, то подавляющее число устройств, поставляемых сегодня, имеют волоконно-оптические физические интерфейсы.

Буферный повторитель. Устройства Ethernet поддерживают дуплексный режим как на физическом уровне, так и на уровне

MAC. Традиционные повторители с портами RJ-45 (10Base-T, 100Base-TX) хотя и имеют дуплексную связь на физическом уровне из-за логической топологии шина внутри себя могут поддерживать только полудуплексный режим, благодаря чему создается коллизионный домен ограниченного диаметра. Хотя в стандарте Gigabit Ethernet допускается использование традиционных повторителей, представляется более эффективным новое устройство — буферный повторитель. Протокол CSMA/CD реализует метод доступа к сети но не к сегменту.

Буферный повторитель — это многопортовое устройство с дуплексными каналами связи, (рис. 5.28). Каждый порт его имеет входной и выходной буферы. Разумеется удаленное устройство, подключено к повторителю также должно поддерживать дуплексную связь на физическом и MAC уровнях. Очередной кадр, прибывая на входной порт, размещается в очереди входного буфера порта и далее пересылается в выходные буферы остальных портов (за исключением выходного буфера этого порта). Внутри повторителя отрабатывается протокол CSMA/CD, на основе которого кадры из входных буферов переходят в выходные буферы других портов.

Поскольку в сегментах нет коллизий, ограничения на их длину могут возникать только из-за физических характеристик кабельной системы. В этой связи ВОК представляется более перспективным, чем витая пара ограниченная длиной 100 м.

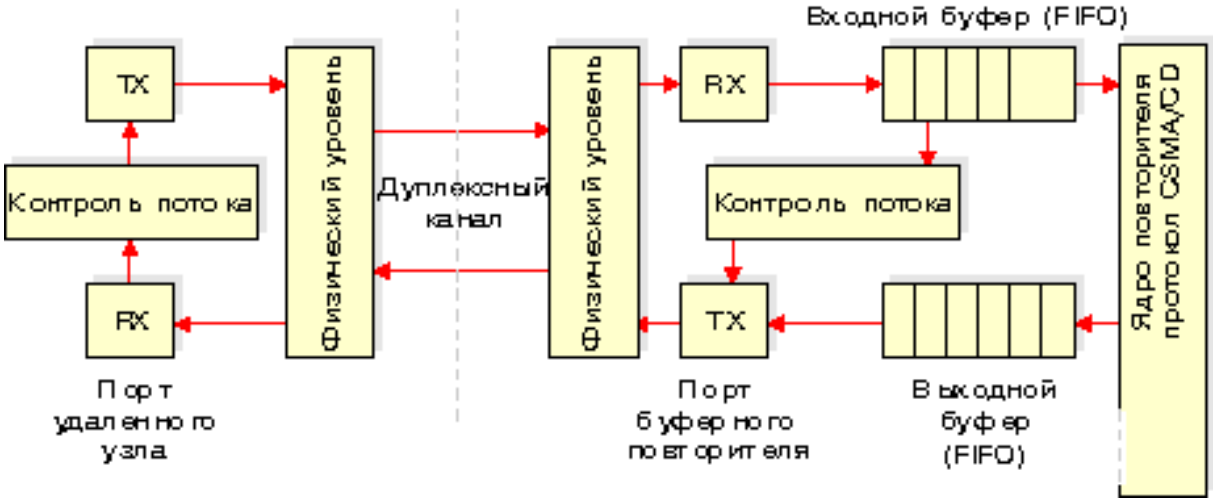


Рис. 5.28 — Архитектура буферного повторителя Gigabit Ethernet

Удаленный узел, передавая серию кадров способен переполнить входной буфер порта повторителя, что может привести к потере кадров. Во избежании этого стандартизован основанный на кадрах контроль потока (frame based flow control), известный как 802.3x. Протокол работает на уровне MAC и предназначен для использования в дуплексных линиях. Буферный повторитель обеспечивает дуплексную связь как и коммутатор, но не такой дорогой, поскольку является просто расширением традиционного повторителя.

Коммутаторы. Перечислим наиболее важные черты коммутаторов Gigabit Ethernet:

- поддержка дуплексного режима по всем портам;
- поддержка контроля потока основанного на кадрах IEEE 802.3x;
- наличие портов или модулей для организации каналов Ethernet, Fast Ethernet;
- поддержка физического интерфейса на одномодовый ВОК;
- возможность коммутации уровня 3;
- поддержка механизма QoS и протокола RSVP;
- поддержка стандарта IEEE 802.1Q/p для организации распределенных виртуальных сетей.

5.8 Технология 100VG-AnyLAN

В технологии 100VG-AnyLAN определены новый метод доступа ***Demand Priority*** и новая схема квартетного кодирования ***Quartet Coding***, использующая избыточный код ***5B/6B***.

Метод доступа Demand Priority основан на передаче концентратору функций арбитра, решающего проблему доступа к разделяемой среде. Метод Demand Priority повышает коэффициент использования пропускной способности сети за счет введения простого, детерминированного метода разделения общей среды, использующего два уровня приоритетов: низкий — для обычных приложений и высокий — для мультимедийных.

Технология 100VG-AnyLAN имеет меньшую популярность среди производителей коммуникационного оборудования, чем конкурирующее предложение — технология Fast Ethernet. Ком-

пании, которые не поддерживают технологию 100VG-AnyLAN, объясняют это тем, что для большинства сегодняшних приложений и сетей достаточно возможностей технологии Fast Ethernet, которая не так заметно отличается от привычной большинству пользователей технологии Ethernet. В более далекой перспективе эти производители предлагают использовать для мультимедийных приложений технологию ATM, а не 100VG-AnyLAN.

И хотя в число сторонников технологии 100VG-AnyLAN одно время входило около 30 компаний, среди которых Hewlett-Packard и IBM, Cisco Systems и Cabletron, общим мнением сетевых специалистов является констатация отсутствия дальнейщих перспектив у технологии 100VG-AnyLAN.

Структура сети 100VG-AnyLAN

Сеть 100VG-AnyLAN всегда включает центральный концентратор, называемый концентратором уровня 1 или корневым концентратором.

Корневой концентратор имеет связи с каждым узлом сети, образуя топологию типа звезда. Этот концентратор представляет собой интеллектуальный центральный контроллер, который управляет доступом к сети, постоянно выполняя цикл кругового сканирования своих портов и проверяя наличие запросов на передачу кадров от присоединенных к ним узлов. Концентратор принимает кадр от узла, выдавшего запрос, и передает его только через тот порт, к которому присоединен узел с адресом, совпадающим с адресом назначения, указанным в кадре.

Каждый концентратор может быть сконфигурирован на поддержку либо кадров 802.3 Ethernet, либо кадров 802.5 Token Ring. Все концентраторы, расположенные в одном и том же логическом сегменте (не разделенном мостами, коммутаторами или маршрутизаторами), должны быть сконфигурированы на поддержку кадров одного типа. Для соединения сетей 100VG-AnyLAN, использующих разные форматы кадров 802.3, нужен мост, коммутатор или маршрутизатор. Аналогичное устройство требуется и в том случае, когда сеть 100VG-AnyLAN должна быть соединена с сетью FDDI или ATM.

Концентратор циклически выполняет опрос портов. Станция, желающая передать пакет, посылает специальный низкочас-

тотный сигнал концентратору, запрашивая передачу кадра и указывая его приоритет. В сети l00VG-AnyLAN используются два уровня приоритетов — низкий и высокий. Низкий уровень приоритета соответствует обычным данным (файловая служба, служба печати и т.п.), а высокий приоритет соответствует данным, чувствительным к временным задержкам (например, мультимедиа). Приоритеты запросов имеют статическую и динамическую составляющие, то есть станция с низким уровнем приоритета, долго не имеющая доступа к сети, получает высокий приоритет.

Если сеть свободна, то концентратор разрешает передачу пакета. После анализа адреса получателя в принятом пакете концентратор автоматически отправляет пакет станции назначения. Если сеть занята, концентратор ставит полученный запрос в очередь, которая обрабатывается в соответствии с порядком поступления запросов и с учетом приоритетов. Если к порту подключен другой концентратор, то опрос приостанавливается до завершения опроса концентратором нижнего уровня. Станции, подключенные к концентраторам различного уровня иерархии, не имеют преимуществ по доступу к разделяемой среде, так как решение о предоставлении доступа принимается после проведения опроса всеми концентраторами опроса всех своих портов.

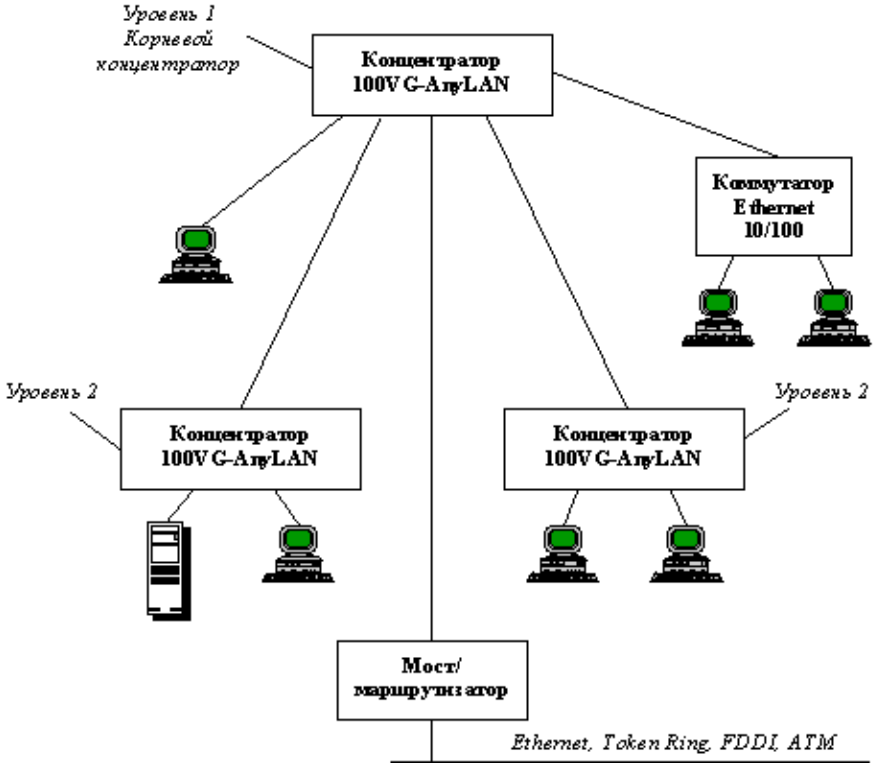


Рис. 5.29 — Архитектура сети 100VG-AnyLAN

Остается неясным вопрос — каким образом концентратор узнает, к какому порту подключена станция назначения? Во всех других технологиях кадр просто передавался всем станциям сети, а станция назначения, распознав свой адрес, копировала кадр в буфер. Для решения этой задачи концентратор узнает адрес MAC станции в момент физического присоединения ее к сети кабелем. Если в других технологиях процедура физического соединения выясняет связность кабеля (link test в технологии l0Base-T), тип порта (технология FDDI), скорость работы порта (процедура auto-negotiation в Fast Ethernet), то в технологии l00VG-AnyLAN концентратор при установлении физического соединения выясняет адрес MAC станции. И запоминает его в таблице адресов MAC, аналогичной таблице моста/коммутатора. Отличие концентратора l00VG-AnyLAN от моста/коммутатора в том, что у него нет внутреннего буфера для хранения кадров. Поэтому он принимает от станций сети только один кадр, отправляет его на порт назначения и, пока этот кадр не будет полностью принят станцией назначения, новые кадры концентратор не принимает. Так что эффект разделяемой среды сохраняется. Улучшается только безопасность сети — кадры не попадают на чужие порты, и их труднее перехватить.

Каждый концентратор имеет один ***восходящий*** (***up-link***) порт и N ***нисходящих*** портов (***down-link***).

Восходящий порт работает как порт узла, но он зарезервирован для присоединения в качестве узла к концентратору более высокого уровня. Нисходящие порты служат для присоединения узлов, в том числе и концентраторов нижнего уровня. Каждый порт концентратора может быть сконфигурирован для работы в нормальном режиме или в режиме монитора. Порт, сконфигурированный для работы в нормальном режиме, передает только те кадры, которые предназначены узлу, подключенному к данному порту. Порт, сконфигурированный для работы в режиме монитора, передает все кадры, обрабатываемые концентратором. Такой порт может использоваться для подключения анализатора протоколов.

Узел представляет собой компьютер или коммуникационное устройство технологии 100VG-AnyLAN — мост, коммутатор, маршрутизатор или концентратор. Концентраторы, подключае-

мые как узлы, называются концентраторами 2-го и 3-го уровней. Всего разрешается образовывать до трех уровней иерархии концентраторов.

Связь, соединяющая концентратор и узел, может быть образована либо четырьмя парами неэкранированной витой пары категорий 3, 4 или 5 (4 × UTP Cat 3, 4, 5), либо 2 парами неэкранированной витой пары категории 5 (2 × UTP Cat 5), либо 2 парами экранированной витой пары типа 1 (2 × STP Type 1), либо 2 парами многомодового оптоволоконного кабеля.

Варианты кабельной системы могут использоваться любые, но ниже будет рассмотрен вариант 4×UTP, который был разработан первым и получил наибольшее распространение.

В заключение раздела приведем таблицу, составленную компанией Hewlett-Packard, в которой приводятся результаты сравнения этой технологии с технологиями 10Base-T и 100Base-T.

Характеристика	**10Base-T**	**100VG-AnyLAN**	**100Base-T**
Топология			
Макс. диаметр сети	2500 м	8000 м	412 м
Каскадирование концентраторов	Да; 3 уровня	Да; 5 уровней	Макс. два концентратора
Кабельная система			
UTP Cat 3,4	100 м	100 м	100 м
UTP Cat 5	150 м	200 м	100 м
STP Type 1	100 м	100 м	100 м
Оптоволокно	2000 м	2000 м	412 м
Производительность			
При длине сети 100 м	80%	95%	80%
При длине сети 2500 м	80%	80%	Не поддерживается
Технология			
Кадры IEEE 802.3	Да	Да	Да
Кадры 802.5	Нет	Да	Нет
Метод доступа	CSMA/CD	Demand Priority	CSMA/CD + Reconciliation sublayer

5.9 Технология FDDI

Технология FDDI во многом основывается на технологии Token Ring, развивая и совершенствуя ее основные идеи.

Сеть FDDI строится на основе двух оптоволоконных колец, которые образуют основной и резервный пути передачи данных между узлами сети. Использование двух колец — это основной способ повышения отказоустойчивости в сети FDDI, и узлы, которые хотят им воспользоваться, должны быть подключены к обоим кольцам. В нормальном режиме работы сети данные проходят через все узлы и все участки кабеля ***первичного*** (***Primary***) кольца, поэтому этот режим назван режимом ***Thru*** (***сквозным*** или ***транзитным***). Вторичное кольцо (***Secondary***) в этом режиме не используется.

В случае какого-либо вида отказа, когда часть первичного кольца не может передавать данные (например, обрыв кабеля или отказ узла), первичное кольцо объединяется со вторичным, образуя вновь единое кольцо. Этот режим работы сети называется ***Wrap*** (***свертывание*** или ***сворачивание***) колец. Операция свертывания производится силами концентраторов и/или сетевых адаптеров FDDI. Для упрощения этой процедуры данные по первичному кольцу всегда передаются против часовой стрелки, а по вторичному — по часовой. Поэтому при образовании общего кольца из двух колец передатчики станций по-прежнему остаются подключенными к приемникам соседних станций, что позволяет правильно передавать и принимать информацию соседними станциями.

В стандартах FDDI отводится много внимания различным процедурам, которые позволяют определить наличие отказа в сети, а затем произвести необходимую реконфигурацию. Сеть FDDI может полностью восстанавливать свою работоспособность в случае единичных отказов ее элементов. При множественных отказах сеть распадается на несколько не связанных сетей.

Кольца в сетях FDDI рассматриваются как общая разделяемая среда передачи данных, поэтому для нее определен специальный метод доступа. Этот метод очень близок к методу доступа сетей Token Ring и также называется методом маркерного кольца — ***token ring***.

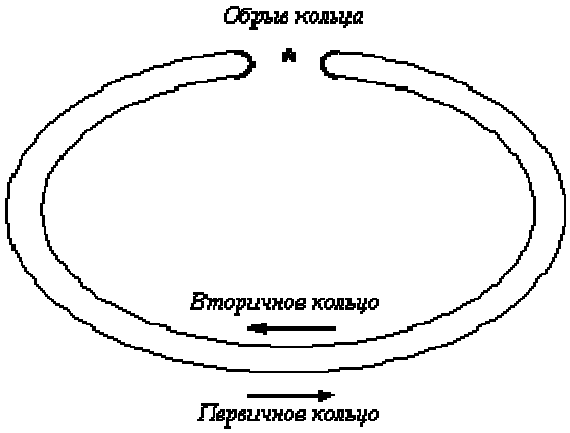


Рис. 5.30 — Режим WRAP
технологии FDDI

Станция может начать передачу своих собственных кадров данных только в том случае, если она получила от предыдущей станции специальный кадр — *токен_доступа*. После этого она может передавать свои кадры, если они у нее имеются, в течение времени, называемого временем удержания токена — ***Token Holding Time (THT)***. После истечения времени THT станция обязана завершить передачу своего очередного кадра и передать токен доступа следующей станции. Если же в момент принятия токена у станции нет кадров для передачи по сети, то она немедленно транслирует токен следующей станции. В сети FDDI у каждой станции есть предшествующий сосед (***upstream neighbor***) и последующий сосед (***downstream neighbor***), определяемые ее физическими связями и направлением передачи информации.

Каждая станция в сети постоянно принимает передаваемые ей предшествующим соседом кадры и анализирует их адрес назначения. Если адрес назначения не совпадает с ее собственным, то она транслирует кадр своему последующему соседу. Нужно отметить, что, если станция захватила токен и передает свои собственные кадры, то на протяжении этого периода времени она не транслирует приходящие кадры, а удаляет их из сети.

Если же адрес кадра совпадает с адресом станции, то она копирует кадр в свой внутренний буфер, проверяет его корректность (в основном по контрольной сумме), передает его поле данных для последующей обработки протоколу лежащего выше над FDDI уровня (например, IP), а затем передает исходный кадр

дальше по сети последующей станции. В передаваемом в сеть кадре станция назначения помечает три признака: распознавания адреса, копирования кадра и отсутствия или наличия в нем ошибок.

После этого кадр продолжает путешествовать по сети, транслируясь каждым узлом. Станция, являющаяся источником кадра для сети, ответственна за то, чтобы удалить кадр из сети, после того, как он, совершив полный оборот, вновь дойдет до нее. При этом исходная станция проверяет признаки кадра, дошел ли он до станции назначения и не был ли при этом поврежден. Процесс восстановления информационных кадров не входит в обязанности протокола FDDI, этим должны заниматься протоколы более высоких уровней.

На рисунке ниже приведена структура протоколов технологии FDDI в сравнении с семиуровневой моделью OSI. FDDI определяет протокол физического уровня и протокол подуровня доступа к среде (MAC) канального уровня. Как и многие другие технологии локальных сетей, технология FDDI использует протокол 802.2 подуровня управления каналом данных (LLC), определенный в стандартах IEEE 802.2 и ISO 8802.2. FDDI использует первый тип процедур LLC, при котором узлы работают в дейтаграммном режиме — без установления соединений и без восстановления потерянных или поврежденных кадров.

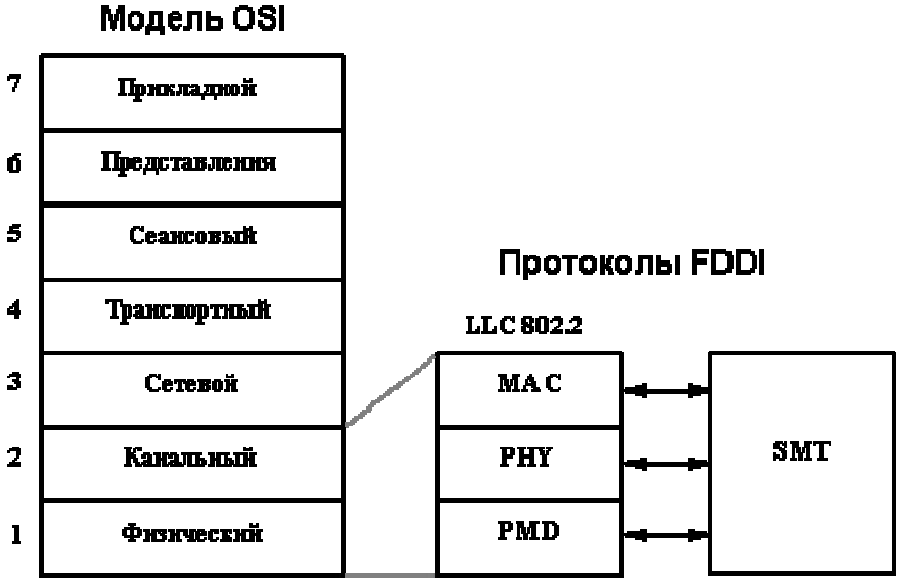


Рис. 5.31 — Структура протоколов технологии FDDI

Физический уровень разделен на два подуровня: независимый от среды подуровень ***PHY (Physical),*** и зависящий от среды подуровень ***PMD (Physical Media Dependent)***. Работу всех уровней контролирует протокол управления станцией ***SMT (Station Management)***.

Уровень PMD обеспечивает необходимые средства для передачи данных от одной станции к другой по оптоволокну. В его спецификации определяются:

- Требования к мощности оптических сигналов и к многомодовому оптоволоконному кабелю 62.5/125 мкм.
- Требования к оптическим обходным переключателям (***optical bypass switches***) и оптическим приемопередатчикам.
- Параметры оптических разъемов ***MIC*** (***Media Interface Connector***), их маркировка.
- Длина волны в 1300 нанометров, на которой работают приемопередатчики.
- Представление сигналов в оптических волокнах в соответствии с методом ***NRZI***.

Спецификация ***TP-PMD*** определяет возможность передачи данных между станциями по витой паре в соответствии с методом ***MLT-3*** аналогично технологии Fast Ethernet.

Уровень PHY выполняет кодирование и декодирование данных, циркулирующих между MAC-уровнем и уровнем PMD, а также обеспечивает тактирование информационных сигналов. В его спецификации определяются:

- кодирование информации в соответствии со схемой 4B/5B;
- правила тактирования сигналов;
- требования к стабильности тактовой частоты 125 МГц;
- правила преобразования информации из параллельной формы в последовательную.

Уровень MAC ответственен за управление доступом к сети, а также за прием и обработку кадров данных. В нем определены следующие параметры:

- Протокол передачи токена.
- Правила захвата и ретрансляции токена.
- Формирование кадра.
- Правила генерации и распознавания адресов.

- Правила вычисления и проверки 32-разрядной контрольной суммы.

Уровень SMT выполняет все функции по управлению и мониторингу всех остальных уровней стека протоколов FDDI. В управлении кольцом принимает участие каждый узел сети FDDI. Поэтому все узлы обмениваются специальными кадрами SMT для управления сетью. В спецификации SMT определено следующее:

- Алгоритмы обнаружения ошибок и восстановления после сбоев.
- Правила мониторинга работы кольца и станций.
- Управление кольцом.
- Процедуры инициализации кольца.

Отказоустойчивость сетей FDDI обеспечивается за счет управления уровнем SMT другими уровнями: с помощью уровня PHY устраняются отказы сети по физическим причинам, например, из-за обрыва кабеля, а с помощью уровня MAC — логические отказы сети, например, потеря нужного внутреннего пути передачи токена и кадров данных между портами концентратора.

Характеристика	**FDDI**	**Ethernet**	**Token Ring**
Битовая скорость	100 Мб/с	10 Мб/с	16 Мб/с
Топология	Двойное кольцо деревьев	Шина/звезда	Звезда/кольцо
Метод доступа	Доля от времени оборота токена	CSMA/CD	Приоритетная система резервирования
Среда передачи данных	Многомодовое оптоволокно, неэкранированная витая пара	Толстый коаксиал, тонкий коаксиал, витая пара, оптоволокно	Экранированная и неэкранированная витая пара, оптоволокно
Максимальная длина сети (без мостов)	200 км (100 км на кольцо)	2500 м	1000 м
Максимальное расстояние между узлами	2 км (–11 dB потерь между узлами)	2500 м	100 м
Максимальное количество узлов	500 (1000 соединений)	1024	260 для экранированной витой па-

Характеристика	FDDI	Ethernet	Token Ring
			ры, 72 для неэкранированной витой пары
Тактирование и восстановление после отказов	Распределенная реализация тактирования и восстановления после отказов	Не определены	Активный монитор

5.10 Технология ATM

Гетерогенность — неотъемлемое качество любой крупной вычислительной сети, и на согласование разнородных компонент системные интеграторы и администраторы тратят большую часть своего времени. Поэтому любое средство, сулящее перспективу уменьшения неоднородности сети хотя бы в каком-либо одном аспекте, привлекает всеобщее внимание и возбуждает ажиотаж. Технология асинхронного режима передачи (***Asynchronous Transfer Mode, ATM***) предназначена ввести единообразие сразу в нескольких аспектах:

- Общие транспортные протоколы для локальных и глобальных сетей.
- Совмещение в рамках одной транспортной системы компьютерного и мультимедийного трафика, причем для каждого вида трафика качество обслуживания будет соответствовать его потребностям.
- Иерархию скоростей передачи данных, от десятков мегабит до гигабит в секунду с гарантированной пропускной способностью для ответственных приложений.

ATM решает эти проблемы за счет деления информации любого типа на небольшие ячейки фиксированной длины. Ячейка ATM имеет размер 53 байта, пять из которых составляют заголовок, оставшиеся 48 — собственно информацию. В сетях ATM данные должны вводиться в форме ячеек или преобразовываться в ячейки с помощью функций адаптации. Сети ATM состоят из коммутаторов, соединенных транковыми каналами ATM. Краевые коммутаторы, к которым подключаются пользовательские устройства, обеспечивают функции адаптации, если ATM не ис-

пользуется вплоть до пользовательских станций. Другие коммутаторы, расположенные в центре сети, обеспечивают перенос ячеек, разделение транков и распределение потоков данных. В точке приема функции адаптации восстанавливают из ячеек исходный поток данных и передают его устройству-получателю.

Передача данных в коротких ячейках позволяет ATM эффективно управлять потоками различной информации и обеспечивает возможность приоритизации трафика.

Пусть два устройства передают в сеть ATM данные, срочность доставки которых различается (например, голос и трафик ЛВС). Сначала каждый из отправителей делит передаваемые данные на ячейки. Даже после того, как данные от одного из отправителей будут приниматься в сеть, они могут чередоваться с более срочной информацией. Чередование может осуществляться на уровне целых ячеек и малые размеры последних обеспечивают в любом случае непродолжительную задержку. Такое решение позволяет передавать срочный трафик практически без задержек, приостанавливая на это время передачу некритичной к задержкам информации. В результате ATM может обеспечивать эффективную передачу всех типов трафика.

Даже при чередовании и приоритизации ячеек в сетях ATM могут наступать ситуации насыщения пропускной способности. Для сохранения минимальной задержки даже в таких случаях ATM может отбрасывать отдельные ячейки при насыщении. Реализация стратегии отбрасывания ячеек зависит от производителя оборудования ATM, но в общем случае обычно отбрасываются ячейки с низким приоритетом (например, данные) для которых достаточно просто повторить передачу без потери информации. Коммутаторы ATM с расширенными функциями могут при отбрасывании ячеек, являющихся частью большого пакета, обеспечить отбрасывание и оставшихся ячеек из этого пакета — такой подход позволяет дополнительно снизить уровень насыщения и избавиться от излишнего объема повторной передачи. Правила отбрасывания ячеек, задержки данных и т.п. определяются набором параметров, называемым качеством обслуживания (Quality of Service) или QoS. Разным приложениям требуется различный уровень QoS и ATM может обеспечить этот уровень.

Поскольку приходящие из разных источников ячейки могут содержать голос, данные и видео, требуется обеспечить независимый контроль для передачи всех типов трафика. Для решения этой задачи используется концепция виртуальных устройств. Виртуальным устройством называется связанный набор сетевых ресурсов, который выглядит как реальное соединение между пользователями, но на самом деле является частью разделяемого множеством пользователей оборудования. Для того, чтобы сделать связь пользователей с сетями АТМ как можно более эффективной, виртуальные устройства включают пользовательское оборудование, средства доступа в сеть и собственно сеть АТМ.

В заголовке АТМ виртуальный канал обозначается комбинацией двух полей — ***VPI*** (***идентификатор виртуального пути***) и ***VCI*** (***идентификатор виртуального канала***). Виртуальный путь применяется в тех случаях, когда 2 пользователя АТМ имеют свои собственные коммутаторы на каждом конце пути и могут, следовательно, организовывать и поддерживать свои виртуальные соединения. Виртуальный путь напоминает канал, содержащий множество кабелей, по каждому из которых может быть организовано виртуальное соединение.

Поскольку виртуальные устройства подобны реальным, они также могут быть *выделенными* или *коммутируемыми*. В сетях АТМ *выделенные* соединения называются ***постоянными виртуальными устройствами*** (***PVC***), создаваемыми по соглашению между пользователем и оператором (подобно выделенной телефонной линии). Коммутируемые соединения АТМ используют ***коммутируемые виртуальные устройства*** (***SVC***), которые устанавливаются путем передачи специальных сигналов между пользователем и сетью. Протокол, используемый АТМ для управления виртуальными устройствами подобен протоколу ISDN. Вариант для ISDN описан в стандарте Q.931, АТМ — в Q.2931.

Виртуальные устройства АТМ поддерживаются за счет мультиплексирования трафика, что существенно снижает расходы на организацию и поддержку магистральных сетей. Если в одном из виртуальных устройств уровень трафика невысок, другое устройство может использовать часть свободных возможностей. За счет этого обеспечивается высокий уровень эффективно-

сти использования пропускной способности ATM и снижаются цены. Небольшие ячейки фиксированной длины позволяют сетям ATM обеспечить быструю передачу критичного к задержкам трафика (например, голосового). Кроме того, фиксированный размер ячеек обеспечивает практически постоянную задержку, позволяя эмулировать устройства с фиксированной скоростью передачи типа T1-E1. Фактически, ATM может эмулировать все существующие сегодня типы сервиса и обеспечивать новые услуги. ATM обеспечивает несколько классов обслуживания, каждый из которых имеет свою спецификацию QoS.

Класс QoS	**Класс обслуживания**	**Описание**
1	A	производительность частных цифровых линий (эмуляция устройств или CBR)
2	B	пакетные аудио/видео-конференции и multimedia (rt-VBR)
3	C	ориентированные на соединения протоколы типа frame relay (nrt-VBR)
4	D	протоколы без организации соединений типа IP, эмуляция ЛВС (ABR)
5	Unspecified	наилучшие возможности в соответствии с определением оператора (UBR)

Технология ATM изначально создавалась как часть сервиса ***Broadband ISDN*** под эгидой CCITT (сейчас ITU). Однако возможности ATM можно эффективно использовать и в локальных сетях.

Фактически, использование ATM обеспечивает сразу множество преимуществ. Во-первых, высокая скорость доступа за приемлемую цену, во-вторых, возможность организации компактных магистралей на базе ATM (***collapsed backbone***). Наконец, эта архитектура обеспечивает сквозное повышение эффективности использования сетевых ресурсов.

Стандарт, определяющий интерфейс между операторами и пользователями ATM называется ***Public User Network Interface*** или ***Public UNI.*** Этот интерфейс определяется для различных значений скорости. Первые услуги ATM предлагались в основном со скоростью T3 (45 Мбит/с). Сейчас многие операторы

предлагают скорость 155 Мбит/с и выше, но такая полоса обычно не требуется пользователям, да и стоимость подобных услуг весьма высока. Для большинства пользователей, планирующих организовать доступ к ATM или создать частную сеть ATM основной проблемой является стоимость оборудования.

Форум ATM — организация производителей оборудования ATM и пользователей работает в направлении развития стандартов и обеспечения интероперабельности оборудования. В конечном итоге это не может не привести к снижению цен. Кроме обеспечения интероперабельности ATM ведется большая работа по реализации ATM на скоростях меньше T3. Здесь возможно несколько вариантов:

- Полнофункциональные решения ATM при скорости T1. Один стандарт для ATM T1 уже утвержден, но некоторые производители и пользователи считают, что связанные с реализацией этого стандарта накладные расходы слишком велики — канал T1 с полосой 1.544 Мбит/с может обеспечить полезную полосу только около 1.1 Мбит/с.
- Так называемый dixie-стандарт (DXI — Data eXchange Interface). DXI был разработан как способ использования ATM в кадровом режиме с маршрутизаторами и другими устройствами передачи данных и специальными устройствами DSU, обеспечивающими преобразование кадров в реальные ячейки ATM. DXI работает через стандартные интерфейсы типа V.35 и HSSI.
- Интерфейс пользователь — сеть Frame Relay или F-UNI (произностися как FOONY), являющийся стандартом использования frame relay для доставки кадров данных ATM в сеть, которая будет конвертировать их в ячейки непосредственно на границе сети.
- Инверсное мультиплексирование ATM или AIM — стандарт для инверсного мультиплексирования множества линий T1 в один транк с полосой между T1 и T3. Такая полоса обеспечивает поддержку ATM для приложений, где скоростные запросы незначительно превышают возможности T1.

Стек протоколов ATM

Стек протоколов ATM соответствует нижним уровням семиуровневой модели ISO/OSI и включает адаптационные уровни

ATM, называемые ***AAL1-AAL5,*** и собственно уровень ATM. Адаптационные уровни транслируют пользовательские данные от верхних уровней коммуникационных протоколов в пакеты, формат и размеры которых соответствуют стандарту ATM. Каждый уровень AAL обрабатывает пользовательский трафик с определенными характеристиками. ***Уровень AAL 1*** занимается трафиком с ***постоянной битовой скоростью*** (***CBR***), который характерен, например, для цифрового видео и цифровой речи и чувствителен как к потере ячеек, так и к временным задержкам. Этот трафик передается в сетях ATM так, чтобы эмулировать обычные выделенные цифровые линии. ***Уровень AAL 3/4*** обрабатывает пульсирующий трафик с ***переменной битовой скоростью*** (***VBR***), обычно характерный для трафика локальных сетей. Этот трафик обрабатывается так, чтобы не допустить потерь ячеек, но ячейки могут задерживаться коммутатором. ***Уровень AAL 3/4*** выполняет сложную процедуру контроля ошибок при передаче ячеек для их гарантированной безошибочной доставки. Уровень ***AAL5*** является упрощенным вариантом уровня ***AAL4***, он работает быстрее.

<table>
<tr><td rowspan="3">Верхние уровни сети Уровни адаптации ATM(AAL1-5)</td><td rowspan="2">Подуровень конвергенции (CS)</td><td>Общая часть подуровня конвергенции</td></tr>
<tr><td>Специфическая для сервиса часть</td></tr>
<tr><td colspan="2">Подуровень сегментации и реассемблирования</td></tr>
<tr><td colspan="3">Уровень ATM (маршрутизация пакетов, мультиплексирование, управление потоком, обработка приоритетов)</td></tr>
<tr><td rowspan="2">Физический уровень</td><td colspan="2">Подуровень согласования передачи</td></tr>
<tr><td colspan="2">Подуровень, зависящий от физической среды</td></tr>
</table>

Рис. 5.32 — Иерархия уровней протокола ATM

Введение различных классов сервисов, реализуемых в стеке протоколов ATM адаптационными уровнями AAL, а также самим протоколом ATM, и позволяет реализовать в сетях ATM совместное сосуществование трафиков разной природы. Коммутаторы ATM, получая в поле типа данных ячейки (поле PTI) информа-

цию о классе сервиса, принимает решение о приоритете обслуживания данной ячейки. Для того, чтобы каждый класс сервиса выполнялся с нужным уровнем качества, в технологии ATM предусмотрены достаточно сложные процедуры заказа качества обслуживания, которые выполняются между станцией и сетью при установлении соединения.

Классы сервиса

В сети ATM каждый раз, когда приложению необходимо установить соединение между двумя пользователями, оно должно заказать вид сервиса, в соответствии с которым будет обслуживать трафик по данному соединению. Классы сервиса ATM содержат ряд параметров, которые определяют гарантии качества сервиса. В спецификациях форума ATM предусмотрено несколько классов сервиса — ***CBR, VBR, UBR и ABR*** (появился совсем недавно). Гарантии качества сервиса могут определять минимальный уровень доступной пропускной способности и предельные значения задержки ячейки и вероятности потери ячейки.

Сервис CBR (***constant bit rate***, ***сервис с постоянной битовой скоростью***) представляет собой наиболее простой класс сервиса ATM. Когда сетевое приложение устанавливает соединение CBR, оно заказывает пиковую скорость трафика ячеек (***peak cell rate, PCR***), которая является максимальной скоростью, которое может поддерживать соединение без риска потерять ячейку.

Класс сервиса	Гарантии пропускной способности	Гарантии изменения задержки	Обратная связь при переполнении
CBR	+	+	–
VBR	+	+	–
UBR	–	–	–
ABR	+	+	+

Затем данные передаются по этому соединению с запрошенной скоростью — не более и, в большинстве случаев, не менее. Любой трафик, передаваемый станцией с большей скоростью, может сетью просто отбрасываться, а передача трафика сетью со скоростью, ниже заказанной, не будет удовлетворять приложение. CBR-соединения должны гарантировать пропускную

способность с минимальной вероятностью потери ячейки и низкими изменениями задержки передачи ячейки. Когда приложение заказывает CBR сервис, то оно требует соблюдения предела изменения задержки передачи ячейки. Сервис CBR предназначен специально для передачи голоса и видео в реальном масштабе времени. Сервис CBR также подходит для эмуляции цифровых каналов типа T1/E1.

Для соединений CBR нет определенных ограничений на скорость передачи данных, и каждое виртуальное соединение может запросить различные постоянные скорости передачи данных. Сеть должна резервировать полную полосу пропускания, запрашиваемую конкретным соединением.

Класс трафика VBR (***variable bit rate, сервис с переменной битовой скоростью***) включает два подкласса: трафик ***VBR реального времени*** (***VBR-RT)*** и трафик ***VBR не реального времени*** (***VBT-NRT***). Трафик VBR-RT допускает очень узкие границы для задержки передачи ячеек и может использоваться для передачи данных приложений реального времени, которые позволяют небольшое изменение задержки передачи ячеек, таких как видео, генерируемое кодеком с переменной скоростью данных или компрессированный видеотрафик, в котором удалены промежутки молчания. Трафик VBR-NRT в свою очередь предъявляет менее жесткие требования к задержке передачи ячеек. Он специально предназначен для передачи коротких, пульсирующих сообщений, таких как сообщения, возникающие при обработке транзакций системами управления базами данных.

По сравнению с сервисом CBR, VBR требует более сложной процедуры заказа соединения между сетью и приложением. В дополнение к пиковой скорости приложение VBR заказывает еще и другой параметр: ***PCR — длительно поддерживаемую скорость*** (***sustained rate***), которая представляет собой среднюю скорость передачи данных, которая разрешена приложению. Пользователь может превышать скорость вплоть до величины ***PCR***, но только на короткие периоды времени, а соединение VBR будет использовать среднее значение SCR для управления трафиком, снижая его интенсивность на соответствующие периоды времени.

Как и при CBR-соединении, приложение и сеть должны прийти к соглашению относительно пиковой скорости PCR и до-

пустимости задержек передачи ячеек. Но в отличии от CBR, соединение VBR должно установить временной предел — как долго могут передаваться данные на скорости PCR.

Когда этот предел, известный как допустимая пульсация, превышается, за ним должен следовать период более низкой активности станции, чтобы обеспечить заданный уровень SCR. Эти периоды низкой активности дают возможность другим видам трафика, таким как ABR, получить доступ к сети.

Как и в случае CBR, пользователи VBR получают гарантированное обслуживание в отношении потерь ячеек, изменения задержек передачи ячеек и доступной полосы пропускания до тех пор, пока трафик удовлетворяет определенным при соединении требованиям. Однако для многих приложений, которые могут быть чрезвычайно «взрывными» в отношении интенсивности трафика, невозможно точно предсказать параметры трафика, оговариваемые при установлении соединения. Например, обработка транзакций и трафик двух взаимодействующих локальных сетей непредсказуемы по своей природе, изменения трафика слишком велики, чтобы заключить с сетью какое-либо разумное соглашение.

В результате администраторы сетей, ответственные за такие приложения, имеют три возможности. Они могут заплатить за дополнительную пропускную способность, которая может оказаться неиспользованной. Они могут попытаться управлять пульсациями трафика более тонко (сложная задача для большинства приложений). Или же они могут превысить скорость, оговоренную при установлении соединения, пренебрегая гарантированным качеством обслуживания.

Для тех, кто выбирает последний вариант, последствия скорее всего будут и самыми тяжелыми — потеря ячеек. Потерянные ячейки должны быть повторно переданы узлом-отправителем. Для ответственных приложений это серьезная проблема, для низкоприоритетных приложений, таких как электронная почта, повторная передача ячеек является досадной потерей времени.

В отличие от CBR и VBR, ***сервис UBR*** (***unspecified bit rate, неопределенная битовая скорость***) не определяет ни битовую скорость, ни параметры трафика, ни качество сервиса. Сервис UBR предлагает только доставку «по возможности», без гарантий по утере ячеек, задержке ячеек или границам изменения задерж-

ки. Разработанный специально для возможности превышения полосы пропускания, сервис UBR представляет собой частичное, но неадекватное решение для тех непредсказуемых «взрывных» приложений, которые не готовы согласиться с фиксацией параметров трафика.

Главными недостатками подхода UBR являются отсутствие управления потоком данных и неспособность принимать во внимание другие типы трафика. Когда сеть становится перегруженной, UBR-соединения продолжают передавать данные. Коммутаторы сети могут буферизовать некоторые ячейки поступающего трафика, но в некоторый момент буфера переполняются и ячейки теряются. А так как UBR-соединения не заключали никакого соглашения с сетью об управлении трафиком, то их ячейки отбрасываются в первую очередь. Потери ячеек UBR могут быть так велики, что «выход годных» ячеек может упасть ниже 50%, что совсем неприемлемо.

Сервис ABR (***available bit rate***), подобно сервису UBR, использует превышение полосы пропускания, но он использует технику управления трафиком для оценки степени переполнения сети и избегает потерь ячеек. ABR — это первый класс сервиса технологии ATM, который действительно обеспечивает надежный транспорт для приложений с пульсирующим трафиком за счет того, что он может находить неиспользуемые интервалы времени в трафике и заполнять их своими пакетами, если другим классам сервиса эти интервалы не нужны.

Как и в сервисах CBR и VBR, при установлении соединения ABR заключается соглашение о пиковой скорости PCR. Однако, соглашение о пределах изменения задержки передачи ячеек или о параметрах пульсации не заключается. Вместо этого сеть и приложение заключают соглашение о требуемой минимальной скорости трафика. Это гарантирует приложению небольшую пропускную способность, обычно минимально необходимую для того, чтобы приложение работало. Пользователь соединения ABR соглашается не передавать данные со скоростью выше пиковой, то есть PCR, а сеть соглашается всегда обеспечивать минимальную скорость передачи ячеек — ***MCR*** (***minimum cell rate***).

Скорость MCR вычисляется в ячейках в секунду, на основании способности приложения выдержать определенную задерж-

ку. Например, если приложению нужно передать файл в 1 Мбайт (около 20000 ячеек ATM) по крайней мере за 2 секунды, то требуемая скорость MCR для приложения составит 10000 ячеек в секунду.

Если приложение при установлении ABR-соединения не определяет максимальную и минимальную скорости, то по умолчанию они принимаются равными скорости линии доступа станции к сети (для PCR) и нулю для MCR.

Пользователь соединения ABR получает гарантированное качество сервиса в отношении потери ячеек и пропускной способности. Что касается задержек передачи ячеек, то хотя они и сводятся к минимуму, но сервис ABR не дает абсолютных гарантий. Следовательно, сервис ABR не предназначен для приложений реального времени, а предназначен для приложений, в которых поток данных не очень чувствителен к задержкам в передаче.

<table>
<tr><td></td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td></td></tr>
<tr><td rowspan="6">заголовок</td><td colspan="4">Управление потоком (GFC)</td><td colspan="4">Идентификатор виртуального пути (VPI)</td><td>1</td></tr>
<tr><td colspan="4">Идентификатор виртуального пути (продолжение)</td><td colspan="4">Идентификатор виртуального канала (VCI)</td><td>2</td></tr>
<tr><td colspan="8">Идентификатор виртуального канала (продолжение)</td><td>3</td></tr>
<tr><td colspan="4">Идентификатор виртуального канала (продолжение)</td><td colspan="2">Тип данных (PTI)</td><td colspan="2">Приоритет потери пакета</td><td>4</td></tr>
<tr><td colspan="8">Управление ошибками в заголовке (HEC)</td><td>5</td></tr>
<tr><td>данные</td><td colspan="8">Данные пакета</td><td>6
...
53</td></tr>
</table>

Рис. 5.33 — Формат ячейки ATM

Стандарты физического уровня, используемые в сетях ATM

Стандарт ATM не включает спецификацию физического уровня, а пользуется спецификациями стандарта физического уровня на передачу данных по оптическим линиям связи ***SONET*** (***Synchronous Optical Network***) и ***SDH*** (***Synchronous Digital Hierarchy***). Стандарт SONET устанавливает скорости передачи данных с дискретностью 51.84 Мб/с до 2.488 Гб/с и может быть

расширен до 13 Гб/с, а SDH — с дискретностью 155.52 Мб/с. Базовая скорость 51.84 Мб/с была выбрана так, чтобы включить в себя скорости линий Т-3 и Е-3. Для линий со стандартными скоростями SONET введены следующие обозначения:

Витая пара	Оптоволокно	Скорость
STS-1	OC-1	51,84 Мб/с
STS-2	OC-3	155,520 Мб/с
	OC-9	466,560 Мб/с
	OC-12	622,080 Мб/с
	OC-18	933,120 Мб/с
	OC-24	1,244 Гб/с
	OC-36	1,866 Гб/с
	OC-48	2,488 Гб/с

Для связи станции с коммутатором используются скорости передачи данных до 155 Мб/с. Скорость 25 Мб/с не входит в разрешенные наборы скоростей стандартов SONET и SDH, поэтому она была отвергнута в свое время организаций Forum ATM. Однако, из-за того, что для такой скорости было найдено удачное и недорогое решение построения сетевого адаптера ATM на основе набора микросхем Token Ring, некоторые производители поддерживают эту скорость и она также, очевидно, будет узаконена в окончательной форме стандарта UNI.

Спецификация ATM LAN emulation

Технология ATM позволяет добиться высоких и сверхвысоких скоростей для связи отдельных узлов вычислительной сети, предоставляя каждому виду трафика сервис с заданными параметрами качества. Однако эта технология разрабатывалась сначала как *вещь в себе*, не учитывая тот факт, что в существующие технологии сделаны большие вложения, и поэтому никто не станет сразу отказываться от установленного и работающего оборудования, даже если появляется новое, более совершенное. Для территориальных сетей, для которых в первую очередь разрабатывалась технология ATM, это не так страшно, так как сети ATM могут использовать те же оптоволоконные каналы, что и существующие цифровые телефонные и вычислительные сети, а стоимость высокоскоростных оптоволоконных каналов, проложенных

на большие расстояния, превышает стоимость коммутаторов сети, так что в этом случае легче пойти на замену коммутаторов новыми.

Для локальных сетей дела обстоят по другому, поэтому желательно, чтобы новая технология могла работать в одной сети со старыми, улучшая характеристики сети там, где это нужно, и оставляя сети рабочих групп или отделов в прежнем виде.

Сейчас форум ATM разработал первую спецификацию, называемую ***LAN emulation*** (то есть ***эмуляция локальных сетей***), которая призвана обеспечить совместимость традиционных протоколов и оборудования локальных сетей с технологией ATM. Эта спецификация обеспечивает совместную работу этих технологий на канальном уровне. При таком подходе коммутаторы ATM работают в качестве высокоскоростных коммутирующих мостов магистрали локальной сети, обеспечивая не только скорость, но и гибкость соединений ATM-коммутаторов между собой, так что эта магистраль не обязательно образуется на внутренней шине одного устройства, а может быть и распределенной.

Спецификация LAN emulation определяет способ преобразования пакетов и адресов MAC-уровня традиционных технологий локальных сетей в пакеты и коммутируемые *виртуальные соединения SVC* технологии ATM, и обратное преобразование. Всю работу по преобразованию протоколов выполняют специальные компоненты, встраиваемые в обычные концентраторы, коммутаторы и маршрутизаторы, поэтому ни коммутаторы ATM, ни рабочие станции локальных сетей не замечают того, что они работают с чуждыми им технологиями. Такая прозрачность была одной из главных целей разработчиков спецификации LAN emulation.

Так как эта спецификация определяет только канальный уровень взаимодействия, то с помощью ATM-коммутаторов и компонент LAN эмуляции можно образовать только виртуальные сегменты, а для их соединения нужно использовать обычные маршрутизаторы.

Основными элементами, реализующими спецификацию, являются программные компоненты LEC и LES. ***LEC*** (***LAN Emulation Client***) — это посредник, работающий между ATM-коммутаторами (и ATM-станциями) и станциями локальной сети,

а ***LES*** (***LAN Emulation Server***) преобразует MAC-адреса в ATM-адреса.

Клиентские части (LEC'и) назначают каждой присоединенной локальной сети ATM-адрес. Клиентские части динамически регистрируют MAC-адрес станции присоединенной локальной сети в сервере LES, который ведет общую таблицу соответствия MAC-адресов станций и ATM-адресов присоединенных локальных сетей.

Компоненты LEC и LES могут быть реализованы в любых устройствах: концентраторах, коммутаторах, маршрутизаторах или ATM-рабочих станциях.

Когда элемент LEC хочет послать пакет через сеть ATM станции другой локальной сети, также присоединенной к сети ATM, то он посылает запрос на разрешение ***адресов MAC-ATM*** серверу LES. Этот запрос иногда называют протоколом ***LE ARP*** (***LAN Emulation address resolution protocol***). Сервер LES отвечает на запрос, указывая ATM-адрес элемента LEC присоединенной сети назначения. Затем LEC исходной сети самостоятельно устанавливает ***виртуальное SVC-соединение*** через сеть ATM обычным способом, описанным в спецификации UNI. После установления связи MAC-кадры локальной сети преобразуются в ячейки ATM каждым элементом LEC с помощью стандартных функций сборки-разборки пакетов (функции SAR) стека ATM.

В спецификации LAN emulation также определен сервер для эмуляции в сети ATM широковещательных пакетов локальных сетей, а также пакетов с неизвестными адресами, так называемый сервер ***BUS*** (***Broadcast and Unknown Server***).

6 СТЕК СЕТЕВЫХ ПРОТОКОЛОВ TCP/IP

Стеки сетевых протоколов обычно разрабатываются по уровням модели OSI, причем каждый уровень отвечает за собственную фазу коммуникаций. Семейства протоколов, такие как TCP/IP, это комбинации различных протоколов на различных уровнях. TCP/IP состоит из четырех уровней, как показано в табл. 6.1.

Таблица 6.1

Прикладной	Telnet, FTP, SNMP, WWW и т.д.
Транспортный	TCP,UDP
Сетевой	IP, ICMP, IGMP
Канальный	Ethernet, FDDI, Token Ring, ARCNet и т.д.

Каждый уровень несет собственную функциональную нагрузку.

1. ***Канальный уровень*** (***link layer***). Еще его называют уровнем сетевого интефейса. Обычно включает в себя драйвер устройства в операционной системе и соответствующую сетевую интерфейсную плату в компьютере. Вместе они обеспечивают аппаратную поддержку физического соединения с сетью (с кабелем или с другой используемой средой передачи).

2. ***Сетевой уровень*** (***network layer***), иногда называемый уровнем межсетевого взаимодействия, отвечает за передачу пакетов по сети. Маршрутизация пакетов осуществляется именно на этом уровне. ***IP*** (Internet Protocol), ***ICMP*** (Internet Control Message Protocol — протокол управления сообщениями Internet) и ***IGMP*** (Internet Group Management Protocol — протокол управления группами Internet) обеспечивают сетевой уровень в семействе протоколов TCP/IP.

3. ***Транспортный уровень*** (***transport layer***) отвечает за передачу потока данных между двумя компьютерами и обеспечивает работу прикладного уровня, который находится выше. В семействе протоколов TCP/IP существует два транспортных протокола: ***TCP*** (Transmission Control Protocol) и ***UDP*** (User Datagram Protocol). TCP осуществляет надежную передачу данных между

двумя компьютерами. Он обеспечивает деление данных, передающихся от одного приложения к другому, на пакеты подходящего для сетевого уровня размера, подтверждение принятых пакетов, установку тайм-аутов, в течение которых должно прийти подтверждение на пакет, и так далее. Так как надежность передачи данных гарантируется на транспортном уровне, на прикладном уровне эти детали игнорируются. UDP предоставляет более простой сервис для прикладного уровня. Он просто отсылает пакеты, которые называются дейтаграммами (*datagram*) от одного компьютера к другому. При этом нет никакой гарантии, что датаграмма дойдет до пункта назначения. За надежность передачи данных, при использовании датаграмм отвечает прикладной уровень. Для каждого транспортного протокола существуют различные приложения, которые их используют.

4. ***Прикладной уровень*** (***application layer***) определяет детали каждого конкретного приложения. Существует несколько распространенных приложений TCP/IP, которые присутствуют практически в каждой реализации:

- Telnet — удаленный терминал.
- FTP, File Transfer Protocol — протокол передачи файлов.
- SMTP, Simple Mail Transfer Protocol — простой протокол передачи электронной почты.
- SNMP, Simple Network Management Protocol — простой протокол управления сетью.

6.1 Архитектура TCP/IP

На рис. 6.1 приведена схема взаимодействия двух сетевых приложений (FTP клиента с FTP сервером), расположенных в локальной сети на базе Ethernet. Сервер предоставляет некоторые типы сервиса клиентам. В данном случае это доступ к файлам на сервере.

С правой стороны на рис. 6.1 показано, что прикладной уровень обеспечивается пользовательским процессом, тогда как три нижних уровня обычно встроены в ядро операционной системы или реализованы аппаратно. Прикладной уровень обычно является приложением и взаимодействует с пользователем, а не занимается передачей данных по сети. Три нижних уровня ничего не

знают о работающих над ними приложениях, однако отвечают за все детали коммуникаций.

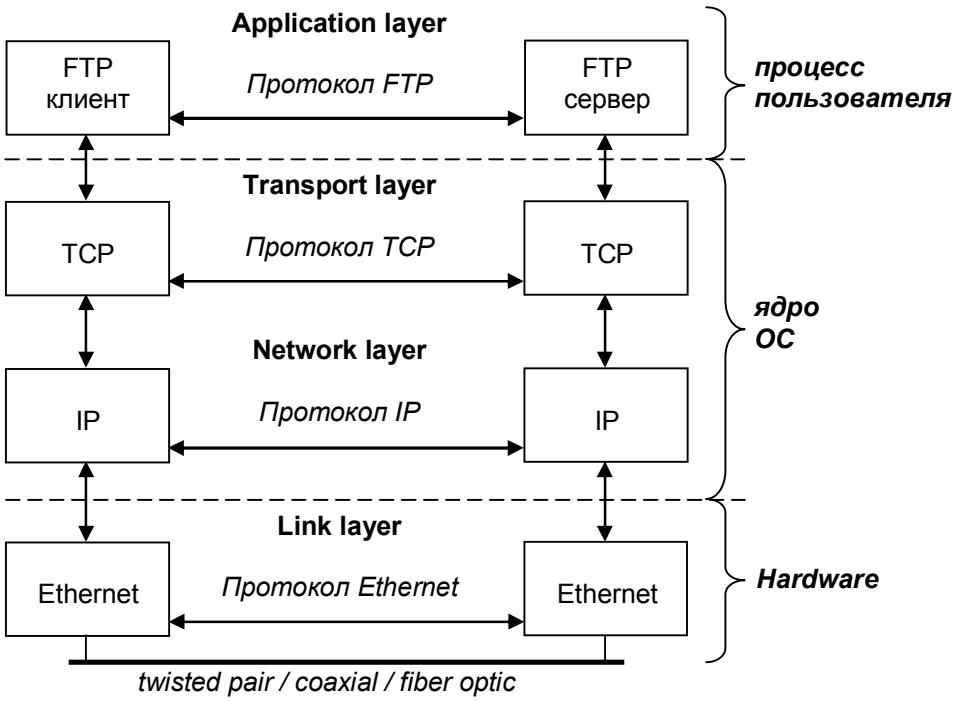


Рис. 6.1 — Два хоста в локальной сети с работающим FTP

Наиболее простой путь осуществить межсетевое взаимодействие — это объединить две или более сетей с помощью ***маршрутизатора*** (***router***). Маршрутизатор представляет из себя программно-аппаратное устройство с собственной операционной системой. Огромное достоинство маршрутизаторов заключается в том, что они могут объединить сети, построенные на различных физических принципах: Ethernet, Token Ring, Point-to-Point, FDDI и т.п.

Маршрутизаторы обеспечивают взаимодействие на уровне IP-протокола, что иллюстрирует разницу между конечной системой (***end system***) — это два компьютера на каждой стороне (рис. 6.1), и промежуточной системой (***intermediate system***), в данном случае это использование маршрутизатора для межсетевого взаимодействия (рис. 6.2).

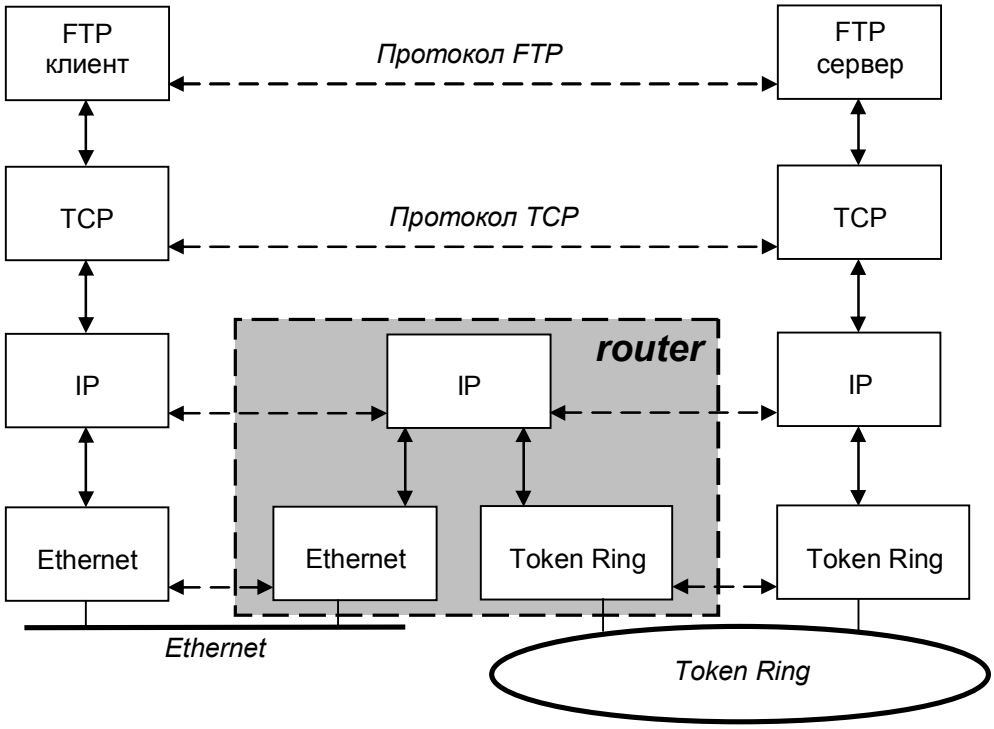


Рис. 6.2 — Две сети, соединенные через маршрутизатор

Маршрутизатор, по определению, имеет два или несколько интерфейсов сетевого уровня (если он объединяет две или более сетей). Любая система с несколькими интерфейсами называется ***многоинтерфейсной*** (***multihomed***). Большинство реализаций TCP/IP позволяют компьютерам с несколькими интерфейсами функционировать в качестве маршрутизаторов. Однако компьютеры должны быть специально сконфигурированы, чтобы решать задачи маршрутизации.

Одна из основных задач объединения сетей заключается в том, чтобы скрыть все детали физического процесса передачи информации между приложениями, находящимися в разных сетях. Поэтому нет ничего удивительного в том, что в объединенных сетях (см. рис. 6.2), прикладные уровни не заботятся (и не должны заботиться) о том, что один компьютер находится в сети Ethernet, а другой в сети Token Ring с маршрутизатором между ними. Даже если бы между сетями было 20 маршрутизаторов и различные типы физического соединения, приложения работали бы точно так же. Подобная концепция, при которой детали физического объединения сетей скрыты от приложений, определяет мощность и гибкость такой технологии объединения сетей.

Существует еще один метод объединения сетей — с помощью ***мостов*** (***bridge***). В этом случае сети объединяются на канальном уровне, тогда как маршрутизаторы объединяют сети на сетевом уровне. Стоит отметить, что объединение TCP/IP сетей в настоящее время осуществляется в основном с помощью маршрутизаторов, а не с помощью мостов. Поэтому мы более подробно рассмотрим маршрутизаторы.

6.2 Протоколы, пакеты и инкапсуляция TCP/IP

В действительности, семейство протоколов TCP/IP объединяет значительно больше протоколов. TCP и UDP — два основных протокола транспортного уровня. Оба используют IP в качестве сетевого уровня (рис. 6.3). Протокол IP предоставляет ненадежный сервис. Это означает, что в процессе своей работы протокол передает пакет от источника к пункту назначения, однако не предоставляет никаких гарантий того, что пакет дойдет по назначению.

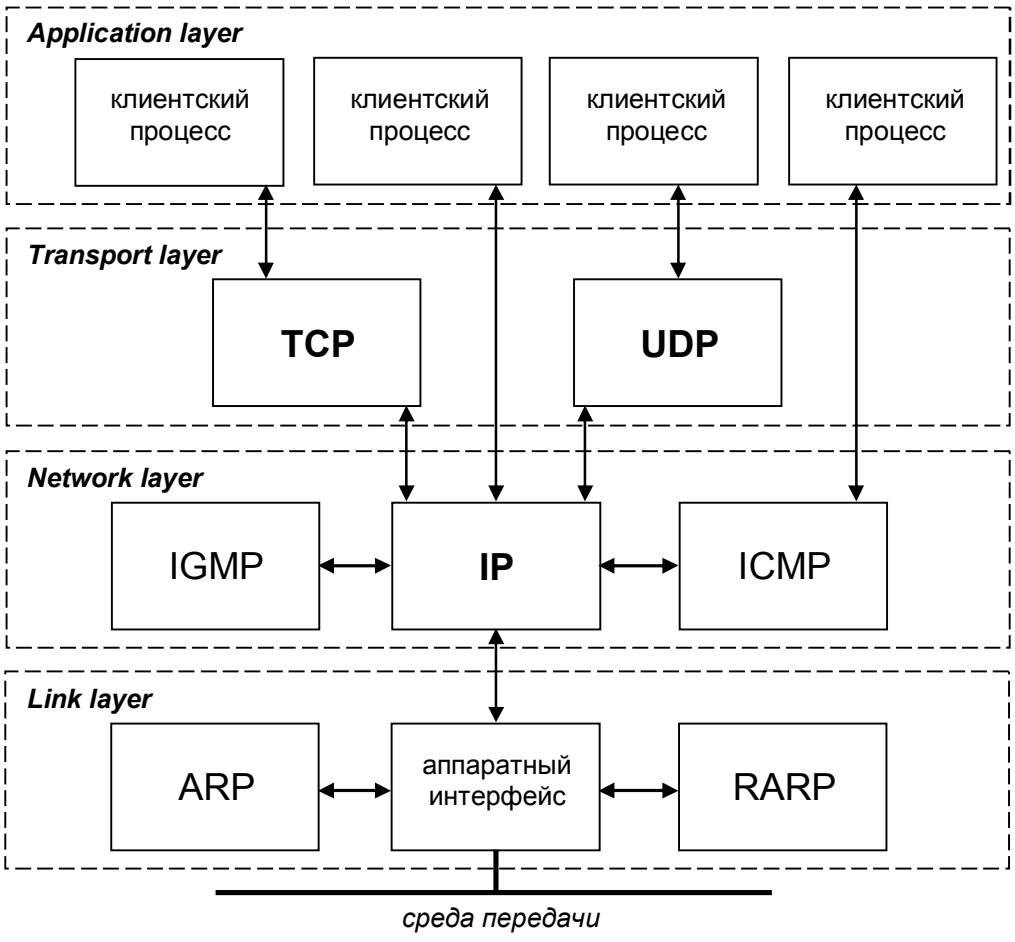


Рис. 6.3 — Различные протоколы на разных уровнях стека TCP/IP

TCP, с другой стороны, предоставляет надежный транспортный уровень, который пользуется ненадежным сервисом IP. Чтобы обеспечить подобный сервис, TCP выставляет тайм-ауты и осуществляет повторные передачи, отсылает и принимает подтверждения и так далее. Транспортный уровень и сетевой уровень несут различную ответственность за передачу данных.

UDP отправляет и принимает дейтаграммы (*datagram*). Дейтаграмма это блок информации (определенное количество байт информации, которое указывается отправителем), который отправляется от отправителя к приемнику. В отличие от TCP, UDP является ненадежным протоколом. Не существует гарантий, что дейтаграмма достигнет конечной точки назначения.

IP это основной протокол сетевого уровня. Он используется как TCP, так и UDP. Каждый блок информации TCP и UDP, который передается по объединенным сетям, проходит через IP уровень в каждой конечной системе и в каждом промежуточном маршрутизаторе. На рис. 6.3 показаны приложения, которые имеют прямой доступ к IP. Такой доступ используется довольно редко, но существует возможность его осуществить (некоторые ранние протоколы маршрутизации были разработаны именно подобным образом). Также в процессе экспериментов при создании новых транспортных уровней используется возможность доступа к протоколу IP.

ICMP является дополнением к протоколу IP. Он используется IP уровнем для обмена сообщениями об ошибках и другой жизненно важной информацией уровня IP. Несмотря на то, что ICMP используется в основном IP уровнем, приложения также могут получить доступ к ICMP. Протокол управления группами Internet (***IGMP*** — *Internet Groupe Management Protocol*), используется при групповой адресации: при этом UDP датаграммы рассылаются нескольким получателям.

Протокол определения адреса (***ARP*** — *Address Resolution Protocol*) и обратный протокол определения адреса (***RARP*** — *Reverse Address Resolution Protocol*) это специализированные протоколы, используемые только с определенным типом сетевых интерфейсов (такие как Ethernet и Token ring). Они применяются для преобразования формата адресов, используемого IP уровнем в формат адресов, используемый сетевым интерфейсом.

Инкапсуляция

Когда приложение посылает данные с использованием TCP, данные опускаются вниз по стеку протоколов, проходя через каждый уровень, до тех пор пока они не будут отправлены в виде потока битов по сети. Каждый уровень добавляет свою информацию к данным путем пристыковки заголовков (а иногда завершителей). На рис. 6.4 показан этот процесс. Пакет который TCP посылает в IP, называется *TCP-сегментом*, если данные с этого уровня отправляет UDP, то они называются *UDP-дейтаграммой*. Блок данных, который IP посылает в сетевой интерфейс, называется *IP-дейтаграммой*. Поток битов, который передается по Ethernet, называется *фреймом* (*frame*).

Одной из физических характеристик фрейма Ethernet является та, что размер данных должен быть в диапазоне между 46 и 1500 байт, это может быть как IP дейтаграмма, так и фрагмент IP дейтаграммы.

Что касается UDP данных, то картина там практически идентичная TCP-сегменту, единственное различие заключается в том, что размер UDP заголовка составляет 8 байт.

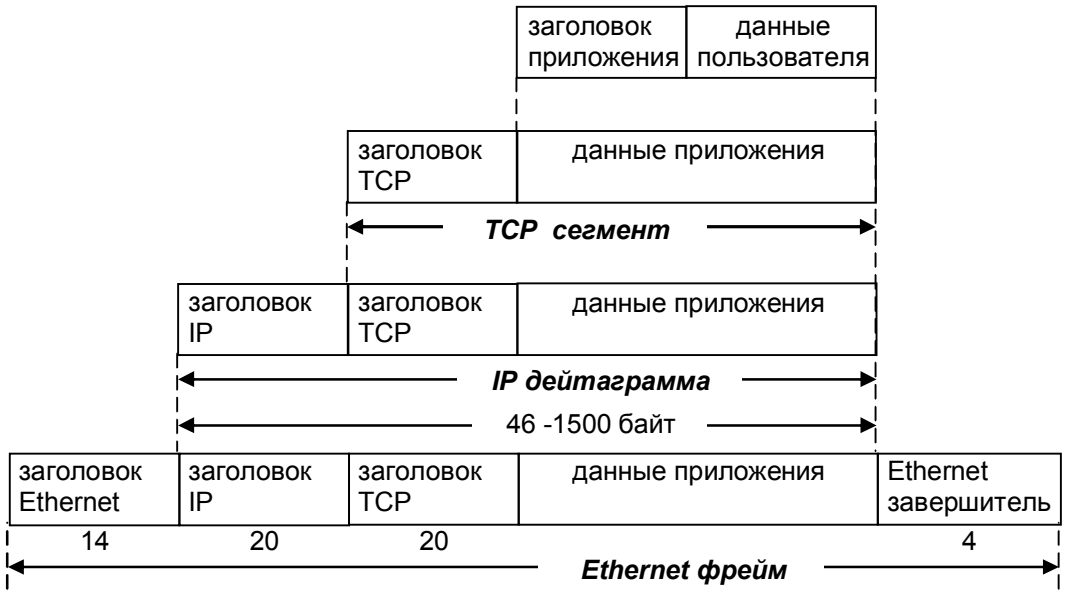


Рис. 6.4 — Инкапсуляция данных стека протоколов TCP/IP

Снова обратимся к рис. 6.3, на котором показано как различные приложения могут использовать TCP или UDP. Протоколы транспортного уровня сохраняют в заголовке идентификатор

приложения, которое их использует. TCP и UDP используют шестнадцатибитный ***номер порта*** (***port number***) источника и номер порта назначения в своих заголовках, чтобы указать на требуемые приложения.

Таблица 6.2 — Номера портов популярных протоколов стека TCP/IP

TCP Transmission Control Protocol			**UDP User Datagram Protocol**	
FTP File Transfer Protocol	21		**TFTP** Trivial File Transfer Protocol	69
HTTP Hypertext Transfer Protocol			**SNMP** Simple Network Management Protocol	161
SMTP Simple Mail Transfer Protocol	25		**DHCP** Dynamic Host Control Protocol	
TELNET Terminal Connection	23		**DNS** Domain Name System	53
			RIP	520

Точно так же TCP, UDP, ICMP и IGMP посылают данные в IP. Протокол IP должен добавить какой-либо идентификатор к IP заголовку, который он генерирует, чтобы указать какому из протоколов принадлежат данные. IP делает это путем сохранения восьмибитного значения в своем заголовке, которое называется ***полем протокола*** (***protocol***). Это значение равно 1 для ICMP, 2 для IGMP, 6 для TCP и 17 для UDP.

Сетевой интерфейс посылает и принимает фреймы, принадлежащие IP, ARP и RARP. Должна существовать форма идентификации в заголовке Ethernet, которая бы указывала, какой сетевой уровень сгенерировал данные. Для этого существует шестнадцатибитное поле ***типа фрейма*** (***frame type***) в заголовке Ethernet.

Демультиплексирование (Demultiplexing)

Когда фрейм Ethernet принимается компьютером приемником, он начинает свой путь вверх по стеку протоколов, при этом все заголовки удаляются в соответствующих уровнях. Каждый протокол просматривает определенные идентификаторы в заголовке, чтобы определить, какой следующий верхний уровень должен получить данные. Этот процесс называется ***демультиплексированием*** (***demultiplexing***) см. рис. 6.5.

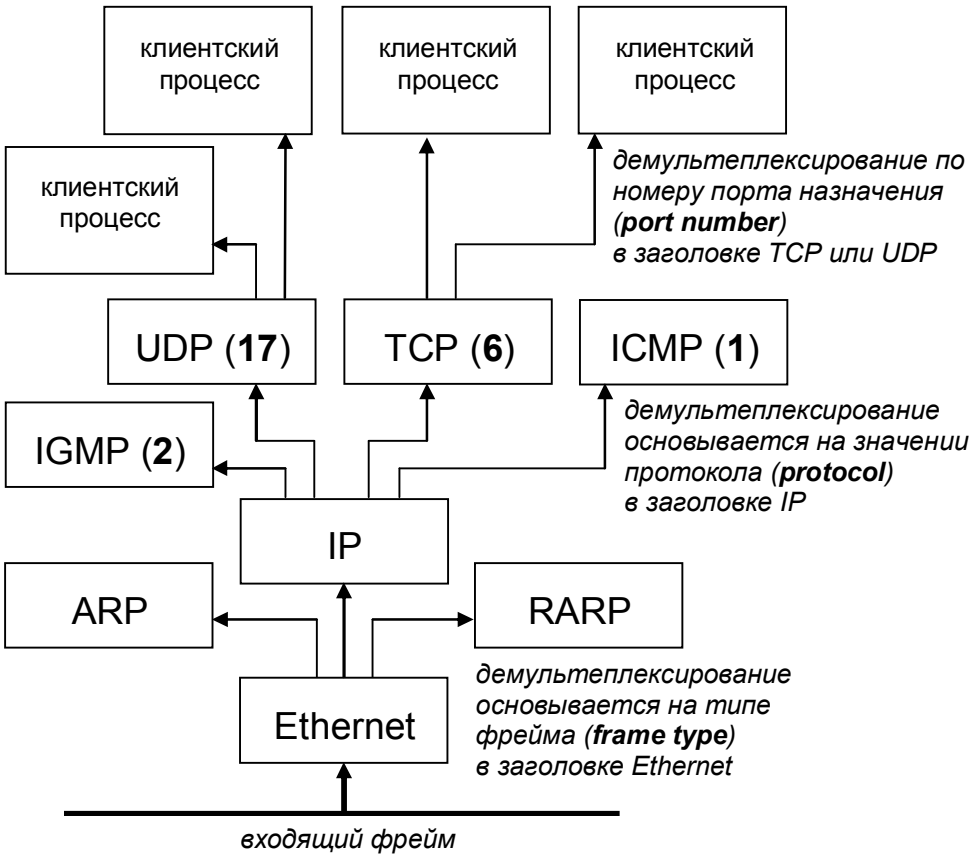


Рис. 6.5 — Демультиплексирование полученного Ethernet фрейма

Когда мы будем рассматривать TCP более подробно, мы увидим, что в действительности демультиплексирование входящих сегментов использует номер порта назначения, IP адрес источника и номер порта источника.

6.3 Протоколы транспортного уровня

Двумя основными протоколами транспортного уровня являются надежный протокол управления передачей данных **TCP** (**Transmission Control Protocol**) и быстрый протокол дейтаграмм пользователя **UDP** (**User Datagram Protocol**). TCP реализует сетевое взаимодействие в режиме с установлением логического (виртуального) соединения, а UDP — без оного. Функции каждого протокола реализуются компонентой программного обеспечения (обычно входящей в состав операционной системы), которую будем называть модулем. Взаимодействие модулей соседних

уровней осуществляется через стандартизированный интерфейс, имеющий, как правило, процедурный характер.

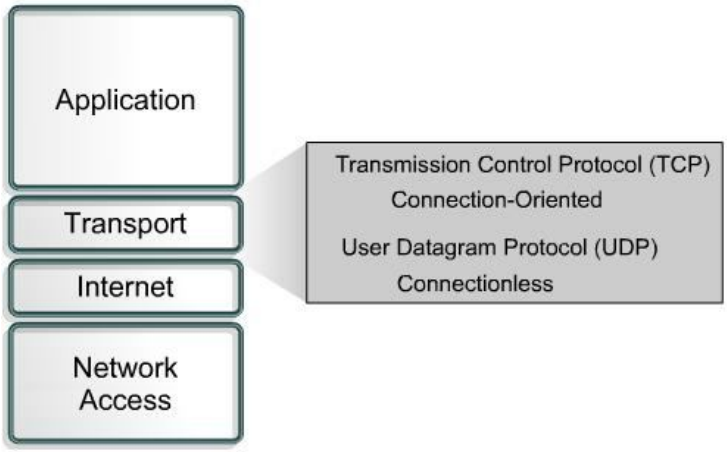


Рис. 6.6 — Транспортный уровень TCP/IP

Протокол управления передачей TCP

Протокол управления передачей TCP является протоколом транспортного уровня и базируется на возможностях, предоставляемых межсетевым протоколом IP. Основная задача TCP — обеспечение надежной передачи данных в сети. Он осуществляет доставку дейтограмм, называемых сегментами, в виде байтовых потоков с установлением соединения. Он использует контрольные суммы пакетов для проверки их целостности и освобождает прикладные процессы от необходимости таймаутов и повторных передач для обеспечения надежности. Для отслеживания подтверждения доставки в TCP реализуется алгоритм «скользящего» окна.

Описание протокола TCP дано в RFC 793. Протокол TCP взаимодействует с одной стороны с пользователем или прикладной программой, а с другой — с протоколом более низкого уровня, таким как протокол IP.

Основные характеристики протокола TCP перечислены ниже:

- реализует взаимодействие в режиме с установлением логического (виртуального) соединения;
- обеспечивает двунаправленную дуплексную связь;
- организует потоковый (с точки зрения пользователя) тип передачи данных;
- дает возможность пересылки части данных, как «экстренных»;

- для идентификации партнеров по взаимодействию на транспортном уровне использует 16-битовые «номера портов»;
- реализует принцип «скользящего окна» (***sliding window***) для повышения скорости передачи;
- поддерживает ряд механизмов для обеспечения надежной передачи данных.

Несмотря на то, что для пользователя передача данных с использованием протокола TCP выглядит как потоковая, на самом же деле обмен между партнерами осуществляется посредством пакетов.

Заголовок TCP-пакета

Рис. 6.7 — Формат TCP пакета

Порт источника и порт приемника — 16-битовые поля, содержащие номера портов, соответственно, источника и адресата TCP-пакета.

Номер в последовательности (sequence number) — 32-битовое поле, содержимое которого определяет (косвенно) положение данных TCP-пакета внутри исходящего потока данных, существующего в рамках текущего логического соединения.

В момент установления логического соединения каждый из двух партнеров генерирует свой начальный «номер в последовательности», основное требование к которому — не повторяться в

промежутке времени, в течение которого TCP-пакет может находиться в сети (по сути, это время жизни IP-сегмента). Партнеры обмениваются этими начальными номерами и подтверждают их получение. Во время отправления TCP-пакетов с данными поле «номер в последовательности» содержит сумму начального номера и количества байт ранее переданных данных.

Номер подтверждения (acknowledgement number) — 32-битовое поле, содержимое которого определяет (косвенно) количество принятых данных из входящего потока к TCP-модулю, формирующему TCP-пакет.

Смещение данных — четырехбитовое поле, содержащее длину заголовка TCP-пакета в 32-битовых словах и используемое для определения начала расположения данных в TCP-пакете.

Флаг URG — бит, установленное в 1 значение которого означает, что TCP-пакет содержит важные ***(urgent)*** данные.

Флаг ACK — бит, установленное в 1 значение которого означает, что TCP-пакет содержит в поле «номер подтверждения» верные данные.

Флаг PSH — бит, установленное в 1 значение которого означает, что данные содержащиеся в TCP-пакете должны быть немедленно переданы прикладной программе, для которой они адресованы. Подтверждение для TCP-пакета, содержащего единичное значение во флаге PSH, означает, что и все предыдущие TCP-пакеты достигли адресата.

Флаг RST — бит, устанавливаемый в 1 в TCP-пакете, отправляемом в ответ на получение неверного TCP-пакета. Также может означать запрос на переустановление логического соединения.

Флаг SYN — бит, установленное в 1 значение которого означает, что TCP-пакет представляет собой запрос на установление логического соединения. Получение пакета с установленым флагом SYN должно быть подтверждено принимающей стороной.

Флаг FIN — бит, установленное в 1 значение которого означает, что TCP-пакет представляет собой запрос на закрытие логического соединения и является признаком конца потока данных, передаваемых в этом направлении. Получение пакета с ус-

тановленым флагом FIN должно быть подтверждено принимающей стороной.

Размер окна — 16-битовое поле, содержащее количество байт информации, которое может принять в свои внутренние буфера TCP-модуль, отправляющий партнеру данный TCP-пакет. Данное поле используется принимающим поток данных TCP-модулем для управления интенсивностью этого потока: так, установив значение поля в 0, можно полностью остановить передачу данных, которая будет возобновлена только, когда размер окна примет достаточно большое значение. Максимальный размер окна зависит от реализации, в некоторых реализациях максимальный размер может устанавливаться системным администратором (типичное значение максимального размера окна — 4096 байт).

Контрольная сумма — 16-битовое поле, содержащее Internet-контрольную сумму, подсчитанную для TCP-заголовка, данных пакета и псевдозаголовка.

Указатель — 16-битовое поле, содержащее указатель (в виде смещения) на первый байт в теле TCP-пакета, начинающий последовательность важных (urgent) данных.

Дополнительные данные заголовка — последовательность полей произвольной длины, описывающих необязательные данные заголовка. Протокол TCP определяет только три типа дополнительных данных заголовка:

- конец списка полей дополнительных данных;
- пусто (No Operation);
- максимальный размер пакета.

Дополнительные данные последнего типа посылаются в TCP-заголовке в момент установления логического соединения для выражения готовности TCP-модулем принимать пакеты длиннее 536 байтов. В UNIX-реализациях длина пакета обычно определяется максимальной длиной IP-сегмента для сети.

Номер порта

Номера портов играют роль адресов транспортного уровня, идентифицируя на конкретных узлах сети, по сути дела, потребителей транспортных услуг, предоставляемых как протоколом TCP, так и протоколом UDP. При этом протоколы TCP и UDP

имеют свои собственные адресные пространства: например, порт номер 513 для TCP не идентичен порту номер 513 для UDP.

Своя собственная адресация на транспортном уровне стека протоколов сетевого взаимодействия необходима для обеспечения возможности функционирования на узле сети одновременно многих сетевых приложений. Наличие в TCP-заголовке номера порта позволяет TCP-модулю, получающему последовательности TCP-пакетов, формировать раздельные потоки данных к прикладным программам.

Взаимодействие прикладных программ, использующих транспортные услуги протокола TCP (или UDP), строится согласно модели ***клиент-сервер***, которая подразумевает, что одна программа (сервер) всегда пассивно ожидает обращения к ней другой программы (клиента). Связь программы-клиента и сервера идентифицируется пятеркой объектов:

- используемый транспортный протокол (TCP или UDP);
- IP-адрес сервера;
- номер порта сервера;
- IP-адрес клиента;
- номер порта клиента.

Для того, чтобы клиент мог обращаться к необходимому ему серверу, он должен знать номер порта, по которому сервер ожидает обращения к нему (слушает сеть). Для прикладных программ, получивших наибольшее распространение в сетях на основе TCP/IP, номера портов фиксированы и носят название ***хорошо известных номеров портов*** (***well-known port numbers***). Ниже приводятся примеры хорошо известных номеров портов для некоторых серверов (служб).

Служба	Номер порта	Протокол
ftp-data	20	TCP
ftp	21	TCP
telnet	23	TCP
smtp	25	TCP
time	37	TCP
time	37	UDP
finger	79	TCP
portmap	111	TCP

```
portmap      111          UDP
exec         512          TCP
login        513          TCP
shell        514          TCP
who          513          UDP
talk         517          UDP
route        520          UDP
Xserver     6000          TCP
```

Программы-клиенты, являющиеся активной стороной во взаимодействии клиент-сервер, могут использовать произвольные номера портов, назначаемые динамически непосредственно перед обращением к серверу (как любые свободные на данном узле).

Любая прикладная программа (будь то клиент или сервер) может открывать для взаимодействия любое количество портов для использования любых транспортных протоколов.

Принцип «скользящего окна»

Протоколы транспортного уровня, обеспечивающие надежную передачу данных, предполагают обязательное подтверждение принимающей стороной правильности полученных данных.

В простых протоколах сторона, отправляющая данные, отсылает пакет с данными принимающей стороне и переходит в состояние ожидания подтверждения получения правильных данных. Только после приема подтверждения становится возможной следующая посылка. Очевидно, что такой подход использует пропускную способность сети неэффективно.

В протоколе TCP используется более совершенный принцип ***скользящего окна (sliding window),*** который заключается в том, что каждая сторона может отправлять партнеру максимум столько байт, сколько партнер указал в поле «размер окна» заголовка TCP-пакета, подтверждающего получение предыдущих данных.

Принцип скользящего окна обеспечивает *опережающую* посылку данных с *отложенным* их подтверждением. Следует отметить недостаток этого механизма: если в течение некоторого времени не будет получено *отсроченное* подтверждение ранее отправленного пакета, то отправляющий TCP-модуль будет вынужден повторить посылку всех TCP-пакетов, начиная с неподтвер-

жденного. Размер окна, как правило, определяется объемом свободного места в буферах принимающего TCP-модуля.

Важные данные

Протокол TCP предусматривает возможность информирования принимающей стороны взаимодействия отправляющей стороной о наличии в TCP-пакете важных данных (***urgent data***), требующих особого внимания согласно логике прикладной задачи. Отличие важных данных от данных основного потока заключается в том, что принимающая сторона должна, как правило, обработать их прежде ранее полученных, но еще не обработанных данных потока.

Для индикации наличия в TCP-пакете важных данных используется *флаг URG* TCP-заголовка, местоположение важных данных в теле TCP-пакета определяется полем *Указатель* TCP-заголовка — оно задает смещение (в стиле языка программирования C) первого байта важных данных в теле TCP-пакета. Ниже на рисунке иллюстрируется расположение важных данных в теле TCP-пакета.

Рис. 6.8 — Поле указатель содержит адрес (смещение) важных данных

Протокол TCP предусматривает передачу важных (urgent) данных в рамках общего потока данных (***in-band***). Существуют протоколы (например, ISO), поддерживающие режим передачи важных (***expedited***) данных вне общего потока данных (***out-band***), что в общем случае быстрее.

Этапы TCP-взаимодействия

Взаимодействие партнеров с использованием протокола TCP строится в три этапа:

- установление логического соединения;
- обмен данными;
- закрытие соединения.

Этап установления соединения реализуется в виде *трехшагового рукопожатия* (*three-way handshake*). На первом шаге TCP-модуль A посылает TCP-модулю B пакет с установленным флагом SYN и некоторым начальным значением *номера_в_последовательности* равным, например, 1000. TCP-модуль B, будучи готов установить соединение, отвечает TCP-пакетом, подтверждающим правильный прием запроса и информирующим о готовности установить соединение. В нем поле *номер_подтверждения* на 1 больше начального *номера_в_последовательности* для TCP-модуля A (в нашем примере 1001), *флаг_ACK* и *флаг_SYN* установлены в 1, установлен в 5000 начальный *номер_в_последовательности*. На третьем шаге TCP-модуль A подтверждает правильность приема TCP-пакета от B.

Этап двустороннего обмена данными. TCP-модуль, принимающий адресованные ему данные, всегда подтверждает их прием, вычисляя значение поля *номер_подтверждения* в заголовке ответного TCP-пакета как сумму пришедшего *номера_в_последовательности* и длины правильно принятых данных. Отметим, что посылка данных к партнеру и подтверждение принятых от него данных реализуются в рамках одного TCP-пакета.

Закрытие соединения. Передающий TCP-модуль, посылает партнеру TCP-пакет с установленным *флагом_FIN*. Прием запроса на закрытие соединения принимающий TCP-модуль подтверждает пакетом, содержащем в своем заголовке поле *номер_подтверждения,* значение которого на 1 больше значения принятого *номера_в_последовательности*. После этого посылка каких-либо данных передающим TCP-модулем становится невозможной, однако принимающий модуль имеет данные для передачи, которые он отправляет исходному TCP-модулю и получает подтверждение на их прием. Затем TCP-модуль приемник формирует пакет с *флагом_FIN*, после подтверждения его приема соединение считается закрытым.

Таймеры

Таймер повторной передачи

Данный таймер взводится значением *RTO* (*Retransmission TimeOut* — интервал до повторной передачи) в момент посылки

TCP-пакета адресату. Если таймер окажется сброшенным в ноль до момента получения подтверждения пакета, то этот пакет должен быть послан вновь.

Ясно, что величина RTO не может быть фиксированной, т.к. TCP-пакеты до разных адресатов следуют по различным маршрутам через сети, скорость передачи данных в которых может различаться более чем в тысячи раз. Для вычисления оптимального значения RTO в каждом логическом соединении используется специальная процедура, специфицированная в RFC 793. Согласно этой процедуре, для каждого TCP-пакета измеряется величина ***RTT*** (***Round Trip Time*** — интервал времени от момента посылки TCP-пакета до момента получения подтверждения на него). На основе измеренных RTT вычисляется величина ***SRTT*** (***Smoothed RTT*** — сглаженный RTT) по следующей формуле:

```
SRTT = k*SRTT + (1 – k)*RTT,
```

где **k** — сглаживающий коэффициент (например, 0.9). Приведенная формула обеспечивает фильтрацию нетипичных (пиковых) значений измеренной величины RTT. Оптимальное значение ***RTO*** вычисляется по формуле:

```
RTO = min(U, max(L, p*SRTT)),
```

где **U** — ограничение сверху на значение RTO (например, 30 секунд); **L** — ограничение снизу на значение RTO (например, 1 секунда); **p** — коэффициент запаса (например, 2).

Если после повторной посылки TCP-пакета, опять не будет получено его подтверждение за интервал времени RTO, то попытки послать TCP-пакеты будут повторены (до 12 раз), но каждый раз с экспоненциально возрастающим значением RTO. Только после неудачи всей серии повторных посылок связь между партнерами будет считаться аварийно закрытой.

Таймер возобновления передачи

В ходе взаимодействия двух TCP-модулей (**A** и **B**) вполне возможна следующая ситуация:

- TCP-модуль B уведомляет TCP-модуль A о невозможности приема от него данных, определяя *размер_окна* равным 0;
- TCP-модуль A, имея данные для передачи, переходит в состояние ожидания от TCP-модуля B пакета с ненулевым *размером_окна*;

- TCP-модуль B, у которого освободилось некоторое пространство в буферах, посылает модулю A TCP-пакет с ненулевым *размером_окна*;
- адресованный модулю A пакет теряется по какой-либо причине и оба TCP-модуля переходят в состояние бесконечного ожидания.

Средством выхода из такого тупикового состояния и служит ***таймер возобновления передачи*** (***persistence timer*** — настойчивый таймер). Он взводится в момент получения TCP-пакета с нулевым значением поля *размер_окна* в его заголовке (типичное начальное значение для этого таймера — 5 секунд). Если до момента обнуления таймера не будет получено разрешение на возобновление передачи данных, то ожидающий разрешения TCP-модуль отправляет партнеру пакет, содержащий всего лишь 1 байт данных. По реакции партнера, возвращающего пакет с нулевым/ненулевым значением размера окна, TCP-модуль продолжает ожидание или возобновляет посылку данных.

Таймер закрытия связи

Протокол TCP предусматривает следующий простой прием предотвращения появления в сети TCP-пакетов, не имеющих адресатов: после закрытия логического соединения между партнерами номера портов, использовавшихся в этом соединении, остаются еще некоторый интервал времени действительными, что дает возможность долго блуждавшим по сети TCP-пакетам добраться до места назначения (где они будут просто проигнорированы). Величина этого интервала равна удвоенному времени жизни IP-сегмента (обычно, 2*15=30 секунд).

Таймеры поддержки соединения

Ниже описывается механизм, используемый для проверки ненарушенности логического соединения между TCP-модулями. Каждый TCP-модуль, участвующий в логическом соединении, через фиксированный промежуток времени (***keep-alive timer***), равный обычно 45 секундам, периодически отправляет партнеру пустые TCP-пакеты и ждет их подтверждения. Каждое полученное подтверждение говорит о ненарушенности соединения. Если же в течении определенного интервала времени (***idle timer***), рав-

ного обычно 360 секудам, не будет получено ни одного подтверждения, то логическое соединение считается оборванным.

Очевидно, что данный механизм имеет смысл включать в работу только тогда, когда партнеры по TCP-взаимодействию приостановили по какой-либо причине обмен данных на достаточно длительный срок (более 45 секунд).

Алгоритмы повышения эффективности

Задержка подтверждения

Задержка отсылки подтверждения принятого пакета используется для сокращения числа TCP-пакетов, которыми обмениваются партнеры по взаимодействию. Поясним эффект от такой задержки следующим примером.

Пусть клиентская часть некоторого приложения направляет серверной части некоторые данные. Серверная часть, получив данные и обработав их, должна вернуть клиенту результат. В ситуации без задержки TCP-модуль на стороне сервера, приняв пакет с данными и разместив их в своем буфере, сразу же отвечает подтверждающим пакетом, содержащим в своем заголовке и некоторый (уменьшенный) размер окна для приема последующих данных. Спустя некоторое (обычно, очень короткое) время данные из буфера передаются серверной части прикладной программы. Освобождение места в буфере заставляет TCP-модуль отправлять партнеру на стороне клиента TCP-пакет с новым (увеличившимся) размером окна. Тем временем прикладная программа, обработав полученные данные (часто за небольшое время), передает результат TCP-модулю для отсылки его клиенту, для чего модуль формирует еще один пакет. Итого: одна транзакция потребовала от TCP-модуля на стороне сервера посылки трех TCP-пакетов.

Введение же задержки при отсылке подтверждающего TCP-пакета позволяет в ряде случаев уменьшить количество пакетов с трех до одного, содержащего сразу подтверждение, новый размер окна и результирующие данные. Экспериментальные исследования показали, что во многих случаях оптимальным значением задержки является 0.2 секунды. Для того, чтобы введение задержки сказывалось минимальным образом на приложения, предъяв-

ляющие жесткие требования к пропускной способности сети, задержка устанавливается нулевой при условии, что размер окна изменяется более чем на 35% или (в абсолютном исчислении) на удвоенный максимальный размер TCP-пакета.

Исключение малых окон

Возможны ситуации, когда прикладная программа, использующая TCP-сервис, выбирает из буфера обслуживающего ее TCP-модуля пришедшие для нее данные малыми порциями. Это приводит к генерации TCP-модулем большого количества TCP-пакетов, содержащих в своих заголовках малую величину размера окна, что в свою очередь приводит к генерации на передающей данные стороне многих TCP-пакетов с короткими данными. Как результат — засорение сети короткими пакетами и снижение ее пропускной способности.

Во избежание деградации сети вследствие описанного явления используется следующий прием: TCP-пакет, информирующий посылающую данные сторону об увеличении размера окна, формируется только при выполнении одного из двух условий:

- свободное место в буфере принимающего данные TCP-модуля увеличилось по крайней мере на четверть размера этого буфера;
- свободное место увеличилось по крайней мере на максимальный размер TCP-пакета.

Кроме того, TCP-модуль, отправляющий данные, должен делать это большими порциями.

Исключение коротких TCP-пакетов

Засорение сети короткими TCP-пакетами возможно и в ситуации, когда прикладная программа, отправляющая данные партнеру по взаимодействию, делает это короткими порциями.

Для борьбы с этим используется следующий прием:

- самая первая порция данных отправляется TCP-модулем сразу же при поступлении коротким TCP-пакетом;
- все последующие накапливаются в буфере TCP-модуля, пока их общий объем не составит максимального размера TCP-пакета или не будет получено подтверждение предыдущей посылки.

Однако этот подход может сказаться на быстродействии некоторых приложений, чтобы избежать этого прикладной программе предоставляются средства для принудительного выталкивания буферизованных данных в необходимых случаях. Кроме того, существует возможность отключения описанного механизма.

Алгоритм медленного старта.

Опыт эксплуатации сетей на основе TCP/IP показал, что с повышением загрузки сети (особенно, сети со шлюзом) ее пропускная способность падает (хотя, казалось бы, она должна оставаться постоянной). Исследования показали, что падение обусловлено появлением в сети большого числа TCP-пакетов, повторно посылаемых к активно используемому узлу сети (обычно это шлюз в другие сети). Дело в том, что приемный буфер TCP-модуля на шлюзе очень быстро заполняется, и TCP-модуль вынужден сбрасывать поступающие к нему пакеты.

Для предупреждения подобной ситуации необходимо согласование темпа передачи TCP-пакетов с возможностями их приема на узле-адресате. Задачу согласования решает алгоритм медленного старта, постепенно повышающий темп передачи данных от медленного до оптимального, при котором нет повторных передач TCP-пакетов. Алгоритм использует так называемое ***окно перегруженности*** (***congestion window***), используемое на передающей стороне для определения максимального объема передаваемых данных вместо размера, получаемого от принимающей стороны в поле окна подтверждающего пакета.

Размер окна перегруженности определяется на передающей стороне путем постепенного его увеличения до момента появления повторных передач (ясно, что размер этого окна никогда не превышает размера окна на принимающей стороне). Однажды определенный размер окна перегруженности остается неизменным, пока вновь не появятся повторные передачи, однако периодически делаются осторожные попытки и увеличить этот размер.

Эксперименты показали, что данный алгоритм позволяет уменьшить количество повторно передаваемых TCP-пакетов на 50 % и повысить пропускную способность сети на 30 %.

Протокол дейтаграмм пользователя UDP

Протокол дейтаграмм пользователя ***UDP*** (***User Datagram Protocol***) является протоколом транспортного уровня и базируется на возможностях, предоставляемых межсетевым протоколом IP. Основная задача UDP — обеспечение быстрой передачи данных в сети. Его транспортный адрес в заголовке IP-сегмента равен 17. Описание протокола UDP дано в RFC 768.

Его основные характеристики перечислены ниже:

- реализует взаимодействие в режиме без установлениz логического (виртуального) соединения;
- организует поблочный (дейтаграммный, пакетный) тип передачи данных;
- для идентификации партнеров по взаимодействию на транспортном уровне использует 16-битовые *номера_портов*;
- не гарантирует надежной передачи данных (возможна как потеря UDP-пакетов, так и их дублирование);
- не имеет средств уведомления источника UDP-пакета о правильности или ошибочности приема пакетов адресатом;
- не обеспечивает правильный порядок доставки UDP-пакетов от источника к приемнику;
- может гарантировать целостность данных в UDP-пакете за счет использования контрольной суммы;
- очень прост (особенно, по сравнению с протоколом TCP).

Следует отметить, что, по сути дела, протокол транспортного уровня UDP играет роль интерфейса для прикладных программ к средствам протокола межсетевого уровня IP. Ниже приведен формат заголовка UDP-пакета.

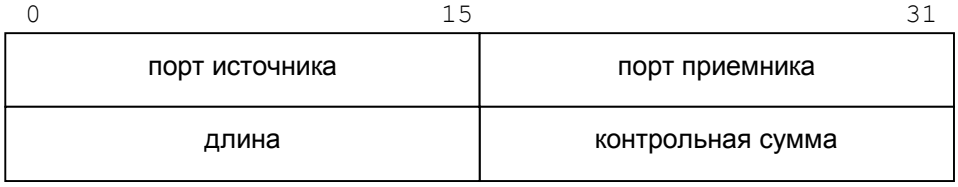


Рис. 6.9 — Формат UDP-пакета (дейтаграммы)

Порт источника и порт приемника — 16-битовые поля, содержащие номера портов, соответственно, источника и адресата UDP-пакета.

Длина — 16-битовое поле, содержащее длину (в байтах) всего UDP-пакета, включая заголовок и данные.

Контрольная сумма — 16-битовое поле, содержащее Internet-контрольную сумму, подсчитанную для UDP-заголовка, данных пакета и псевдозаголовка. Если поле *контрольная_сумма* UDP-заголовка содержит нулевое значение, это означает, что источник UDP-пакета контрольную сумму не подсчитывал, и приемник выполнять ее проверку не должен. Некоторые реализации протокола UDP (например, в SunOS — клоне ОС UNIX от Sun Microsystems) контрольную сумму не подсчитывают в принципе, полагаясь на возможности контроля целостности данных, реализованные в протоколах сетевого уровня (например, в Ethernet).

Модель Клиент-Сервер

Большинство сетевых приложений написано таким образом, что с одной стороны присутствует клиент, а с другой — сервер. При этом сервер предоставляет определенные сервисы клиентам.

Можно подразделить серверы на два класса: последовательные (***iterative***) и конкурентные (***concurrent***).

В процессе выполнения шага П2 работы последовательного сервера (см. рис. 6.10, *а*) часто возникает проблема, заключающаяся в том, что в это время никакие другие клиенты не могут быть обслужены.

Конкурентный сервер позволяет избежать ее (рис. 6.10, *б*). Запуск нового сервера на шаге К2 для обработки запроса клиента может выглядеть как создание нового процесса, задачи, в зависимости от того какая операционная система лежит в основе этого сервера. Новый сервер обрабатывает поступивший запрос клиента целиком. По завершении сервер уничтожается.

Преимущество конкурентного сервера заключается в том, что он просто запускает другие сервера для обработки запросов от клиентов. В подобном случае каждый клиент имеет собственный сервер. Предполагается, что операционная система поддерживает многозадачность и обслуживание нескольких клиентов одновременно.

В общем случае, серверы TCP — конкурентные, а серверы UDP — последовательные. Однако из этого правила могут быть исключения.

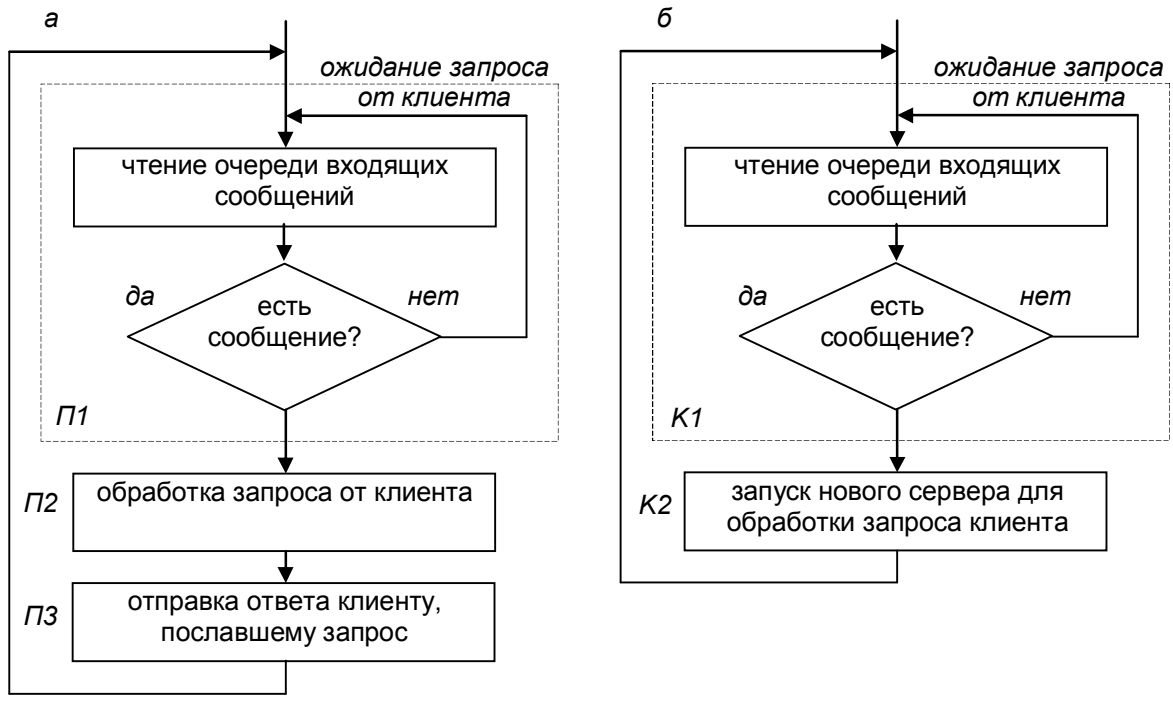


Рис. 6.10 — Алгоритм работы последовательного и конкурентного сервера

6.4 Адресация Internet

Протокол IP в процессе своей работы передает пакет от источника к пункту назначения, для этого в заголовке каждой IP-дейтаграммы размещаются уникальные IP-адреса: источника и назначения. Эти адреса представляют собой 32-разрядные номера, уникально идентифицирующие узел (компьютер или сетевое устройство — принтер или маршрутизатор). Необходимо отметить, что сетевые устройства с несколькими интерфейсами (*multihomed*) имеют несколько IP адресов: по одному на каждый интерфейс.

IP-адреса обычно представлены в виде 4-х байт (***октетов***), разделенных точками, например 192.168.123.132. Существует определенная структура адреса Internet. На рис. 6.6 показано 5 классов адресов Internet.

Чтобы глобальная сеть TCP/IP работала эффективно как совокупность сетей, маршрутизаторы, обеспечивающие обмен пакетами данных между сетями, не знают точного расположения узла, для которого предназначен пакет. Маршрутизаторы знают только, к какой сети принадлежит узел, и используют сведения,

хранящиеся в таблицах маршрутизации, чтобы доставить пакет в сеть узла назначения. Как только пакет доставлен в необходимую сеть, он доставляется в соответствующий узел. Для осуществления этого процесса IP-адрес состоит из двух частей. Первая часть IP-адреса обозначает адрес сети, последняя часть — адрес узла. IP-адрес 192.168.123.132, например, разделяется на следующие две части: 192.168.123.0 — адрес сети. 0.0.0.132 — адрес узла.

Определить класс адреса, или класс сети, можно по первому числу в адресе, на рис. 6.11 показаны различные классы, причем первый октет выделен.

Так как каждый интерфейс, подключенный к сети, должен иметь уникальный адрес, встает вопрос распределения IP адресов в глобальной сети Internet. Этим занимается сетевой информационный центр (*Internet Network Information Center* или ***InterNIC***) http://www.internic.net. InterNIC назначает только сетевые идентификаторы, назначением идентификаторов хостов в сети занимаются системные администраторы.

Класс A	**0**.0.0.0 — **127**.255.255.255		
0	идентификатор сети	идентификатор хоста	
	7 бит	24 бита	
Класс B	**128**.0.0.0 — **191**.255.255.255		
1	0	идентификатор сети	идентификатор хоста
		14 бит	16 бит
Класс C	**192**.0.0.0 — **223**.255.255.255		
1	1	0	идентификатор сети	идентификатор хоста
			21 бит	8 бит
Класс D	**224**.0.0.0 — **239**.255.255.255		
1	1	1	0	идентификатор группы (multicast)
				28 бит
Класс E	**240**.0.0.0 — **247**.255.255.255		
1	1	1	1	0	зарезервировано для дальнейшего использования
					27 бит

Рис. 6.11 — Классы IP-адресов и их структура

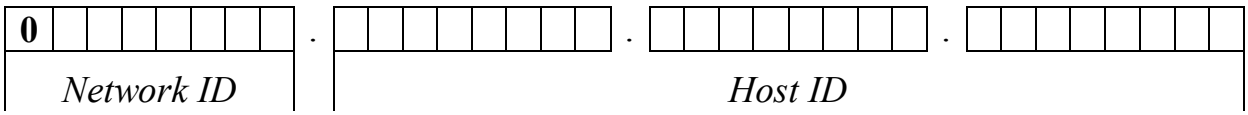
Существует три типа IP адресов: ***персональный адрес*** (***unicast***) — указывает на один хост, ***широковещательный адрес*** (***broadcast***) — указывает на все хосты в указанной сети, и ***группо-***

вой адрес (***multicast***) — указывает на группу хостов, принадлежащей к группе адресации.

Классы сетей

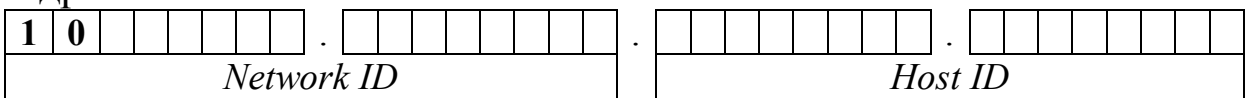
Интернет-адреса распределяются организацией InterNIC по классам (см. рис. 6.11). Наиболее распространены классы A, B и C. Классы D и E существуют, но обычно не используются конечными пользователями. Каждый из классов адресов имеет свою маску подсети по умолчанию. Определить класс IP-адреса можно по его первому октету:

- Сети ***класса A*** имеют в первом октете значение от 0 до 127, что позволяет адресовать $2^7 - 1 = 127$ сетей и 16 777 216 узлов:



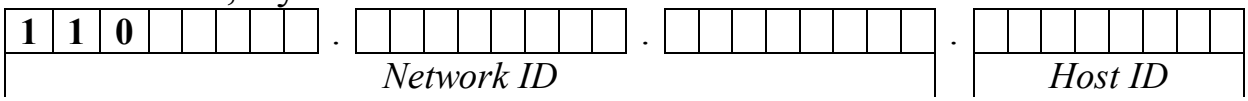
*IP-адреса:*0.0.0.1 — 126.255.255.254
Subnet Mask: 255.0.0.0
*Broadcast IP:*0-126.x.x.255

- Сети ***класса B*** имеют в первом октете значение от 128 до 191. Что определяет 14 бит для адреса сети и позволяет задавать адреса $2^{14} = 16\,384$ сетей и 65 535 хостов:



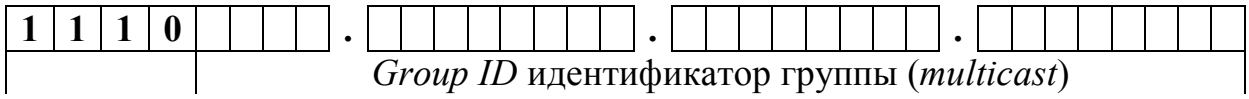
*IP-адреса:*128.0.0.0 — 191.255.255.254
Subnet Mask: 255.255.0.0
*Broadcast IP:*128-191.x.255.255

- Сети ***класса C*** имеют в первом октете значение от 192 до 223. В адресе сети класса C 21 бит отводится под адрес сети и 8 бит — под адрес хоста. Сетей этого класса может быть $2^{21} = 2\,097\,152$, а узлов — всего 254:



*IP-адреса:*192.0.0.0 — 223.255.255.254
Subnet Mask: 255.255.255.0
*Broadcast IP:*191-223.255.255.255

- Сети ***класса D*** (224.0.0.0 — 239.255.255.255) планировалось использовать для групповой адресации, они практически не используются, и имеют специфический формат:



- Сети ***класса E*** (240.0.0.0 — 247.255.255.255) зарезервированы и нигде не используются.

Маска подсети

Следующий элемент, необходимый для работы протокола TCP/IP, — это ***маска подсети*** (***subnet mask***). Протокол TCP/IP использует маску подсети, чтобы определить, в какой сети находится узел: в локальной подсети или удаленной сети.

В протоколе TCP/IP части IP-адреса, используемые в качестве адреса сети и узла, не зафиксированы, следовательно, указанные выше адреса сети и узла невозможно определить без наличия дополнительных сведений. IP-адрес 192.168.123.132, например, разделяется на следующие две части: 192.168.123.0 — адрес сети. 0.0.0.132 — адрес узла при помощи маски подсети 255.255.255.0. Чтобы понять использование масок подсетей для распознавания узлов, сетей и подсетей достаточно представить маску в двоичном обозначении:
11111111.11111111.11111111.00000000.

При помощи маски подсети, можно выделить в IP-адресе составляющие сети и узла. Первые 24 разряда (число единиц в маске подсети) распознаются как адрес сети, а последние 8 разрядов (число оставшихся нолей в маске подсети) — адрес узла:
11000000.10101000.01111011.10000100 — IP-адрес (192.168.123.132) 11111111.11111111.11111111.00000000 — маска подсети (255.255.255.0) 11000000.10101000.01111011.00000000 — адрес сети (192.168.123.0)
00000000.00000000.00000000.10000100 — адрес узла (000.000.000.132)

Когда пакет с конечным адресом 192.168.123.132 доставляется в сеть 192.168.123.0 (из локальной подсети или удаленной сети), компьютер получит его из сети и обработает.

В некоторых случаях значения маски подсети того или иного класса по умолчанию не соответствует потребностям организации из-за физической топологии сети или потому, что количество сетей (или узлов) не соответствует ограничениям маски под-

сети. Этого неудобства можно избежать, разделяя сети на подсети (***subnet***) с помощью масок подсети.

Для упрощения записи маски подсети применяется ***префикс сети***, показывающий, сколько бит в *IP-адресе* отводится под *Network ID*, следующие записи эквивалентны:
192.168.123.132 (255.255.255.0) и 192.168.123.132 /24.

Подсети

TCP/IP-сеть класса A, B или C может еще быть разбита на подсети системным администратором. Для этого можно «занять» несколько разрядов, из адреса узла и использовать их для адреации подсети в адресе.

Например, использование маски подсети 255.255.255.192 (то же самое, что и 1111111.11111111.1111111.**11**000000) преобразует сеть 192.168.123.0 в четыре сети:

1) сеть 192.168.123.0, с узлами из диапазона 192.168.123.1 — 62;
2) сеть 192.168.123.64, с узлами 192.168.123.65 — 126;
3) сеть 192.168.123.128, с узлами 192.168.123.129 — 190 и
4) сеть 192.168.123.192, с узлами 192.168.123.193 — 254.

Первые две цифры последнего октета становятся адресами сети, поэтому появляются дополнительные сети, в которых последние 6 двоичных цифр можно использовать в качестве адресов узлов. Маска подсети 255.255.255.192 позволяет создать четыре сети с 62-мя узлами в каждой.

Двоичные адреса узлов с одними только единицами и нолями недействительны, поэтому нельзя использовать адреса со следующими числами в последнем октете: 0, 63, 64, 127, 128, 191, 192, или 255.

Основные шлюзы

Связь между узлами из различных TCP/IP сетей осуществляется через маршрутизатор. С точки зрения TCP/IP маршрутизатор, указанный на узле, связывающем подсеть узла с другими сетями, называется ***основным шлюзом.*** Протокол TCP/IP определяет, отправлять или нет пакеты данных на основной шлюз, чтобы связаться с другим хостом в сети.

При попытке установления связи между узлом и другим устройством с помощью протокола TCP/IP узел сопоставляет

маску подсети и IP-адрес назначения (из передаваемого пакета) с маской подсети и своим собственным IP-адресом. В результате этого сопоставления компьютер узнает, для локального или для удаленного узла предназначен данный пакет.

Если в результате этого процесса назначением является локальный узел, то компьютер просто отправляет пакет в локальную подсеть. Если в результате сопоставления выясняется, что назначением является удаленный узел, компьютер направляет пакет на основной шлюз, определенный в свойствах TCP/IP. Таким образом, именно маршрутизатор отвечает за отправку пакета в правильную подсеть.

IP маршрутизация

IP маршрутизация это довольно простой процесс, особенно с точки зрения хоста. Если пункт назначения напрямую подключен к хосту (например канал точка-точка) или хост включен между несколькими сетями (Ethernet или Token ring), IP дейтаграмма направляется непосредственно в пункт назначения, иначе хост посылает дейтаграмму на маршрутизатор по умолчанию, тем самым предоставляя маршрутизатору решать как доставить дейтаграмму в пункт назначения. Эту простую схему реализуют практически все хосты.

Большинство многопользовательских систем в настоящее время могут быть сконфигурированы таким образом, чтобы выступать в роли маршрутизатора (например, рабочая станция с несколькими сетевыми адаптерами — с несколькими сетевыми интерфейсами). Существует возможность указать простой алгоритм маршрутизации, который будет использоваться как хостом, так и маршрутизатором. Основная и фундаментальная разница между хостом и маршрутизатором заключается в том, что хост никогда не перенаправляет дейтаграммы с одного своего интерфейса на другой, тогда как маршрутизатор перенаправляет.

В соответствии с общей схемой, IP может получать дейтаграммы от собственных уровней TCP, UDP, ICMP и IGMP (это дейтаграммы, формирующиеся здесь же), которые необходимо отправить, однако дейтаграммы могут быть приняты с какого-либо сетевого интерфейса (эти дейтаграммы должны быть перенаправлены). IP уровень имеет в памяти таблицу маршрутизации,

которую он просматривает каждый раз при получении дейтаграммы, которую необходимо перенаправить. Когда дейтаграмма принята с сетевого интерфейса, IP, во-первых, проверяет, не принадлежит ли ему указанный IP адрес назначения или не является ли этот IP адрес широковещательным. Если это так, то дейтаграмма доставляется в модуль протокола, указанный в поле протокола в IP заголовке. Если дейтаграмма не предназначается для этого IP уровня, если IP уровень был сконфигурирован для того чтобы работать как маршрутизатор, пакет перенаправляется (в этом случае дейтаграмма обрабатывается как исходящая, что будет описано ниже), иначе дейтаграмма уничтожается.

Каждый пункт таблицы маршрутизации содержит следующую информацию:

- IP адрес назначения. Это может быть как полный адрес хоста (host address) или адрес сети (network address), что указывается в поле флагов. Адрес хоста имеет ненулевое значение идентификатора хоста и указывает на один конкретный хост, тогда как адрес сети имеет идентификатор хоста, установленный в 0, и указывает на все хосты, включенные в определенную сеть (Ethernet, Token ring).
- IP адрес маршрутизатора следующей пересылки (next-hop router), или, иначе говоря, IP адрес непосредственно подключенной сети. Маршрутизатор следующей пересылки принадлежит одной из непосредственно подключенных сетей, в которую мы можем отправить дейтаграммы для их доставки. Маршрутизатор следующей пересылки это не конечный пункт назначения, однако он принимает дейтаграммы, которые мы посылаем, и перенаправляет их в направлении конечного пункта.
- Флаги. Один флаг указывает, является ли IP адрес пункта назначения, адресом сети или адресом хоста. Другой флаг указывает на то, является ли маршрутизатор следующей пересылки действительно маршрутизатором или это непосредственно подключенный интерфейс.
- Указание на то, на какой сетевой интерфейс должны быть переданы дейтаграммы для передачи.

IP маршрутизация осуществляется по принципу пересылка-за-пересылкой. Как видно из таблицы маршрутизации, IP не знает полный маршрут к пункту назначения (за исключением тех пунк-

тов назначения, которые непосредственно подключены к посылающему хосту). Все что может предоставить IP маршрутизация — это IP адрес маршрутизатора следующей пересылки, на который посылается дейтаграмма. При этом делается предположение, что маршрутизатор следующей пересылки ближе к пункту назначения, чем посылающий хост. Также делается предположение, что маршрутизатор следующей пересылки напрямую подключен к посылающему хосту.

IP маршрутизация осуществляет следующие действия:

- Осуществляется поиск в таблице маршрутизации, при этом ищется пункт, который совпадет с полным адресом пункта назначения (должен совпасть идентификатор сети и идентификатор хоста). Если пункт найден в таблице маршрутизации, пакет посылается на указанный маршрутизатор следующей пересылки или на непосредственно подключенный интерфейс (в зависимости от поля флагов). Как правило, так определяются каналы точка-точка, при этом другой конец такого канала, как правило, является полным IP адресом удаленного хоста.
- Осуществляется поиск в таблице маршрутизации пункта, который совпадет, как минимум, с идентификатором сети назначения. Если пункт найден, пакет посылается на указанный маршрутизатор следующей пересылки или на непосредственно подключенный интерфейс (в зависимости от поля флагов). Маршрутизация ко всем хостам, находящимся в сети назначения, осуществляется с использованием этого единственного пункта таблицы маршрутизации. Например, все хосты локальной сети Ethernet представляются в таблицах маршрутизации именно таким образом. Эта проверка совпадения идентификатора сети осуществляется с использованием возможной маски подсети, которую мы опишем в следующем разделе.
- В таблице маршрутизации ищется пункт, помеченный «по умолчанию» (default). Если пункт найден, пакет отсылается на указанный маршрутизатор по умолчанию.

Если ни один из шагов не дал положительного результата, дейтаграмма считается недоставленной. Если недоставленная дейтаграмма была сгенерирована данным хостом, то обычно возвращается ошибка «хост недоступен» (host unreachable) или «сеть

недоступна» (network unreachable). Этот код ошибки возвращается приложению, которое сгенерировало дейтаграмму.

В начале всегда осуществляется сравнение на совпадение полного адреса хоста, после чего осуществляется сравнение идентификатора сети. Только в том случае, если результат обеих сравнений отрицательный, используется маршрут по умолчанию. Маршруты по умолчанию и сообщения ICMP о перенаправлении, отправляемые на маршрутизатор следующей пересылки (если для дейтаграммы выбрано неверное направление по умолчанию), являются довольно мощными характеристиками IP маршрутизации.

Еще одна фундаментальная характеристика IP маршрутизации заключается в возможности указать маршрут к сети, вместо того, чтобы указывать маршрут к каждому отдельно взятому хосту. Именно поэтому хосты включенные в Internet, например, имеют в своих таблицах маршрутизации тысячи пунктов, вместо того чтобы содержать в них не более чем миллионы пунктов.

Косвенная маршрутизация

На рис. 6.11 представлен фрагмент сети Internet. В данном случае сеть Internet состоит из трех сетей Ethernet, на базе которых работают три IP-сети, объединенные шлюзом D. Каждая IP-сеть включает четыре машины; каждая машина имеет свои собственные IP- и Ethernet адреса.

Шлюз D соединяет все три сети и, следовательно, имеет три IP-адреса и три Ethernet-адреса. Машина D вместо двух модулей ARP и двух драйверов содержит три модуля ARP и три драйвера Ethernet. Обратим внимание на то, что машина D имеет только один модуль IP. Системным администратором присвоен каждой сети уникальный номер, называемый IP-номером сети (Network ID).

Когда машина A посылает IP-пакет машине B, то процесс передачи идет в пределах одной сети. При всех взаимодействиях между машинами, подключенными к одной IP-сети, используется прямая маршрутизация, базирующаяся на MAC-адресах. Когда машина D взаимодействует с машиной A, то это прямое взаимодействие. Когда машина D взаимодействует с машиной E, то это прямое взаимодействие. Когда машина D взаимодействует с машиной H, то это прямое взаимодействие. Это так, поскольку каждая пара этих машин принадлежит одной IP-сети. Маршрутиза-

ция IP-пакетов выполняется модулями IP и является прозрачной для модулей TCP, UDP и прикладных процессов.

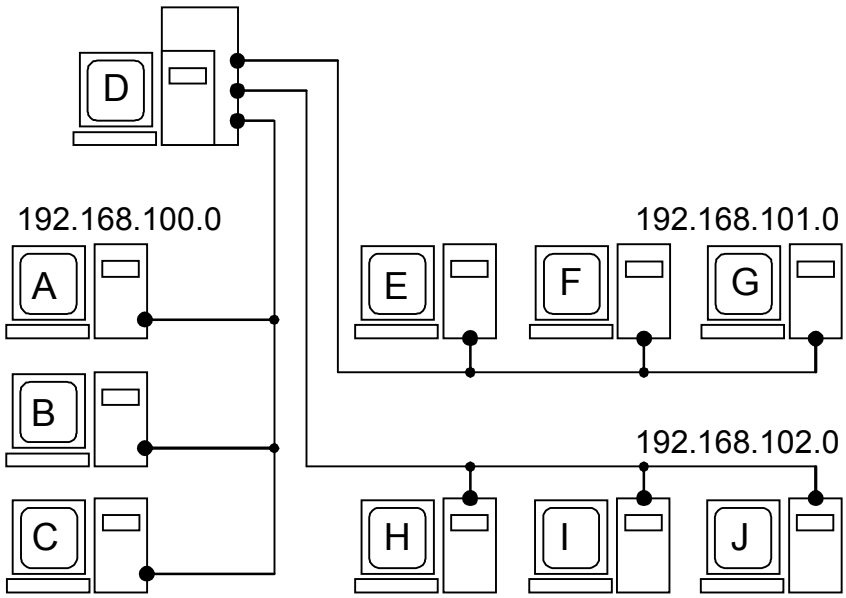


Рис. 6.11 — Сеть Internet, состоящая из трех IP-сетей

Однако, когда машина A взаимодействует с машинами, включенными в другую IP-сеть, то взаимодействие уже не будет прямым. Машина A должна использовать шлюз D для ретрансляции IP-пакетов в другую IP-сеть. Такое взаимодействие называется ***косвенным***.

Если машина A посылает машине E IP-пакет, то IP-адрес и Ethernet-адрес отправителя соответствуют адресам A. IP-адрес места назначения является адресом E, но поскольку модуль IP в A посылает IP-пакет через D, Ethernet-адрес места назначения является адресом D:

адрес	отправитель	получатель
IP-заголовок	A	E
Ethernet-заголовок	A	D

Модуль IP в машине D получает IP-пакет и проверяет IP-адрес места назначения. Определив, что это не его IP-адрес, шлюз D посылает этот IP-пакет прямо к E:

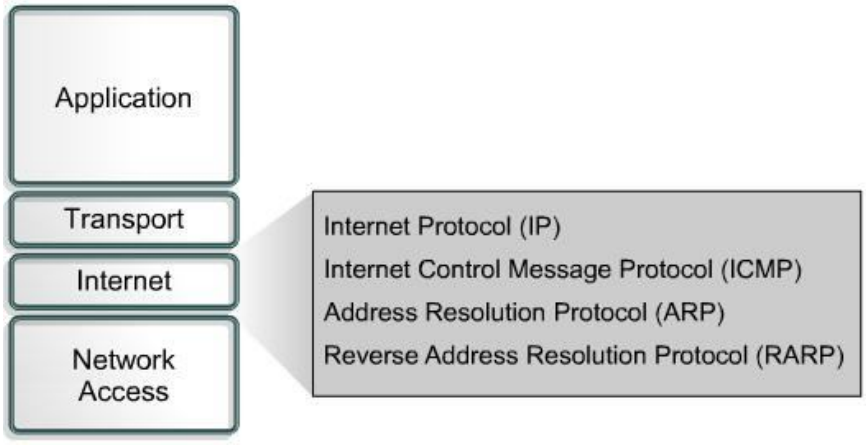
адрес	отправитель	получатель
IP-заголовок	A	E
Ethernet-заголовок	D	E

Итак, при прямой маршрутизации IP- и Ethernet-адреса отправителя соответствуют адресам того узла, который послал IP-пакет, а IP- и Ethernet-адреса места назначения соответствуют адресам получателя. При косвенной маршрутизации IP- и Ethernet-адреса не образуют таких пар.

В данном примере (рис. 6.11) сеть Internet является очень простой. Реальные сети могут быть гораздо сложнее, так как могут содержать несколько шлюзов и несколько типов физических сред передачи. В приведенном примере несколько сетей Ethernet объединяются шлюзом для того, чтобы локализовать широковещательный трафик в каждой сети.

6.5 Протоколы уровня IP

К протоколам Internet уровня относятся протоколы IP, ICMP, ARP и RARP. Ниже детально разобраны структура пакетов и функции данных протоколов.



Формат заголовка IP-дейтаграммы

Заголовок дейтаграммы состоит из 32-разрядных слов и имеет переменную длину, зависящую от размера поля *Options*, но всегда кратную 32 битам. За заголовком непосредственно следуют данные, передаваемые в дейтаграмме. Формат заголовка:

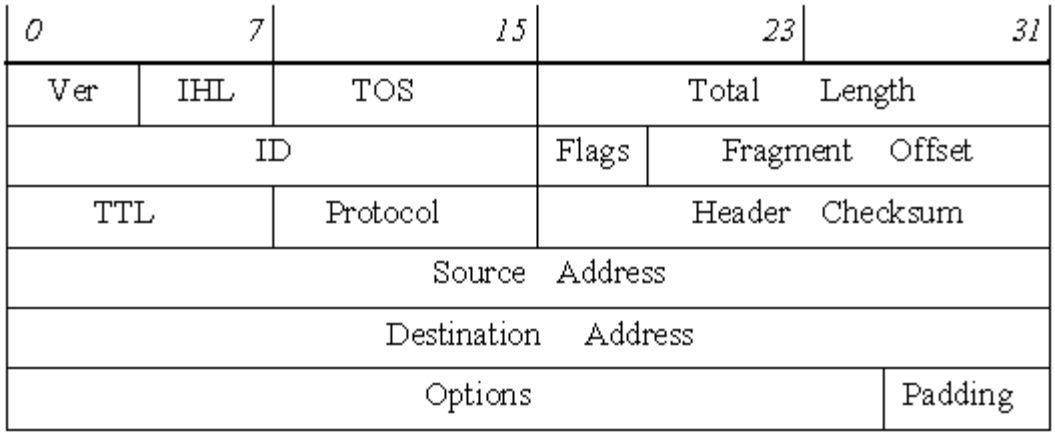


Рис. 6.12 — Формат заголовка IP-дейтаграммы

Значения полей заголовка следующие:

Ver (4 бита) — версия протокола IP, в настоящий момент используется версия 4, новые разработки имеют номера версий 6-8.

IHL (***Internet Header Length***) (4 бита) — длина заголовка в 32-битных словах; диапазон допустимых значений от 5 (минимальная длина заголовка, поле *Options* отсутствует) до 15 (т.е. может быть максимум 40 байт опций).

TOS (***Type Of Service***) (8 бит) — значение поля определяет приоритет дейтаграммы и желаемый тип маршрутизации.

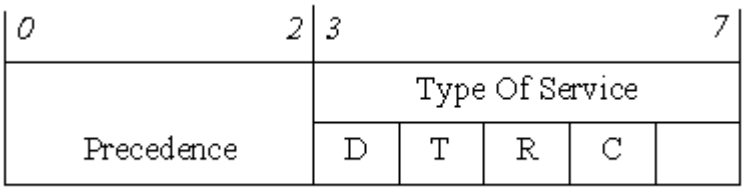


Рис. 6.13 — Структура байта TOS

Три младших бита (*Precedence*) определяют приоритет дейтаграммы:

- 111 — управление сетью;
- 110 — межсетевое управление;
- 101 — CRITIC-ECP;
- 100 — более чем мгновенно;
- 011 — мгновенно;
- 010 — немедленно;
- 001 — срочно;

- 000 — обычно.

Биты D,T,R,C определяют желаемый тип маршрутизации:

- ***D*** (***Delay***) — выбор маршрута с минимальной задержкой,
- ***T*** (***Throughput***) — выбор маршрута с максимальной пропускной способностью,
- ***R*** (***Reliability***) — выбор маршрута с максимальной надежностью,
- ***C*** (***Cost***) — выбор маршрута с минимальной стоимостью.

В дейтаграмме может быть установлен только один из битов D,T,R,C. Старший бит байта не используется.

Реальный учет приоритетов и выбора маршрута в соответствии со значением байта TOS зависит от маршрутизатора, его программного обеспечения и настроек. Маршрутизатор может поддерживать расчет маршрутов для всех типов TOS, для части или игнорировать TOS вообще. Маршрутизатор может учитывать значение приоритета при обработке всех дейтаграмм или при обработке дейтаграмм, исходящих только из некоторого ограниченного множества узлов сети, или вовсе игнорировать приоритет.

Total Length (16 бит) — длина всей дейтаграммы в октетах, включая заголовок и данные, максимальное значение 65535, минимальное — 21 (заголовок без опций и один октет в поле данных).

ID (***Identification***) (16 бит), ***Flags*** (3 бита), ***Fragment Offset*** (13 бит) используются для фрагментации и сборки дейтаграмм.

TTL (***Time To Live***) (8 бит) — «время жизни» дейтаграммы. Устанавливается отправителем, измеряется в секундах. Каждый маршрутизатор, через который проходит дейтаграмма, переписывает значение TTL, предварительно вычтя из него время, потраченное на обработку дейтаграммы. Так как в настоящее время скорость обработки данных на маршрутизаторах велика, на одну дейтаграмму тратится обычно меньше секунды, поэтому фактически каждый маршрутизатор вычитает из TTL единицу. При достижении значения TTL=0 дейтаграмма уничтожается, при этом отправителю может быть послано соответствующее ICMP-сообщение. Контроль TTL предотвращает зацикливание дейтаграммы в сети.

Protocol (8 бит) — определяет программу (вышестоящий протокол стека), которой должны быть переданы данные дейта-

граммы для дальнейшей обработки. Коды некоторых протоколов приведены в таблице 6.3.

Таблица 6.3 — **Коды IP-протоколов**

Код	Протокол	Описание
1	ICMP	Протокол контрольных сообщений
2	IGMP	Протокол управления группой хостов
3	IP	IP поверх IP (инкапсуляция)
4	TCP	TCP
5	EGP	Протокол внешней маршрутизации (устарел)
6	IGP	Протокол внутренней маршрутизации (устарел)
7	UDP	UDP
8	RSVP	Протокол резервирования ресурсов при мультикастинге
9	IGRP	Протокол внутренней маршрутизации от фирмы cisco
10	OSPF	Протокол внутренней маршрутизации

Header Checksum (16 бит) — контрольная сумма заголовка, представляет из себя 16 бит, дополняющие биты в сумме всех 16-битовых слов заголовка. Перед вычислением контрольной суммы значение поля *Header Checksum* обнуляется. Поскольку маршрутизаторы изменяют значения некоторых полей заголовка при обработке дейтаграммы (как минимум, поля *TTL*), контрольная сумма каждым маршрутизатором пересчитывается заново. Если при проверке контрольной суммы обнаруживается ошибка, дейтаграмма уничтожается.

Source Address (32 бита) — IP-адрес отправителя.

Destination Address (32 бита) — IP-адрес получателя.

Options — опции, поле переменной длины. Опций может быть одна, несколько или ни одной. Опции определяют дополнительные услуги модуля IP по обработке дейтаграммы, в заголовок которой они включены.

Padding — выравнивание заголовка по границе 32-битного слова, если список опций занимает нецелое число 32-битных слов. Поле «Padding» заполняется нулями

Фрагментация дейтаграмм

Различные среды передачи имеют различный максимальный размер передаваемого блока данных (***MTU — Media Transmission Unit***), это число зависит от скоростных характеристик среды и вероятности возникновения ошибки при передаче. Например, размер MTU в 10Мбит/с Ethernet равен 1536 октетам, в 100 Мбит/с FDDI — 4096 октетам.

При передаче дейтаграммы из среды с большим MTU в среду с меньшим MTU может возникнуть необходимость во фрагментации дейтаграммы. Фрагментация и сборка дейтаграмм осуществляются модулем протокола IP. Для этого применяются поля ***ID*** (***Identification***), ***Flags*** и ***Fragment Offset*** заголовка дейтаграммы.

Flags — поле состоит из 3 бит, младший из которых всегда сброшен:

0	DF	MF

Значения бита ***DF*** (***Don't Fragment***):

0 — фрагментация разрешена,

1 — фрагментация запрещена (если дейтаграмму нельзя передать без фрагментации, она уничтожается).

Значения бита ***MF*** (***More Fragments***):

0 — данный фрагмент последний (единственный),

1 — данный фрагмент не последний.

ID (***Identification***) — идентификатор дейтаграммы, устанавливается отправителем; используется для сборки дейтаграммы из фрагментов для определения принадлежности фрагментов одной дейтаграмме.

Fragment Offset — смещение фрагмента, значение поля указывает, на какой позиции в поле данных исходной дейтаграммы находится данный фрагмент. Смещение считается 64-битовыми порциями, т.е. минимальный размер фрагмента равен 8 октетам, а следующий фрагмент в этом случае будет иметь смещение 1. Первый фрагмент имеет смещение нуль.

Опции IP

Опции определяют дополнительные услуги протокола IP по обработке дейтаграмм. Опция состоит, как минимум, из октета «Тип опции», за которым могут следовать октет «Длина опции» и октеты с данными для опции.

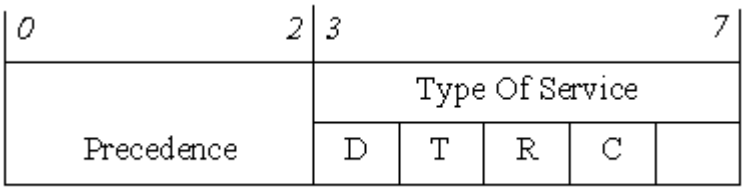


Рис. 6.14 — Структура октета *Тип опции*

Значения бита C:

1 — опция копируется во все фрагменты;

0 — опция копируется только в первый фрагмент.

Определены два класса опций: 0 — *Управление* и 2 — *Измерение и отладка*. Внутри класса опция идентифицируется номером. Ниже приведены опции, описанные в стандарте протокола IP.

Таблица 6.4 — Опции IP

Класс	**Номер**	**Октет длины**	**Опция**
0	0	–	Конец списка опций
0	1	–	Нет операции
0	2	+(11)	Безопасность
0	3	+	Loose Source Routing (свободное исполнение маршрута отправителя)
0	9	+	Strict Source Routing (строгое исполнение маршрута отправителя)
0	7	+	Запись маршрута
0	8	+(4)	Stream ID
2	4	+	Internet Timestamp (временной штамп)

При обнаружении в списке опции *Конец списка опций* разбор опций прекращается, даже если длина заголовка (IHL) еще не исчерпана. Опция *Нет операции* обычно используется для выравнивания между опциями по границе 32 бит.

Большинство опций в настоящее время не используются. Опции *Stream ID* и *Безопасность* применялись в ограниченном круге экспериментов, функции опций *Запись маршрута* и *Internet Timestamp* выполняет программа traceroute.

Применение опций в дейтаграммах замедляет их обработку. Поскольку большинство дейтаграмм не содержат опций, то есть имеют фиксированную длину заголовка, их обработка макси-

мально оптимизирована именно для этого случая. Появление опции прерывает этот скоростной процесс и вызывает стандартный универсальный модуль IP, способный обработать любые стандартные опции, но за счет существенной потери в быстродействии.

ARP: протокол определения адреса

IP адреса имеют какое-либо значение только в семействе протоколов TCP/IP. Канальные уровни, такие как Ethernet или Token ring, имеют собственную схему адресации (в основном 48-битные адреса); сетевые уровни, в свою очередь, используют эти канальные уровни. Сеть Ethernet, может быть использована различными сетевыми уровнями в одно и то же время. Компьютеры использующие разные сетевые протоколы могут находиться на одном и том же физическом кабеле.

Когда фрейм Ethernet отправляется от одного хоста по локальной сети к другому, по его MAC-адресу определяется, к какому интерфейсу он должен быть доставлен. Драйвер сетевой платы никогда не смотрит на IP адрес назначения в IP дейтаграмме.

Другими словами возникает необходимость установить соответствие между двумя различными формами адресов: 32-битными IP адресами и каким-либо типом адресов канального уровня. RFC 826 [Plummer 1982] — официальная спецификация ARP.

Для этого используются в основном два базовых протокола: протокол определения адреса (ARP — address resolution protocol) и обратный протокол определения адреса (RARP — reverse address resolution protocol).

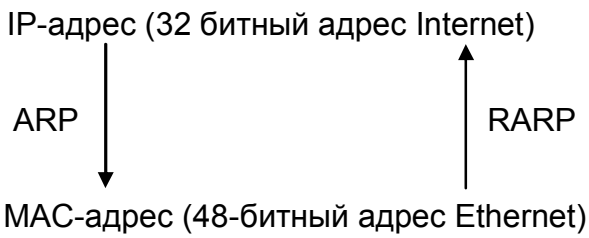


Рис. 6.15 — Транспортный уровень TCP/IP

ARP предоставляет динамическое сопоставление IP адресов и соответствующих аппаратных адресов. Термин динамическое используется тут в том смысле, что это происходит автоматически и не зависит от используемых прикладных программ или воли системного администратора.

Фундаментальная концепция, заложенная в ARP, заключается в следующем. Сетевой интерфейс имеет аппаратный адрес (48-битное значение для Ethernet или Token ring). Фреймы, которыми обмениваются на аппаратном уровне, должны адресоваться к корректному интерфейсу. Однако TCP/IP испоьзует собственную схему адрессации: 32-битные IP адреса. Знание IP адреса хоста не позволяет ядру послать дейтаграмму этому хосту. Драйвер Ethernet должен знать аппаратный адрес пункта назначения, чтобы послать туда данные. В задачу ARP входит обеспечение динамического соответствия между 32-битными IP адресами и аппаратными адресами, используемыми различными сетевыми технологиями.

Каналы точка-точка не используют ARP. Когда эти каналы конфигурируются (обычно во время загрузки), ядру необходимо сказать IP адрес для каждого конца канала. Аппаратные адреса, такие как Ethernet адреса, в данном случае не используются.

Эффективность функционирования ARP во многом зависит от ARP кэша (области памяти ОС выделенной под хранение ARP таблиц), который присутствует на каждом хосте. В кэше содержатся Internet адреса и соответствующие им аппаратные адреса. Стандартное время жизни каждой записи в кэше составляет 20 минут с момента создания записи.

Содержимое ARP кэша можно увидеть с использованием команды arp. Опция -a показывает все записи, содержащиеся в кэше:

На рис. 6.16 показан формат ARP запроса и формат ARP отклика, в случае использования Ethernet и IP адресов. (ARP можно использовать в других сетей, при этом он способен устанавливать соответствие не только для IP адресов. Первые четыре поля, следующие за полем типа фрейма, указывают на типы и размеры заключительных четырех полей.)

Два первых поля в Ethernet заголовке — поля источника и назначения Ethernet. Специальный адрес назначения Ethernet, со-

стоящий из всех единиц, означает широковещательный адрес. Фреймы с таким адресом будут получены всеми Ethernet интерфейсами на кабеле.

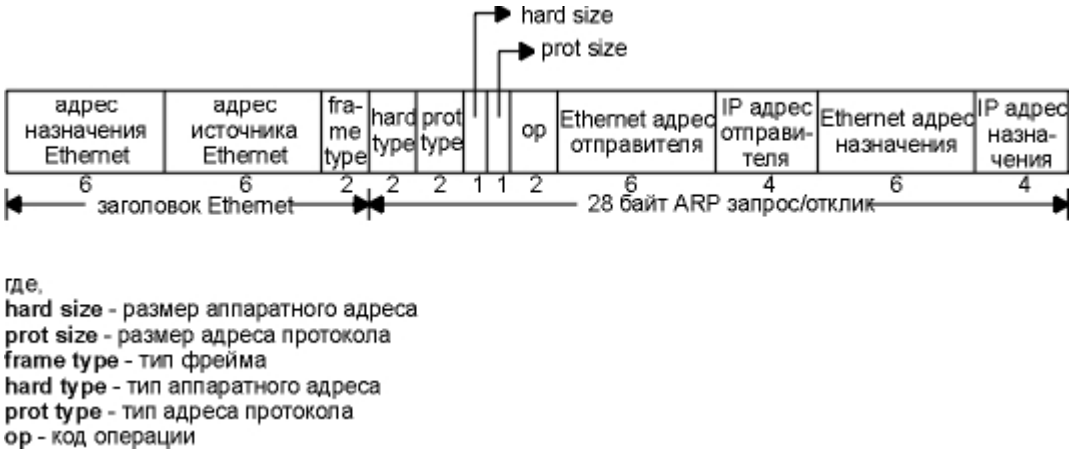


Рис. 6.16 — Формат ARP запроса или отклика при работе с Ethernet

Двухбайтовый тип фрейма (frame type) Ethernet указывает, данные какого типа, пойдут следом. Для ARP запроса или ARP отклика это поле содержит 0x0806.

Выражения аппаратный (hardware) и протокол (protocol) используются для описания полей в пакетах ARP. Например, ARP запрос запрашивает аппаратный адрес (в данном случае Ethernet адрес) соответствующий адресу протокола (в данном случае IP адрес).

Поле hard type указывает на тип аппаратного адреса. Для Ethernet это значение равно единице. Prot type указывает тип адреса протокола, к которому будет приведено соответствие. Для IP адресов используется значение 0x0800. По своему целевому назначению это значение соответствует полю типа во фрейме Ethernet, который содержит IP дейтаграмму.

Два следующих однобайтных поля, hard size и prot size, указывают на размеры в байтах аппаратного адреса и адреса протокола. В ARP запросах и откликах они составляют 6 для Ethernet и 4 для IP адреса.

Поле *op* указывает на тип операции: ARP запрос (значение устанавливается в 1), ARP отклик (2), RARP запрос (3) и RARP отклик (4). Это поле необходимо, так как поля типа фрейма (frame type) одинаковы для ARP запроса и ARP отклика.

Следующие четыре поля: аппаратный адрес отправителя (Ethernet MAC-адрес в данном примере), адрес протокола (IP адрес), аппаратный адрес назначения и адрес протокола назначения. Обратите внимание, что в данном случае происходит некоторое дублирование информации: аппаратный адрес отправителя может быть получен как из Ethernet заголовка, так и из ARP запроса.

Для ARP запроса все поля заполнены, за исключением аппаратного адреса назначения. Когда система получает ARP запрос, который предназначается ей, она вставляет свой аппаратный адрес, меняет местами адреса источника и назначения, устанавливает поле op в значение 2 и отправляет отклик.

RARP: обратный протокол определения адреса

Когда загружается система с локальным диском, она обычно получает свой IP адрес из конфигурационного файла, который считывается с диска. Однако для систем, не имеющих диска, таких как X терминалы или бездисковые рабочие станции, требуются другой способ определения собственного IP адреса.

Каждая система в сети имеет уникальный аппаратный адрес (MAC-адрес), который назначается производителем сетевого интерфейса (сетевой платы). Принцип работы RARP заключается в том, что бездисковая система может считать свой уникальный аппаратный адрес с интерфейсной платы и послать RARP запрос (широковещательный фрейм в сеть), где потребует кого-нибудь откликнуться и сообщить IP адрес (с помощью RARP отклика).

Несмотря на то что концепция довольно проста, ее реализация как правило значительно сложнее чем ARP. Официальная спецификация RARP находится в RFC 903 [Finlayson et al. 1984].

Формат пакета RARP практически идентичен пакету ARP (рис. 6.16). Единственное отличие заключается в том, что поле тип фрейма (frame type) для запроса или отклика RARP установлено в 0x8035, а поле op имеет значение 3 для RARP запроса и значение 4 для RARP отклика.

RARP запрос является широковещательным, а RARP отклик обычно персональный.

Дисковая рабочая станция, способная ответить на RARP-запрос, называется RARP сервером. Основная задача RARP сервера заключается в том, чтобы предоставить соответствие между

аппаратными адресами и IP адресами для множества хостов (все бездисковые системы в сети). Необходимая информация содержится в дисковом файле (обычно /etc/ethers в UNIX системах). Так как ядро обычно не читает дисковые файлы, функция RARP сервера реализуется с использованием пользовательского процесса, который не является частью ядра TCP/IP.

Еще раз подчеркнем, что RARP запросы передаются в качестве Ethernet фреймов со специфическим полем типа фрейма Ethernet (0x8035). Это означает, что RARP сервер должен обладать способностью отправлять и принимать Ethernet фреймы подобного типа.

Еще одна особенность заключается в том, что RARP запросы посылаются в виде широковещательных запросов аппаратного уровня, что означает, что они не перенаправляются маршрутизаторами. Чтобы позволить бездисковым системам загружаться, даже если RARP сервер выключен, в сети обычно существуют несколько RARP серверов (на одном и том же кабеле).

По мере того как количество серверов растет (чтобы повысить надежность), увеличивается сетевой траффик, так как каждый сервер посылает RARP отклик на каждый RARP запрос. Бездисковые системы, которые посылают RARP запросы, обычно используют первый полученный ими RARP отклик. (Мы никогда не имели подобных проблем с ARP, потому что только один хост посылает ARP отклик.) Более того, существует вероятность, что несколько RARP серверов отправят отклики одновременно, увеличивая тем самым количество коллизий в Ethernet.

ICMP: протокол управления сообщениями Internet

Обычно считается, что ICMP это часть IP уровня. С его помощью передаются сообщения об ошибках и сообщения о возникновении условий и ситуаций, которые требуют к себе особого внимания. ICMP сообщения обрабатываются IP уровнем или более высокими уровнями (TCP или UDP). При появлении некоторых ICMP сообщений генерируются сообщения об ошибках, которые передаются пользовательским процессам. ICMP сообщения передаются внутри IP дейтаграмм, как показано на рис. 6.17.

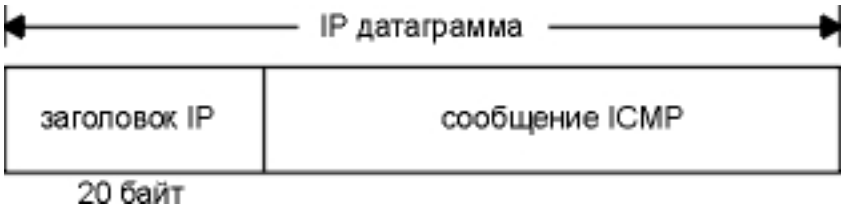


Рис. 6.17 — Инкапсуляция ICMP
сообщений в IP дейтаграммы

Официальная спецификация ICMP находится в RFC 792 [Postel 1981b].

На рис. 6.18 показан формат ICMP сообщения. Первые 4 байта одинаковы для всех сообщений, однако остальные отличаются в зависимости от типа сообщения. Мы будем показывать точный формат каждого сообщения, по мере того как будем их описывать.

Существует 15 различных значений для поля типа (type), которые указывают на конкретныей тип ICMP сообщения. Для некоторых ICMP сообщений используются различные значения в поле кода (code), подобным образом осуществляется дальнейшее подразделение ICMP сообщений. Поле контрольной суммы (checksum) охватывает ICMP сообщения целиком. Алгоритм, используемый при этом, такой же как тот, что используется при расчете контрольной суммы IP заголовка. Контрольная сумма ICMP присутствует всегда.

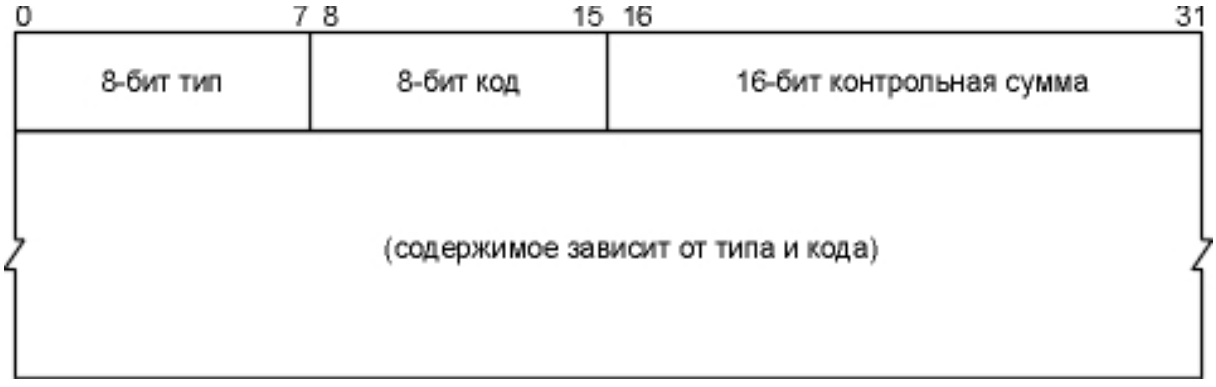


Рис. 6.18 — ICMP сообщение

Типы сообщений ICMP

В таблице ниже приведены возможные типы ICMP сообщений, как они определяются полями типа (type) и кода (code).

Последние две колонки на рисунке указывают, является ли ICMP сообщение запросом (query) или сообщением об ошибке (error). Подобное разделение необходимо, потому что сообщения об ошибках ICMP иногда обрабатываются специальным образом. Например, ICMP сообщение об ошибке никогда не генерируется в ответ на ICMP сообщение об ошибке. (Если не придерживаться этого правила, то ошибка будет генерироваться на ошибку до бесконечности.)

Когда посылается ICMP сообщение об ошибке, оно всегда содержит IP заголовок и первые 8 байт IP дейтаграммы, которая вызвала генерацию ICMP ошибки. Это позволяет принимающему ICMP модулю установить соответствие между полученным сообщением, одним из конкретных протоколов (TCP или UDP из поля протоколов в IP заголовке) и с одним из конкретных пользовательских процессов (с помощью номера порта TCP или UDP, который содержится в TCP или UDP заголовке в первых 8 байтах IP дейтаграммы).

Сообщение об ошибке ICMP никогда не генерируется в ответ на:

1. ICMP сообщение об ошибке. (ICMP сообщение об ошибке, однако, может быть сгенерировано в ответ на ICMP запрос.)
2. Дейтаграмму, направляющуюся на широковещательный IP адрес или групповой адрес IP.
3. Дейтаграмму, которая посылается широковещательным запросом на канальном уровне.
4. Фрагмент, который не является первым.
5. Дейтаграмму, адрес источника которой не указывает на конкретный хост. Это означает, что адрес источника не может быть нулевым, loopback адресом, широковещательным или групповым адресом.

Эти правила введены для того, чтобы предотвратить лавинообразный рост количества широковещательных сообщений, который может произойти, если ICMP сообщения об ошибках будут отправляться в ответ на широковещательные пакеты.

Таблица 6.5 — **Типы сообщений ICMP**

Тип	Код	Описание
0	0	эхо-отклик (отклик-Ping, глава 7)
3		назначение недоступно:
	0	сеть недоступна — network unreachable (раздел «ICMP ошибки о недоступности хоста и сети» главы 9)
	1	хост недоступен — host unreachable (раздел «ICMP ошибки о недоступности хоста и сети» главы 9)
	2	протокол недоступен — protocol unreachable
	3	порт недоступен — port unreachable (раздел «ICMP ошибка недоступности порта (ICMP Port Unreachable Error)» главы 6)
	4	необходима фрагментация, однако установлен бит «не фрагментировать» — fragmentation needed but don't-fragment bit set (раздел «ICMP ошибки о недоступности» главы 11)
	5	не работает маршрутизация от источника — source route failed (глава 8, раздел «Опция IP маршрутизации от источника»)
	6	неизвестна сеть назначения — destination network unknown
	7	неизвестен хост назначения — destination host unknown
	8	хост источник изолирован — source host isolated
	9	сеть назначения закрыта администратором — destination network administrativrly prohibited
	10	хост назначения закрыт администратором — destination host administrativrly prohibited
	11	сеть недоступна для TOS — network unreachable for TOS (глава 9, раздел «ICMP ошибки о недоступности хоста и сети»)
	12	хост недоступен для TOS — host unreachable for TOS (глава 9, раздел «ICMP ошибки о недоступности хоста и сети»)
	13	связь административно закрыта путем фильтрации — communication administratively prohibited by filtering
	14	нарушено старшинство для хоста — host precedence violation
	15	старшинство разъединено — precedence cutoff in effect
4	0	подавление источника (элементарное управление потоком данных) — source quench (глава 11, раздел «ICMP ошибка подавления источника»)
5		перенаправление — redirect (глава 11, раздел «ICMP ошибка подавления источника»):
	0	перенаправление в сеть — redirect for network
	1	перенаправление в хост — redirect for host
	2	перенаправление для типа сервиса и сети — redirect for type-of-service and network

Окончание табл. 6.5

Тип	Код	Описание
	3	перенаправление для типа сервиса и хоста — redirect for type-of-service and host
8	0	эхо запрос — echo request (Ping запрос, глава 7)
9	0	объявление маршрутизатора — router advertisement (глава 9, раздел «ICMP сообщения поиска маршрутизатора»)
10	0	запрос к маршрутизатору — router solicitation (глава 9, раздел «ICMP сообщения поиска маршрутизатора»)
11		время истекло — time exceeded:
	0	время жизни стало равным 0 в процессе передачи — time-to-live equals 0 during transit (Traceroute, глава 8)
	1	время жизни стало равным 0 в процессе повторной сборки — time-to-live equals 0 during reassembly (глава 11, раздел «Фрагментация IP»)
12		проблемы с параметрами — parameter problem:
	0	неверный IP заголовок — IP header bad
	1	отсутствует необходимая опция — required option missing
13	0	запрос временной марки — timestamp request (глава 6, раздел «ICMP запрос и отклик временной марки»)
14	0	отклик с временной маркой — timestamp reply (глава 6, раздел «ICMP запрос и отклик временной марки»)
15	0	информационный запрос — information request
16	0	информационный отклик — information reply
17	0	запрос маски адреса — address mask request (глава 6, раздел «ICMP запрос и отклик маски адреса»)
18	0	отклик с маской адреса — address mask reply (глава 6, раздел «ICMP запрос и отклик маски адреса»)

ICMP запрос и отклик маски адреса

ICMP запрос маски адреса используется бездисковыми системами, чтобы получить маску подсети во время загрузки. Система посылает широковещательный ICMP запрос.

Рис. 6.19 — ICMP запрос и отклик маски адреса

Поля идентификатора и номера последовательности в ICMP сообщении могут быть установлены по выбору отправителя, эти же значения будут возвращены в отклике. Именно таким образом отправитель идентифицирует отклик на свой запрос.

ICMP запрос и отклик времени

ICMP запрос времени позволяет системе запросить другую систему о текущем времени. Рекомендуемое значение, которое должно быть возвращено, это количество миллисекунд, которые прошли с полуночи в формате UTC, (Универсальное согласованное время — Coordinated Universal Time). (В старых руководствах UTC называется Среднее время по Гринвичу — Greenwich Mean Time.) Одна из основных особенностей ICMP сообщения заключается в том, что оно предоставляет время в с точностью до миллисекунд.

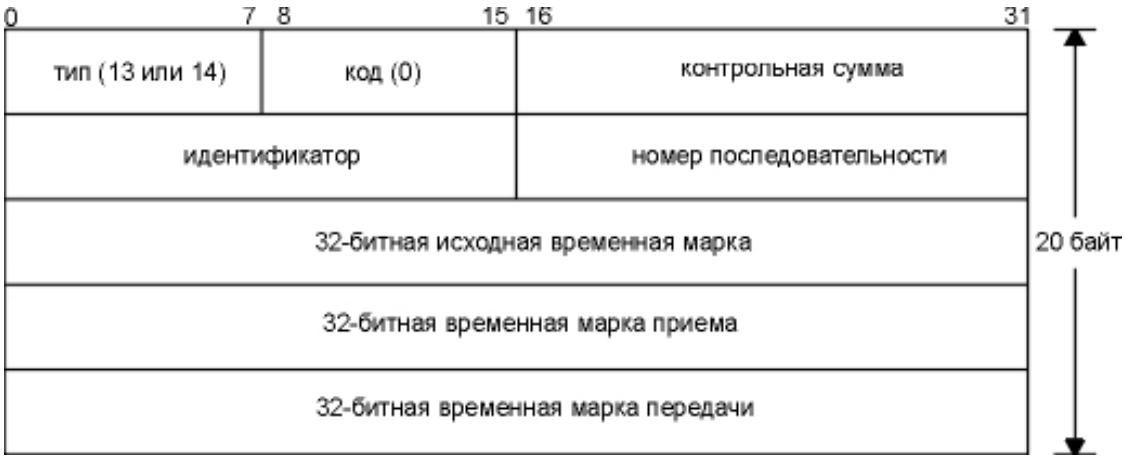


Рис. 6.20 — ICMP запрос и отклик времени

Запрашивающий заполняет исходное время и отправляет запрос. Отвечающая система заполняет время приема, когда получает запрос, и время передачи, когда отправляет отклик. Большинство реализаций устанавливают в два последних поля одно и то же значение. (Причина, по которой существуют три поля, заключается в том, что отправителю необходимо вычислить время, которое потребовалось на отправку запроса, и отдельно рассчитать время, которое потребуется на отправку отклика).

ICMP ошибка недоступности порта (ICMP Port Unreachable Error)

В двух последних разделах описаны запросы ICMP — маски адреса, и запросы и отклики временной марки. Сейчас мы рассмотрим ICMP сообщения об ошибках, а именно: сообщение о недоступности порта, подкод сообщения ICMP о недоступности пункта назначения. Также рассмотрим дополнительную информацию, которая возвращается в ICMP сообщении об ошибке.

6.6 Некоторые важные сетевые службы прикладного уровня

Система имен доменов (DNS — Domain Name System)

Несмотря на то, что каждый сетевой интерфейс компьютера имеет свой собственный IP адрес, пользователи привыкли работать с именами хостов. Существует распределенная мировая база данных TCP/IP, называемая системой имен доменов (***DNS — Domain Name System***), которая позволяет установить соответствие между IP адресами и именами хостов.

Для того чтобы определить IP адрес (или адреса, соответствующие данному имени хоста) приложение должно вызвать функцию из стандартной библиотеки, которая предоставляет возможность осуществить и обратную процедуру, то есть по заданному IP адресу определить соответствующее имя хоста. Большинство приложений, которые воспринимают имя хоста в качестве аргумента, также воспринимают и IP адреса.

Протокол DNS является служебным протоколом прикладного уровня. Этот протокол несимметричен — в нем определены DNS-серверы и DNS-клиенты. DNS-серверы хранят часть распределенной базы данных о соответствии символьных имен и IP-адресов. Эта база данных распределена по административным доменам сети Internet. Клиенты сервера DNS знают IP-адрес сервера DNS своего административного домена и по протоколу IP передают запрос, в котором сообщают известное символьное имя и просят вернуть соответствующий ему IP-адрес.

Если данные о запрошенном соответствии хранятся в базе данного DNS-сервера, то он сразу посылает ответ клиенту, если же нет — то он посылает запрос DNS-серверу другого домена,

который может сам обработать запрос, либо передать его другому DNS-серверу. Все DNS-серверы соединены иерархически, в соответствии с иерархией доменов сети Internet. Процесс поиска ускоряется из-за того, что серверы имен постоянно кэшируют информацию, предоставляемую по запросам. Клиентские компьютеры могут использовать в своей работе IP-адреса нескольких DNS-серверов, для повышения надежности своей работы.

База данных DNS имеет структуру дерева, называемого доменным пространством имен, в котором каждый домен (узел дерева) имеет имя и может содержать поддомены. Имя домена идентифицирует его положение в этой базе данных по отношению к родительскому домену, причем точки в имени отделяют части, соответствующие узлам домена.

Корень базы данных DNS управляется центром ***Internet Network Information Center.*** Домены верхнего уровня назначаются для каждой страны, а также на организационной основе. Имена этих доменов должны следовать международному стандарту ISO 3166. Для обозначения стран используются трехбуквенные и двухбуквенные аббревиатуры, а для различных типов организаций используются следующие аббревиатуры:

- com — коммерческие организации (например, microsoft.com);
- edu — образовательные (например, mit.edu);
- gov — правительственные организации (например, nsf.gov);
- org — некоммерческие организации (например, fidonet.org);
- net — организации, поддерживающие сети (например, nsf.net).

Каждый домен DNS администрируется отдельной организацией, которая обычно разбивает свой домен на поддомены и передает функции администрирования этих поддоменов другим организациям. Каждый домен имеет уникальное имя, а каждый из поддоменов имеет уникальное имя внутри своего домена. Имя домена может содержать до 63 символов. Каждый хост в сети Internet однозначно определяется своим полным доменным именем (***fully qualified domain name, FQDN***), которое включает имена всех доменов по направлению от хоста к корню.

Автоматизация процесса назначения IP-адресов узлам сети — протокол DHCP

Как уже было сказано, IP-адреса могут назначаться администратором сети вручную. Это представляет для администратора утомительную процедуру. Ситуация усложняется еще тем, что многие пользователи не обладают достаточными знаниями для того, чтобы конфигурировать свои компьютеры для работы в интерсети и должны поэтому полагаться на администраторов.

Протокол ***Dynamic Host Configuration Protocol*** (***DHCP***) был разработан для того, чтобы освободить администратора от этих проблем. Основным назначением DHCP является динамическое назначение IP-адресов. Однако, кроме динамического, DHCP может поддерживать и более простые способы ручного и автоматического статического назначения адресов.

При автоматическом статическом способе выделения адресов DHCP-сервер присваивает IP-адрес (и, возможно, другие параметры конфигурации клиента) из пула наличных IP-адресов без вмешательства оператора. Границы пула назначаемых адресов задает администратор при конфигурировании DHCP-сервера. Между идентификатором клиента и его IP-адресом по-прежнему, как и при ручном назначении, существует постоянное соответствие. Оно устанавливается в момент первичного назначения сервером DHCP IP-адреса клиенту. При всех последующих запросах сервер возвращает тот же самый IP-адрес.

При динамическом распределении адресов DHCP-сервер выдает адрес клиенту на ограниченное время, что дает возможность впоследствии повторно использовать IP-адреса другими компьютерами. Динамическое разделение адресов позволяет строить IP-сеть, количество узлов в которой намного превышает количество имеющихся в распоряжении администратора IP-адресов.

DHCP обеспечивает надежный и простой способ конфигурации сети TCP/IP, гарантируя отсутствие конфликтов адресов за счет централизованного управления их распределением. Администратор управляет процессом назначения адресов с помощью параметра ***продолжительности аренды*** (***lease duration***), которая определяет, как долго компьютер может использовать назначен-

ный IP-адрес, перед тем как снова запросить его от сервера DHCP в аренду.

Примером работы протокола DHCP может служить ситуация, когда компьютер, являющийся клиентом DHCP, удаляется из подсети. При этом назначенный ему IP-адрес автоматически освобождается. Когда компьютер подключается к другой подсети, то ему автоматически назначается новый адрес. Ни пользователь, ни сетевой администратор не вмешиваются в этот процесс. Это свойство очень важно для мобильных пользователей.

Протокол DHCP использует модель клиент-сервер. Во время старта системы компьютер-клиент DHCP, находящийся в состоянии *инициализация*, посылает сообщение ***discover*** (***исследовать***), которое широковещательно распространяется по локальной сети и передается всем DHCP-серверам частной интерсети. Каждый DHCP-сервер, получивший это сообщение, отвечает на него сообщением ***offer*** (***предложение***), которое содержит IP-адрес и конфигурационную информацию.

Компьютер-клиент DHCP переходит в состояние *выбор* и собирает конфигурационные предложения от DHCP-серверов. Затем он выбирает одно из этих предложений, переходит в состояние *запрос* и отправляет сообщение ***request*** (***запрос***) тому DHCP-серверу, чье предложение было выбрано.

Выбранный DHCP-сервер посылает сообщение ***DHCP-acknowledgment*** (***подтверждение***), содержащее тот же IP-адрес, который уже был послан ранее на стадии исследования, а также параметр аренды для этого адреса. Кроме того, DHCP-сервер посылает параметры сетевой конфигурации. После того, как клиент получит это подтверждение, он переходит в состояние *связь*, находясь в котором он может принимать участие в работе сети TCP/IP. Компьютеры-клиенты, которые имеют локальные диски, сохраняют полученный адрес для использования при последующих стартах системы. При приближении момента истечения срока аренды адреса компьютер пытается обновить параметры аренды у DHCP-сервера, а если этот IP-адрес не может быть выделен снова, то ему возвращается другой IP-адрес.

Однако использование DHCP несет в себе и некоторые проблемы. Во-первых, это проблема согласования информационной адресной базы в службах DHCP и DNS. Как известно, DNS слу-

жит для преобразования символьных имен в IP-адреса. Если IP-адреса будут динамически изменятся сервером DHCP, то эти изменения необходимо также динамически вносить в базу данных сервера DNS. Хотя протокол динамического взаимодействия между службами DNS и DHCP уже реализован некоторыми фирмами (так называемая служба Dynamic DNS), стандарт на него пока не принят.

Во-вторых, нестабильность IP-адресов усложняет процесс управления сетью. Системы управления, основанные на протоколе SNMP, разработаны с расчетом на статичность IP-адресов. Аналогичные проблемы возникают и при конфигурировании фильтров маршрутизаторов, которые оперируют с IP-адресами.

Служба каталогов Active Directory

Назначение службы каталогов

По своей сути, ***служба каталогов Active Directory*** это средство для именования, хранения и выборки информации в некоторой распределенной среде, доступное для приложений, пользователей и различных клиентов этой среды. Служба сетевых каталогов хранит информацию об общедоступных приложениях, файлах, принтерах и сведения о пользователях.

Служба каталогов Active Directory обеспечивает эффективную работу сложной корпоративной среды, предоставляя следующие возможности:

- ***Единая регистрация в сети.*** Пользователи могут регистрироваться в сети *с одним* именем и паролем и получать при этом доступ ко *всем* сетевым ресурсам (серверам, принтерам, приложениям, файлам и т.д.) независимо от их расположения в сети.
- ***Безопасность информации.*** Средства аутентификации и управления доступом к ресурсам, встроенные в службу Active Directory, обеспечивают централизованную защиту сети. Права доступа можно определять не только для каждого *объекта* каталога, но и каждого *свойства* (атрибута) объекта.
- ***Централизованное управление.*** Администраторы могут централизованно управлять всеми корпоративными ресурсами.

Рутинные задачи администрирования не нужно повторять для многочисленных объектов сети.

- ***Администрирование с использованием групповых политик.*** При загрузке компьютера или регистрации пользователя в системе выполняются требования групповых политик; их настройки хранятся в ***объектах групповых политик*** (***GPO***) и назначаются сайтам, доменам или организационным единицам. Групповые политики определяют, например, права доступа к различным объектам каталога или ресурсам, а также множество других правил работы в системе.
- ***Интеграция с DNS.*** Служба Active Directory тесно связана с DNS. Этим достигается единство в именовании ресурсов локальной сети и сети Интернет, в результате чего упрощается подключение пользовательской сети к Интернету.
- ***Масштабируемость.*** Служба Active Directory может охватывать как один домен, так и множество доменов, один контроллер домена или множество контроллеров домена — т.е. она отвечает требованиям сетей любого масштаба. Несколько доменов можно объединить в дерево доменов, а несколько деревьев доменов можно связать в лес.
- ***Репликация информации.*** В службе Active Directory используется репликация служебной информации в схеме со ***многими ведущими*** (***multi-master***), что позволяет модифицировать каталог на любом контроллере домена. Наличие в домене нескольких контроллеров обеспечивает отказоустойчивость и возможность распределения сетевой нагрузки.
- ***Гибкость запросов к каталогу.*** Пользователи и администраторы сети могут быстро находить объекты в сети, используя *свойства* объекта (например, имя пользователя или адрес его электронной почты, тип принтера или его местоположение и т.п.). Оптимальность процедуры поиска достигается благодаря использованию глобального каталога.
- ***Стандартные интерфейсы.*** Для разработчиков приложений служба каталогов предоставляют доступ ко всем возможностям (средствам) каталога и поддерживают принятые стандарты и интерфейсы программирования (API). Служба каталогов тесно связана с операционной системой, что позволяет избежать

дублирования в прикладных программах функциональных возможностей системы, например, средств безопасности.

Архитектура Active Directory

Служба каталогов нужна многим приложениям. Требуется она и операционным системам, которым удобно хранить в едином каталоге учетные записи пользователей, информацию о файлах и приложениях политики безопасности и многое другое.

Если в некоторой распределенной среде отсутствует главный, центральный каталог, то каждому приложению необходимо иметь собственный каталог, в результате чего появляются различные решения и механизмы хранения информации. Понятно, что наличие нескольких механизмов, реализующих одну и ту же задачу — далеко не самое удачное решение. Гораздо лучше — единственная служба каталогов, доступная всем клиентам и имеющая одну базу данных, общие схему и соглашения об именовании информации, а также возможность централизованного администрирования.

Операционная система Windows хранит в каталоге информацию о пользовательских учетных записях, принтерах и компьютерах сети, а также многое другое. Большое значение Active Directory имеет для ***Windows Management Architecture*** (*Архитектура управления Windows*) — в частности, *с* помощью каталога ищутся серверы, на которых располагаются компоненты приложений.

Служба Active Directory основана на трех базовых технологиях:

- Стандарт X.500.
- Служба DNS.
- Протокол LDAP.

В Active Directory частично реализована модель данных, описываемая стандартом ***X.500.*** Традиционная в сетях TCP/IP служба DNS используется, в частности, для поиска контроллеров домена, а благодаря протоколу ***LDAP*** (***Lightweight Directory Access Protocol***) клиенты могут по имени находить в каталоге Active Directory нужные объекты и получать доступ к их атрибутам.

Объекты и объектные классы

Каталог состоит из ***элементов*** (***entries***), представляющих собой информацию, или ***атрибуты,*** связанные с некоторым реальным ***объектом,*** например компьютером, человеком или организацией. Термины элемент и объект часто используют как взаимозаменяемые, хотя объект — это нечто относящееся к физическому миру, а элемент — его представление в каталоге.

Каждый объект принадлежит, по крайней мере, к одному ***объектному классу,*** представляющему собой некоторое семейство объектов с определенными общими характеристиками. Класс объектов определяет тип информации, содержащейся в Active Directory для экземпляров (объектов) данного класса. В качестве примера объектных классов можно привести два стандартных класса: person и domain. Среди множества атрибутов этих классов — cn (Common-Name), userPassword (User-Password) и dc (Domain-Component), url (WWW-Page-Other), соответственно. Атрибуты могут быть как ***обязательными*** (***mandatory***) для данного класса (например, cn и dc), так и ***дополнительными*** (***optional***) (userPassword и url).

Помимо стандартных объектных классов можно описывать *дополнительные* классы, относящиеся к различным уровням (national и local).

Атрибуты и их типы

Каждый элемент каталога имеет ***атрибуты*** различных ***типов***, характеризующих информацию, содержащуюся в этих атрибутах. Например, атрибут типа commonName представляет собой имя, идентифицирующее некоторый объект. Каждый атрибут может иметь одно или несколько ***значении.***

Помимо атрибутов стандартных типов можно создавать и использовать дополнительные типы атрибутов.

Контейнер

Контейнер (***container***) — это специфический объект службы каталогов, который, в отличие от обычных объектов, не имеет какого-либо физического представления, а служит только структурной организации — группировки — других объектов каталога. Типичным примером контейнеров могут служить *организаци-*

онные единицы, или *подразделения,* используемые для упрощения администрирования отдельных групп ресурсов или пользователей в домене.

Информационное дерево каталога

Элементы каталога организованы в виде *иерархического дерева,* называемого ***Directory Information Tree*** (DIT, Информационное дерево каталога или просто Дерево каталога). Элементы, находящиеся ближе к корню дерева, обычно представляют крупные объекты, например, организации или компании; элементы, располагающиеся на ветвях этого дерева, (листья) представляют более простые объекты — пользователей, устройства, компьютеры.

Схема каталога

Схема каталога (***Directory Schema***) — это набор правил, описывающих структуру дерева каталога, объявления и синтаксис объектных классов и типы атрибутов, входящих в каталог.

Схема каталога гарантирует, что все добавления или изменения каталога соответствуют данным правилам, и препятствует появлению некорректных элементов, ошибочных типов атрибутов или классов.

В Active Directory схема реализована как набор экземпляров объектных классов, хранящийся в самом каталоге. Этим Active Directory отличается от многих каталогов, в которых схема хранится в текстовом файле, считываемом при запуске каталога. Когда схема хранится в каталоге, пользовательские приложения могут обращаться к ней и узнавать об имеющихся объектах и свойствах.

Пространство имен

Любая служба каталога в первую очередь представляет собой некоторое ***пространство имен*** (***namespace***). Пространство имен — это любая ограниченная область, в которой можно по имени обратиться к атрибутам самого объекта или к информации, связанной с этим именем. Процесс преобразования имени в ссылку на объект называется *разрешением имен.* За иерархию пространства имен, за его топологию и структуру отвечает ***служба каталогов Active Directory.*** Основные компоненты любой службы каталога — база данных, содержащая нужную информацию, и

один или несколько протоколов, обеспечивающих доставку данных пользователям.

Active Directory обеспечивает хранение любой общедоступной информации. Как и другие службы каталогов, Active Directory обеспечивает некоторый механизм хранения информации и протоколы для доступа к ней.

Домены и Контроллеры доменов

Домены — это известное решение для администрирования групп, предоставляющее каждому пользователю учетную запись в конкретном домене. Доменная система внедрялась начиная с *Windows NT Server*, где доменам давались простые строковые имена (имена *NetBIOS*), в среде же *Windows 2000 Server* каждый домен должен иметь имя, отвечающее соглашениям именования доменов *Domain Name System* (*DNS*). Так, домен *MainOffice* при обновлении может получить новое имя типа *mainqfflce.company.com.* В каждом домене один или несколько компьютеров должны выполнять функции ***контроллеров домена***. Каждый контроллер домена содержит полную копию базы данных Active Directory этого домена. В Active Directory используются так называемое ядро ***Extended Storage Engine*** (***ESE***) и два различных протокола, обеспечивающих связь между клиентами и базой данных. Для поиска контроллера домена клиент обращается к протоколу, описанному в DNS — стандартной службе каталогов, применяемой в настоящее время для сетей TCP/IP. Для доступа к данным в Active Directory клиент использует протокол ***Lightweight Directory Access Protocol*** (***LDAP***) (рис. 6.21).

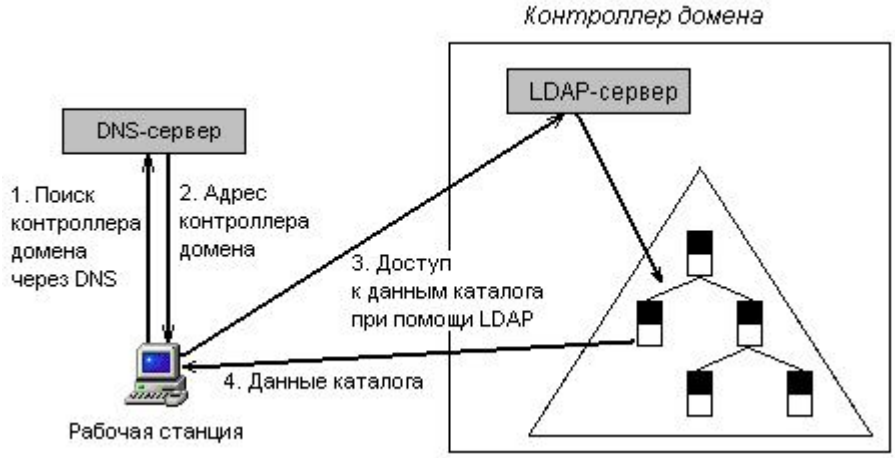


Рис. 6.21

Службы DNS и Active Directory

Интеграцию служб Active Directory и DNS можно рассматривать в трех аспектах:

- Домены Active Directory и домены DNS имеют одинаковую иерархическую структуру и схожее пространство имен.
- Зоны (***zone***) DNS могут храниться в Active Directory. Если используется сервер DNS, входящий в состав Windows 2000 Server, то первичные зоны (primary zone), занесенные в каталог, реплицируются на все контроллеры домена, что обеспечивает лучшую защищенность службы DNS.
- Использование клиентами службы DNS при поиске контроллеров домена.

Active Directory может использовать любую стандартную, законченную реализацию службы DNS: не обязательно задействовать DNS-сервер, входящий в Windows 2000 Server (например; можно использовать BIND 8.1.x). Однако лучше остановить свой выбор на нем, поскольку модули Windows 2000 более согласованы друг с другом (хранение и репликация зон и т.п.), ведь необходимо, чтобы выбранный DNS-сервер соответствовал последним стандартам. Например, для Active Directory нужен DNS-сервер, поддерживающий записи типа SRV. Записи подобного типа (SRV records), в соответствии с *RFC 2052*, позволяют клиентам находить нужные сетевые службы. В Active Directory служба LDAP каждого домена Windows 2000 представлена некоторой SRV-записью службы DNS. Такая запись содержит DNS-имя контроллера этого домена, по которому клиенты Active Directory могут находить IP-адрес компьютера-контроллера домена. После того как нужный контроллер обнаружен, для доступа к данным Active Directory, хранящихся на нем, клиент может использовать протокол LDAP.

Windows 2000 Server поддерживает также ***службу динамического именования хостов, Dynamic DNS***. В соответствии с *RFC 2136* служба Dynamic DNS расширяет протокол DNS, позволяя модифицировать базу данных DNS со стороны удаленных систем. Например, при подключении некоторый контроллер домена может сам добавлять SRV-запись для себя, освобождая администратора от такой необходимости.

7 ОБОРУДОВАНИЕ ИНФОРМАЦИОННЫХ СЕТЕЙ

7.1 Типовой состав оборудования вычислительной сети

Фрагмент вычислительной сети (рис. 7.1) включает основные типы коммуникационного оборудования, применяемого сегодня для образования локальных сетей и соединения их через глобальные связи друг с другом. Для построения локальных связей между компьютерами используются различные виды кабельных систем, сетевые адаптеры, концентраторы-повторители, мосты, коммутаторы и маршрутизаторы. Для подключения локальных сетей к глобальным используются специальные выходы (*WAN-порты*) мостов и маршрутизаторов. Для подключения к цифровым каналам связи используются *TA* — терминальные адаптеры сетей ISDN и устройства обслуживания цифровых выделенных каналов типа CSU/DSU и т.п. При передаче данных по аналоговым линиям применяется модемная связь.

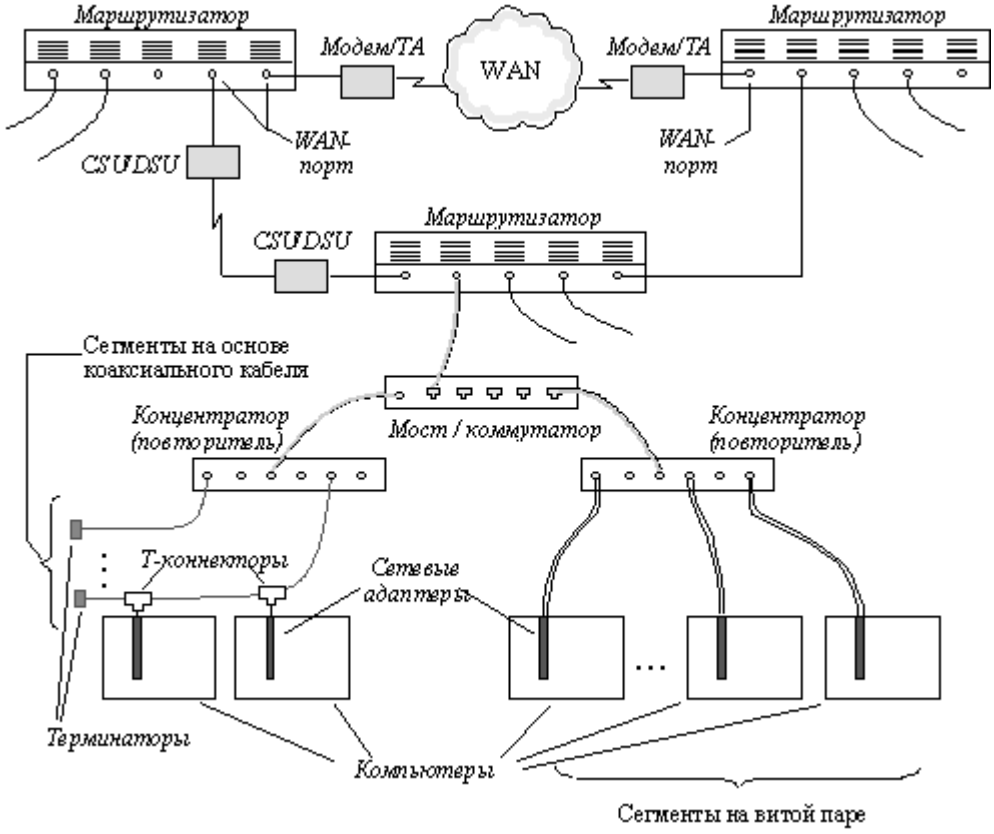


Рис. 7.1 — Фрагмент вычислительной сети

Задачи надежного обмена двоичными сигналами по линиям связи в локальных сетях решают сетевые адаптеры, а в глобальных сетях — аппаратура передачи данных. Это оборудование кодирует и декодирует информацию, синхронизирует передачу электромагнитных сигналов по линиям связи и проверяет правильность передачи.

Важной характеристикой сети является топология — тип графа, вершинам которого соответствуют компьютеры сети (иногда и другое оборудование, например концентраторы), а ребрам — ***физические связи*** между ними. Конфигурация физических связей определяется электрическими соединениями компьютеров между собой и может отличаться от конфигурации логических связей между узлами сети. ***Логические связи*** представляют собой маршруты передачи данных между узлами сети.

Для вычислительных сетей характерны как индивидуальные линии связи между компьютерами, так и разделяемые, когда одна линия связи попеременно используется несколькими компьютерами. В последнем случае возникают как чисто электрические проблемы обеспечения нужного качества сигналов при подключении к одному и тому же проводу нескольких приемников и передатчиков, так и логические проблемы разделения времени доступа к этим линиям

7.2 Кабельная система

Для построения локальных связей в вычислительных сетях в настоящее время используются различные виды кабелей — коаксиальный кабель, кабель на основе экранированной и неэкранированной витой пары и оптоволоконный кабель. Наиболее популярным видом среды передачи данных на небольшие расстояния (до 100 м) становится *неэкранированная витая пара*, которая включена практически во все современные стандарты и технологии локальных сетей и обеспечивает пропускную способность до 100 Мб/с. *Оптоволоконный кабель* широко применяется как для построения локальных связей, так и для образования магистралей глобальных сетей. Оптоволоконный кабель может обеспечить очень высокую пропускную способность канала (до нескольких десятков Гб/с) и передачу на значительные расстояния (до не-

скольких десятков километров без промежуточного усиления сигнала). В качестве среды передачи данных в вычислительных сетях используются также электромагнитные волны различных частот — КВ, УКВ, СВЧ. Однако пока в локальных сетях радиосвязь используется только в тех случаях, когда оказывается невозможной прокладка кабеля. Это объясняется, прежде всего, недостаточной надежностью сетевых технологий, построенных на использовании электромагнитного излучения. Для построения глобальных каналов этот вид среды передачи данных используется шире — на нем построены спутниковые каналы связи и наземные радиорелейные каналы, работающие в зонах прямой видимости в СВЧ-диапазонах.

Согласно зарубежным исследованиям, 70 % времени простоев обусловлено проблемами, возникшими вследствие низкого качества применяемых кабельных систем. Поэтому так важно правильно построить фундамент сети — кабельную систему. В последнее время в качестве такой надежной основы все чаще используется структурированная кабельная система. ***Структурированная кабельная система*** (***Structured Cabling System, SCS***) — это набор коммутационных элементов (кабелей, разъемов, коннекторов, кроссовых панелей и шкафов), а также методика их совместного использования, которая позволяет создавать регулярные, легко расширяемые структуры связей в вычислительных сетях.

Преимущества структурированной кабельной системы:

- *Универсальность.* Структурированная кабельная система при продуманной организации может стать единой средой для передачи компьютерных данных в локальной вычислительной сети, организации локальной телефонной сети, передачи видеоинформации и даже передачи сигналов от датчиков пожарной безопасности или охранных систем. Это позволяет автоматизировать многие процессы по контролю, мониторингу и управлению хозяйственными службами и системами жизнеобеспечения.
- *Увеличение срока службы.* Срок старения хорошо структурированной кабельной системы может составлять 8—10 лет.
- *Уменьшение стоимости добавления новых пользователей и изменения их мест размещения.* Стоимость кабельной системы в основном определяется не стоимостью кабеля, а стоимостью работ по его прокладке. Поэтому более выгодно провести одно-

кратную работу по прокладке кабеля, возможно с большим запасом по длине, чем несколько раз выполнять прокладку, наращивая длину кабеля. Это помогает быстро и дешево изменять структуру кабельной системы при перемещениях персонала или смене приложений.

- *Возможность легкого расширения сети.* Структурированная кабельная система является модульной, поэтому ее легко наращивать, позволяя легко и ценой малых затрат переходить на более совершенное оборудование, удовлетворяющее растущим требованиям к системам коммуникаций.
- *Обеспечение более эффективного обслуживания.* Структурированная кабельная система облегчает обслуживание и поиск неисправностей по сравнению с шинной кабельной системой.
- *Надежность.* Структурированная кабельная система имеет повышенную надежность поскольку обычно производство всех ее компонентов и техническое сопровождение осуществляется одной фирмой-производителем.

7.3 Сетевые адаптеры

Сетевой адаптер (***Network Interface Card, NIC***) — это периферийное устройство компьютера, непосредственно взаимодействующее со средой передачи данных, которая прямо или через другое коммуникационное оборудование связывает его с другими компьютерами. Это устройство решает задачи надежного обмена двоичными данными, представленными соответствующими электромагнитными сигналами, по внешним линиям связи. Как и любой контроллер компьютера, сетевой адаптер работает под управлением драйвера операционной системы и распределение функций между сетевым адаптером и драйвером может изменяться от реализации к реализации. В большинстве современных стандартов для локальных сетей предполагается, что между сетевыми адаптерами взаимодействующих компьютеров устанавливается специальное коммуникационное устройство (*концентратор, мост, коммутатор или маршрутизатор*), которое берет на себя некоторые функции по управлению потоком данных.

Сетевые адаптеры различаются по типу и разрядности внутренней шины данных, используемой в компьютере — ISA, EISA, PCI, MCA.

Сетевой адаптер обычно выполняет следующие ***функции***:

- Оформление передаваемой информации в виде кадров определенного формата. Кадр имеет несколько служебных полей, среди которых имеется адрес компьютера назначения и контрольная сумма кадра, по которой сетевой адаптер станции назначения делает вывод о корректности доставленной по сети информации.
- Получение доступа к среде передачи данных. В последних стандартах и технологиях локальных сетей наметился переход от использования разделяемой среды передачи данных к использованию индивидуальных (иногда виртуальных) каналов связей компьютера с коммуникационными устройствами сети, как это всегда делалось в телефонных сетях. Технологиями, использующими индивидуальные линии связи, являются *100VG-AnyLAN*, *ATM* и коммутирующие модификации традиционных технологий — *switching Ethernet*, *switching Token Ring* и *switching FDDI*. При использовании индивидуальных линий связи в функции сетевого адаптера часто входит установление соединения с коммутатором сети.
- Кодирование последовательности бит кадра последовательностью электрических сигналов при передаче данных и декодирование при их приеме. Кодирование должно обеспечить передачу исходной информацию по линиям связи с определенной полосой пропускания и определенным уровнем помех таким образом, чтобы принимающая сторона смогла распознать с высокой степенью вероятности посланную информацию. Так в ЛВС используются широкополосные кабели, и сетевые адаптеры не используют модуляцию сигнала, необходимую для передачи дискретной информации по узкополосным линиям связи (например, телефонным каналам тональной частоты), а передают данные с помощью импульсных сигналов.
- Преобразование информации из параллельной формы в последовательную и обратно. Эта операция связана с тем, что для упрощения проблемы синхронизации сигналов и удешевления линий связи в вычислительных сетях информация передается в

последовательной форме, бит за битом, а не побайтно, как внутри компьютера.

- Синхронизация битов, байтов и кадров. Для устойчивого приема передаваемой информации необходимо поддержание постоянного синхронизма приемника и передатчика информации. Сетевой адаптер использует для решения этой задачи специальные методы кодирования, не использующие дополнительной шины с тактовыми синхросигналами. Эти методы обеспечивают периодическое изменение состояния передаваемого сигнала, которое используется тактовым генератором приемника для подстройки синхронизма. Кроме синхронизации на уровне битов, сетевой адаптер решает задачу синхронизации и на уровне байтов, и на уровне кадров.

Сетевые адаптеры различаются *по типу сетевой технологии*, принятой в сети — *Ethernet, Token Ring*, *FDDI* и т.п. Как правило, конкретная модель сетевого адаптера работает по определенной сетевой технологии. В связи с тем, что для каждой технологии в настоящее время имеется возможность использования различных сред передачи данных, то сетевой адаптер может поддерживать как одну, так и одновременно несколько сред. В случае, когда сетевой адаптер поддерживает только одну среду передачи данных, а необходимо использовать другую, применяются *трансиверы и конверторы*.

Трансивер (***приемопередатчик, transmitter-receiver***) — это часть сетевого адаптера, его оконечное устройство, выходящее на кабель. В некоторых стандартах, например, работающих на толстом коаксиальном кабеле, трансивер располагался непосредственно на кабеле и связывался с остальной частью адаптера, располагавшейся внутри компьютера, с помощью интерфейса *AUI* (*attachment unit interface*). В настоящее время применяются стандарты сетевых адаптеров (да и другие коммуникационных устройств) с портом *AUI*, к которому можно присоединить трансивер для требуемой среды.

Вместо подбора подходящего трансивера можно использовать ***конвертор***, который может согласовать выход приемопередатчика, предназначенного для одной среды, с другой средой передачи данных (например, выход на витую пару преобразуется в выход на коаксиальный кабель).

7.4 Физическая структуризация локальной сети. Повторители и концентраторы

Для построения простейшей односегментной сети достаточно иметь сетевые адаптеры и кабель подходящего типа. Но даже в этом простом случае часто используются дополнительные устройства — *повторители* сигналов, позволяющие преодолеть ограничения на максимальную длину кабельного сегмента.

Основная функция ***повторителя*** (***repeater***), как это следует из его названия — повторение сигналов, поступающих на один из его портов, на всех остальных портах (*Ethernet*) или на следующем в логическом кольце порте (*Token Ring, FDDI*) синхронно с сигналами-оригиналами. Повторитель так же улучшает электрические характеристики сигналов и их синхронность, и за счет этого появляется возможность увеличивать общую длину кабеля между самыми удаленными в сети станциями.

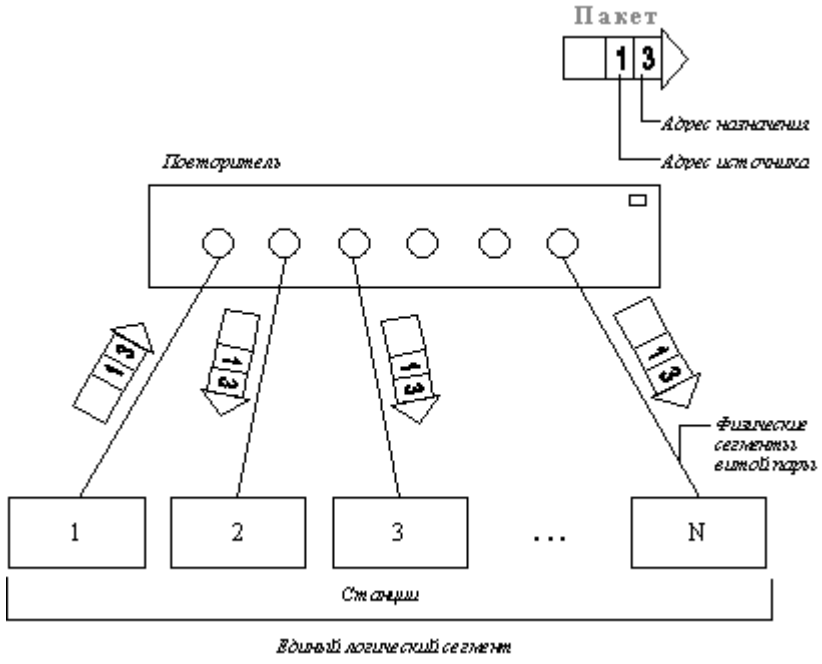


Рис 7.2 — Повторитель Ethernet синхронно повторяет биты кадра на всех своих портах

Многопортовый повторитель часто называют ***концентратором*** (***hub, concentrator***), что отражает тот факт, что данное устройство реализует не только функцию повторения сигналов, но и концентрирует в одном центральном устройстве функции

объединения компьютеров в сеть. Практически во всех современных сетевых стандартах концентратор является необходимым элементом сети, соединяющим отдельные компьютеры в сеть.

Отрезки кабеля, соединяющие два компьютера или какие либо два других сетевых устройства, называются *физическими сегментами*. Таким образом, концентраторы и повторители, которые используются для добавления новых физических сегментов, являются средством физической структуризации сети.

Концентраторы образуют из отдельных физических отрезков кабеля общую среду передачи данных — *логический сегмент*. Логический сегмент также называют ***доменом коллизий***, поскольку при попытке одновременной передачи данных любых двух компьютеров этого сегмента, хотя бы и принадлежащих разным физическим сегментам, возникает блокировка передающей среды. Следует особо подчеркнуть, что какую бы сложную структуру не образовывали концентраторы, например, путем иерархического соединения, все компьютеры, подключенные к ним, образуют единый логический сегмент, в котором любая пара *коллизирующих* компьютеров полностью блокирует возможность обмена данными для других компьютеров.

Использование устройств, централизующих соединения между отдельными сетевыми узлами, позволяет улучшить управляемость сети и ее эксплуатационные характеристики (модифицируемость, ремонтопригодность и т.п.). С этой целью разработчики концентраторов часто встраивают в свои устройства, кроме основной функции повторителя, ряд вспомогательных функций, весьма полезных для улучшения качества сети, наиболее часто встречаются следующие:

- объединение сегментов с различными физическими средами (коаксиальный кабель, витая пара и оптоволоконный кабель) в единый логический сегмент;
- автосегментация портов — автоматическое отключение порта при его некорректном поведении (повреждение кабеля, интенсивная генерация пакетов ошибочной длины и т.п.);
- поддержка между концентраторами резервных связей, которые используются при отказе основных;
- защита передаваемых по сети данных от несанкционированного доступа (например, путем искажения поля данных в кад-

рах, повторяемых на портах, не содержащих компьютера с адресом назначения).

Концентратор — многопортовый повторитель сети с *автосегментацией*. Все порты концентратора равноправны. Получив сигнал от одной из подключенных к нему станций, концентратор транслирует его на все свои активные порты. При этом если на каком-либо из портов обнаружена неисправность, то этот порт автоматически отключается (сегментируется), а после ее устранения снова делается активным. Обработка коллизий и текущий контроль состояния каналов связи обычно осуществляется самим концентратором. Автосегментация необходима для повышения надежности сети. Ведь *повторитель*, заставляющий на практике применять звездообразную кабельную топологию, находится в рамках стандарта *IEEE 802.3* и тем самым обязан обеспечивать соединение типа *моноканал*.

Назначение концентраторов — объединение отдельных рабочих мест в ***рабочую группу*** в составе локальной сети. Для рабочей группы характерны следующие признаки:

- определенная территориальная сосредоточенность;
- коллектив пользователей рабочей группы решает сходные задачи, использует однотипное программное обеспечение и общие информационные базы;
- в пределах рабочей группы существуют общие требования по обеспечению безопасности и надежности;
- происходит одинаковое воздействие внешних источников возмущений (климатических, электромагнитных и т.п.);
- совместно используются высокопроизводительные периферийные устройства;
- обычно содержат свои локальные сервера;
- нередко территориально расположенные на территории рабочей группы.

В связи с тем, что концентраторы работают на физическом уровне, они не чувствительны к протоколам верхних уровней. Результатом этого является возможность совместного использования различных операционных систем со своими протоколами. Есть, правда, определенное *давление* на хозяина сети при использовании программ управления: управляющие программы, как правило, используют для связи с SNMP оборудованием протокол

IP. Поэтому в части управления сетью приходится использовать только этот протоколы и соответственно операционные оболочки на станциях управления сетью. Но это не очень серьезное давление, ибо протокол IP является, в настоящее время, самым популярным.

Все концентраторы обладают следующими характерными эксплуатационными признаками:

- оснащены светодиодными индикаторами, указывающими состояние портов (*Port Status*), наличие коллизий (*Collisions*), активность канала передачи (*Activity*), наличие неисправности (*Fault*) и наличие питания (*Power*), что обеспечивает быстрый контроль состояния всего концентратора и диагностику неисправностей;
- при включении электропитания выполняют процедуру самотестирования, а в процессе работы — функцию самодиагностики;
- имеют стандартный размер по ширине — 19";
- обеспечивают автосегментацию портов для изоляции неисправных портов и улучшения сохранности сети (*network integrity*);
- обнаруживают ошибку полярности при использовании кабеля на витой паре и автоматически переключают полярность для устранения ошибки монтажа;
- поддерживают конфигурации с применением нескольких концентраторов, соединенных друг с другом либо посредством специальных кабелей и *stack*-портов, либо тонкой коаксиальной магистрали, включенной между портами *BNC*, либо посредством оптоволоконного или толстого коаксиального кабеля подключенного через соответствующие трансиверы к порту *AUI*, либо посредством *UTP* кабелей, подключенных между портами концентраторов;
- поддерживают речевую связь и передачу данных через один и тот же кабельный жгут;
- прозрачны для программных средств сетевой операционной системы;
- могут быть смонтированы и введены в действие в течении нескольких минут.

Концентраторы подразделяются на следующие основные классы:

- ***Концентраторы начального уровня*** — 8-ми, 5-ти, реже 12-ти и 16-ти портовые концентраторы. Часто имеют дополнительный *BNC*, реже *AUI* порт. Не обеспечивает возможности управления ни через консольный порт (в виду его отсутствия), ни по сети (по причине отсутствия *SNMP* модуля). Являются простым и дешевым решением для организации рабочей группы небольшого размера.
- ***Концентраторы среднего класса*** — 12-ти, 16-ти, 24-х портовые концентраторы. Имеют консольный порт, часто дополнительные *BNC* и *AUI* порты. Этот тип концентраторов предоставляет возможности для внеполосного управления сетью (*out-of-band management*) через консольный порт *RS-232* под управлением какой-либо стандартной терминальной программы, что дает возможность конфигурировать другие порты и считывать статистические данные концентратора. Этот тип концентраторов используется для построения сетей в диапазоне от малых до средних.
- ***SNMP-управляемые концентраторы*** — 12-ти, 16-ти, 24-х и 48-ми портовые концентраторы. Их отличает не только наличие консольного порта *RS-232* для управления, но и возможность осуществления управление и сбор статистики по сети используя протоколы *SNMP/IP* или *IPX*. Владельцу подобного *hub*-а становятся доступными следующие сбор статистики на узлах сети (концентраторах), ее первичная обработка и анализ: идентифицируются главные источники сообщений (*top talkers*), наиболее активные пользователи (*heavy users*), источники ошибок и коммуникационные пары (*communications pairs*). Эти типы концентраторов целесообразно применять для построения *LAN*-сетей в диапазоне от средних и выше.
- ***BNC-концентраторы или концентраторы ThinLAN*** — многопортовые повторители для тонких коаксиальных кабелей, используемых в сетях стандартов ***10Base2***. Они имеют в своем составе порты *BNC* и, как правило, один порт *AUI*, часто поддерживают *SNMP* протоколы. Они, как и *hub*-ы ***10Base-T***, сегментируют порты (отключая при этом не одну станцию, а абонентов всего луча) и транслируют входящие пакеты во все порты. На каждый *BNC*-порт распространяются все те же ограничения, что

и на фрагмент сети стандарта *10Base-2*: поддерживается работа сегментов тонкого коаксиального кабеля протяженностью до 185 м на каждый порт, обеспечивается до 30 сетевых соединений на сегмент, включая *«пустые Т-коннекторы»*. Если произойдет нарушение целостности кабельного сегмента, этот сегмент исключается из работы, но остальная часть концентратора будет продолжать функционировать. Сфера применения концентраторов данного типа — модернизация старых сетей стандарта ***10Base2*** с целью повышения их надежности, модернизация сетей, достигших ограничений на применение повторителей и не требующих частых изменений.

- ***10/100Hub***-ы появились в последнее время. Упомянутым устройствам присущ серьезный недостаток: концентраторы данного типа не умеют буферизировать пакеты, а следовательно, не умеют согласовывать разные скорости. Поэтому, если к такому hub-у подключена хотя бы одна станция стандарта ***10Base-T***, то все порты будут работать на 10 скорости.

Redundant link. Концентраторы среднего класса и SNMP-управляемые концентраторы поддерживают одну избыточную связь (***redundant link***) на каждый концентратор для создания резервных связей (***back up link***) между любыми двумя концентраторами. Это обеспечивает отказоустойчивость сети на аппаратном уровне. Резервная связь представляет собой отдельный кабель, смонтированный между двумя концентраторами. Используя консольный порт концентратора, надо просто задать конфигурацию основного канала связи и резервного канала связи одного из концентраторов. Резервный канал связи автоматически деблокируется при отказе основного канала связи двух концентраторов. После устранения неисправности на основном кабельном сегменте, основная связь автоматически не возобновляет работу. Для возобновления работы главной связи придется использовать консоль концентратора или нажать кнопку *Reset* на концентраторе.

Связной бит у концентраторов представляет собой периодический импульс длительностью 100 нс, посылаемый через каждые 16 мс. Он не влияет на трафик сети. Связной бит посылается в тот период, когда сеть не передает данные. Эта функция осуществляет текущий контроль сохранности *UTP* канала. Данную функцию следует использовать во всех возможных случаях и

блокировать ее только тогда, когда к порту концентратора подсоединяется устройство, не поддерживающее ее.

Обеспечение безопасности в сетях, построенных с использованием концентраторов, довольно неблагодарное занятие, т.к. повторители по определению является широковещательным устройством. Но, при необходимости, могут быть доступны следующие средства:

- блокирование неиспользуемых портов,
- установка пароля на консольный порт,
- установка шифрования информации на каждом из портов (некоторые модели имеют эту возможность).

7.5 Логическая структуризация сети. Мосты и коммутаторы

Несмотря на появление новых дополнительных возможностей, основной функцией концентраторов остается передача пакетов по общей разделяемой среде. Коллективное использование многими компьютерами общей кабельной системы в режиме разделения времени приводит к существенному снижению производительности сети при интенсивном трафике. Общая среда перестает справляться с потоком передаваемых кадров и в сети возникает очередь компьютеров, ожидающих доступа. Это явление характерно для всех технологий, использующих разделяемые среды передачи данных, независимо от используемых алгоритмов доступа (хотя наиболее страдают от перегрузок трафика сети *Ethernet* с методом случайного доступа к среде).

Поэтому сети, построенные на основе концентраторов, не могут расширяться в требуемых пределах — при определенном количестве компьютеров в сети или при появлении новых приложений всегда происходит насыщение передающей среды, и задержки в ее работе становятся недопустимыми. Эта проблема может быть решена путем логической структуризации сети с помощью *мостов, коммутаторов и маршрутизаторов*.

Мост (***bridge***), а также его быстродействующий функциональный аналог — ***коммутатор*** (***switching hub***), делит общую среду передачи данных на логические сегменты. Логический сегмент образуется путем объединения нескольких физических сег-

ментов с помощью одного или нескольких концентраторов. Каждый логический сегмент подключается к отдельному порту моста/коммутатора (рис. 7.3). При поступлении кадра на какой-либо из портов мост/коммутатор повторяет этот кадр, но не на всех портах, как это делает концентратор, а только на том порту, к которому подключен сегмент, содержащий компьютер-адресат.

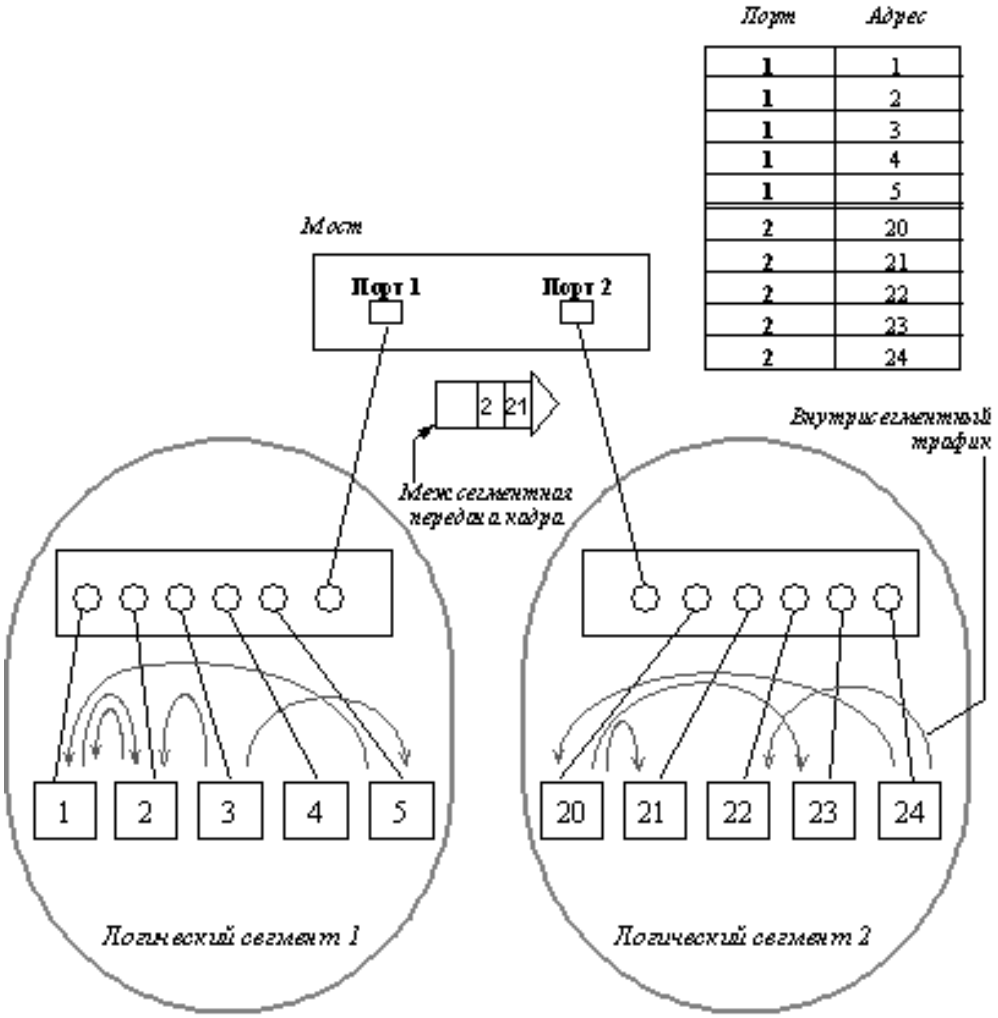


Рис. 7.3 — Разделение сети на логические сегменты посредством моста

Разница между мостом и коммутатором состоит в том, что мост в каждый момент времени может осуществлять передачу кадров только между одной парой портов, а коммутатор одновременно поддерживает потоки данных между всеми своими портами. Другими словами, мост передает кадры последовательно, а коммутатор параллельно. Следует отметить, что в последнее

время локальные мосты полностью вытеснены коммутаторами. Мосты используются только для связи локальных сетей с глобальными, то есть как средства удаленного доступа, поскольку в этом случае необходимость в параллельной передаче между несколькими парами портов просто не возникает.

При работе коммутатора среда передачи данных каждого логического сегмента остается общей только для тех компьютеров, которые подключены к этому сегменту непосредственно. Коммутатор осуществляет связь сред передачи данных различных логических сегментов. Он передает кадры между логическими сегментами только при необходимости, то есть только тогда, когда взаимодействующие компьютеры находятся в разных сегментах.

Деление сети на логические сегменты улучшает производительность сети, если в сети имеются группы компьютеров, преимущественно обменивающиеся информацией между собой. Если же таких групп нет, то введение в сеть коммутаторов может только ухудшить общую производительность сети, так как принятие решения о том, нужно ли передавать пакет из одного сегмента в другой, требует дополнительного времени. Однако даже в сети средних размеров такие группы, как правило, имеются. Поэтому разделение ее на логические сегменты дает выигрыш в производительности — трафик локализуется в пределах групп, и нагрузка на их разделяемые кабельные системы существенно уменьшается.

Коммутаторы принимают решение о том, на какой порт нужно передать кадр, анализируя адрес назначения, помещенный в кадре, а также на основании информации о принадлежности того или иного компьютера определенному сегменту, подключенному к одному из портов коммутатора, то есть на основании информации о конфигурации сети. Для того чтобы собрать и обработать информацию о конфигурации подключенных к нему сегментов, коммутатор должен пройти стадию *«обучения»*, то есть самостоятельно проделать некоторую предварительную работу по изучению проходящего через него трафика. Определение принадлежности компьютеров сегментам возможно за счет наличия в кадре не только адреса назначения, но и адреса источника, сгенерировавшего пакет. Используя информацию об адресе источника,

коммутатор устанавливает соответствие между номерами портов и адресами компьютеров. В процессе изучения сети мост/коммутатор просто передает появляющиеся на входах его портов кадры на все остальные порты, работая некоторое время повторителем. После того, как мост/коммутатор узнает о принадлежности адресов сегментам, он начинает передавать кадры между портами только в случае межсегментной передачи. Если, уже после завершения обучения, на входе коммутатора вдруг появится кадр с неизвестным адресом назначения, то этот кадр будет повторен на всех портах.

Мосты/коммутаторы, работающие описанным способом, обычно называются *прозрачными* (*transparent*), поскольку появление таких мостов/коммутаторов в сети совершенно не заметно для ее конечных узлов. Это позволяет не изменять их программное обеспечение при переходе от простых конфигураций, использующих только концентраторы, к более сложным, сегментированным.

Существует и другой класс мостов/коммутаторов, передающих кадры между сегментами на основе полной информации о межсегментном маршруте. Эту информацию записывает в кадр станция-источник кадра, поэтому говорят, что такие устройства реализуют ***алгоритм маршрутизации от источника*** (***source routing***). При использовании мостов/коммутаторов с маршрутизацией от источника конечные узлы должны быть в курсе деления сети на сегменты и сетевые адаптеры, в этом случае должны в своем программном обеспечении иметь компонент, занимающийся выбором маршрута кадров.

За простоту принципа работы прозрачного моста/коммутатора приходится расплачиваться ограничениями на топологию сети, построенной с использованием устройств данного типа — такие сети не должны иметь замкнутых маршрутов — *петель*. Мост/коммутатор не может правильно работать в сети с петлями, при этом сеть засоряется зацикливающимися пакетами и ее производительность снижается.

Для автоматического распознавания петель в конфигурации сети разработан ***алгоритм покрывающего дерева*** (***Spanning Tree Algorithm, STA***). Этот алгоритм позволяет мостам/коммутаторам адаптивно строить дерево связей, когда они изучают топологию

связей сегментов с помощью специальных тестовых кадров. При обнаружении замкнутых контуров некоторые связи объявляются резервными. Мост/коммутатор может использовать резервную связь только при отказе какой-либо основной. В результате сети, построенные на основе мостов/коммутаторов, поддерживающих алгоритм покрывающего дерева, обладают некоторым запасом надежности, но повысить производительность за счет использования нескольких параллельных связей в таких сетях нельзя.

В отличие от мостов, ряд коммутаторов не помещает все приходящие пакеты в буфер. Это происходит лишь тогда, когда надо согласовать скорости передачи, или адрес назначения не содержится в адресной таблице, или когда порт, куда должен быть направлен пакет, занят, а коммутирует пакеты «на лету». Коммутатор лишь анализирует адрес назначения в заголовке пакета и, сверившись с адресной таблицей, тут же (время задержки около 30—40 микросекунд) направляет этот пакет в соответствующий порт. Таким образом, когда пакет еще целиком не прошел через входной порт, его заголовок уже передается через выходной. Типичные коммутаторы работают по алгоритму «устаревания адресов». Это означает, что, если по истечении определенного промежутка времени, не было обращений по этому адресу, то он удаляется из адресной таблицы.

Коммутаторы поддерживают при соединении друг с другом режим полного дуплекса. В таком режиме данные передаются и принимаются одновременно, что невозможно в обычных сетях *Ethernet*. При этом скорость передачи данных повышается в два раза, а при соединении нескольких коммутаторов можно добиться и большей пиковой производительности.

Несколько особняком стоят коммутаторы серии *SmartSwitch* фирмы *Cabletron Systems*. Эта серия коммутаторов поддерживает технологию SNS, которая ранее называлась SFS, Одна из ее особенностей заключается в том, что коммутаторы, составляющие сеть, хранят таблицу адресов *«вечно»* и обмениваются ими друг с другом, могут выгружать их на специальный сервер. Это позволяет не только сократить время прохождения пакета по сети, но и решить ряд специфических проблем, особенно связанных с безопасностью.

7.6 Маршрутизаторы

Маршрутизатор (***router***) позволяет обрабатывать в сети избыточные связи, образующие петли. Он справляется с этой задачей за счет того, что принимает решение о передаче пакетов на основании более полной информации о графе связей в сети, чем мост или коммутатор. Маршрутизатор имеет в своем распоряжении базу топологической информации, которая говорит ему, например, о том, между какими подсетями общей сети имеются связи и в каком состоянии (работоспособном или нет) они находятся. Имея такую карту сети, маршрутизатор может выбрать один из нескольких возможных маршрутов доставки пакета адресату. В данном случае под маршрутом понимают последовательность прохождения пакетом маршрутизаторов.

В отличие от моста/коммутатора, который не знает, как связаны сегменты друг с другом за пределами его портов, маршрутизатор видит всю картину связей подсетей друг с другом, поэтому он может выбрать правильный маршрут и при наличии нескольких альтернативных маршрутов. Решение о выборе того или иного маршрута принимается каждым маршрутизатором, через который проходит сообщение.

Для того чтобы составить карту связей в сети, маршрутизаторы обмениваются специальными служебными сообщениями, в которых содержится информация о тех связях между подсетями, о которых они знают (эти подсети подключены к ним непосредственно или же они узнали эту информацию от других маршрутизаторов).

Построение графа связей между подсетями и выбор оптимального по какому-либо критерию маршрута на этом графе представляют собой сложную задачу. При этом могут использоваться разные критерии выбора маршрута — наименьшее количество промежуточных узлов, время, стоимость или надежность передачи данных.

Маршрутизаторы позволяют объединять сети с различными принципами организации в единую сеть, которая в этом случае часто называется *интрасеть* (*intranet*). Название интрасеть подчеркивает ту особенность, что образованное с помощью маршрутизаторов объединение компьютеров представляет собой сово-

купность нескольких сетей, сохраняющих большую степень автономности, чем несколько логических сегментов одной сети. В каждой из сетей, образующих интрасеть, сохраняются присущие им принципы адресации узлов и протоколы обмена информацией. Поэтому маршрутизаторы могут объединять не только локальные сети с различной технологией, но и локальные сети с глобальными.

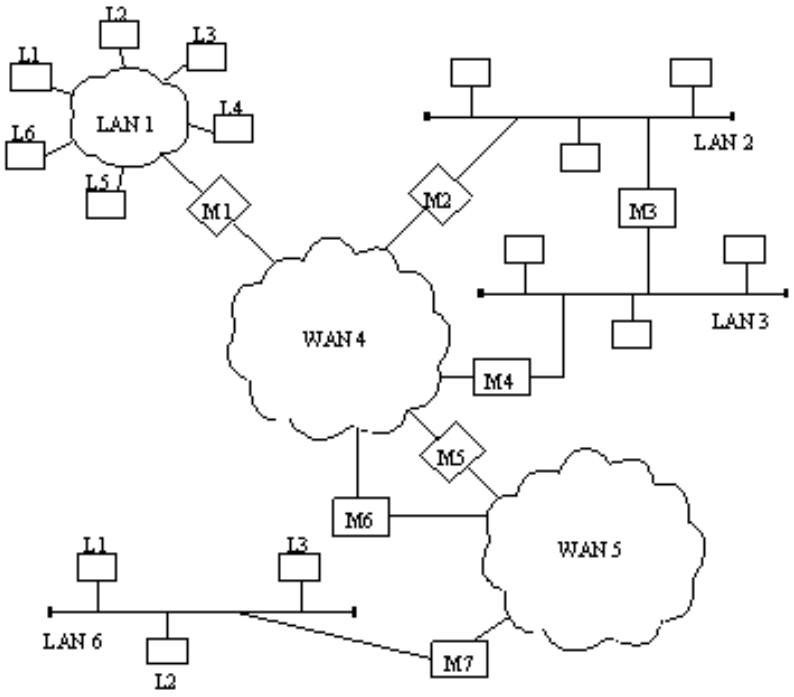


Рис. 7.4 — Структура интерсети,
построенной на основе маршрутизаторов

Маршрутизаторы не только объединяют сети, но и надежно защищают их друг от друга. Причем эта изоляция осуществляется гораздо проще и надежнее, чем с помощью мостов/коммутаторов. Например, при поступлении кадра с неправильным адресом мост/коммутатор обязан повторить его на всех своих портах, что делает сеть незащищенной от некорректно работающего узла. Маршрутизатор же в таком случае просто отказывается передавать *«неправильный»* пакет дальше, изолируя дефектный узел от остальной сети.

Кроме того, маршрутизатор предоставляет администратору удобные средства фильтрации потока сообщений за счет того,

что сам распознает многие поля служебной информации в пакете и позволяет их именовать понятным администратору образом. Нужно заметить, что некоторые мосты/коммутаторы также способны выполнять функции гибкой фильтрации, но задавать условия фильтрации администратор сети должен сам в двоичном формате, что достаточно сложно.

Кроме фильтрации, маршрутизатор может обеспечивать приоритетный порядок обслуживания буферизованных пакетов, когда на основании некоторых признаков пакетам предоставляются преимущества при выборе из очереди. Различные типы *router*-ов отличаются количеством и типами своих портов, что собственно и определяет места их использования. Маршрутизаторы, например, могут быть использованы в локальной сети *Ethernet* для эффективного управления трафиком при наличии большого числа сегментов сети, для соединения сети типа *Ethernet* с сетями другого типа, например *Token Ring*, *FDDI*, а также для обеспечения выходов локальных сетей на глобальную сеть.

Маршрутизаторы не просто осуществляют связь разных типов сетей и обеспечивают доступ к глобальной сети, но и могут управлять трафиком на основе протокола сетевого уровня (третьего в модели OSI), то есть на более высоком уровне по сравнению с коммутаторами. Необходимость в таком управлении возникает при усложнении топологии сети и росте числа ее узлов, если в сети появляются избыточные пути (при поддержке протокола IEEE 802.1 *Spanning Tree*), когда нужно решать задачу максимально эффективной и быстрой доставки отправленного пакета по назначению. При этом существует два основных алгоритма определения наиболее выгодного пути и способа доставки данных: *RIP* и *OSPF*. При использовании протокола маршрутизации ***RIP***, основным критерием выбора наиболее эффективного пути является минимальное число сетевых устройств между узлами (*hops*). Этот протокол минимально загружает процессор маршрутизатора и предельно упрощает процесс конфигурирования, но он не рационально управляет трафиком. При использовании ***OSPF*** наилучший путь выбирается не только с точки зрения минимизации числа *«хопов»*, но и с учетом других критериев: производительности сети, задержки при передаче пакета и т.д.

Сети большого размера, чувствительные к перегрузке трафика и базирующиеся на сложной маршрутизирующей аппаратуре, требуют использования протокола *OSPF*. Реализации этого протокола возможна только на маршрутизаторах с достаточно мощным процессором, т.к. его реализация требует существенных вычислительных затрат.

Маршрутизация в сетях, как правило, осуществляться с применением пяти популярных сетевых протоколов — *TCP/IP*, *Novell IPX*, AppleTalk *II*, *DECnet Phase IV* и *Xerox XNS*. Если маршрутизатору попадается пакет неизвестного формата, он начинает с ним работать как обучающийся мост. Кроме того, маршрутизатор обеспечивает более высокий уровень локализации трафика, чем мост, предоставляя возможность фильтрации широковещательных пакетов, а также пакетов с неизвестными адресами назначения, поскольку умеет обрабатывать адрес сети.

Современные маршрутизаторы обладают следующими свойствами:

- поддерживают коммутацию и высокоскоростную маршрутизацию уровня 3;
- поддерживают передовые технологии передачи данных, такие как *Fast Ethernet*, *Gigabit Ethernet* и *ATM*;
- поддерживают технологии *ATM* с использованием скоростей до 622 Мбит/с;
- поддерживают одновременно разные типы кабельных соединений (медные, оптические и их разновидности);
- поддерживают *WAN*-соединения, включая поддержку технологий *PPP*, *Frame Relay*, *HSSI*, *SONET* и др.;
- поддерживают технологию коммутации уровня 4 (*Layer 4 Switching*), использующую не только информация об адресах отправителя и получателя, но и информацию о типах приложений, с которыми работают пользователи сети;
- обеспечивают возможность использования механизма *«сервис по запросу»* (*Quality of Service — QoS*), позволяющего назначать приоритеты тем или иным ресурсам в сети и обеспечивать передачу трафика в соответствии со схемой приоритетов;
- позволяют управлять шириной полосы пропускания для каждого типа трафика;

- поддерживают основные протоколы маршрутизации, такие как *IP RIP1, IP RIP2, OSPF, BGP-4, IPX RIP/SAP*, а также протоколы *IGMP, DVMPR, PIM-DM, PIM-SM, RSVP*;
- поддерживают несколько *IP* сетей одновременно;
- поддерживают протоколы *SNMP, RMON* и *RMON 2*, что дает возможность осуществлять управление работой устройств, их конфигурированием со станции сетевого управления, а также осуществлять сбор и последующий анализ статистики как о работе устройства в целом, так и его интерфейсных модулей;
- поддерживать как одноадресный (*unicast*), так и многоадресный (*multicast*) трафик;

Маршрутизатор является сложным интеллектуальным устройством, построенным на базе одного, а иногда и нескольких мощных процессоров. Такой специализированный мультипроцессор работает, как правило, под управлением специализированной операционной системы.

7.7 Модульные многофункциональные концентраторы

При построении сложной сети могут быть полезны все типы коммуникационных устройств: и концентраторы, и мосты, и коммутаторы, и маршрутизаторы (сетевые адаптеры исключены из этого списка потому, что они необходимы всегда). Чаще всего отдельное коммуникационное устройство выполняет только одну основную функцию, представляя собой либо повторитель, либо мост, либо коммутатор, либо маршрутизатор. Но это не всегда удобно, так как в некоторых случаях более рационально иметь в одном корпусе многофункциональное устройство, которое может сочетать эти базовые функции и тем самым позволяет разработчику сети использовать его более гибко.

В идеале можно представить себе универсальное коммуникационное устройство, имеющее достаточное количество портов для подключения сетевых адаптеров, которые объединяются в группы с программируемыми функциями взаимоотношений между собой (по алгоритму повторителя, коммутатора или маршрутизатора). Однако известно, что всякая универсализация всегда вредит качеству выполнения узких специальных функций и, воз-

можно поэтому, на современном уровне развития техники такое полностью универсальное устройство пока не появилось, хотя отдельное совмещение функций в одном устройстве иногда выполняется.

Так маршрутизаторы часто могут работать и в качестве мостов, в зависимости от того, как сконфигурировано администратором их программное обеспечение. А вот функции повторителя требуют высокого быстродействия, которое может быть достигнуто только на сугубо аппаратном уровне. Поэтому функции повторителя не объединяются с функциями моста или маршрутизатора.

Для совмещения функций может быть использован другой подход. В специальных устройствах — ***модульных концентраторах*** — отдельные компоненты, выполняющие одну из трех описанных основных функций, реализованы в виде модулей, устанавливаемых в общем корпусе. При этом межмодульные связи организуются не внешним образом, как это делается, когда модули представляют собой отдельные устройства, а по внутренним шинам единого устройства. В зависимости от комплектации модульный многофункциональный концентратор может сочетать функции и повторителя (причем различных технологий), и моста, и коммутатора, и маршрутизатора, а может выполнять и только одну из них.

7.8 Функциональное соответствие видов коммуникационного оборудования уровням модели OSI

Лучшим способом для понимания отличий между сетевыми адаптерами, повторителями, мостами/коммутаторами и маршрутизаторами является рассмотрение их работы в терминах модели OSI. Соотношение между функциями этих устройств и уровнями модели OSI показано на рис. 7.5.

Повторитель, который регенерирует сигналы, за счет чего позволяет увеличивать длину сети, работает на физическом уровне.

Сетевой адаптер работает на физическом и канальном уровнях. К физическому уровню относится та часть функций сетевого адаптера, которая связана с приемом и передачей сигналов по линии связи, а получение доступа к разделяемой среде передачи,

распознавание MAC-адреса компьютера — это уже функция канального уровня.

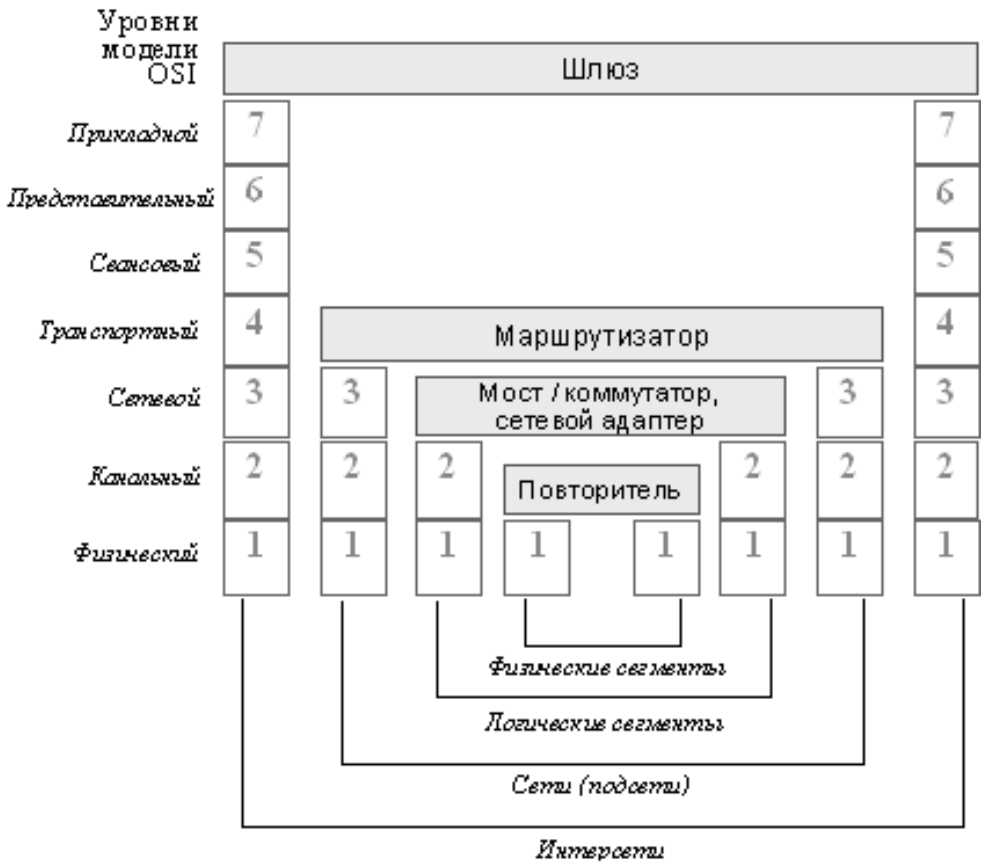


Рис. 7.5 — Соответствие функций коммуникационного оборудования модели OSI

Мосты выполняют большую часть своей работы на канальном уровне. Для них сеть представляется набором MAC-адресов устройств. Они извлекают эти адреса из заголовков, добавленных к пакетам на канальном уровне, и используют их во время обработки пакетов для принятия решения о том, на какой порт отправить тот или иной пакет. Мосты не имеют доступа к информации об адресах сетей, относящейся к более высокому уровню. Поэтому они ограничены в принятии решений о возможных путях или маршрутах перемещения пакетов по сети.

Маршрутизаторы работают на сетевом уровне модели OSI. Для маршрутизаторов сеть — это набор сетевых адресов устройств и множество сетевых путей. Маршрутизаторы анализируют все возможные пути между любыми двумя узлами сети и

выбирают самый короткий из них. При выборе могут приниматься во внимание и другие факторы, например, состояние промежуточных узлов и линий связи, пропускная способность линий или стоимость передачи данных.

Для того, чтобы маршрутизатор мог выполнять возложенные на него функции ему должна быть доступна более развернутая информация о сети, нежели та, которая доступна мосту. В заголовке пакета сетевого уровня кроме сетевого адреса имеются данные, например, о критерии, который должен быть использован при выборе маршрута, о времени жизни пакета в сети, о том, какому протоколу верхнего уровня принадлежит пакет.

Благодаря использованию дополнительной информации, маршрутизатор может осуществлять больше операций с пакетами, чем мост/коммутатор. Поэтому и программное обеспечение, необходимое для работы маршрутизатора, является более сложным.

На рис. 7.5 показан еще один тип коммуникационных устройств — ***шлюз***, который может работать на любом уровне модели OSI. ***Шлюз*** (***gateway***) — это устройство, выполняющее трансляцию протоколов. Шлюз размещается между взаимодействующими сетями и служит посредником, переводящим сообщения, поступающие из одной сети, в формат другой сети. Шлюз может быть реализован как чисто программными средствами, установленными на обычном компьютере, так и на базе специализированного компьютера. Трансляция одного стека протоколов в другой представляет собой сложную интеллектуальную задачу, требующую максимально полной информации о сети, поэтому шлюз использует заголовки всех транслируемых протоколов.

7.9 Магистральные средства и средства удаленного доступа

Магистральные средства используются для образования одноранговых связей между крупными локальными вычислительными сетями. Магистральные территориальные сети должны обеспечивать высокую пропускную способность, так как на магистрали объединяются потоки большого количества подсетей. Кроме того, магистральные сети должны быть постоянно доступны, то есть поддерживать очень высоким коэффициент готовно-

сти. Ввиду особой важности магистральных средств они часто характеризуются высокой стоимостью. К магистральным средствам, как правило, не предъявляются требования поддержания разветвленной инфраструктуры доступа.

Обычно в качестве магистральных средств используются цифровые выделенные каналы со скоростями от 2 Мбит/с до 622 Мбит/с, сети с коммутацией пакетов *frame relay*, *ATM*, *X.25* или *TCP/IP*.

Под ***средствами удаленного доступа*** понимаются средства, необходимые для связи небольших локальных сетей и даже удаленных отдельных компьютеров с центральной *WAN*. В качестве отдельных удаленных узлов могут также выступать банкоматы или кассовые аппараты, требующие доступ к центральной базе данных о легальных клиентах банка, пластиковые карточки которых необходимо авторизовать на месте. Банкоматы или кассовые аппараты обычно рассчитаны на взаимодействие с центральным компьютером по сети *X.25*, которая в свое время специально разрабатывалась как сеть для удаленного доступа неинтеллектуального терминального оборудования к центральному компьютеру.

К средствам удаленного доступа предъявляются требования, существенно отличающиеся от требований к магистральным средствам. Так как точек удаленного доступа у предприятия может быть очень много, то одним из основных требований является наличие разветвленной инфраструктуры доступа, которая может использоваться пользователями, как при работе дома, так и в командировках. Кроме того, стоимость удаленного доступа должна быть умеренной, чтобы экономически оправдать затраты на подключение десятков или сотен удаленных абонентов. При этом требования к пропускной способности у отдельного компьютера или локальной сети, состоящей из двух-трех клиентов, обычно укладываются в диапазон нескольких десятков Кбит/с.

В качестве транспортных средств удаленного доступа используются телефонные аналоговые сети, сети *ISDN* и реже — сети *frame relay*. Качественный скачок в расширении возможностей удаленного доступа произошел в связи со стремительным ростом популярности и распространенности сети *Internet*. Транспортные услуги *Internet* дешевле, чем услуги междугородных и международных телефонных сетей, а их качество быстро улучшается.

Если какое-либо предприятие не строит свою территориальную сеть, а пользуется услугами общественной сети, то внутренняя структура этой сети его не интересует. Для абонента общественной сети главное — это предоставляемый сетью сервис и четкое определение интерфейса взаимодействия с сетью для того, чтобы его оконечное оборудование данных и аппаратура передачи данных корректно сопрягались с соответствующим оборудованием и программным обеспечением общественной сети.

7.10 Типы устройств доступа к территориальным сетям

Независимо от типа коммутации, используемого в территориальной сети, а также от того, относится ли территориальная сеть к магистральным средствам или к средствам удаленного доступа, все абоненты сети присоединяются к ней с помощью оборудования доступа (***Access Devices***), которое позволяет согласовать протоколы и интерфейсы локальных сетей с протоколами и интерфейсами территориальной сети. Обычно в глобальной сети строго описан и стандартизован интерфейс взаимодействия пользователей с сетью — ***User Network Interface***, ***UNI***. Это необходимо для того, чтобы пользователи могли без проблем подключаться к сети с помощью коммуникационного оборудования любого производителя, который соблюдает стандарт *UNI*.

Устройство доступа — это устройство, которое поддерживает на входе интерфейс локальной сети, а на выходе — требуемый интерфейс *UNI*. Интерфейс между локальной и глобальной сетями может быть реализован устройствами разных типов. В первую очередь эти устройства делятся на устройства:

- аппаратуру передачи данных (Data Circuit-terminating Equipment, DCE);
- оконечное оборудование данных (Data Terminal Equipment, DTE).

Устройства ***DCE*** представляют собой аппаратуру передачи данных по территориальным каналам, работающую на физическом уровне. Устройства этого типа имеют выходные интерфейсы физического уровня, согласованные с территориальным каналом передачи данных. Различают аппаратуру передачи данных по

аналоговым и цифровым каналам. Для передачи данных по аналоговым каналам применяются модемы различных стандартов, а по цифровым — устройства *DSU/CSU*.

DTE — это очень широкий класс устройств, которые непосредственно готовят данные для передачи по глобальной сети. *DTE* представляют собой устройства, работающие на границе между локальными и глобальными сетями и выполняющие протоколы уровней более высоких, чем физический уровень OSI. Устройства *DTE* могут поддерживать только канальные протоколы — такими устройствами являются удаленные мосты, либо протоколы канального и сетевого уровней — тогда они являются маршрутизаторами, а могут поддерживать протоколы всех уровней, включая прикладной — в таком случае их называют ***шлюзами***.

Связь компьютера или маршрутизатора с ***цифровой выделенной линией*** осуществляется с помощью пары устройств, обычно выполненных в одном корпусе или же совмещенных с маршрутизатором. Этими устройствами являются: ***устройство обслуживания данных (Data Service Unit — DSU)*** *и* ***устройство обслуживания канала*** (***Channel Service Unit — CSU***). Устройство обслуживания данных *DSU* преобразует сигналы, поступающие от оконечного оборудования данных *DTE* (обычно по интерфейсу *RS-232* или *HSSI*), в биполярные импульсы интерфейса *G.703*. Устройство обслуживания канала *DSU* также выполняет все временные отсчеты, регенерацию сигнала и выравнивание загрузки канала. Устройство *CSU* выполняет более узкие функции, в основном оно занимается созданием оптимальных условий передачи в линии (выравнивание). Эти устройства, как и модуляторы-демодуляторы, часто обозначаются одним словом *DSU/CSU* (рис. 7.6).

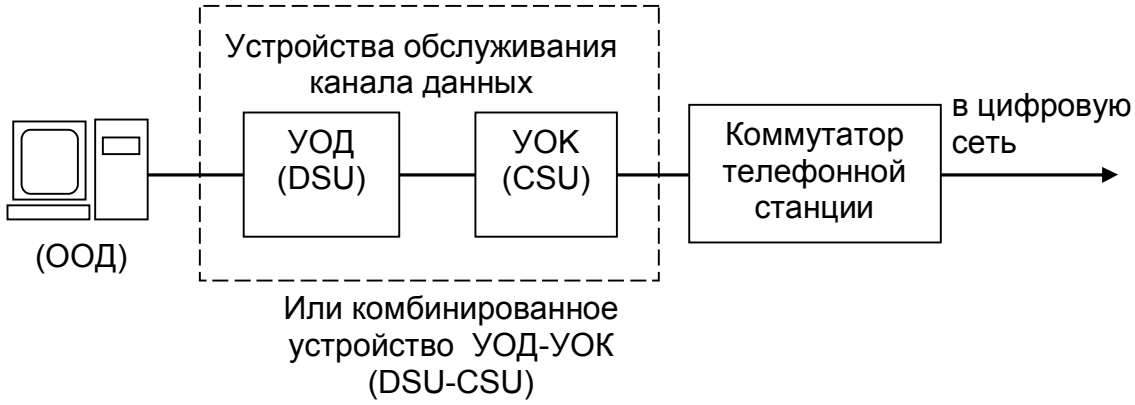


Рис. 7.6 — Связь компьютера с цифровой линией

Оконечное оборудование данных — устройства *DTE* — это устройства, работающие на более высоком уровне, чем физический, которые формируют данные непосредственно для передачи из локальной сети в глобальную. Под названием *DTE* объединяются несколько типов устройств — маршрутизаторы с интерфейсами глобальных сетей, мультиплексоры *«голос — данные»*, устройства доступа к сетям *frame relay* (*FRAD*), устройства доступа к сетям *X.25* (*PAD*), удаленные мосты. Когда к глобальной сети подключается не локальная сеть, а отдельный компьютер, то он при этом сам становится устройством типа *DTE*. *DTE* принимают решения о передаче данных в глобальную сеть, а также выполняют форматирование данных на канальном и сетевом уровнях, а для сопряжения с территориальным каналом используют *DCE*. Такое разделение функций позволяет гибко использовать одно и то же устройство *DTE* для работы с разными глобальными сетями за счет замены только *DCE*.

Маршрутизаторы с интерфейсами глобальных сетей. При передаче данных через глобальную сеть маршрутизаторы работают точно так же, как и при соединении локальных сетей. Если они принимают решение о передаче пакета через глобальную сеть, то упаковывают пакеты принятого в локальных сетях сетевого протокола (например, *IP*) в кадры канального уровня глобальной сети (например, *frame relay*) и отправляют их в соответствии с интерфейсом *UNI* ближайшему коммутатору глобальной сети через устройство *DTE*. Каждый пользовательский интерфейс с глобальной сетью имеет свой собственный адрес в формате, принятом для технологии этой сети. В соответствии с этим адресом коммутаторы глобальной сети передают свои кадры друг другу, пока кадр не дойдет до абонента-получателя. При получении кадра маршрутизатор абонента извлекает из него сетевой пакет и передает его по локальной сети уже в соответствии с ее канальным протоколом.

Когда абонентом глобальной сети является отдельный компьютер, то процедуры интерфейса с сетью реализуются его программным обеспечением, а также устройством *DCE*, подключенным непосредственно к глобальному каналу, в качестве которого обычно выступает *модем*. Иногда компьютер оснащается специальным адаптером (например, адаптером сети *X.25*), который раз-

гружает центральный процессор, выполняя большую часть интерфейсных процедур аппаратно.

Иногда маршрутизаторы оснащаются встроенными устройствами *DCE* — чаще всего такими устройствами являются устройства *DCU/CSU* для цифровых каналов, так как они компактнее, чем аналоговые модемы.

Маршрутизаторы с выходами на глобальные сети характеризуются типом физического интерфейса (*RS-232, RS-422, RS-530, HSSI, SDH*), к которому присоединяется устройство *DCE*, а также поддерживаемыми протоколами территориальных сетей — протоколами коммутации каналов для телефонных сетей или протоколами коммутации пакетов для компьютерных глобальных сетей.

Устройства доступа к сетям frame relay — FRAD (***Frame Relay Access Devices***). Эти устройства представляют собой специализированные маршрутизаторы. Их специализация заключается в том, что среди глобальных интерфейсов они поддерживают только интерфейсы к сетям *frame relay*, а также в усеченности функций маршрутизации — чаще всего такие устройства поддерживают только протоколы *IP* и *IPX*. Появление таких специализированных устройств обусловлено большой популярностью сетей *frame relay*.

Устройства доступа к сетям X.25 — PAD (***Packet Assembler-Disassembler***). Сети *X.25* изначально разрабатывались для связи неинтеллектуальных алфавитно-цифровых терминалов с удаленными компьютерами, поэтому в архитектуру этих сетей были включены специальные устройства — *PAD*'ы, собирающие данные от нескольких медленных асинхронных терминалов в общие пакеты и отсылающие пакеты в сеть.

Удаленные мосты. Эти устройства обычно имеют два интерфейса — один для подключения к локальной сети, а второй — для подключения к глобальной сети. Так как мост работает на канальном уровне и не поддерживает протоколы маршрутизации, то удаленные мосты чаще всего не работают через глобальные сети с коммутацией пакетов, такие как *X.25*, *frame relay* и т.п., так как установление соединения через эти сети требует от моста интеллектуальных способностей устройства третьего уровня. Уда-

ленный мост работает через выделенные каналы или через сеть с коммутацией каналов.

Мультиплексоры «голос — данные» предназначены для совмещения в рамках одной территориальной сети компьютерного и голосового трафиков. Поэтому эти мультиплексоры кроме входных интерфейсов локальных сетей имеют и интерфейсы для подключения телефонов и офисных АТС. Мультиплексоры *«голос — данные»* делятся на две категории в зависимости от типа глобальной сети, на которую они могут работать.

Мультиплексоры *«голос — данные»,* работающие на сети с коммутацией пакетов, упаковывают голосовую информацию в кадры канального протокола такой сети и передают их ближайшему коммутатору точно так же, как и маршрутизаторы. Такой мультиплексор выполняется на базе маршрутизатора, который для голосовых пакетов использует заранее сконфигурированные маршруты. Если глобальная сеть поддерживает приоритеты трафика, то кадрам голосового трафика мультиплексор присваивает наивысший приоритет, чтобы коммутаторы обрабатывали и продвигали их в первую очередь.

Другим типом устройств являются мультиплексоры *«голос — данные»,* работающие на сети с коммутацией каналов или первичные сети выделенных каналов. Эти мультиплексоры нарезают компьютерные пакеты на более мелкие части — например, байты, которые передают в соответствии с техникой мультиплексирования используемой территориальной сети — *FDM* или *TDM*. При использовании *«неделимого»* с точки зрения территориальной сети канала — например, канала 64 Кбит/с цифровой сети или канала тональной частоты аналоговой сети, мультиплексор организует разделение этого канала между голосом и данными нестандартным фирменным способом.

Использование мультиплексоров *«голос — данные»* предполагает на другом конце территориальной сети аналогичного мультиплексора, который выполняет разделение голосового и компьютерного трафика на отдельные потоки.

Модемная связь. Модуляция и демодуляция сигнала для передачи по аналоговым каналам выполняется в устройстве, называемом модемом. Модем выполняет функции аппаратуры окончания канала данных. В качестве оконечного оборудования

обычно выступает компьютер, в котором имеется приемопередатчик — микросхема UART (Universal Asynchronous Reciver/Transmitter). Приемопередатчик подключается к модему через один из последовательных портов компьютера и последовательный интерфейс RS-232C (скорость 9,6 Кбит/с на расстоянии до 15 м). Более высокая скорость (до 1000 Кбит/с на расстоянии до 100 м) обеспечивается интерфейсом RS-422, в котором используется две витые пары проводов с сопротивлениями на концах, образующие сбалансированную линию.

7.11 Серверы удаленного доступа, удаленного управления и терминальные серверы

Существует особый класс устройств, предназначенных для связи удаленных узлов в том случае, когда к сети нужно подключить не другую сеть, а автономный компьютер. В таких случаях в центральной сети устанавливается сервер удаленного доступа, который обслуживает доступ к сети большого числа разрозненных компьютеров.

Обычно, сервер удаленного доступа служит для подключения удаленных клиентов по телефонным сетям — аналоговом или *ISDN*, так как это наиболее распространенные и повсеместно доступные сети. Серверы удаленного доступа обычно имеют большое количество портов для поддержки модемного пула, соединяющего сервер с телефонной городской сетью. Серверы удаленного доступа подразделяются на серверы удаленных узлов, серверы удаленного управления и терминальные серверы.

Серверы удаленных узлов обеспечивают для своих клиентов только транспортный сервис, соединяя их с центральной сетью по протоколам *IP*, *IPX* или *NetBIOS*. В сущности, они выполняют в этом случае роль маршрутизаторов или шлюзов, ориентированных на низкоскоростные модемные соединения.

Серверы удаленного управления, кроме обеспечения транспортного соединения, выполняют и некоторые дополнительные функции — они запускают от имени своих удаленных клиентов приложения на компьютерах центральной сети и эмулируют на экране удаленного компьютера графическую среду этого прило-

жения. Обычно, серверы удаленного управления ориентируются на среду операционных систем персональных компьютеров.

Терминальные серверы выполняют похожие функции, но для многотерминальных операционных систем — Unix, VAX VMS, IBM VM.

8 БЕЗОПАСНОСТЬ КОМПЬЮТЕРНЫХ СЕТЕЙ

Любой компютер в локальной сети, подключенной к Интернет, сегодня подвергается реальной опасности нападения злоумышленника. Определенные уязвимости существуют в программном обеспечении компьютеров и сетевого оборудования, в структуре сетевых протоколов. Знание слабых мест и способов их использования позволяет принять меры противодействия и защититься от атаки хакера. Анализ экспертов показывает, что число таких атак с каждым годом стремительно, — в геометрической прогрессии, — растет. Поэтому защита копьютерной сети сегодня становиться одним из важнейших, ключевых условий нормальной работы ее пользователей.

8.1 Классификация компьютерных атак

Когда мы говорим *компьютерная атака*, мы имеем в виду запуск людьми программ для получения неавторизованного доступа к компьютеру. Формы организации атак весьма разнообразны, но в целом все они принадлежат к одной из следующих категорий:

- ***Удаленное проникновение в компьютер:*** программы, которые получают неавторизованный доступ к другому компьютеру через Интернет (или локальную сеть)
- ***Локальное проникновение в компьютер:*** программы, которые получают неавторизованный доступ к компьютеру, на котором они работают.
- ***Удаленное блокирование компьютера:*** программы, которые через Интернет (или сеть) блокируют работу всего удаленного компьютера или отдельной программы на нем (для восстановления работоспособности чаще всего компьютер надо перезагрузить)
- ***Локальное блокирование компьютера:*** программы, которые блокируют работу компьютера, на котором они работают
- ***Сетевые сканеры:*** программы, которые осуществляют сбор информации о сети, чтобы определить, какие из компьютеров и программ, работающих на них, потенциально уязвимы к атакам.

- ***Сканеры уязвимых мест программ:*** программы, проверяют большие группы компьютеров в Интернете в поисках компьютеров, уязвимых к тому или иному конкретному виду атаки.
- ***Вскрыватели паролей:*** программы, которые обнаруживают легко угадываемые пароли в зашифрованных файлах паролей. Сейчас компьютеры могут угадывать пароли так быстро, что казалось бы сложные пароли могут быть угаданы.
- ***Сетевые анализаторы (снифферы):*** программы, которые слушают сетевой трафик. Часто в них имеются возможности автоматического выделения имен пользователей, паролей и номеров кредитных карт из трафика.

Как защититься от большинства компьютерных атак

Защита сети от компьютерных атак — это постоянная и нетривиальная задача; но ряд простых средств защиты смогут остановить большинство попыток проникновения в сеть. Например, хорошо сконфигурированный межсетевой экран и антивирусные программы, установленные на всех рабочих станциях, смогут сделать невозможными большинство компьютерных атак. Ниже мы кратко опишем 14 различных средств защиты, реализация которых поможет защитить вашу сеть.

1. Оперативная установка исправлений для программ (Patching). Компании часто выпускают исправления программ, чтобы ликвидировать неблагоприятные последствия ошибок в них. Если не внести исправления в программы, впоследствии атакующий может воспользоваться этими ошибками и проникнуть в ваш компьютер. Системные администраторы должны защищать самые важные свои системы, оперативно устанавливая исправления для программ на них. Обычно исправления должны получаться ТОЛЬКО от производителей программ.

2. Обнаружение вирусов и троянских коней. Хорошие антивирусные программы — незаменимое средство для повышения безопасности в любой сети. Они наблюдают за работой компьютеров и выявляют на них вредоносные программы. Единственной проблемой, возникающей из-за них, является то, что для максимальной эффективности они должны быть установлены на всех компьютерах в сети. На установку антивирусных программ на всех компьютерах и регулярное обновление антивирусных баз в

них может уходить достаточно много времени — но иначе это средство не будет эффективным. Пользователей следует учить, как им самим делать эти обновления, но при этом нельзя полностью полагаться на них. Помимо обычной антивирусной программы на каждом компьютере рекомендуется, чтобы организации сканировали приложения к электронным письмам на почтовом сервере. Таким образом можно обнаружить большинство вирусов до того, как они достигнут машин пользователей.

3. Межсетевые экраны. Межсетевые экраны (***firewalls***) — это самое важное средство защиты сети организации. Они контролируют сетевой трафик, входящий в сеть и выходящий из нее. Межсетевой экран может блокировать передачу в сеть какого-либо вида трафика или выполнять те или иные проверки другого вида трафика. Хорошо сконфигурированный межсетевой экран в состоянии остановить большинство известных компьютерных атак.

4. Вскрыватели слабых паролей (Password Crackers). Хакеры часто используют малоизвестные уязвимые места в компьютерах для того, чтобы украсть файлы с зашифрованными паролями. Затем они используют специальные программы для вскрытия паролей, которые могут обнаружить слабые пароли в этих зашифрованных файлах. Как только слабый пароль обнаружен, атакующий может войти в компьютер, как обычный пользователь и использовать разнообразные приемы для получения полного доступа к вашему компьютеру и вашей сети. Хотя это средство используются злоумышленниками, оно будет также полезно и системным администраторам. Они должны периодически запускать эти программы на свои зашифрованные файлы паролей, чтобы своевременно обнаружить слабые пароли.

5. Шифрование. Атакующие часто проникают в сети с помощью прослушивания сетевого трафика в наиболее важных местах и выделения из него имен пользователей и их паролей. Поэтому соединения с удаленными машинами, защищаемые с помощью пароля, должны быть зашифрованы. Это особенно важно в тех случаях, если соединение осуществляется по Интернет или с важным сервером. Имеется ряд коммерческих и бесплатных программ для шифрования трафика TCP/IP (наиболее известен SSH).

6. Сканеры уязвимых мест. Это программы, которые сканируют сеть в поисках компьютеров, уязвимых к определенным видам атак. Сканеры имеют большую базу данных уязвимых мест, которую они используют при проверке того или иного компьютера на наличие у него уязвимых мест.

7. Грамотное конфигурирование компьютеров в отношении безопасности. Компьютеры с заново установленными операционными системами часто уязвимы к атакам. Причина этого заключается в том, что при начальной установке операционной системы обычно разрешаются все сетевые средства и часто разрешаются небезопасным образом. Это позволяет атакующему использовать много способов для организации атаки на машину. Все ненужные сетевые средства должны быть отключены.

8. Боевые дозвонщики (war dialer). Пользователи часто обходят средства защиты сети организации, разрешая своим компьютерам принимать входящие телефонные звонки. Пользователь перед уходом с работы включает модем и соответствующим образом настраивает программы на компьютере, после чего он может позвонить по модему из дома и использовать корпоративную сеть. Атакующие могут использовать программы — боевые диалеры (дозвонщики) для обзвонки большого числа телефонных номеров в поисках компьютеров, обрабатывающих входящие звонки. Так как пользователи обычно конфигурируют свои компьютеры сами, они часто оказываются плохо защищенными и дают атакующему еще одну возможность для организации атаки на сеть. Системные администраторы должны регулярно использовать боевые диалеры для проверки телефонных номеров своих пользователей и обнаружения сконфигурированных подобным образом компьютеров.

9. Рекомендации по безопасности (security advisories). Рекомендации по безопасности — это предупреждения, публикуемые группами по борьбе с компьютерными преступлениями и производителями программ о недавно обнаруженных уязвимых местах. Рекомендации обычно описывают самые серьезные угрозы, возникающие из-за этих уязвимых мест и поэтому являются занимающими мало времени на чтение, но очень полезными. Они описывают вцелом угрозу и дают довольно конкретные советы о том, что нужно сделать для устранения данного уязвимого места.

Найти их можно в ряде мест, но двумя самыми полезными являются те рекомендации, которые публикует группа по борьбе с компьютерными преступлениями CIAC (http://ciac.llnl.gov) и CERT (http://www.cert.org)

10. Средства обнаружения атак (Intrusion Detection). Системы обнаружения атак оперативно обнаруживают компьютерные атаки. Они могут быть установлены за межсетевым экраном, чтобы обнаруживать атаки, организуемые изнутри сети. Или они могут быть установлены перед межсетевым экраном, чтобы обнаруживать атаки на межсетевой экран. Средства этого типа могут иметь разнообразные возможности.

11. Средства выявления топологии сети и сканеры портов. Эти программы позволяют составить полную картину того, как устроена ваша сеть и какие компьютеры в ней работают, а также выявить все сервисы, которые работают на каждой машине. Атакующие используют эти средства для выявления уязвимых компьютеров и программ на них. Системные администраторы должны использовать эти средства для наблюдения за тем, какие программы и на каких компьютерах работают в их сети. С их помощью можно обнаружить неправильно сконфигурированные программы на компьютерах и установить исправления на них.

12. Группа по расследованию происшествий с безопасностью. В каждой сети, независимо от того, насколько она безопасна, происходят какие-либо события, связанные с безопасностью (может быть даже ложные тревоги). Сотрудники организации должны заранее знать, что нужно делать в том или ином случае. Важно заранее определить следующие моменты — когда вызывать правоохранительные органы, когда вызывать сотрудников группы по борьбе с компьютерными преступлениями, когда следует отключить сеть от Internet, и что делать в случае компрометации важного сервера.

13. Политики безопасности. Система сетевой безопасности насколько сильна, насколько сильно защищено самое слабое ее место. Если в рамках одной организации имеется несколько сетей с различными политиками безопасности, то одна сеть может быть скомпрометирована из-за плохой безопасности другой сети. Организации должны разработать политику безопасности, в которой определялся бы ожидаемый уровень защиты, который

должен быть везде единообразно реализован. Самым важным аспектом политики является выработка единых требований к тому, какой трафик должен пропускаться через межсетевые экраны сети. Также политика должна определять как и какие средства защиты (например, средства обнаружения атак или сканеры уязвимых мест) должны использоваться в сети. Для достижения единого уровня безопасности политика должна определять стандартные безопасные конфигурации для различных типов компьютеров.

14. Тестирование межсетевых экранов и WWW-серверов на устойчивость к попыткам их блокирования. Атаки на блокирование компьютера распространены в Интернет. Атакующие постоянно выводят из строя WWW-сайты, перегружают компьютеры или переполняют сети бессмысленными пакетами. Атаки этого типа могут быть очень серьезными, особенно если атакующий настолько умен, что организовал продолжительную атаку, у которой не выявить источник. Сетевые администраторы, заботящиеся о безопасности сети, могут организовать атаки против своей сети сами, чтобы определить, какой ущерб может быть нанесен, и предотвратить его.

8.2 Брандмауэры

Для защиты локальной компьютерной сети от внешних угроз применяются межсетевые экраны — ***брандмауэры*** (***firewall***). Брандмауэр обычно устанавливается на границе локальной сети и отделяет ее от глобальной сети.

От маршрутизатора, функцией которого является максимально быстрая доставка трафика в пункт назначения, брандмауэр, отличается тем, что его функция — контроль доступа в сеть и пропуск только разрешенного трафика из потока данных. Любой маршрутизатор можно настроить на блокировку определенного трафика с помощью ***списков доступа*** (***ACL, Access Control List***) редактируемых администратором конфигурационных файлов. Однако при этом используется только фильтрация по IP-адресам, современные же брандмауэры могут осуществлять намного более интеллектуальную фильтрацию, например, по типу прикладного протокола или даже по содержащимся в сообщении пользовательским данным.

Можно сказать, что брандмауэр запрещает весь трафик, кроме разрешенного, маршрутизатор же, наоборот, разрешает весь трафик, кроме запрещенного.

Сэтим связаны и проблемы помех, создаваемых программным брандмауэром персонального компьютера, например, при попытке создания нового сетевого соединения. Брандмауэр блокирует пакеты, обмен которыми происходит при создании такого соединения. Таким образом, создать его оказывается невозможно. Решить проблему можно отключив брандмауэр, после чего создать новое соединение, а затем вновь включить брандмауэр.

Существуют два основных типа брандмауэров:

- прикладного уровня;
- с пакетной фильтрацией.

Брандмауэр прикладного уровня может быть реализован как программно, так и аппаратно. В брандмауэрах прикладного уровня правила политики безопасности усиливаются посредством использования модулей доступа. Каждому разрешаемому протоколу должен соответствовать свой собственный модуль доступа. Лучшими модулями доступа считаются те, которые построены специально для разрешаемого протокола. Например, модуль доступа FTP предназначен для протокола FTP и может определять, соответствует ли проходящий трафик этому протоколу и разрешен ли этот трафик правилами политики безопасности.

При использовании межсетевого экрана прикладного уровня все соединения проходят через него (см. рис. 8.1).

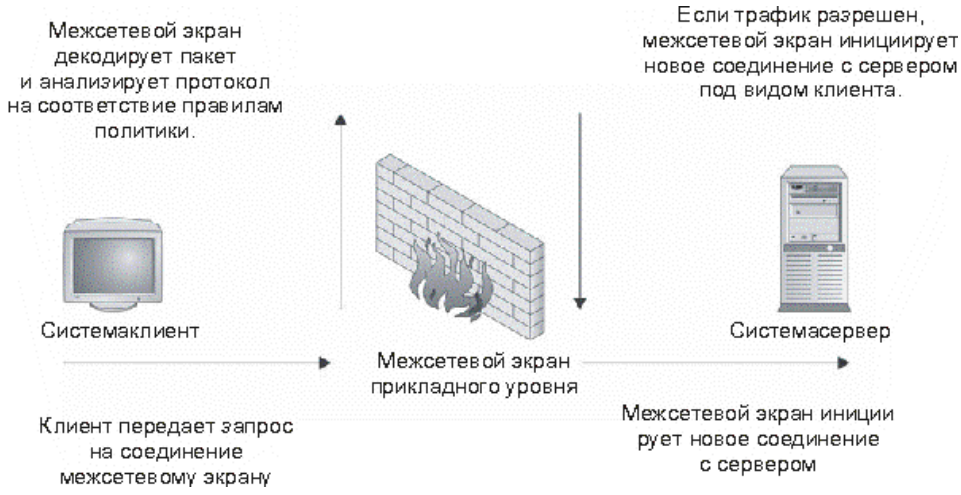


Рис. 8.1 — Соединения модуля доступа брандмауэра прикладного уровня

Как показано на рисунке, соединение инициированное клиентом поступает на внутренний интерфейс брандмауэра. Он принимает запрос, анализирует содержимое пакета и используемый протокол и определяет, соответствует ли этот запрос правилам политики безопасности. Если это так, то брандмауэр, в свою очередь, инициирует новое соединение с сервером.

Брандмауэры прикладного уровня используют также модули доступа и для входящих подключений. Модуль доступа в брандмауэре принимает входящее подключение и обрабатывает его перед отправкой получателю. Если пакет содержит вредоносный код, нацеленный на уязвимость приложения на компьютере конечного пользователя, это будет обнаружено и пакет блокируется (конечно, предполагается, что сам модуль доступа брандмауэра неуязвим). Таким образом, брандмауэр защищает системы от атак, выполняемых посредством приложений.

Брандмауэры прикладного уровня содержат модули доступа для наиболее часто используемых протоколов, таких как HTTP, SMTP, FTP и telnet. Если модуль доступа для какого-то протокола отсутствует, то этот протокол не может использоваться для соединения через брандмауэр. Брандмауэры прикладного уровня скрывает схему внутренней адресации сети. Так как все соединения инициируются и завершаются на интерфейсах брандмауэра, внутренние системы сети не видны напрямую извне.

Брандмауэр с пакетной фильтрацией, как правило, имеет несколько интерфейсов, по одному на каждую из сетей, к которым он подключен. Аналогично брандмауэрам прикладного уровня, доставка трафика из одной сети в другую определяется набором правил политики. Если правило не разрешает явным образом определенный трафик, то соответствующие пакеты будут отклонены или аннулированы.

Правила политики могут быть усилены посредством использования программных фильтров пакетов. Фильтры изучают содержимое пакета и определяют, является ли трафик разрешенным, согласно правилам политики и состоянию протокола (проверка с учетом состояния). Если протокол приложения функционирует через TCP, определить состояние относительно просто, так как TCP сам поддерживает состояния. Это означает, что когда

протокол находится в определенном состоянии, разрешена передача только определенных пакетов.

Например, при установлении TCP-соединения первый ожидаемый пакет — пакет *SYN*. Брандмауэр обнаруживает этот пакет и переводит соединение в состояние SYN. В данном состоянии ожидается один из двух пакетов — либо *SYN ACK* (опознавание пакета и разрешение соединения) или пакет *RST* (сброс соединения по причине отказа в соединении получателем). Если в данном соединении появятся другие пакеты, брандмауэр аннулирует их, так как они не подходят для данного состояния соединения, даже если такое соединение разрешено набором правил.

Для протокола UDP, брандмауэр с пакетной фильтрацией отслеживает состояние трафика UDP. Например, брандмауэр принимает внешний пакет UDP и ожидает ответный пакет от получателя, соответствующий исходному пакету по адресу и порту, в течение определенного времени. Если пакет принимается в течение этого отрезка времени, его передача разрешается. В противном случае брандмауэр определяет, что трафик UDP не является ответом на запрос, и аннулирует его.

Брандмауэр с пакетной фильтрацией не прерывает соединения (см. рис. 8.2). При поступлении пакетов брандмауэр выясняет, разрешен ли данный пакет и состояние соединения правилами политики. Если это так, пакет передается по своему маршруту. В противном случае пакет отклоняется или аннулируется.

Рис. 8.2 — Передача трафика через брандмауэр с фильтрацией пакетов

Брандмауэры с фильтрацией пакетов не используют модули доступа для каждого протокола и поэтому могут работать с любым протоколом, работающим через IP. Некоторые протоколы требуют распознавания брандмауэром выполняемых ими действий. Например, протокол FTP использует одно соединение для начального входа и команд, а другое — для передачи файлов. Соединения, используемые для передачи файлов, устанавливаются как часть соединения FTP, и поэтому брандмауэр должен уметь считывать трафик и определять порты, которые будут использоваться новым соединением. Если брандмауэр не поддерживает эту функцию, передача файлов невозможна.

Обычно считается, что брандмауэры с фильтрацией пакетов имеют возможность поддержки большего объема трафика, т.к. в них отсутствует нагрузка, создаваемая дополнительными процедурами настройки и вычисления, имеющими место в программных модулях доступа брандмауэров прикладного уровня.

К недостаткам брандмауэров этого типа можно отнести то, что компьютер в локальной сети может быть атакован через открытую службу, разрешенную правилами политики и брандмауэр никак не отреагирует на атаку. Кроме того, брандмауэры с фильтрацией пакетов позволяют видеть извне внутреннюю структуру локальной сети, так как соединения не прерываются на межсетевом экране. (По этой причине работу таких брандмауэров всегда совмещают с технологией трансляции сетевых адресов ***NAT*** (***Network Address Translation***), когда в локальной сети используются специальные IP-адреса, не транслируемые маршрутизаторами в Интернет (*RFC 1918*). Трансляция адресов может выполняться как самим брандмауэром, так и пограничным маршрутизатором).

Гибридные брандмауэры. Для поддержки модулями доступа прикладного уровня других протоколов, для которых не существует определенных модулей, производителями брандмауэров прикладного уровня была разработана технология ***Generic Services Proxy*** (***GSP***), обеспечивающая работу брандмауэров прикладного уровня в режиме пакетной фильтрации. Производители брандмауэров с пакетной фильтрацией добавили в них модуль доступа SMTP. В то время как базовая функциональность межсетевых экранов обоих типов осталась прежней, эти нововведения

привели к появлению на рынке гибридных межсетевых экранов. Сейчас практически невозможно найти брандмауэр, функционирование которого построено исключительно на прикладном уровне или фильтрации пакетов. Это позволяет администраторам, отвечающим за безопасность, более гибко настраивать устройство для работы в конкретных условиях.

8.3 Создание демилитаризованных зон

Брандмауэр, обеспечивая определенный уровень защищенности локальной сети, в то же время не является гарантией полной безопасности. Поэтому при соединении локальной сети с внешней сетью вводится промежуточная сеть, получившая название ***демилитаризованная зона*** (***demilitarized zone, DMZ***). Смысл создания такого сетевого сегмента заключается в том, чтобы отделить системы, к которым осуществляют доступ пользователи интернета, от систем, с которыми работают сотрудники организации.

Данная зона также защищается брандмауэром и для нее определяется набор правил, разрешающих определенные виды трафика, рис. 8.3. Внешний запрос может попасть только в демилитаризованную зону и никогда — прямо в локальную сеть.

Рис. 8.3 — DMZ отделяет локальную сеть от доступа извне

Правила доступа для DMZ позволяют внешним пользователям подключаться к соответствующим службам, расположенным на серверах в демилитаризованной зоне. В то же время, на системы в DMZ налагаются строгие ограничения на доступ ко внутренним системам сети. Соединение между компьютером в локальной сети и DMZ инициируется только из локальной сети. Поэтому пользователи локальной сети могут осуществлять доступ к DMZ или в Интернет согласно установленным политикам безопасности, а внешним пользователям доступ локальную сеть запрещен.

В демилитаризованной зоне размещаются как правило почтовые и веб-сервера организации, сервер DNS и т.п. Существует множество архитектур демилитаризованных зон. Как и в большинстве вопросов безопасности, имеют место преимущества и недостатки каждой архитектуры, и для каждой организации следует в отдельном порядке осуществлять выбор конкретной архитектуры DMZ. На рис. 8.4 показан пример реализации DMZ с использованием дублирования серверов DNS и электронной почты.

Такая схема позволяет дополнительно ограничить доступ к внутренним ресурсам сети. Внешний почтовый сервер используется для приема входящей почты и для отправки исходящей почты. Новая почта принимается внешним почтовым сервером и передается на внутренний почтовый сервер. Внутренний почтовый сервер передает исходящую почту на внешний сервер. В идеальном случае все эти действия выполняются внутренним почтовым сервером с запрашиванием почты с внешнего почтового сервера.

Многие веб-сайты предоставляют активное содержимое (интернет-магазины и т.п.), функционирующее на основе вводимых пользователем данных. Эти данные могут вноситься в базу данных или обрабатываться на основе информации из такой базы. База данных содержит важную информацию, и ее не следует располагать в демилитаризованной зоне. Веб-сервер сам по себе мог бы осуществлять обратную связь с сервером базы данных, но веб-сервер доступен из внешней среды и, таким образом, не пользуется полным доверием. В данном случае рекомендуется использовать третью систему для размещения на ней приложения, непосредственно соединяющегося с базой данных. Веб-сервер получает вводимые пользователем данные и предоставля-

ет их серверу приложения для обработки. Сервер приложения запрашивает в базе данных нужную информацию и предоставляет ее веб-серверу для доставки пользователю, рис. 8.4.

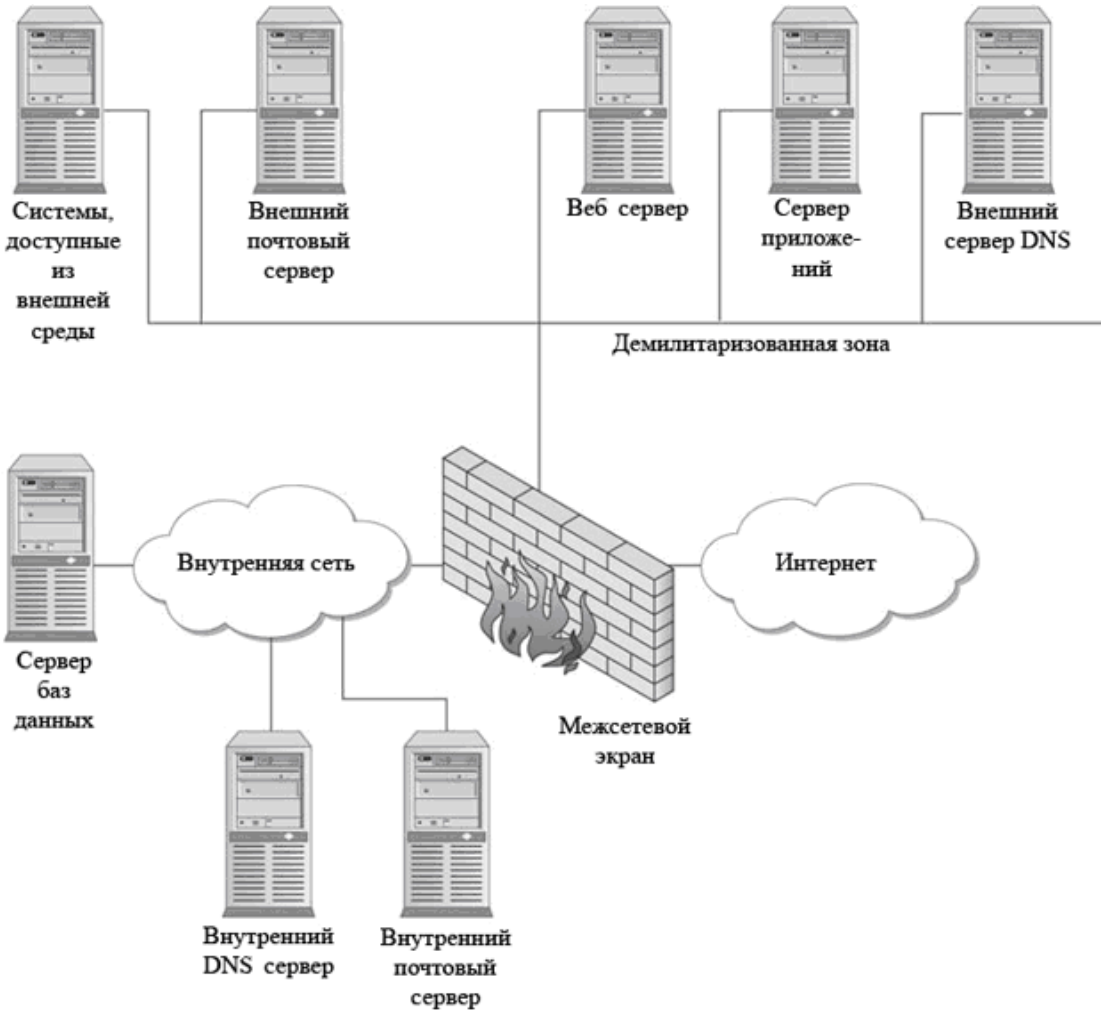


Рис. 8.4 — Пример реализации DMZ

8.4 Системы обнаружения вторжений

Системы обнаружения вторжений (***IDS, Intrusion Detection System***) применяются для обнаружения различной опасной активности на компьютерах локальной сети, которая может быть результатом выполняемой на локальную сеть атаки, а также следствием работы вирусов, компьютерных червей и программ троянских коней. Такая система состоит из программ-датчиков, реагирующих на определенные события, системы оповещения и наблюдения за сетевой активностью и программного

анализатора событий. Датчики размещаются в узких местах сети, через которые проходит основной трафик. Часто это точка соединения локальной сети с Интернет или демилитаризованная зона, рис. 8.5.

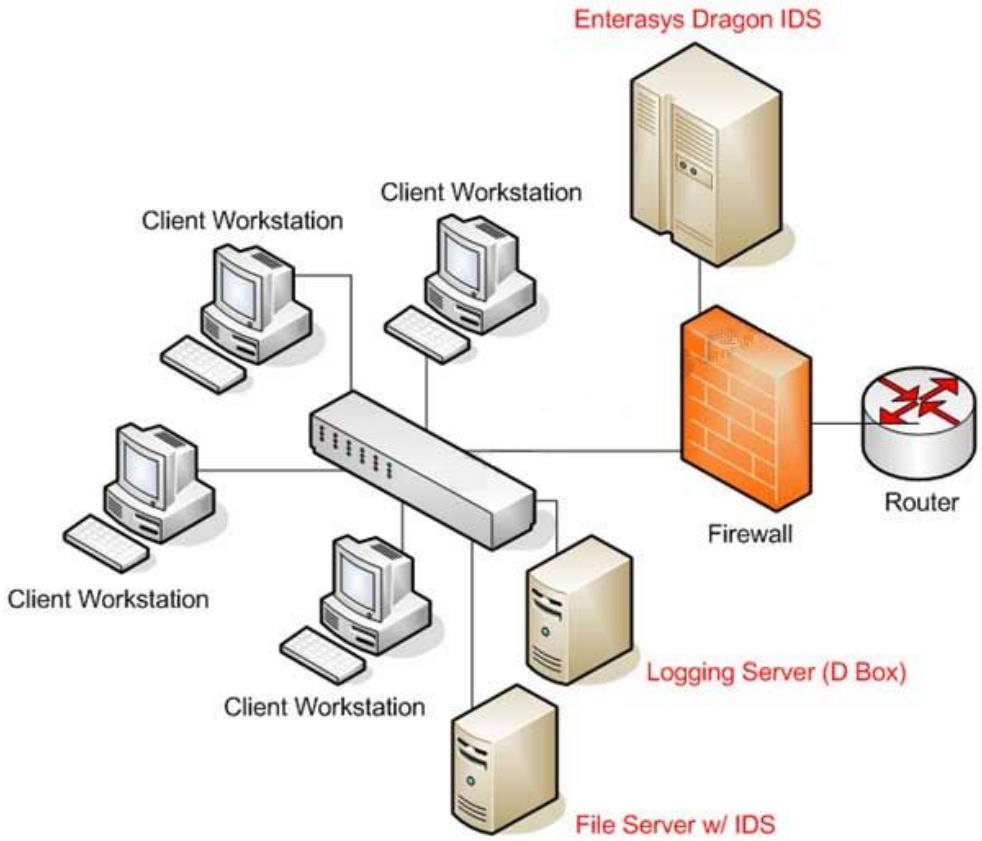


Рис 8.5 — Пример включения IDS.

Различают активные и пассивные системы обнаружения вторжений. Пассивные системы ведут регистрацию событий и сообщают о появлении угрозы, активные блокируют ее, например, разрывают соединение. Активные системы обнаружения вторжений называют также ***IPS, Intrusion Prevention System.*** Работа таких систем близка по своему характеру к функционированию брандмауэров прикладного уровня.

Система обнаружения вторжений позволяет обнаружить попытки несанкционированного доступа не перехваченные брандмауэром, т.к. она реагирует не на отдельный сетевой запрос, а на динамику изменения и характер сетевой активности в целом. Так, одной из функций, выполняемых IDS, может быть просмотр проходящих через сетевые соединения пакетов на предмет обнару-

жения сигнатур известных вирусных программ. Это объединяет подобные системы с антивирусными программами. Обнаружив большое число TCP соединений по большому числу портов пограничного маршрутизатора, IDS определяет, что выполняется попытка сканирования его портов и позволяет предотвратить начинающуюся атаку.

Система обнаружения вторжений может также контролировать внутренние сетевые запросы, проверяя их на лояльность, т.к. многие сетевые атаки начинаются изнутри локальной сети, с помощью внедрения на один из компьютеров вредоносной программы-закладки, проходящей через все ступени системы защиты в зашифрованном файле-контейнере.

8.5 Сетевой компьютер

Сетевой компьютер (сервер или рабочая станция) является конечным объектом атаки злоумышленника. Уязвимости сетевого компьютера определяются типом установленной на него операционной системы (ОС). Для осуществления атаки злоумышленнику необходима информация об установленной ОС, т.к. способы проникновения на компьютеры с различными ОС неодинаковы. Наиболее распространенными сетевыми ОС в настоящее время являются ОС двух типов: семейства Windows и UNIX/Linux. Причем, последние строились изначально как сетевые ОС, устойчивость и безопасность работы которых была для разработчиков определяющим фактором. Защищенность таких систем, как правило, оказывается намного выше. Современные версии ОС Windows значительно приблизились по уровню обеспечения безопасности к UNIX-системам, однако многие из имеющихся возможностей по-умолчанию отключены или требуют настройки, не всегда выполняемой пользователем. Неудивительно, поэтому, что большинство успешных атак выполняется именно на компьютеры, работающие под управлением ОС Windows. При работе в сети Windows на основе контроллера домена, ***служба каталогов*** (***Active Directory***) позволяет настроить общие для всех компьютеров сети параметры безопасности. При модемном подключении к Интернет или работе в одноранговой

сети, необходимо учитывать следующие факторы повышения безопасности Windows-систем:

- Отключение на локальном компьютере учетной записи гостя (Guest).
- Отключение неиспользуемых служб удаленного доступа.
- Применение на жестких дисках файловой системы NTFS. NTFS повилась в версии Windows 2000 и обеспечивает намного более высокий уровень безопасности, по сравнению с использованной ранее системой FAT.
- Применение системы шифрования файлов (EFS). EFS защищает файлы даже в том случае, когда злоумышленнику удалось получить доступ к дискам NTFS из другой ОС, в которой механизмы защиты фаловой системы не работают.
- Настройка параметров безопасности с помощью программной оснастки ***secpol.msc***. Консоль *LSS* (*Local Security Settings*) позволяет задать целый ряд ограничений на сетевой доступ к компьютеру, параметры входа пользователя в систему, пользовательские настройки безопасности (например, минимальная длина и сложность пароля), включить аудит событий и т.п.
- Включение программного брандмауэра ***ICF*** (***Internet Connection Firewall***). Брандмауэр входит в состав обновления *Service Pack 2* для Windows XP и является составной частью Windows Vista.

8.6 Защищенный внешний доступ к локальной сети

Внешний доступ в локальную сеть может быть необходим как для сотрудников организации, так и для ее клиентов. Выбор метода такого соединения влияет на архитектуру локальной сети в организации. Доступ для клиентов следует планировать только к ресурсам, расположенным в демилитаризованной зоне. Доступ сотрудников ко внутренним системам из удаленных местоположений, как правило, осуществляется посредством использования виртуальной частной сети (***VPN, Virtual Private Network***). Виртуальные частные сети обеспечивают поддержку различных прикладных протоколов, что делает их в настоящее время распространенным методом защищенного подключения к внутренней сети. Различают пользовательские и узловые VPN-подключения.

Пользовательское VPN-подключение

При установлении пользовательского VPN-подключения происходит аутентификация сотрудника на VPN-сервере организации, после чего весь передаваемый в направлении сотрудника и от него трафик шифруется, рис. 8.6. Возможно, самой большой проблемой безопасности при использовании пользовательского VPN сотрудником является то, что с компьютера сотрудника выполняется также одновременное соединение с другими сайтами интернета. Как правило, программное обеспечение VPN на компьютере сотрудника определяет, должен ли трафик передаваться через VPN, либо его необходимо отправить на какой-либо другой сайт в открытом виде.

Рис. 8.6 — Подключение к внутренней локальной сети через VPN-канал

Если на компьютер сотрудника была произведена атака с использованием *троянского коня*, то злоумышленник сможет использовать компьютер сотрудника для подключения к внутренней сети организации. Причем, с точки зрения системы защиты, такое подключение будет допустимым, ведь оно выполняется от имени сотрудника. Поэтому одним из ключевых аспектов пользовательской VPN, о котором не следует забывать, является установка хорошей антивирусной программы на компьютере пользователя.

Этот программный пакет должен обеспечивать регулярное обновление своих баз для противостояния вирусам и троянским коням. На компьютере сотрудника необходимо также установить брандмауэр офисного или домашнего уровня. Многие из таких систем могут управляться удаленно, что позволит организации отслеживать и настраивать систему защиты.

Узловое VPN-подключение

В случае узлового VPN-подключения канал VPN соединяет один межсетевой экран или пограничный маршрутизатор с другим аналогичным устройством, рис. 8.7.

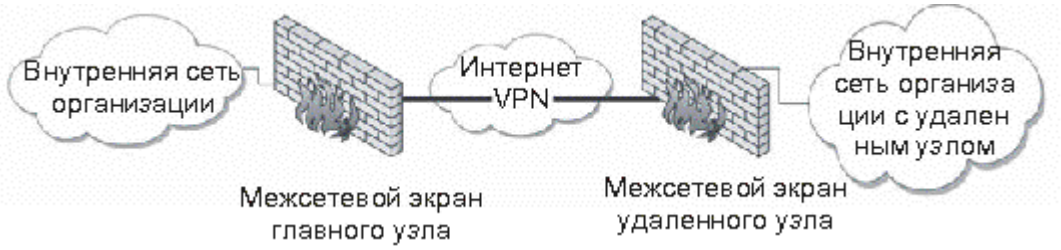


Рис. 8.7 — Организация узлового VPN-канала

Основным преимуществом узлового VPN соединения является экономичность. Организация с небольшими, удаленными друг от друга офисами может создать виртуальную частную сеть, соединяющую все удаленные офисы с центральным узлом (или друг с другом) со значительно меньшими затратами, по сравнению с приобретением для этих целей выделенных линий. Недостатком узловых VPN-подключений является то, что они расширяют периметр безопасности организации, усложняют работу по проведению единой политики безопасности.

ЗАКЛЮЧЕНИЕ

Данное учебное пособие создано как основное для чтения лекционного материала студентам специальности «Промышленная электроника» при изучении курса «Эксплуатация и развитие компьютерных систем и сетей».

Основными задачами данного учебно-методического пособия являются:

1. Оказание помощи студентам очной формы обучения в более глубоком изучении основ компьютерных технологий и информационных сетей.

2. Помощь студентам при изучении известных стеков сетевых протоколов, описывающих их стандартов, форматов пакетов, интерфейсов и протоколов.

3. Систематизация знаний студентов в части информационных сетей, структурирование по уровням OSI и формирование у студентов четкого понимания способов взаимодействия программно-аппаратных функций сетевой аппаратуры.

4. Помощь в освоении современного оборудования информационных сетей, выполняемых им функций и их устройств и структур.

5. Формирование навыков разработки, наладки и настройки программного обеспечения и операционных систем сетевых устройств, серверов и рабочих станций.

Практическое закрепление знаний по современным и классическим сетевым технологиям, способам построения компьютерных сетей и проведения необходимых расчетов при их проектировании реализуется в рамках лабораторных работ по курсу. Авторами подготовлено так же Учебно-методическое пособие под одноименным названием для систематизации самостоятельной практической работы студентов и проведения лабораторных работ.

Самостоятельная работа над курсом «Эксплуатация и развитие компьютерных систем и сетей» распределяется по времени следующим образом: 32 часа — на изучение лекционного курса; оформление отчетов по лабораторным работам — 40 часов; выполнение творческих заданий — 10 часов; подготовка к экзамену — 10 часов.

Работа студентов над материалом дисциплины «Эксплуатация и развитие компьютерных систем и сетей» может оцениваться по рейтинговой системе. Максимальный рейтинг при этом составляет 120 баллов и определяется в соответствии с положением: для получения оценки «отлично» требуется набрать не менее 100 баллов, «хорошо» — 80 баллов. Для допуска к экзамену требуется набрать не менее 60 баллов. Таблица

Творческое задание выполняется по желанию самостоятельно. Тема творческого задания выбирается студентом самостоятельно по тематике курса. Тема творческого задания должна быть обязательно согласована с преподавателем.

РЕКОМЕНДУЕМАЯ ЛИТЕРАТУРА

Основная литература

1. Норенков И.П., Трудоношин В.А. Телекоммуникационные технологии и сети. — 2-е изд., испр. и доп. — М.: Изд-во МГТУ им. Н. Э. Баумана, 2000. — 248 с.

2. Обрусник П.В. Эксплуатация и развитие компьютерных сетей и систем. Часть 2: Учебно-методическое пособие. — Томск: Томский межвузовский центр дистанционного образования, 2001. — 21 с.

Дополнительная литература

1. Танненбаум Э. Компьютерные сети: Пер. с англ. — 4-е изд. — СПб.: Питер, 2003. — 991 с.

2. Соловьева Л.Ф. Сетевые технологии: учебник-практикум. — СПб.: БХВ-Перербург, 2004. — 397 с.

3. Нанс Б. Компьютерные сети: Пер. с англ. — М.: Бином, 1995. — 400 с.

4. Вишневский В.М. Теоретические основы проектирования компьютерных сетей: Монография / РАН. Институт проблем передачи информации. — М.: Техносфера, 2003. — 506 с.

5. Корнеев В.В. Вычислительные системы. — М.: Гелиос АРВ, 2004. — 510 с.

6. Компьютерные сети. Принципы, технологии, протоколы: Учебник / В.Г. Олифер, Н.А. Олифер. — СПб: Изд-во «Питер», 2000. — 672 с.