МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Озерский технологический институт — филиал НИЯУ МИФИ

Кафедра прикладной математики

Вл. Пономарев

КОМПЬЮТЕРНЫЕ СЕТИ

Учебно-методическое пособие

Озерск, 2019

УДК 681.3.06
П56

Пономарев В.В. Компьютерные сети. Учебно-методическое пособие по курсу «Сети и телекоммуникации». Озерск: ОТИ НИЯУ МИФИ, 2019. — 84 с., ил.

В пособии рассматриваются компьютерные сети, сетевые технологии и телекоммуникационное оборудование.

В качестве вспомогательного материала пособие предназначено для студентов старших курсов, обучающихся по направлению подготовки 09.03.01 «Информационная и вычислительная техника», или по специальности 09.05.01 «Применение и эксплуатация автоматизированных систем специального назначения».

Рецензенты:
1.
2.

УТВЕРЖДЕНО
Редакционно-издательским
Советом ОТИ НИЯУ МИФИ

Содержание

1. Базовые сведения

1.1. Понятие сети

Компьютерные сети являются объединением компьютеров и технологий связи. Два компьютера находятся в сети, если они могут обмениваться данными при помощи устройств связи. Компьютеры занимаются обработкой информации, а устройства связи осуществляют ее передачу.

Сети объединяют не только компьютеры, но и другие устройства, такие, как принтеры. Отдельное устройство называют хостом, узлом или станцией, узлом называют также устройство в сети. Границы сети находятся в ее устройствах, называемых DTE (Data Terminal Equipment), оконечное оборудование данных. Через DTE данные поступают в сеть из хоста, или наоборот, из сети в хост (рисунок 1). Обычным устройством DTE является сетевой адаптер, встроенный в хост.

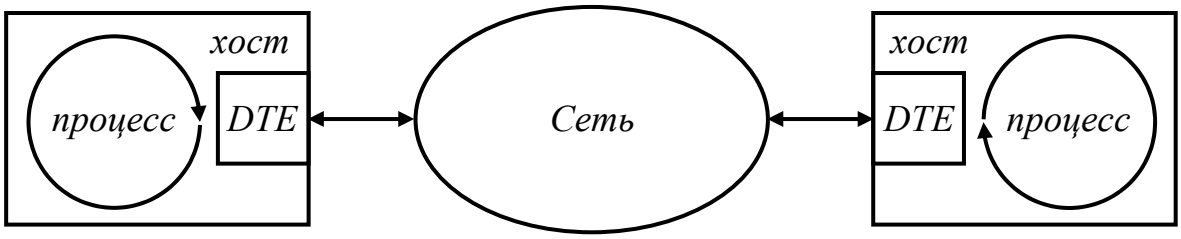


Рисунок 1 — Оконечные устройства сети

Внутри хоста с DTE взаимодействует один или несколько процессов, принимающих или передающих данные. Программное взаимодействие с сетью происходит через *сокет* (*socket*, гнездо). Прикладные программы, использующие сеть, обращаются к постоянно действующему процессу, или непосредственно к сети через сокет.

Собственно сеть состоит из линий связи и передающих устройств.

Линии связи передают данные бит за битом, октет за октетом. Октет представляет байт, передаваемый по сети. Кроме бит данных, по линиям связи передаются биты, необходимые для организации передачи.

Передающие устройства сети (телекоммуникационные устройства) усиливают и преобразуют сигналы, структурируют сеть, локализуют трафик, направляют трафик по определенным маршрутам, соединяют и разъединяют потоки данных с целью их уплотнения.

Частью сетевого взаимодействия являются также интерфейсы и протоколы. Интерфейсом являются элементы, соединяющие оборудование и (или) программы друг с другом. Протокол — это порядок действий, такой, как последовательность установления соединения или обмена данными по линии связи.

1.2. Цели сетей

Сети дают возможность совместно использовать информационные и аппаратные ресурсы конечными пользователями.

Цели использования сетей в организации многочисленны.

1. Совместное использование данных, организованных в базы данных или в хранилища общих файлов. Для этих целей существуют, соответственно, серверы баз данных и файловые серверы.

2. Совместное использование таких ресурсов, как принтер, сканер, устройство резервного копирования.

3. Обмен сообщениями и документами по электронной почте.

4. Мгновенный обмен личными сообщениями при помощи программ типа Skype способствует и общению и совместной работе.

5. Совместная работа над документами и проектами повышает производительность и эффективность труда.

6. IP-телефония решает проблему телефонных счетов.

7. Видеоконференции предотвращают передвижение сотрудников с целью координации действий и обсуждения проблем.

8. Использование цифровых подписей уменьшает объем печатных документов и необходимость их доставки.

9. Корпоративный сайт доступен на любом рабочем месте.

10. Электронная коммерция сокращает затраты на обслуживание.

11. Видеонаблюдение и видеорегистрация повышают безопасность.

12. Удаленная форма работы создает удобство для работников.

13. Удаленный прием заявок на обслуживание.

14. Удаленное обслуживание.

Несомненно, что роль сетей в организации расширяется с течением времени. Например, на нашем градообразующем предприятии появился удаленный контроль оплаты за спецпитание работников.

Цели использования сетей частными лицами тоже разнообразны.

1. Получение информации в сети Интернет.

2. Общение в социальных сетях.

4. Общение с людьми по всему миру.

3. Обмен сообщениями и документами по электронной почте.

5. Получение новостей в электронных газетах и журналах.

6. Получение телевизионных передач в любое удобное время.

7. Доступ к электронным библиотекам и фильмотекам.

8. Удаленная подача заявок на обслуживание, например, покупка в электронном магазине, запись на прием.

9. Удаленное обслуживание, например, покупка билетов.

10. Удаленное управление предметами (домом).

Это не полный перечень возможностей, и он тоже расширяется.

1.3. Виды сетей

В первом приближении все сети делятся на локальные и глобальные. Классификационным признаком принимается размер сети, иногда называемый ее диаметром. Это деление сложилось исторически.

1.3.1. Глобальные сети

Первые ЭВМ являлись независимыми научно-исследовательскими центрами, целью которых было выявление возможностей новых аппаратов для быстрого счета. С появлением многотерминальных систем и операционных систем разделения времени впервые, видимо, возникла необходимость передачи данных по линии связи.

Терминал подключается к мэйнфрейму при помощи двухпроводной линии связи. Передача ведется в текстовом режиме в коде ASCII, разработанном для телетайпных систем. Для организации передачи используются управляющие символы ASCII. Такая передача использовалась в первых сетях, и повлияла на развитие сетей в дальнейшем.

Многотерминальные системы вполне удовлетворяли текущие нужды организаций, сосредоточенных в переделах здания. Для доступа к мэйнфрейму из удаленных мест, например, из других городов, требовалось проложить линию связи. Вместо этого были использованы имеющиеся телефонные линии, а для передачи цифрового сигнала по аналоговой линии связи разработали модем. Модем (от модулятор — демодулятор) преобразует цифровой сигнал в сигнал звуковой частоты или наоборот. Так появились первые сети, связывавшие компьютеры, расположенные на значительном удалении друг от друга (рисунок 2).

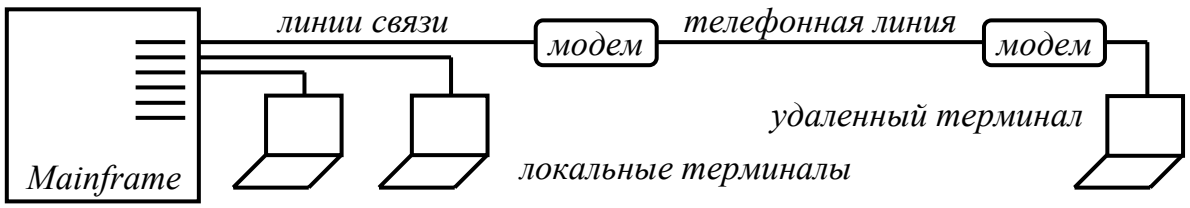


Рисунок 2 — Многотерминальная система и удаленная связь

Говорят, что первая такая связь произошла в 1969 году, когда было передано сообщение «login» по сети ARPANET между двумя компьютерами, расположенными на расстоянии 640 км. Сеть ARPANET создавалась по заказу военного ведомства США, и считается первой в мире сетью, а также прародительницей сетей Интернет.

Сети, объединяющие компьютеры на значительном удалении, называют глобальными сетями, Wide Area Network, WNA.

1.3.2. Локальные сети

По мере развития компьютерной индустрии компьютеры стали доступны не только специалистам вычислительных центров, но и бизнесу, науке и производству. С появлением компьютеров в офисах предприятий возникает необходимость связи между ними для совместного использования, например, дорогих в то время принтеров, для передачи файлов с одного компьютера на другой, для доступа к базам данных.

Между отдельными ЭВМ в различных организациях создаются местные сети из линий связи и устройств сопряжения вида «линия связи — конкретная ЭВМ» или «линия связи — конкретное устройство». Развитие таких местных сетей в конечном итоге привело к формированию сетевых технологий локальных сетей, их унификации, стандартизации, и широкому распространению.

Если в глобальной сети можно использовать имеющуюся телефонную линию, то для создания локальной сети линии связи нужно создавать. Это дает возможность использовать кабель, допускающий высокую скорость обмена информацией, сопоставимую со скоростью работы устройств компьютера. Вследствие этого в локальных сетях появляются различные службы для передачи файлов, факсимильных и электронных сообщений, доступа к дискам других компьютеров, к базам данных. Наиболее простая схема локальной сети показана на рисунке 3.

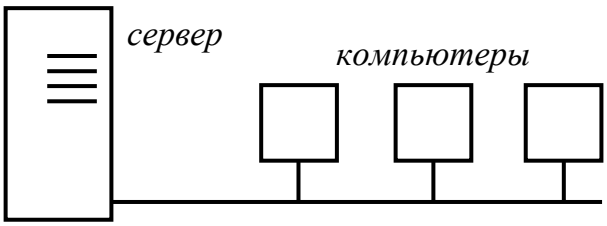


Рисунок 3 — Локальная сеть с сервером

Здесь один из компьютеров предназначен для хранения общей информации в виде файлов или баз данных, а пользователи прочих компьютеров могут использовать эту информацию совместно.

Таким образом, все многообразие сетей условно можно поделить на две важные и значительно различающиеся категории.

Локальная вычислительная сеть (Local Area Network, LAN, ЛВС) объединяет компьютеры организации, такой, как университет, завод или торговая точка. Глобальная вычислительная сеть объединяет компьютеры, расположенные, как минимум, в разных городах. Глобальной также называется вполне конкретная сеть, имеющая определенное название и назначение. Примерами являются сети ARPANET, USENET, NSFNET, FIDONET, MILNET и другие.

1.3.3. Подсети, интерсети, опорные сети и магистрали

В связи с глубоким проникновением Интернет и предоставляемых с его помощью услуг в повседневную жизнь, потребность в глобальных линиях связи резко возросла. В ответ на эту потребность созданы глобальные сети, предназначенные не для соединения компьютеров конечных пользователей, а для передачи трафика на большие расстояния. Для обозначения такой сети используется термин подсеть (subnet). Подсеть состоит обычно из линий связи и маршрутизаторов (рисунок 4).

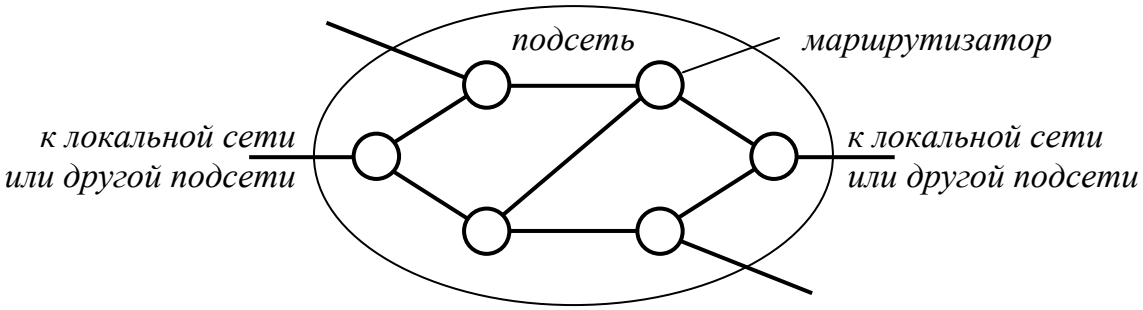


Рисунок 4 — Подсеть

Глобальную сеть можно рассматривать как подсеть или несколько соединенных подсетей, к которым подключены компьютеры пользователей. Вследствие протяженности отдельные линии связи, а также подсети иногда называют магистралями, магистральными сетями, магистральными каналами, или опорными сетями (backbone).

Интерсетью (internetwork, internet с маленькой буквы) называют соединение нескольких сетей, когда создание и поддержку сетей оплачивают разные организации, и (или) сети используют разные технологии.

Тогда сеть — это подсеть и присоединенные к ней хосты, когда хосты не принадлежат подсети, точно так же, как телефоны на принадлежат телефонной компании, содержащей телефонную сеть.

Интернет (Internet с большой буквы) — это объединение множества существующих разнородных сетей в одну большую гетерогенную глобальную сеть. Интернет часто для простоты называют сетью.

1.3.4. Другие сети

Национальные сети — сети масштаба страны. Региональные сети — сети масштаба округа, области. Эти сети обычно глобальные. Муниципальные и городские сети — сети масштаба города, они обычно локальные. Сеть кампуса, а также корпоративная сеть (enterprise network) может соединять как глобальные сети, так и локальные сети. Персональная сеть принадлежит одному владельцу, это может быть один компьютер.

1.4. Коммутация каналов и пакетов

Передача сообщений в сетях производится небольшими порциями, называемых пакетами. Каждому пакету присваивается порядковый номер для того, чтобы на приемной стороне было понятно, какой пакет за каким следует. Размер пакета в сетях разных технологий различается, ориентировочно это 1000-1500 байт.

В сетях, построенных на основе телефонных сетей, передача пакетов происходит по линиям связи, которые устанавливаются в результате телефонного соединения. Соединение устанавливается не для одного пакета, а на все время сеанса связи. Поэтому все пакеты в сеансе следуют один за другим по одним и одним и тем же линиям связи. По завершении сеанса проложенный маршрут разрывается, а освободившиеся линии используются для установления новых соединений. Такой способ передачи пакетов называется коммутацией каналов (рисунок 5).

Рисунок 5 — Коммутация каналов

Если во время сеанса связи по линиям ничего не передается, линия простаивает, а трафик сети имеет пульсирующий характер. Заметим также, что обрыв канала прерывает связь. Для повышения надежности сети и эффективности использования линий связи для компьютерных сетей был разработан метод коммутации пакетов (рисунок 6).

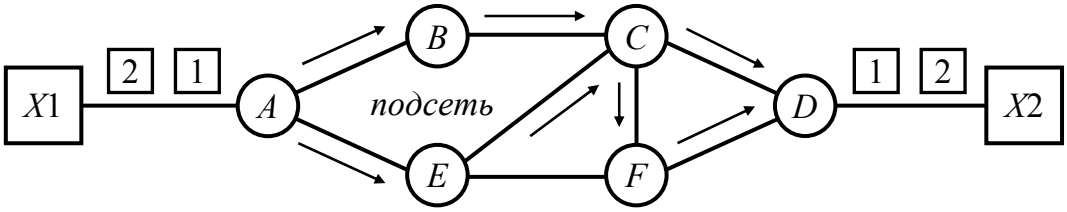


Рисунок 6 — Коммутация пакетов

При коммутации пакетов каждый пакет следует своим маршрутом. Один пакет, например, проходит маршрутом AECFD, а другой — ABCD. Скорость доставки пакетов при этом разная, и пакеты могут перепутываться на приемной стороне, однако обрыв одной линии не нарушает возможность доставки. Если пакет не удалось доставить, он посылается повторно, а маршрут для него прокладывается заново. На приемной стороне пакеты упорядочиваются.

1.5. Передача пакетов

В самом общем случае сети делятся на широковещательные и сети с передачей от узла к узлу. В широковещательных сетях для передачи используется единый канал связи, совместно используемый всеми устройствами сети (рисунок 7).

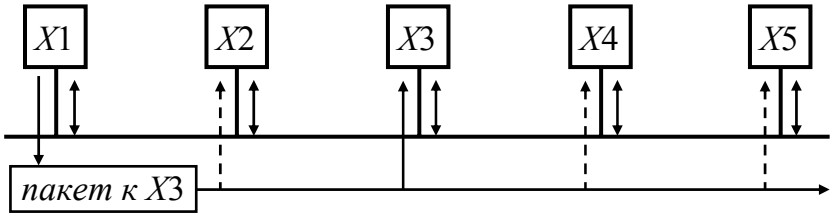


Рисунок 7 — Широковещательная сеть

Пакет, посылаемый в сеть одним из узлов, получают одновременно все устройства, с учетом задержки распространения сигнала. В пакете записан адрес. Узел, адрес которого совпадает с адресом пакета, помещает пакет в свой буфер и обрабатывает его, а другие узлы игнорируют пакет. Передача пакетов разными узлами происходит попеременно.

В широковещательных сетях пакет может быть адресован одновременно всем узлам при помощи специальной формы адреса. Такие пакеты принимаются и обрабатываются всеми машинами, а передача называется широковещательной.

В некоторых широковещательных сетях есть возможность посылать пакет подмножеству узлов. Тогда эти узлы приписываются к группам рассылки, и такая передача называется *многоадресной*.

Сети с передачей от узла к узлу состоят из множества попарно соединенных станций (рисунок 8). В такой сети пакет делает прыжок, называемый *хоп* (hop), от одного узла к другому, пока не достигнет пункта назначения, определяемого адресом, записанным в пакете.

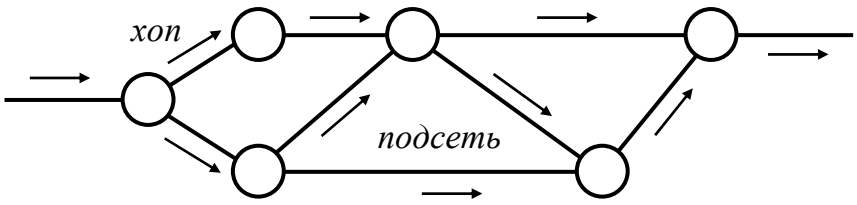


Рисунок 8 — Сеть с передачей от узла к узлу

Поскольку часто существует несколько путей, по которым пакет может достичь цели, в этих сетях одной из основных задач является выбор маршрута движения пакета, — задача маршрутизации. Она решается при помощи различных протоколов и алгоритмов.

1.6. Топологии сетей

Топологией сети называют способ организации физических связей между узлами (компьютерами). В ЛВС используется одна из топологий под названием «общая шина», «кольцо» или «звезда» (рисунок 9).

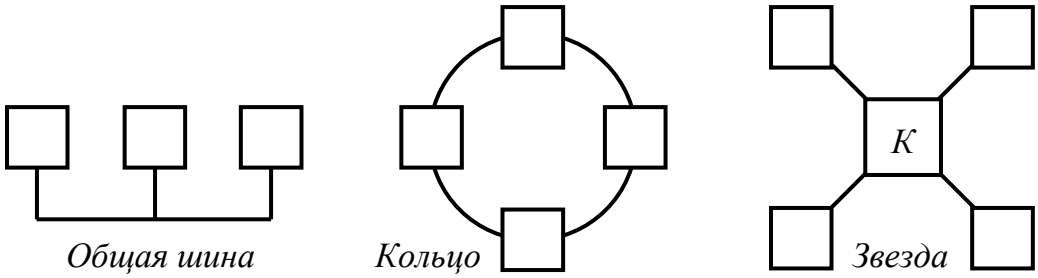


Рисунок 9 — Топологии локальных сетей

В топологии «общая шина» все узлы присоединяются к одному общему проводнику. Это самая простая и дешевая топология. В топологии «кольцо» узлы соединены проводниками по кругу. Недостатком общей шины и кольца является отказ всей сети при отказе общего проводника.

В топологии «звезда» узлы присоединяются к одному общему устройству — концентратору (обозначен буквой К). В современных ЛВС используется топология «звезда», а также «иерархическая звезда», в которой несколько концентраторов соединяются между собой. Эта топология похожа на структуру телефонной сети, и обладает большей надежностью и безопасностью.

В подсетях в узлах находится коммутационное оборудование. Целью подсетей является обеспечение множества связей между узлами для повышения надежности. Поэтому в подсетях чаще образуются произвольные топологии — полносвязная или ячеистая (рисунок 10).

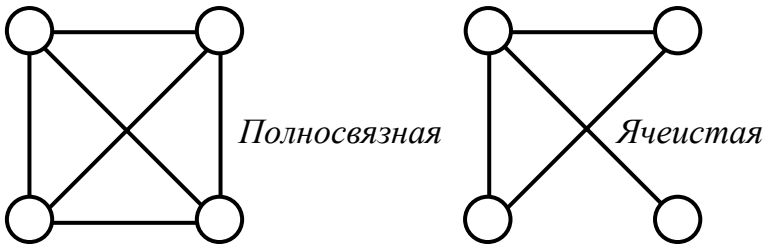


Рисунок 10 — Топологии глобальных сетей

Если каждый узел сети соединен с каждым другим узлом линией связи, топология полносвязная. Ее недостатки — большое число связей и необходимость иметь множество входов в узле. Вместо полносвязной топологии обычно получается ячеистая топология, когда часть связей в полносвязной топологии отсутствует.

1.7. Ethernet

Согласованный набор протоколов и реализующих их программно-аппаратных средств, достаточный для построения базиса вычислительной сети, называется сетевой технологией. Самой распространенной технологией ЛВС является Ethernet. Другие известные технологии ЛВС — это Token Ring (маркерное кольцо) и FDDI (Fiber Distributed Data Interface, волоконно-оптический распределенный интерфейс данных).

Ethernet разработан в компании Xerox, получил распространение, и в 1978 г. компании DEC, Intel и Xerox разработали стандарт DIX, описывающий Ethernet, работающий на скорости 10 Мбит/с, с использованием толстого коаксиального кабеля в качестве линий связи. С небольшими изменениями этот стандарт в 1983 г. был преобразован в стандарт IEEE 802.3. Соединение компьютеров в сети Ethernet показано на рисунке 11.

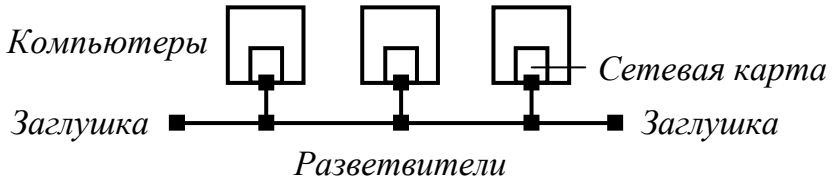


Рисунок 11 — Сегмент Ethernet на коаксиальном кабеле

Ethernet — это широковещательная сеть топологии «общая шина» со случайным методом доступа. Все компьютеры (хосты) своими сетевыми картами присоединены при помощи разветвителей к одному общему кабелю. На концах коаксиального кабеля должны быть заглушки.

Передача данных ведется кадрами Ethernet. Вести передачу кадра может любой хост в любое время. Сначала хост прослушивает линию на предмет того, что на линии ничего не передается. Если линия свободна, хост начинает передавать свой кадр. При этом может случиться, что два хоста одновременно начнут передачу. Эта ситуация называется коллизией, и является для сети Ethernet обычным, нормальным явлением. При обнаружении коллизии оба хоста прекращают передачу. Затем они выдерживают некоторое случайное время, и повторяют попытку.

Для указания адресата используется MAC-адрес сетевой карты. Тот хост, адрес которого совпадает с адресом в кадре, считывает кадр с линии во внутренний буфер, другие хосты ничего не предпринимают.

В современных сетях Ethernet в качестве среды передачи используются кабели на основе витой пары и оптоволоконные кабели, обладающие лучшими характеристиками, а вместо общей шины используется топология звезды. Современные сети Ethernet работают на скоростях 100 Мбит/с (Fast Ethernet), 1000 Мбит/с (Gigabit Ethernet) и выше.

1.8. Структуризация сетей

1.8.1. Сетевое оборудование

Сеть передает информацию при помощи линий связи и передающих устройств. Одни устройства усиливают сигнал, другие структурируют сеть, третьи выбирают маршрут движения. Все вместе называется сетевым оборудованием, которое делится на активное и пассивное.

Все то, что является частью сети, но работает без источника питания, является пассивным оборудованием. Это собственно линии связи и различные соединительные устройства (разъемы).

Активное оборудование имеет электронные схемы, оно может не только усиливать, но и преобразовывать сигналы. Активные устройства сети могут быть управляемыми и неуправляемыми. Алгоритм работы неуправляемого устройства жестко задан. Управляемые устройства могут быть настроены на выполнение разных алгоритмов, и часто имеют встроенную операционную систему.

Повторитель (repeater) просто усиливает передаваемый сигнал. Это устройство уровня L1 позволяет увеличить длину линий связи.

Концентратор или хаб (hub) также усиливает сигнал, но имеет несколько портов (от 4 до 32), позволяя структурировать, разветвлять сеть.

Мост (bridge) локализует трафик, направляя его в только тот сегмент, для которого предназначен пакет. Это неуправляемое устройство уровня L2 с одним процессором, с 2-4 портами.

Коммутатор (switch, switching hub) — это мост, имеющий процессор на каждом своем порту. Часто коммутатор управляется встроенной операционной системой. Некоторые коммутаторы могут выполнять функции маршрутизации, работая на уровне L3. Имеют 4-32 порта.

Маршрутизатор (router) соединяет разные сети или сегменты сетей. Его основное назначение заключается в выборе маршрута движения передаваемых данных. Фактически это специализированный компьютер (или компьютер, выполняющий функции маршрутизации). Это управляемое устройство уровня L3. Маршрутизаторы многообразны, как и их функции. Одни маршрутизаторы предназначены для локальных сетей, другие для магистральных. Первые имеют, как правило, два порта и несколько слотов для расширения, вторые имеют несколько портов для формирования многосвязной топологии. Пограничные маршрутизаторы, соединяющие подсети, имеют два порта, но разнообразные интерфейсы.

Маршрутизаторы часто выполняют функции межсетевых экранов, а беспроводные маршрутизаторы, обычно используемые в квартирах или небольших офисах, позволяют построить небольшую локальную сеть.

В отличие от других устройств, которые являются прозрачными для сети, маршрутизаторы являются основными ее узлами, между которыми происходит передача, и каждый порт маршрутизатора имеет свой адрес. Пакеты, проходящие по сети от одного хоста к другому, фактически совершают прыжки от одного маршрутизатора к другому, пока не достигнут маршрутизатора, с которым прозрачно связан хост назначения.

Межсетевой экран (firewall) — это специализированный маршрутизатор или компьютер, фильтрующий пакеты при их передаче из одной сети в другую.

Шлюз (gateway) — это маршрутизатор, соединяющий сети, использующие разные технологии.

Модем (modem) занимается модуляцией и демодуляцией данных для того, чтобы их можно было передать по определенной линии связи. Модем является промежуточным устройством между DTE и сетью. Это управляемое устройство уровня L1 без операционной системы.

Мультиплексор соединяет несколько потоков данных в один с целью уплотнения канала передачи данных. Демультиплексор выполняет обратное разъединение потоков.

1.8.2. Структуризация сети Ethernet

При построении протяженной локальной сети Ethernet возникают проблемы, связанные с затуханием сигнала в линиях и с возникновением множества коллизий, снижающих производительность. Поэтому в сетях Ethernet существуют ограничения по длине сегмента общей шины и количеству подключаемых к сегменту хостов. Эти ограничения формулируют в виде «5-4-3», что означает максимум 5 сегментов, максимум 4 повторителя, максимум 3 нагруженных сегмента. При этом к одному сегменту можно подключить не более 30 хостов (рисунок 12).

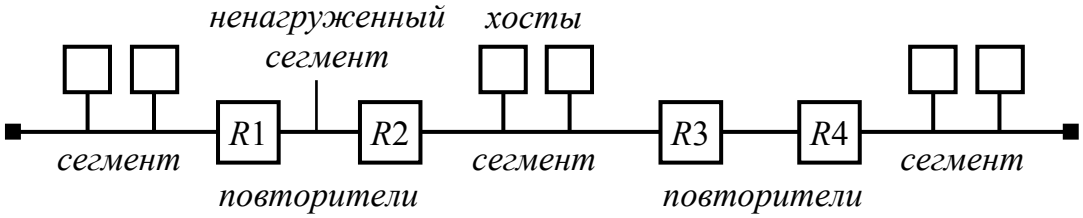


Рисунок 12 — Сегменты и повторители Ethernet

Повторитель передает кадры из одного сегмента в другой (и обратно), улучшая его форму и амплитуду. Длина одного сегмента зависит от кабеля. На толстом коаксиальном кабеле максимальная длина сегмента равна 500 м, а общая длина сети 2500 м. На тонком коаксиальном кабеле длина сегмента не должна превышать 185 м.

Технология Ethernet на толстом кабеле обозначается 10Base-5, на тонком кабеле — 10Base-2, где 10 обозначает скорость передачи в Мбит/с, 5 обозначает 500 м, 2 — 185 м.

Современные сети строятся на основе кабелей на витой паре. При этом изменяется физическая топология подключения хостов, но логическая топология остается прежней — общая шина (рисунок 13).

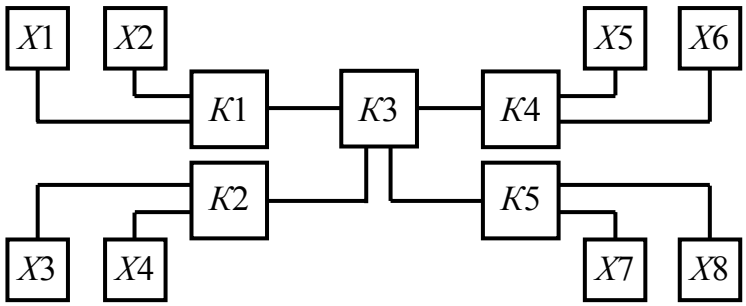


Рисунок 13 — Концентраторы Ethernet

Физическая топология схемы «звезда» формируется при помощи концентраторов. Сигнал, пришедший на один из портов концентратора, повторяется на всех других портах. Устройство называется концентратором потому, что оно соединяет линии от хостов и других концентраторов в одну точку, которая и является общей шиной.

1.8.3. Логическая структуризация сети

Рассмотрим, как работает сеть Ethernet, приведенная на рисунке 13.

Пусть хосты X1 и X2 интенсивно обмениваются информацией. Поскольку сеть является широковещательной, все другие хосты вынуждены простаивать. Если предположить, что группы хостов, соединенные с коммутаторами К1-К4 образуют локальные объединения подразделений (отделов), то было бы удачным решением локализовать трафик этих групп в пределах ограниченной области. Эта задача может быть решена при помощи мостов и коммутаторов.

Мост анализирует проходящий через него трафик и запоминает, какие адреса хостов находятся в одном сегменте сети, а какие — в другом. Когда на мост поступает пакет от хоста X1 к хосту X2, мост запоминает адрес хоста X1, и в дальнейшем направляет кадры, адресованные хосту X1, только на соответствующий порт. При этом кадр не появляется на других портах моста, и в других сегментах сети.

Таким образом, при помощи мостов и коммутаторов происходит логическая структуризация сети, — разбиение сети на сегменты, в которых трафик локализован.

Еще одним устройством, локализующим трафик, является маршрутизатор. Маршрутизатор использует для управления пакетами адрес, записанный в пакете. Он направляет пакеты в соответствующие сегменты, и определяет оптимальный маршрут в сети с несколькими маршрутизаторами, соединенными произвольным образом.

В случае, если нужно соединить сети, построенные по разным технологиям, например, Ethernet и Token Ring, используется многопротокольный маршрутизатор, или шлюз (рисунок 14).

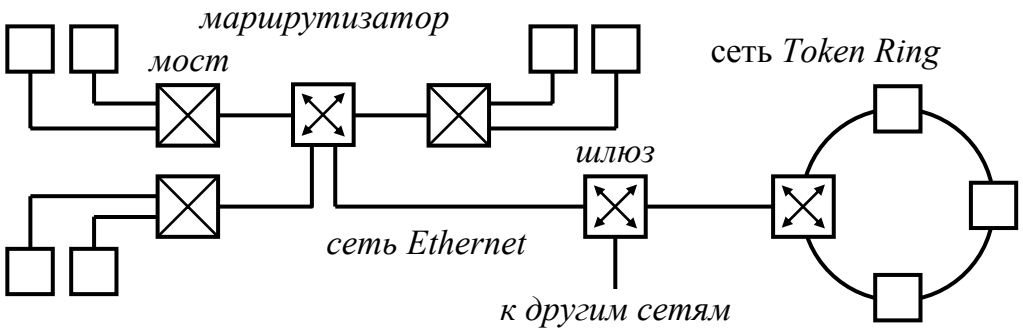


Рисунок 14 — Логическая структуризация сети

Многопротокольный маршрутизатор не только пересылает пакеты из одной сети в другую, но и при необходимости переформатирует их.

1.9. Адресация в сетях

Для передачи данных в сети используются адреса. К адресу предъявляются такие требования, как уникальность в сети любого масштаба, компактность представления, удобство использования, иерархическая структура. В сетях используются числовые и строковые адреса. Числовые адреса удобнее для программного обеспечения и аппаратуры. Для человека удобнее адреса в виде строки.

В локальных сетях Ethernet используется MAC-адрес (media access control, управление доступом к среде), называемый также аппаратным. Он присваивается активному оборудованию на заводе, состоит из шести байт, первые три байта — код производителя, следующие три байта — уникальный номер устройства, например, 90-E6-BA-DA-F0-61. Команда getmac показывает MAC-адреса устройств вашего компьютера.

Для маршрутизации используется IP-адрес. IPv4 — это 32-битное число из четырех групп, например 127.0.0.1, IPv6 — 128-битное число из 8 групп, например, 2001:0db8:11a3:09d7:1f34:8a2e:07a0:765d.

Для адресации в Web используется DNS-адрес (domain name system), состоящий из имен доменов через точку, например, revol.ponocom.ru. Команда ipconfig /all показывает IP и DNS-адреса.

2. Стеки протоколов

Сети являются сложными аппаратно-программными системами, для построения которых используют многоуровневый подход.

2.1. Иерархия протоколов

Сетевое программное обеспечение организуется в наборы уровней или слоёв, каждый из которых отвечает за определенный аспект передачи данных (рисунок 15).

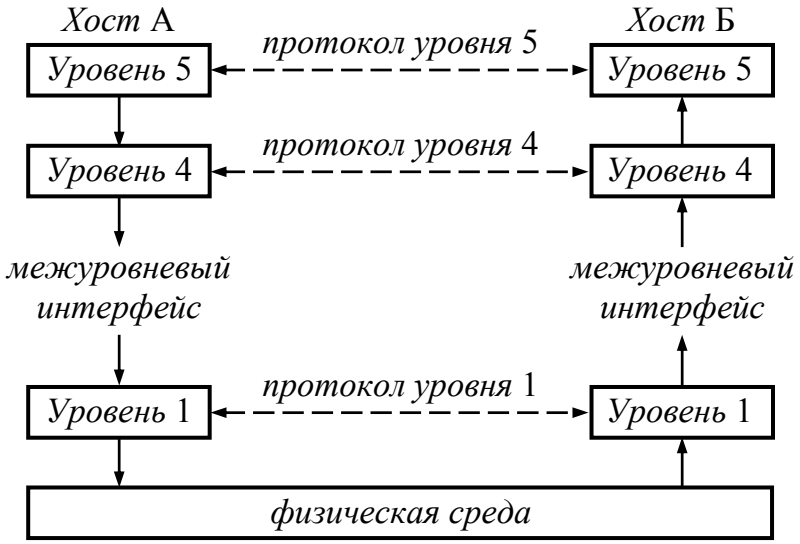


Рисунок 15 — Иерархия уровней

Набор уровней и протоколов называется архитектурой сети. Список протоколов, по одному на уровень, называется стеком протоколов.

Элементами иерархии являются сервисы, интерфейсы и протоколы.

Сервисы — это службы уровня, образующие его сущность. Их можно рассматривать как набор функций, реализованных в процессах уровня. Между каждой парой уровней располагается межуровневый интерфейс, изолирующий уровни друг от друга. Он определяет сервисы, которые нижележащий уровень предоставляет вышележащему уровню.

Протоколы связывают сущности одного уровня. Протокол — это договоренность о том, как, в каком порядке происходит общение.

Сущности одного уровня разных машин называются одноранговыми. В процессе обмена данными между одноранговыми сущностями происходит виртуальное общение в соответствии с протоколом. Данные перемещаются от уровня к уровню с использованием сервисов, предоставляемых межуровневыми интерфейсами.

На самом нижнем уровне находится физическая среда, по которой происходит реальная передача бит.

Роль интерфейсов и протоколов можно проследить на схеме общения директоров двух компаний (рисунок 16).

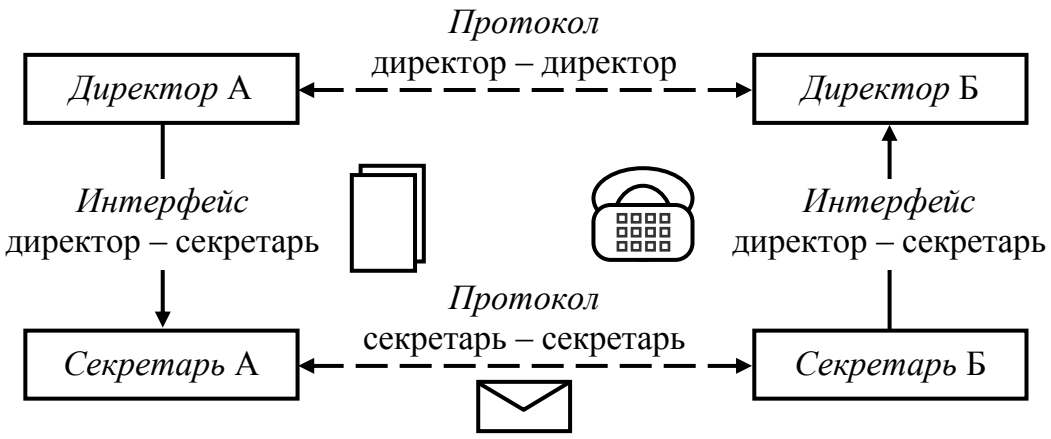


Рисунок 16 — Протоколы и интерфейсы

Между компаниями А и Б существуют договор о поставке продукции от Б к А. Директор А регулярно посылает директору Б запрос на поставку продукции, директор Б в ответ посылает заявку на требуемое количество продукции. Директоры выполняют установленный порядок обмена — протокол «директор-директор».

В компании А обмен документами между директором и секретарем производится при помощи специальной папки, а директор Б общается с секретарем по телефону. Интерфейсы «директор-секретарь» в компаниях А и Б, таким образом, разные.

После того, как директор передал документ секретарю, он «забывает» о нем, а передачей документа занимается секретарь. Для пересылки используются, например, почтовые конверты, или нарочный.

При передаче информации между уровнями сети к пересылаемым данным на каждом уровне могут добавляться дополнительные служебные поля в соответствии с протоколом уровня (рисунок 17).

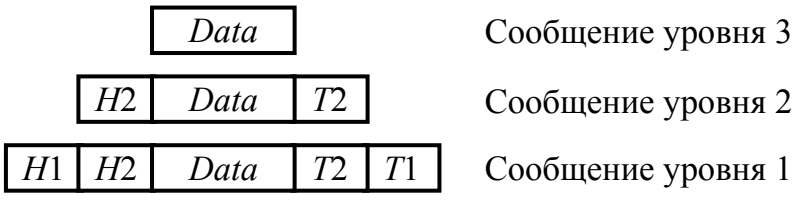


Рисунок 17 — Вложенность сообщений

Здесь H2 и T2 — это заголовок (header) и хвост (trailer), добавленные уровнем 2. Заголовок H1 и хвост T1 добавлены уровнем 1. При движении вверх по иерархии каждый уровень удаляет свои служебные поля, при этом происходит виртуальное общение уровней. Добавление служебных полей уровней называется инкапсуляцией.

2.2. Сетевые службы

Сетевые уровни предлагают вышестоящим уровням услуги двух типов: с установлением соединения и без установления соединения.

Примером сервиса с установлением соединения является телефонная связь. Чтобы поговорить с кем-нибудь, нужно набрать номер, в результате чего происходит соединение телефонных линий одного и другого абонентов. При этом, независимо от того, разговаривают ли абоненты, или нет, соединение телефонов не нарушается до момента, когда один из абонентов не повесит трубку. Использование линий связи установленного соединения для передачи других разговоров невозможно.

В компьютерных сетях для установления соединения используются специальные протоколы, предполагающие обмен служебными пакетами между инициатором передачи и приемником, в результате чего стороны договариваются об условиях передачи и устанавливается канал на все время передачи. Для завершения передачи стороны также обмениваются служебными пакетами, и установленное соединение разрывается.

Примером сервиса без установления соединения является почтовая служба. Мы опускаем письмо в почтовый ящик, а далее оно следует своим маршрутом, пока не достигнет точки назначения. При этом может оказаться, что два письма, опущенных в ящик один за другим с интервалом, скажем, в один день, не обязательно поступят в пункт назначения в том же порядке, и может также случиться, что одно из писем, или оба, затеряются. В компьютерных сетях сервис без установления соединения работает примерно так же: каждый пакет следует по свободным линиям связи по собственному маршруту, при этом доставка пакетов в порядке их передачи не гарантируется, равно как и собственно доставка.

Сетевые службы характеризуются также качеством обслуживания.

Одни службы являются надежными, — они никогда не теряют данных. Надежная служба реализуется посредством подтверждений получения пакетов и, при необходимости, их повторной отправкой. При этом обычно устанавливается соединение, а надежность обеспечивается за счет избыточности пересылаемой информации.

Не всегда при передаче информации требуется высокая надежность. Например, при передаче видео важнее получать данные в реальном времени, при этом потеря части информации не столь важна. Ненадежная служба без подтверждений пакетов называется службой дейтаграмм. Соединение в этом случае не требуется. Иногда используется служба дейтаграмм с подтверждениями, она соответствует отправке заказного письма с уведомлением. Существует также служба запросов и ответов, в которой отправитель посылает дейтаграммы, содержащие запрос, а получатель отправляет ответ. Эти службы обычно используются СУБД.

2.3. Эталонная модель OSI

В 1983 г. международной организацией по стандартизации ISO была разработана «эталонная модель взаимодействия открытых систем ISO», «ISO OSI Reference Model» (OSI — Open System Interconnection). Для краткости ее называют «модель OSI», используется также сокращение ЭМВОС. Модель OSI определяет структуру, назначение и названия уровней, стандартные протоколы уровней.

В модели OSI семь уровней (рисунок 18).

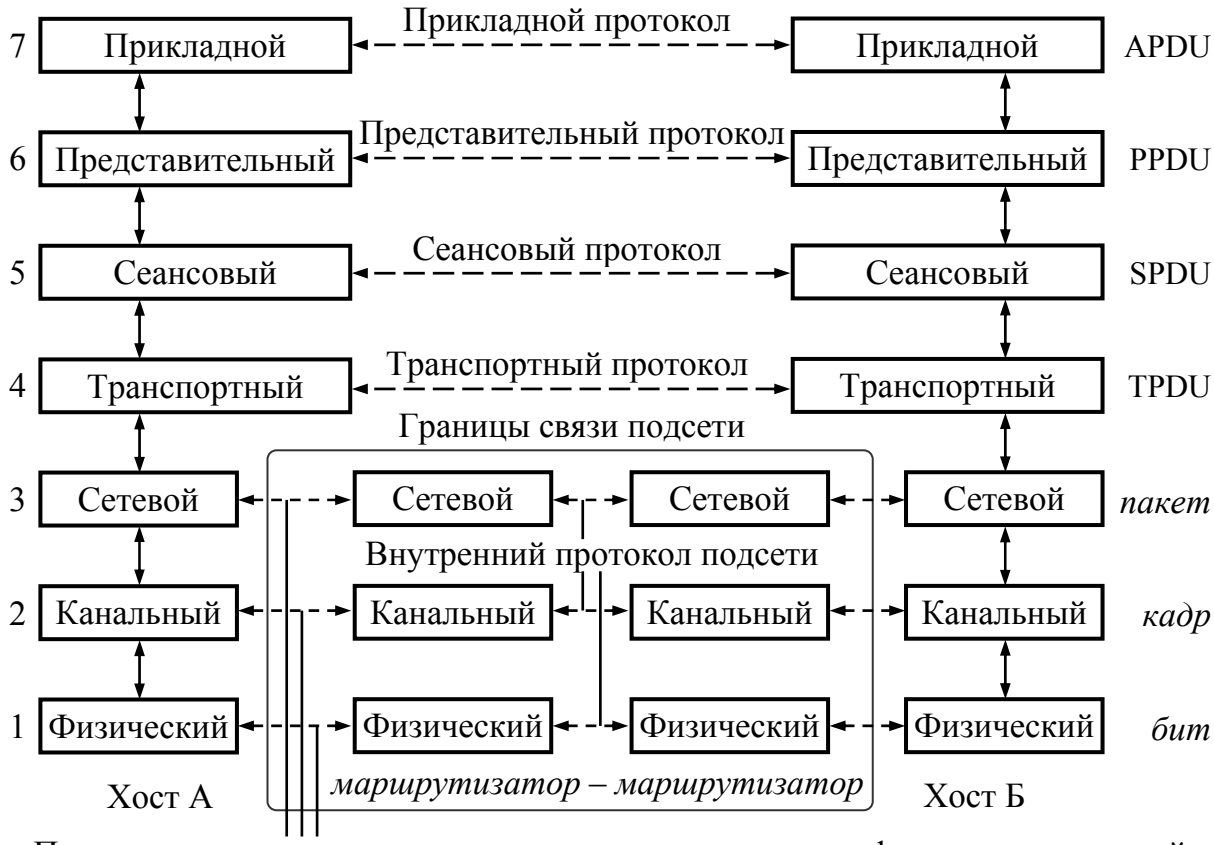


Рисунок 18 — Эталонная модель OSI

Оригинальные названия уровней: Application Layer (прикладной), Presentation Layer (представительный, уровень представления), Session Layer (сеансовый), Transport Layer (транспортный), Network Layer (сетевой или уровень intranet), Data Link Layer (канальный, уровень передачи данных), Physical Layer (физический).

Для каждого уровня определено название единицы передачи данных Protocol Data Unit, PDU. Для сетевого уровня единицей данных является пакет, для канального — кадр, на физическом уровне передаются биты.

На трех нижних уровнях между одноранговыми сущностями сети находится подсеть (intranet), и прямого диалога уровней нет. Сквозной диалог между определен только между уровнями 4–7.

На физическом уровне определяются характеристики среды передачи данных, электрические, механические, оптические. Здесь рассматриваются кабели, разъемы, кодирование, полоса пропускания и т.п.

На канальном уровне решаются задачи: а) предоставление канала в широковещательных сетях и б) контроль и коррекция ошибок. Здесь пересылаемые данные группируются в контролируемые кадры, а уровень обеспечивает передачу кадров с определенной надежностью.

Сетевой уровень соединяет подсети и определяет маршрут движения пакетов. Подсети соединяются маршрутизаторами. Задачей этого уровня является также адресация.

Транспортный уровень обеспечивает доставку данных с заданной надежностью, и изолирует верхние уровни от изменений в технологии доставки. Четыре нижних уровня формируют сетевой транспорт, доставляющий сообщения с заданным уровнем качества в составных сетях с произвольной топологией и различными технологиями.

Сеансовый уровень управляет диалогом для обеспечения очередности передачи данных, синхронизирует действия сторон.

Представительный уровень занимается формой представления данных — синтаксисом и семантикой. Здесь могут выполняться шифрация и дешифрация, преобразование кодировок.

Прикладной уровень является набором широко используемых протоколов, таких, как telnet, ftp, http и других.

Три нижних уровня являются сетезависимыми, три верхних — сетенезависимыми. Транспортный уровень промежуточный, он скрывает детали реализации сетей от верхних уровней. Верхние уровни мало зависят от технических особенностей построения сети. Это позволяет изменять технологии доставки, не заботясь об изменении всего ПО.

Эталонная модель OSI описывает наиболее общую структуру сетей. Она опирается на понятие «открытой системы», означающей открытость спецификаций. Модель OSI и является набором спецификаций, описывающих модули сетевой иерархии и их взаимодействие при помощи протоколов. Открытость спецификаций позволяет сторонним организациям создавать аппаратное и программное обеспечение, и строить сети из компонентов различных производителей.

Примером открытой системы является Интернет, объединяющий самое разнообразное сетевое оборудование и программное обеспечение огромного числа сетей, расположенных в различных географических точках.

2.4. Стек OSI

Модель OSI описывает концептуальную схему взаимодействия открытых систем, а стек OSI является набором спецификаций конкретных протоколов, которые эту схему реализуют. Из-за своей сложности протоколы OSI требуют значительных вычислительных ресурсов, что ограничивает их применение мощными компьютерами.

Стек OSI в точности соответствует модели OSI, — для каждого уровня в нем существуют спецификации протоколов.

На физическом уровне стек OSI поддерживает все существующие виды физических сред, — кабели, оптоволокно и радиоволны.

На канальном уровне стек OSI поддерживает протоколы локальных сетей типа Ethernet, Token Ring, FDDI, а также протоколы глобальных сетей, такие, как X.25, IDSN, ATM.

На сетевом уровне стек OSI предусматривает протоколы маршрутизации ES-IS и IS-IS, где IS обозначает промежуточную систему (Intermediate System), а ES — оконечную систему (End System).

На транспортном уровне в стеке OSI предусмотрено пять протоколов TP0, TP1, TP2, TP3, TP4, предлагающие различное качество услуг.

Протоколы TP0–TP3 обеспечивают сервис с установлением соединения, а протокол TP4 — без установления соединения.

Протокол TP0 выполняет только сегментацию и повторную сборку. Протокол TP1 устраняет базовые ошибки. Протокол TP2 может мультиплексировать и демультиплексировать потоки данных через отдельную виртуальную цепь. Протокол TP3 комбинирует характеристики TP1 и TP2. Протокол TP4 обеспечивает надежные услуги, соответствует протоколу TCP стека TCP/IP.

Протокол сеансового уровня управляет диалогом посредством маркера (token), дающего право на связь.

Протоколом представительного уровня принято считать протокол ASN.1 (Abstract Syntax Notation). Его назначение — выражение форматов данных в независимом от машины формате.

На прикладном уровне определены такие протоколы, как:

- протокол общей информации управления CMIP (Common Management Information Protocol),
- протокол службы каталогов DS (Directory Service) на основе спецификации X.500,
- протокол услуг по передаче файлов FATM (File Transfer Access and Management),
- протокол виртуальных терминалов VTP (Virtual Terminal Protocol),
- протокол, описывающий системы обработки сообщений MHS (Message Handling Systems) и другие.

2.5. Стек TCP/IP

Стек TCP/IP был разработан для создания связи экспериментальной сети ARPANET с другими сетями, как набор общих протоколов разнородной вычислительной среды. Основными протоколами стека являются TCP и IP, давшие название стеку. Этот стек один из наиболее распространенных, он используется в сетях Интернет, а также в локальных сетях, которые в этом случае называют IP-сетями.

Большой вклад в развитие и распространение стека TCP/IP внес университет Беркли, реализовавший протоколы в своей версии операционной системы UNIX. В университете Беркли были также изобретены сокеты, — программные структуры, предназначенные для работы с сетью.

В отличие от стека OSI, в стеке TCP/IP четыре уровня (рисунок 19).

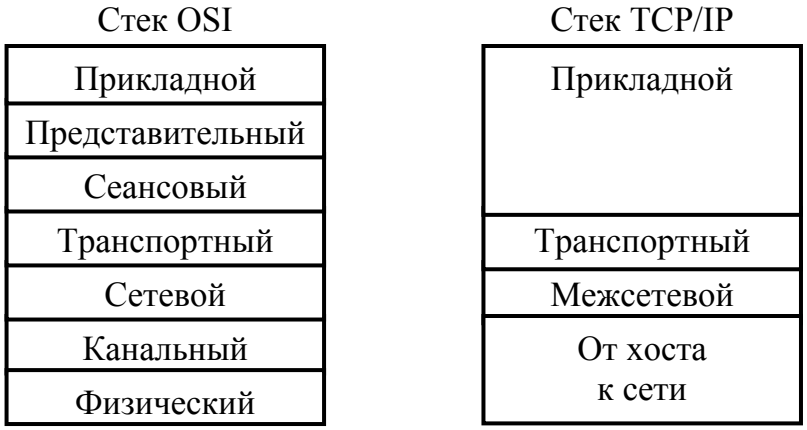


Рисунок 19 — Стеки OSI и TCP/IP

Уровень от хоста к сети описывает соединение хоста с сетью при помощи любого протокола, позволяющего пересылать IP-пакеты.

Межсетевой уровень составляет основу архитектуры. Его задача — обеспечить передачу пакетов в любую сеть. Определяет формат пакета и протокол IP (internet protocol). Этот протокол доставляет IP-пакеты, выполняя маршрутизацию и борьбу с пробками (заторами).

Транспортный уровень обеспечивает прямую связь однорагновых объектов при помощи двух протоколов: TCP (Transmission Control Protocol, протокол управления передачей), надежный протокол с установлением связи, восстановлением кадров и управлением потоком, и UDP (User Data Protocol, пользовательский протокол данных), протокол дейтаграмм без установления связи.

На прикладном уровне располагаются протоколы высокого уровня, такие, как telnet, ftp, smtp, nmtp, dns, http и другие. Представительный и сеансовый уровни модели OSI включены в прикладной уровень, так как особой необходимости в их выделении на практике не возникает.

2.6. Стек IPX/SPX

Это стек протоколов фирмы Novell, разработанный для сетевой операционной системы NetWare в начале 80-х годов XX века. Долгое время эта операционная система являлась лидирующей, а ее стек проколов был основным в локальных сетях организаций.

В основе стека лежат два протокола. Протокол IPX (Internetworking Packet Exchange) соответствует сетевому уровню модели OSI, а протокол SPX (Sequenced Packet Exchange) соответствует сеансовому уровню модели OSI. Стек IPX/SPX реализован во многих операционных системах, таких, как Windows NT, SCO UNIX, Sun Solaris.

2.7. Стек NetBIOS/SMB

Этот стек используется в продуктах IBM и Microsoft.

NetBIOS (Network BIOS), — это расширение стандартной BIOS компьютера IBM PC для сетевой программы PC Network (IBM). В дальнейшем этот протокол был заменен протоколом расширенного пользовательского интерфейса NetBEUI (NetBIOS Extended User Interface). Этот протокол содержит много полезных функций, относящихся к сетевому, транспортному и сеансовому уровням, однако с его помощью нельзя выполнять маршрутизацию, что ограничивает его применение локальными сетями, например, сетями Windows.

Протокол SMB (Server Message Block) выполняет функции сеансового, представительного и прикладного уровней. На его основе реализуется файловая служба, служба печати, служба передачи сообщений.

2.8. Сравнение популярных стеков

В следующей таблице сравниваются уровни различных стеков.

<table>
<tr><td>Модель OSI</td><td>Стек OSI</td><td>TCP/IP</td><td>IPX/SPX</td><td>IBM</td></tr>
<tr><td>Прикладной</td><td>X.400
X.500
FTAM</td><td rowspan="2">TELNET
FTP
SNMP
SMTP
WWW</td><td rowspan="3">NCP
SAP</td><td rowspan="2">SMB</td></tr>
<tr><td>Представительный</td><td></td></tr>
<tr><td>Сеансовый</td><td></td><td rowspan="2">TCP</td><td rowspan="2">NetBIOS</td></tr>
<tr><td>Транспортный</td><td>TP0–TP5</td><td>SPX</td></tr>
<tr><td>Сетевой</td><td>ES-IS
IS-IS</td><td>IP
RIP
OSPF</td><td>IPX
RIP
NLSP</td><td></td></tr>
<tr><td>Канальный</td><td colspan="4">Ethernet, Token Ring, FDDI, X.25, ATM, LAP, PPP</td></tr>
<tr><td>Физический</td><td colspan="4">Кабели, оптоволоконные линии, радиосвязь</td></tr>
</table>

2.9. Стандартизация в области коммуникационных технологий

Стандартизацией сетей занимаются множество организаций. В зависимости от их статуса можно выделить следующие виды стандартов:

- стандарты отдельных фирм;
- стандарты специальных комитетов и объединений;
- национальные стандарты;
- международные стандарты.

Наиболее известные организации, занимающиеся стандартизацией в области вычислительных сетей следующие.

1. Международная организация по стандартизации, ISO.

2. Международный союз электросвязи (International Telecommunications Union, ITU). В рамках этой организации значительную роль играет Международный консультативный комитет по телеграфии и телефонии (Consultative Committee on International Telegraphy and Telephony, CCITT). В 1993 г. в результате реорганизации ITU этот комитет стал называться сектором телекоммуникационной стандартизации ITU-T (ITU Telecommunications Standardization Sector).

3. Институт инженеров по электротехнике и радиоэлектронике (Institute of Electrical and Electronic Engineers, IEEE). В 1981 г. рабочая группа 802 этого института сформулировала основные требования к ЛВС, в результате чего появились стандарты 802.1, 802.2, 802.3, 802.5 и другие.

4. Европейская ассоциация производителей компьютеров (European Computer Manufacturers Association, ECMA).

5. Ассоциация производителей компьютеров и оргтехники (Computer and Business Equipment Manufacturers Association, CBEMA).

6. Ассоциация электронной промышленности (Electronic Industries Association, EIA).

7. Министерство обороны США (Department of Defense, DoD). Наиболее известной разработкой является стек TCP/IP.

8. Американский национальный институт стандартов (American National Standards Institute, ANSI).

Выработкой открытых стандартов Интернет занимаются: профессиональное сообщество Internet Society (ISOC), под управлением которого работает организация Internet Architecture Board (IAB), являющаяся конечной инстанцией при определении новых стандартов Интернет.

В IAB входит группа Internet Engineering Task Force (IETF), занимающаяся решением ближайших технических проблем и разработкой спецификаций, и группа Internet Research Task Force (IRTF), координирующая долгосрочные проекты по протоколам TCP/IP. Стандарты Интернет называются RFC (Request For Comments), каждому такому документу присваивается номер.

3. Физический уровень

На самом нижнем уровне иерархии компьютерных сетей определяются их механические, электрические и временные характеристики. К этому уровню относятся физические среды, используемые для передачи данных, такие, как кабели, оптоволоконные каналы или радиоволны, методы кодирования, оконечное оборудование сетей.

3.1. Гармонический анализ

При передаче дискретных данных по линии связи применяются два вида физического кодирования — на основе синусоидального несущего сигнала, и на основе последовательности прямоугольных импульсов. В первом случае говорят о модуляции сигнала, например, об амплитудной модуляции. Второй способ называют цифровым кодированием. Эти два способа различаются шириной спектра передаваемого сигнала, а также сложностью аппаратуры, которая кодирует и декодирует информацию.

Как известно, любая периодическая функция $y(t)$ с периодом T может быть разложена в ряд Фурье по следующей формуле:

$$y(t) = \frac{1}{2}c + \sum_{n=1}^{\infty} a_n \cos(2\pi nft) + \sum_{n=1}^{\infty} b_n \sin(2\pi nft)$$

где $f = 1/T$ — основная частота, an и b_n — амплитуды синусов и косинусов частоты f_n (n — натуральное число), c — константа. Значения a_n, b_n и c могут быть получены из следующих выражений:

$$a_n = \frac{2}{T}\int_0^T y(t)\cos(2\pi nft)dt,\ b_n = \frac{2}{T}\int_0^T y(t)\sin(2\pi nft)dt,\ c = \frac{2}{T}\int_0^T y(t)dt.$$

Любой периодический сигнал можно представить как сумму синусоид разной частоты, амплитуды и фазы, соответствующих элементам ряда Фурье, и называемых гармониками.

При передаче информации используется тот факт, что на приемной стороне линии связи исходный сигнал можно восстановить, если каналом передается достаточное число гармоник. Для примера рассмотрим передачу кода 01100010 буквы *b*. На рисунке 20 изображен передаваемый сигнал в виде последовательности импульсов, где единице двоичного представления соответствует высокий потенциал, нулю — низкий.

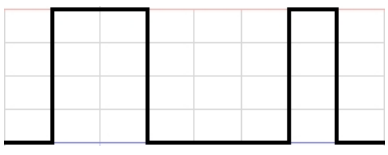
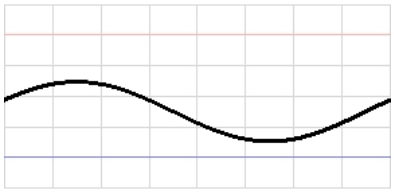
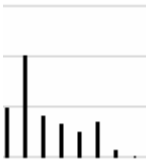


Рисунок 20 — Передаваемый сигнал

На рисунке 21 показаны восстановленные сигналы при количестве передаваемых гармоник 1, 2, 4, и 8. Период T и частота *f* здесь равны 1.

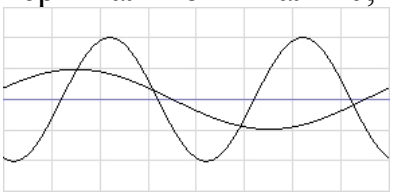


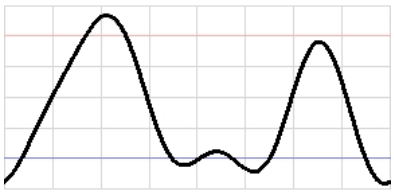
1 гармоника



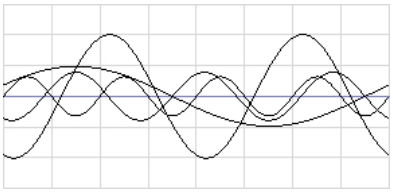
Амплитуды гармоник, цена деления вертикальной шкалы 0,25

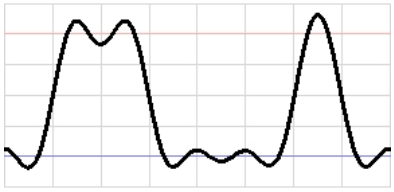
2 гармоники





4 гармоники





8 гармоник

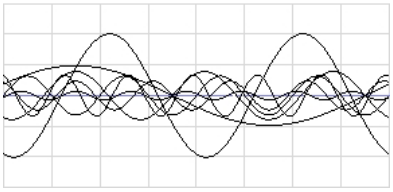


Рисунок 21 — Гармоники передаваемого сигнала

Для данного сигнала значения a_n, b_n, и c вычисляются по формулам:

$$a_n = \frac{1}{\pi n}\left(+ \sin\frac{3\pi n}{4} - \sin\frac{\pi n}{4} + \sin\frac{7\pi n}{4} - \sin\frac{6\pi n}{4}\right),$$

$$b_n = \frac{1}{\pi n}\left(- \cos\frac{3\pi n}{4} + \cos\frac{\pi n}{4} - \cos\frac{7\pi n}{4} + \cos\frac{6\pi n}{4}\right),$$

$$c = \frac{3}{4}.$$

Заметим, что не все гармоники имеют одинаковый вклад в формирование сигнала. Амплитуды гармоник показаны на рисунке 21 справа от первой гармоники. Они определяются по формуле

$$A_n = \sqrt{a_n^2 + b_n^2}$$

Наибольший вклад в данном случае имеет гармоника 2, а гармоники, номер которых кратен 8, вообще не влияют на восстановленный сигнал.

3.2. Полоса пропускания и скорость передачи

Как показано на рисунке 21, по мере роста количества передаваемых гармоник форма сигнала приближается к передаваемой, однако каналы имеют ограничения на максимально возможную полосу передаваемых частот. Поэтому количество информации, которую можно передать в единицу времени по определенному каналу, ограничено, и возникает естественный вопрос о максимально возможной скорости передачи.

Количество информации, передаваемой по линии связи в единицу времени, измеряют двумя способами: а) количеством передаваемых бит в секунду, и б) количеством бод в секунду. При этом 1 Кбит/с соответствует 1000 битам/с, 1 Мбит/с соответствует 1000000 бит/с и т.д.

Бод (baud) — это количество изменений информационного параметра несущего сигнала в секунду (название произошло от фамилии изобретателя телеграфного кода Эмиля Бодо, Baudot). При передаче двоичных сигналов как есть скорость в битах и в бодах совпадают. Однако изменение информационного параметра сигнала может нести более одного бита информации. Например, существуют способы кодирования, в которых одно изменение амплитуды сигнала передает два бита. В этом случае скорость передачи в битах равна двум за одно изменение сигнала, и скорость в бодах равна единице. Скорость в бодах называют также символьной скоростью, которую нужно понимать как количество передаваемых в единицу времени символов.

В 1948 г. Клод Шеннон показал, что пропускная способность линии связи не может превышать некоторого значения, зависящего от отношения «сигнал/шум», обозначаемого SNR (signal-noise relation):

$C = B \log_2 (1 + SNR)$,

где C — пропускная способность, бит/с, B — полоса пропускания, Гц, SNR = S/N, S — мощность сигнала, Вт, N — мощность шума, Вт.

Эта пропускная способность является максимально возможной для линии связи независимо от способа кодирования сигналов. Важное значение в теории связи имеет также формула Гарри Найквиста, который в 1927 г. показал, что максимальная скорость передачи двоичных импульсов по идеальному, без шумов, телеграфному каналу связи не превышает значения, равному удвоенной полосе пропускания:

$C = 2B \log_2 M$,

где M — количество дискретных уровней сигнала. При передаче двоичных импульсов M = 2 (ноль и единица), тогда C = 2B.

Из сравнения формул следует $M = (1 + SNR)^{1/2}$, то есть максимально возможное число уровней приблизительно равно отношению амплитуд сигнала и шума (взятие корня возвращает отношение мощностей к отношению напряжений).

Аналогичные результаты были получены В. Котельниковым, одним из создателей советской секретной связи. В 1933 г. он доказал теорему, согласно которой «любую функцию $F(t)$, состоящую из частот от 0 до f_1, можно непрерывно передавать с любой точностью при помощи чисел, следующих друг за другом через $1/(2f_1)$ секунд [0]» (теорема отсчетов).

По теореме Котельникова непрерывный сигнал $x(t)$ можно представить в виде интерполяционного ряда

$$x(t) = \sum_{k=-\infty}^{\infty} x(k\Delta)\mathrm{sinc}\left[\frac{\pi}{\Delta}(t - k\Delta)\right]$$

где $\mathrm{sinc}(x) = \sin(x)/x$ — функция sinc. Интервал дискретизации находится в пределах $0 < \Delta < 1/(2f_1)$, мгновенные значения ряда есть дискретные отсчеты сигнала $x(k\Delta)$. Теорема доказывает, что возможна полная реконструкция аналогового сигнала по дискретным отсчетам, если частота сигнала не меньше половины частоты дискретизации.

3.3 Телефонный канал

Телефонные каналы использовались в первых компьютерных сетях, и оказали значительное влияние на их развитие. В аналоговом канале частоту сигналов принудительно ограничивают фильтрами, пропуская частоты 300-3400 Гц, и ширина полосы пропускания B равна 3100 Гц. Этой ширины достаточно для передачи разборчивой речи.

Принимая SNR = 125 (что соответствует 21 дБ), по формуле Шеннона получим максимальную скорость передачи цифровых данных в 21600 бит/с. Для реальной коммутируемой телефонной линии это предел. Если предпринять меры, противостоящие помехам, можно добиться устойчивой передачи при SNR = 625 с максимумом скорости в 28800 бит/с.

Принимая частоту дискретизации равной 2400 (частота отсчетов в модемах), максимальная скорость передачи по критерию Найквиста равна $2400 \times \log_2(M)$ бит/с. Для скорости 21600 бит/с получим M = 29, то есть 9 бит на один отсчет, и это очень сложно технически реализовать.

В цифровой телефонии ширина полосы пропускания принимается равной 4 кГц с учетом так называемых защитных полос, и в соответствии с критерием Найквиста и теоремой Котельникова частота дискретизации равна 8 кГц. Период этой частоты равен 125 мкс, и поэтому все временные интервалы в цифровой телефонии кратны этому числу.

Максимальная скорость передачи данных при этом равна 64 Кбит/с, если передается 8 бит данных за один отсчет. Это число также важное, его обозначают DS0 (digital signal 0), он определяет *формат цифровой передачи телефонного звонка*. Заметим, что иногда только семь бит из восьми передают данные, и тогда скорость передачи равна 56 Кбит/с.

3.4. Характеристики линий связи

Важнейшими характеристиками линии связи являются амплитудно-частотная характеристика (АЧХ), пропускная способность, помехоустойчивость, достоверность передачи данных, стоимость и другие.

Амплитудно-частотная характеристика показывает уменьшение (затухание) амплитуды сигнала в зависимости от частоты (рисунок 22).

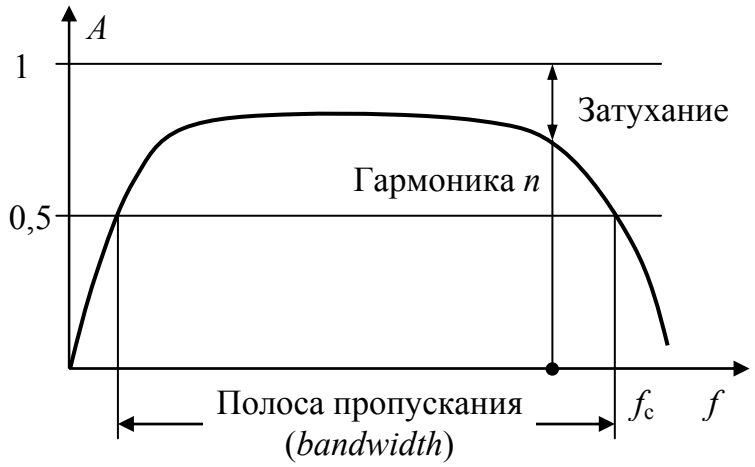


Рисунок 22 — Амплитудно-частотная характеристика

Предполагается, что амплитуда исходного сигнала равна единице.

Затухание измеряется в децибелах, и вычисляется по формуле

$A = 10 \log10(P_{вых}/P_{вх})$,

где $P_{вых}$ — мощность сигнала на выходе линии, а $P_{вх}$ — мощность сигнала на входе линии. АЧХ показывает, что гармоники разной частоты имеют разное затухание, что ведет к искажению сигнала. Принято считать, что затухание –3дБ, соответствующее снижению амплитуды в два раза, не вносит сильных искажений в сигнал.

Таким образом, на амплитудно-частотной характеристике можно выделить полосу частот, называемую полосой пропускания (bandwidth), в пределах которой гармоники передаются более-менее без искажений. Граничную справа частоту f_c называют частотой среза. Гармоники, частота которых превышает f_c, не передаются каналом.

Пропускная способность линии связи (throughput) зависит не только от характеристик линии связи, но и от спектра передаваемых сигналов.

Помехоустойчивость — это способность снижать уровень помех, создаваемых внешней средой. Помехоустойчивость зависит от физической среды, и от экранирующих и подавляющих средств самой линии.

Наиболее помехоустойчивыми являются оптические кабельные системы. Кабели на основе витой пары и коаксиальные кабели обладают хорошей помехоустойчивостью. Для радиолиний характерно сильное

влияние волн друг на друга из-за отражения, поэтому радиолинии обладают самой низкой помехоустойчивостью.

Кроме внешних помех, в линиях связи возникают также внутренние помехи вследствие взаимных наводок параллельных проводников в кабельных линиях, или отраженных волн в радиолиниях.

Для кабельных линий устойчивость к внутренним помехам оценивается параметром «перекрестные наводки на ближнем конце» (Near End Cross Talk, NEXT). Эта характеристика выражается в децибелах и вычисляется по формуле 10 log ($P_{вых}/P_{нав}$), где $P_{вых}$ — мощность выходного сигнала, $P_{нав}$ — мощность наводок.

Достоверность передачи данных характеризует вероятность искажения каждого передаваемого бита данных. Иногда этот показатель называют интенсивностью битовых ошибок (Bit Error Rate, BER). Величина BER составляет примерно 10–4—10–6 для линий на основе медных проводников, и примерно 10–9 для оптоволоконных линий.

3.5. Физические среды передачи данных

Физической средой, передающей сигнал, является медный кабель, оптоволоконный кабель, или радиоволна.

Кабель состоит из медных проводников, изоляции, слоев экрана, слоев уплотнения, вспомогательных элементов, оболочки. Один кабель может содержать несколько пар проводников. Например, кабель телефонной сети общего пользования содержит несколько десятков пар проводников, а коаксиальный кабель содержит только одну пару проводников (центральный медный проводник и экранирующую оплетку).

Наиболее важные электрические характеристики кабеля:

1. Затухание (Attenuation). Измеряется в децибелах на метр длины для определенной частоты сигнала.

2. Импеданс (волновое сопротивление) — полное (активное и реактивное) сопротивление в электрической цепи. Измеряется в Омах.

3. Активное сопротивление — сопротивление постоянному току в электрической цепи. Измеряется в Омах на метр длины.

4. Емкость — способность проводников накапливать энергию. Высокое значение емкости в кабеле приводит к искажению сигнала.

5. Перекрестные наводки на ближнем конце (NEXT). Измеряются в децибелах на метр для определенной частоты.

6. Уровень внешнего электромагнитного излучения или электромагнитный шум — нежелательное переменное напряжение в проводнике.

7. Диаметр или площадь поперечного сечения проводника. Определяет его активное сопротивление.

3.5.1. Кабели на витой паре

Кабели на витой паре описываются стандартом EIA/TIA-568A. Они представляют собой пары скрученных изолированных медных проводников, помещенные в пластиковую оболочку. В сетях часто применяют неэкранированные кабели, маркированные UTP (unshielded twisted pair), в которых в одной оболочке может находиться две или четыре пары.

Отдельные пары имеют общепринятую цветную маркировку. Одна из пар имеет коричневый цвет (при этом один проводник имеет коричневый изоляционный слой, а второй — бело-коричневый), другие пары имеют зеленый, синий и оранжевый цвета. Половина пар используются для передачи дискретных данных в две стороны, другая половина пар — для передачи аналоговых данных тоже в две стороны.

Кабели UTP делятся на категории, обозначаемые CATN*x*, где N — номер категории от 1 до 7, *x* — дополнительная буква. Для присоединения к устройствам на концах кабеля обжимают разъемы, имеющие обозначение 8P8C, называемые обычно RJ45.

Примерная пропускная способность кабелей UTP разных категорий приведена в следующей таблице.

Категория	Пропускная способность, Мбит/с
1	0,1
2	1
3	16
4	20
5	100
6	250
7	600

Характеристики кабеля определяют максимальную его длину, при которой затухание сигнала не превышает допускаемого значения. Для широко используемого в локальных сетях кабеля UTP категории 5 эта длина составляет не более 100 м на частоте 100 МГц, а величина затухания при этом не превышает –22 дБ. Напомним, что величина затухания разная на разных частотах.

Кабели категории 2 использовались для соединения с терминалами, а также в старых сетях Token Ring и ArcNet. Кабели категории 1 и 3 используются в телефонной связи, хотя кабели категории 3 могут быть использованы в Ethernet (10Base-T4).

В современных ЛВС используются кабели категорий 5–7. В этих кабелях пары скрученных проводников имеют разный шаг скрутки (для уменьшения наводок сигналов одних пар на другие), который примерно равен 12–28 мм.

В кабелях категории 6 все пары дополнительно укладываются по спирали вокруг витого сердечника. Витые пары кабеля категории 7 экранируются, экранируется также сам кабель.

Кабели на основе экранированных витых пар имеют лучшую помехозащищенность при условии правильно выполненного заземления экранов. Они имеют обозначение STP (shielded twisted pair), и используются в сетях IBM, а также в сетях Token Ring и FDDI. Эти кабели делятся на типы Type 1…Type 9, включают в себя также неэкранированный кабель Type 3 и оптоволоконный кабель Type 5. Кабели типа STP имеют волновое сопротивление 150 Ом и являются несовместимыми с кабелями типа UTP.

Современная классификация кабелей на витой паре предусматривает альтернативную маркировку кабелей как UTP, так и STP.

3.5.2. Коаксиальные кабели

Коаксиальный (coaxial — соосный) кабель состоит из центрального, достаточно толстого медного проводника (диаметром около 1-2 мм), вокруг которого располагается толстый (несколько миллиметров) слой изолятора из полиэтилена или фторопласта (тефлона), вокруг которого располагается второй проводник в виде медной оплетки, заключенные в полиэтиленовую оболочку. Второй проводник выполняет роль экрана. Иногда между изолятором и оплеткой размещается экранирующий слой в виде тонкой фольги.

Коаксиальный кабель обладает хорошей помехоустойчивостью, малым затуханием, полосой пропускания до единиц гигагерц, но в современных сетях уступил соперничество кабелю на основе витой пары.

3.5.3. Оптоволоконные кабели

Оптоволоконная система передачи данных состоит источника света, носителя, по которому распространяется световой сигнал, и детектора оптического сигнала. Носителем является сверхтонкое стеклянное волокно диаметром 6–10 или 50–60 мкм. Оптическое волокно помещается в стеклянную оболочку с более низким коэффициентом преломления, и свет целиком отражается от этой оболочки внутрь проводника вследствие эффекта полного внутреннего отражения. Оболочку оптоволокна помещают в защитную пластиковую оболочку, несколько таких оптических проводников помещают в общую внешнюю оболочку.

Различают одномодовые и многомодовые оптические волокна. Мода (mode) — это режим распространения света, отличающийся углом падения и отражения, и другими характеристиками.

Если по оптическому волокну одновременно распространяется несколько лучей с разными углами, оно называется многомодовым (Multi Mode Fiber, MMF). Если волокно имеет диаметр около 10 мкм и сопоставим с длиной волны, свет распространяется прямолинейно, и тогда говорят об одномодовом волокне (Single Mode Fiber, SMF, рисунок 23).

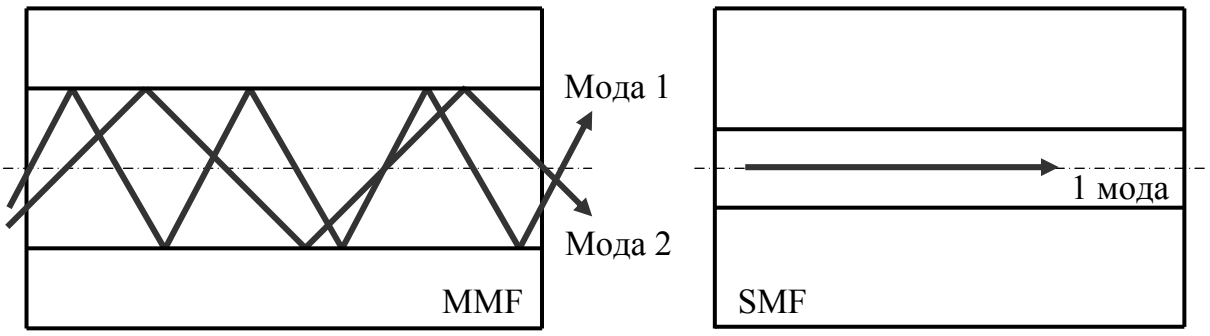


Рисунок 23 — Многомодовое и одномодовое оптическое волокно

Многомодовое оптическое волокно имеет диаметр 50 или 62,5 мкм, а одномодовое — менее 10 мкм (обычно 9 мкм). Оптическая оболочка имеет диаметр 100 или 125 мкм. Полоса пропускания волокна MMF составляет 500–800 МГц/км, а волокна SMF — десятки и сотни гигагерц.

В качестве источника света используют светодиоды и полупроводниковые лазеры. Светодиоды излучают длины волн 850 и 1300 нм, при этом излучатели с длиной волны 850 нм дешевле, но полоса пропускания при этом снижается до 200 МГц/км.

Лазеры излучают длины волн 1300 и 1550 нм. Они позволяют модулировать световой поток с частотой порядка 10 ГГц, кроме того, лазеры создают направленный поток света, за счет чего снижаются потери.

Использование волн длиной 850, 1300 и 1550 нм связано с особенностью амплитудно-частотной характеристики световых волн. Для указанных значений на АЧХ наблюдаются выраженные максимумы. В следующей таблице приведены сравнительные характеристики светодиодных и лазерных излучателей.

Характеристика	Светодиод	Лазер
Скорость передачи данных	Низкая	Высокая
Тип волокна	MMF	MMF и SMF
Расстояние	Короткое	Длинное
Срок службы	Долгий	Короткий
Стоимость	Низкая	Высокая

Приемником (детектором) светового луча является фотодиод. Время срабатывания фотодиода порядка 1 нс, что соответствует скорости передачи данных 1 Гбит/с, и это ограничивает пропускную способность.

Оптоволоконные линии связи не подвержены влиянию электромагнитных помех. К ним сложно подключиться для прослушивания. Однако и соединение оптических волокон требует специального оборудования, при помощи которого оптические кабели сваривают с большой точностью. Оптоволоконные кабели достаточно хорошо гнутся, обладают небольшим весом по сравнению с медными кабелями, а по стоимости незначительно отличаются от кабелей на витой паре.

3.5.4. Беспроводная связь

Беспроводные линии связи состоят из антенн и приемопередающей аппаратуры. Физическим носителем является электромагнитная волна некоторой частоты. На практике вместо частоты используют длину волны, так как длина волны зависит от среды распространения.

Длина волны и ее частота связаны между собой уравнением $\lambda f = c$, где λ — длина волны в метрах, f — частота волны в герцах, c — скорость распространения электромагнитной волны в вакууме, равная примерно 3·108 м/с (скорость света).

Весь электромагнитный спектр делится на участки диапазонов длин волн примерно следующим образом (рисунок 24).

Рисунок 24 — Электромагнитный спектр

В следующей таблице приведены диапазоны электромагнитных волн, соответствующие участкам «радиоволны» и «микроволны».

Диапазон	Длины волн	Диапазон частот	Применение
LF, ДВ	10 – 1 км	30 – 300 кГц	радиовещание
MF, СВ	1000 – 100 м	300 – 3000 кГц	радиовещание
HF, КВ	100 – 10 м	3 – 30 МГц	радиовещание
VHF, УКВ	10 – 1 м	30 – 300 МГц	радио, телевещание
UHF, СВЧ	1000 – 100 мм	300 – 3000 МГц	теле, сотовая связь
SHF, СВЧ	100 – 10 мм	3 – 30 ГГц	компьютерные сети
EHF, СВЧ	10 – 1 мм	30 – 300 ГГц	компьютерные сети
THF, СВЧ	1 – 0,1 мм	300 – 3000 ГГц	компьютерные сети

Волны разной длины обладают разными характеристиками, обуславливающими их применение в той или иной области. Имеет значение распространение, поглощение и отражение волн.

Радиоволны диапазонов ДВ и СВ огибают земную поверхность, и их можно использовать для радиовещания на длинных расстояниях. Радиоволны диапазона КВ поглощаются землей, но отражаются от ионосферы, поэтому связь на длинные расстояния также возможна. Волны более высоких частот распространяются по прямой, поэтому приемная и передающая антенны должны находиться в пределах прямой видимости.

Эти диапазоны мало подходят для сетей в основном потому, что все они уже распределены, и найти здесь достаточный по ширине диапазон частот для передачи данных не представляется возможным. В компьютерных сетях используются микроволны, инфракрасное излучение и видимый свет. Волны, длина которых меньше длины волн видимого света, нельзя использовать, так как они небезопасны для человека.

Некоторые микроволны плохо проходят сквозь здания. Микроволны могут отражаться атмосферными слоями, из-за чего может возникать эффект многолучевого затухания, когда в точку приема поступают прямые и отраженные волны, имеющие разные фазы. На частотах свыше 4 ГГц микроволны поглощаются водой (дождем), что также создает проблемы.

На гигагерцовых частотах работают спутниковые и наземные радиорелейные линии связи. Радиорелейная линия связи состоит из антенн, расположенных в пределах прямой видимости (примерно каждые 50 км), приемо-передающего оборудования и ретрансляторов. Ретрансляторы передают сигнал от одной антенны к другой, таким образом образуется цепочка сегментов прямой видимости. В цепочку ретрансляторов могут включаться также и спутники (рисунок 25).

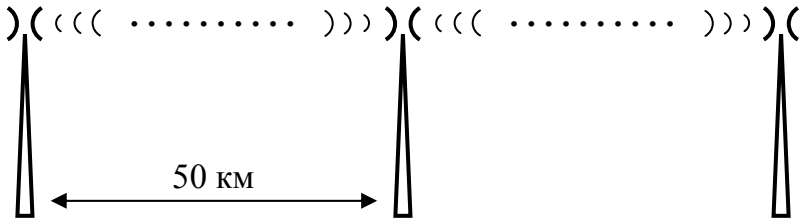


Рисунок 25 — Ретрансляционная линия

На спутниках располагаются транспондеры (transmitter-responder, передатчик-ответчик), принимающие восходящий сигнал на одной частоте, и передающие нисходящий сигнал на другой частоте.

В диапазоне 2,4 ГГц работают сети стандарта 802.11 (Wi-Fi).

Инфракрасные и миллиметровые волны используются для связи на небольшие расстояния (например, в пультах дистанционного управления, в сетях Bluetooth). Волны этих длин не проходят сквозь твердые предметы, что позволяет легко изолировать сети друг от друга.

3.6. Кодирование сигналов

По линиям связи всегда передается аналоговый сигнал, так как носителем информации является электромагнитная волна. Электромагнитная волна модулируется некоторым образом для передачи аналоговой информации, и некоторым другим образом для передачи информации в цифровой форме. Для передачи аналоговой и цифровой информации используется амплитудная, частотная и фазовая манипуляция синусоидального сигнала. Для передачи цифровой информации используется кодирование. Когда говорят, что канал связи цифровой, то имеют ввиду, что передается аналоговый сигнал, модулированный цифровой информацией, при этом сигнал не синусоидальный, имеющий очень широкий спектр, так как требуется передавать несколько гармоник.

3.6.1. Передача данных по аналоговой линии связи

Преобразование цифровых данных в аналоговый сигнал и обратно выполняют модемы, при этом используется аналоговая линия телефонной связи. В модемах используется частота отсчетов, равная 2400, при этом один отсчет передает символ с числом бит, зависящим от метода кодирования. Для кодирования символов используется амплитудная и (или) фазовая манипуляция несущей частоты (рисунок 26).

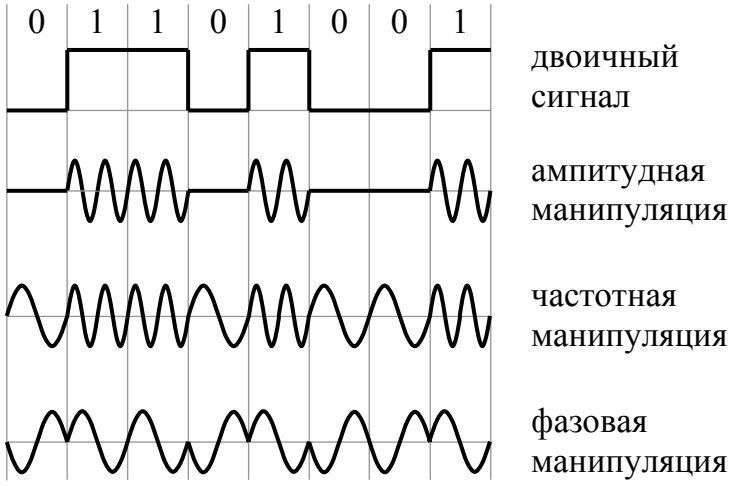


Рисунок 26 — Виды модуляции синусоидального сигнала

Первый модем (1958 г) использовал две разных частоты для передачи нуля и единицы, и имел скорость передачи 300 бит/с (стандарт V.21). Метод кодирования называется FSK (Frequency Shift Keying).

При квадратурной фазовой модуляции (Quadrature Phase Shift Keying, QPSK) 4 разных фазы передают символ из двух бит с получением битовой скорости 4800 бит/с при символьной скорости 2400 бод.

На рисунке 27 показаны амплитудно-фазовые диаграммы (звездчатые диаграммы), где точки соответствуют комбинациям амплитуды и фазы. Левая диаграмма соответствует QPSK.

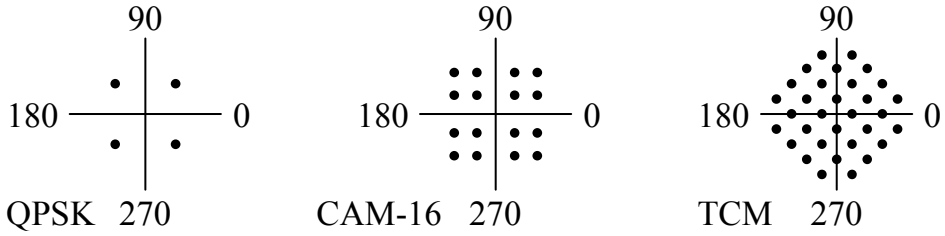


Рисунок 27 — Амплитудно-фазовые диаграммы модемов

В методе QAM-16 (Quadrature Amplitude Modulation) используются одновременно четыре разные фазы и две разных амплитуды, что дает 16 комбинаций и скорость 9600 бит/с. В методе QAM-64 используется 64 разных комбинации, что дает скорость передачи 14400 бит/с. Однако следует учитывать, что при большой плотности точек на диаграмме увеличивается вероятность ошибки. Для коррекции ошибок часть передаваемых бит используется для контроля. Схема решетчатого кодирования TCM (Trellis-Coded Modulation) приведена на рисунке 27 справа, диаграмма повернута по техническим соображениям. Она соответствует стандарту V.32, использует QAM-32 с 32 точками и передает за один отсчет 4 бита данных и один контрольный бит. Скорость передачи данных с коррекцией ошибок составляет 9600 бит/с. Если использовать 128 точек, то можно передать 6 бит данных и 1 контрольный, с получением битовой скорости 14400 бит/с (стандарт V.32bis). Эти скорости можно получить на реальной аналоговой телефонной линии.

Дальнейшее повышение скорости передачи требует сложных технических решений (например, сжатие), теоретически она не может превышать 36600 бит/с, и это с учетом идеального состояния линии, при SNR около 1632 (32 дБ), повышения частоты дискретизации до 3429 Гц, и небольшого увеличения полосы пропускания против 3100 Гц (стандарт V.34bis). Получить такую скорость на реальной линии чрезвычайно сложно, поэтому модемы автоматически переходят на более низкую скорость в случае, если линия связи имеет низкое качество.

Есть еще один момент, касающийся связи не только в модемах. Это симплексная, дуплексная и полудуплексная передача. Передача называется симплексной, если направление передачи по каналу только в одну сторону. При дуплексной связи передача по каналу ведется в обе стороны одновременно. Полудуплексный канал передает данные в обе стороны, но в один момент времени в одну сторону. Большинство протоколов модемов имеют дуплексный режим.

3.6.2. Передача данных по цифровой линии связи

При цифровой передаче используют потенциальные и импульсные коды. В потенциальных кодах для представления нуля и единицы используется значение сигнала в период передачи бита. Импульсные коды представляют ноль и единицу переходом потенциала. Сигнал в виде импульсов имеет бесконечный спектр, но основная энергия сосредоточена в диапазоне частот от 0 до $f = 1/t_0$, где t_0 — длительность импульса.

Метод кодирования должен:

- иметь наименьшую ширину спектра для данной скорости передачи;
- минимизировать величину постоянной составляющей;
- автоматически синхронизироваться;
- быть простым для реализации.

Сначала заметим, что для распознавания потенциала в битовом интервале требуется дополнительный сигнал, называемый тактирующим сигналом или стробом (clock). Он необходим для синхронизации передатчика и приемника. Если можно обойтись без тактирующего сигнала, код называется самосинхронизирующимся.

На рисунке 28 приведены основные потенциальные коды.

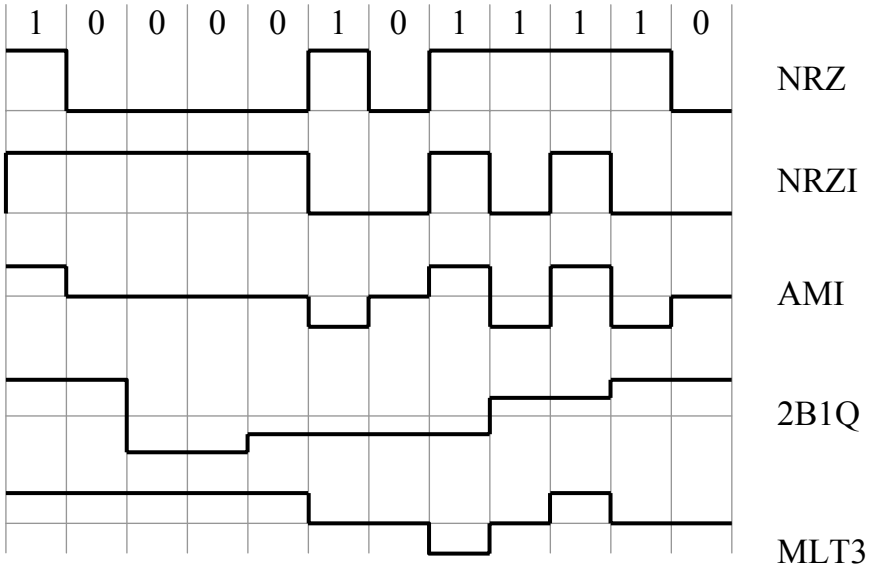


Рисунок 28 — Потенциальные коды

Потенциальный код NRZ (Non Return to Zero, без возврата к нулю) кодирует единицу высоким потенциалом, ноль — нулевым (или наоборот, это несущественно). При передаче последовательностей нулей или единиц сигнал — это постоянный ток, поэтому говорят, что сигнал содержит постоянную составляющую, поэтому код не самосинхронизирующийся. В сетях этот код применяется только в сочетании с логическим кодированием, которое устраняет указанные недостатки.

Потенциальный код NRZI (Non Return to Zero with ones Inverted, без возврата к нулю с инверсией единиц) при передаче нуля не меняет уровень сигнала, при передаче единицы меняет на противоположный. Постоянная составляющая проявляется только при передаче нулей. Используется в USB, в сетях Ethernet (в сочетании с логическим кодированием), в оптоволоконных каналах.

Код AMI (Alternate Mark Inversion, биполярное кодирование с альтернативной инверсией) использует три уровня сигнала. Нулю соответствует нулевой потенциал, единице положительный или отрицательный, при этом каждая новая единица имеет противоположный потенциал. Длинная последовательность нулей сбивает синхронизацию. Код требует более мощного передатчика из-за трех уровней сигнала, но частично выявляет ошибки. Используется в форматах DS1-DS4, в сетях ISDN.

В потенциальном коде 2B1Q два бита кодируются одним четверичным символом (Quarter-nary symbol). В сетях ISDN паре бит 00 соответствует потенциал –2,5В, паре 01 соответствует –0,833В, паре 10 соответствует 0,833В, и паре 11 соответствует 2,5В. Этот код требует мощного передатчика, но занимает меньшую полосу частот.

Код MLT3 (Multi-Level Transmission-3) использует три уровня. При передаче нуля потенциал не изменяется. При передаче единицы потенциал меняется на следующий в цикле: +V, 0, –V. Код не самосинхронизирующийся, имеет постоянную составляющую, но узкую полосу. Применяется вместе с логическим кодированием в 100Base-TX (Ethernet).

Из импульсных кодов рассмотрим манчестерский код (рисунок 29).

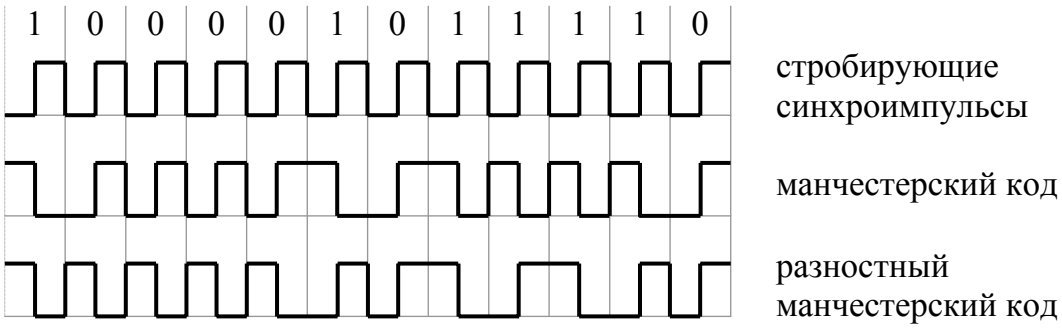


Рисунок 29 — Импульсные коды

Сигнал формируется сложением по модулю два потока бит данных с тактовой последовательностью бит. Переход 0–1 в середине интервала кодирует ноль, переход 1-0 единицу. Код самосинхронизирующийся, но требуется широкая полоса пропускания. Используется в классическом Ethernet. В разностном манчестерском коде ноль кодируется изменением потенциала в начале интервала, единица — его сохранением, в середине обязательный переход. Применяется в сетях Token Ring.

3.6.3. Логическое кодирование

Логическое кодирование используется для улучшения потенциальных кодов типа AMI, NRZI или 2B1Q с целью устранения постоянной составляющей и улучшения синхронизации. Логическое кодирование сначала формирует улучшенные битовые последовательности, которые затем передаются по линиям связи при помощи простых методов физического кодирования. Для логического кодирования применяются избыточные коды и скремблирование

Код **4B/5B** заменяет группу из четырех бит группой из пяти бит, и из пятизначных кодов выбираются те, которые содержат не более двух нулей подряд. Например, группа 0000 заменяется группой 11110, а группа 0001 заменяется группой 01001. Неиспользуемые пятизначные слова используются в качестве служебных. Такая замена гарантирует, что в последовательности бит окажется не более трех нулей подряд, платой за это является снижение скорость передачи на 25%. В сочетании с кодом NRZI такое кодирование применяется в 100Base-FX (оптоволокно), а в сочетании с кодом MLT3 — в 100Base-TX (UTP категории 5e).

Код **8B/10B** заменяет группу из восьми бит группой из десяти бит. В заменяющей последовательности бит нет четырех подряд идущих одинаковых бит, и в ней не более шести подряд идущих нулей или единиц. В сочетании с кодом NRZI используется в 1000Base-X (оптоволокно).

В коде **8B/6T** каждые восемь бит заменяются шестью троичными цифрами "̃", "̣" и "0". Например, 00000000 заменяется на̣ 0̃0̣. Избыточность кода составляет 36/28 = 729/256 = 2,85. Используется в устаревшей спецификации 100Base-T4 (UTP категории 3).

При скремблировании биты данных перемешиваются перед физическим кодированием, при этом используются несколько уже переданных бит. Пример формулы скремблирования:

$B_i = A_i \oplus B_{i-3} \oplus B_{i-5}$.

здесь B_i — передаваемый, A_i — исходный бит. При входной последовательности бит 11000000 получим последовательность 11011000, в которой нет шести подряд идущих нулей (первые три бита не изменяются):

$B_4 = A_4 \oplus B_1 = 0 \oplus 1 = 1$

$B_5 = A_5 \oplus B_2 = 0 \oplus 1 = 1$

$B_6 = A_6 \oplus B_3 = 0 \oplus 0 = 0$

$B_7 = A_7 \oplus B_4 \oplus B_1 = 0 \oplus 1 \oplus 1 = 0$

$B_8 = A_8 \oplus B_5 \oplus B_2 = 0 \oplus 1 \oplus 1 = 0$

Для обратного преобразования используется похожая формула:

$C_i = B_i \oplus B_{i-3} \oplus B_{i-5}$.

3.7. Уплотнение каналов

Физическая линия связи обычно имеет полосу пропускания шире, чем полоса, занимаемая передаваемыми данными, и пропускная способность линии превосходит скорость передачи данных. Это дает возможность передавать по одной линии одновременно несколько потоков. Для примера, в диапазоне FM несколько радиостанций работают, не мешая друг другу, потому что для передачи стерео сигнала требуется примерно 300 кГц, а ширина FM-диапазона составляет примерно 20 МГц. Спектр сигнала распределяется равномерно слева и справа (или с одной стороны) от основной частоты, называемой несущей (поднесущей). Выбирая основные частоты станций с интервалом, немного превосходящим ширину спектра, FM-диапазон можно поделить на независимые каналы.

3.7.1. Уплотнение с разделением частоты

Мультиплексирование с разделением частоты, или частотное уплотнение (FDM, Frequency Division Multiplexing) использовалось в старых аналоговых телефонных сетях. Телефонная линия имеет полосу пропускания минимум 1 МГц, а сигнал тональный частоты занимает полосу всего 3100 Гц. Чтобы по одной линии (кабеля между станциями) передать несколько разговоров, каждому разговору выделяют полосу 4 кГц, с учетом так называемых защитных полос по 450 Гц с каждой стороны. При этом АЧХ каналов частично перекрываются и возможно небольшое влияние каналов друг на друга. По рекомендации CCITT, в диапазоне 60-108 кГц через каждые 4 кГц размещается 12 каналов, называемые первичной группой (рисунок 30).

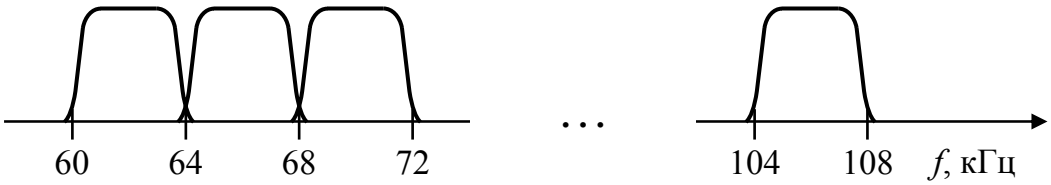


Рисунок 30 — Частотное уплотнение телефонных разговоров

Из первичных групп формируются вторичные группы из 12×5 = 60 каналов, которые размещаются в диапазоне 312-552 кГц. Из вторичных групп могут формироваться третичные группы и т.д. Мультиплексирование выполняет уплотняющая аппаратура на станциях.

В ADSL (Asymmetric Digital Subscriber Line — асимметричная цифровая абонентская линия) полоса шириной 1,1 МГц делится на 256 полос с поднесущими, разнесенными на примерно 4 кГц, метод называется дискретная мультитональная модуляция (Discrete MultiTone, DMT).

Полоса с номером 0 используется как обычный телефонный канал, по которому абонент может разговаривать одновременно с передачей данных. Пять следующих полос не используются, они защищают телефонный канал от интерференции с цифровыми данными. Оставшиеся 250 каналов работают независимо по методу CAM с символьной скоростью примерно 4000 символов/с, каждый со своей скоростью передачи, зависящей от текущего состояния линии (рисунок 31).

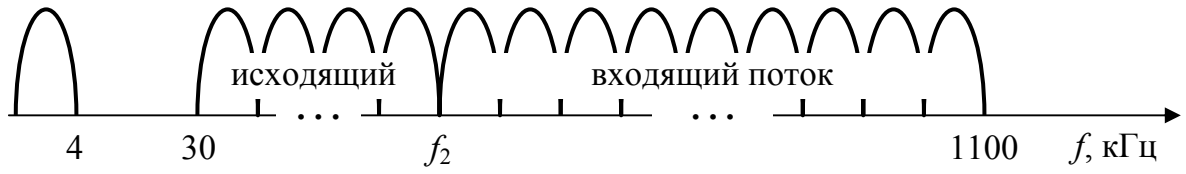


Рисунок 31 — Частотное уплотнение в ADSL

Исходящий трафик несут первые 32 канала, остальные — входящий, поэтому канал ADSL асимметричный. Это обеспечивает скорость до 1 Мбит/с на исходящий трафик и до 8 Мбит/с на входящий. У абонента находится ADSL-модем, на стороне провайдера не аналоговое, а цифровое оборудование. На участке от абонента до провайдера используется существующая аналоговая телефонная линия. Заметим, что ADSL и другие DSL методы не уплотняют канал для нескольких абонентов, а предоставляют его для одного, но здесь присутствует разделение частот.

3.7.2. Уплотнение с разделением времени

Мультиплексирование с разделением времени (TDM, Time Division Multiplexing) используется для уплотнения магистралей в цифровых телефонных сетях. Аналоговый сигнал, приходящий на станцию, оцифровывает устройство кодек (кодер-декодер). АЦП кодека передающей стороны каждые 125 мкс формирует 8-битный отсчет амплитуды. Слева на рисунке 32 точками показаны отсчеты двух синусоид разной амплитуды с частотой 1 Гц и периодом 1 мс, в который умещается 8 отсчетов.

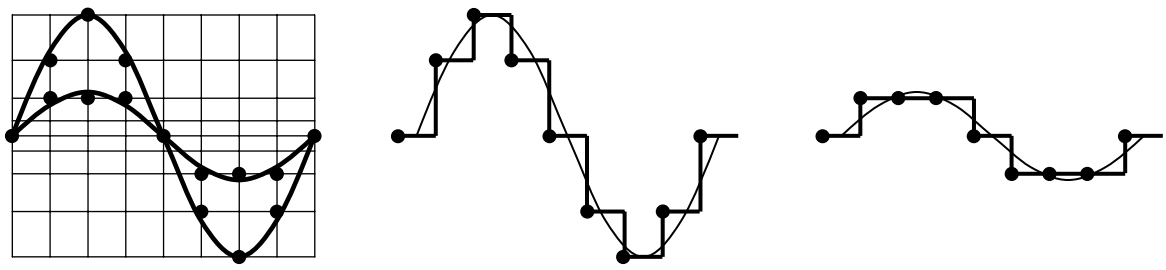


Рисунок 32 — Импульсно-кодовая модуляция

Для наглядности число уровней отсчетов здесь равно 9, а не 256.

На приемной стороне сигнал восстанавливает 8-битный ЦАП. На рисунке 32 справа показаны сигналы на приемной стороне, отсчетам соответствуют горизонтальные черточки. Сигнал восстанавливается при помощи фильтров, которые сглаживают прямоугольный сигнал.

Для сжатия и расширения динамического диапазона, называемого компандингом (companding), используются две формы логарифмического преобразования, называемые μ-закон (μ-law, используется в Северной Америке и Японии), и A-закон (A-law, используется в Европе), которые компенсируют низкую точность 8-битного оцифровывания. Этот метод называется импульсно-кодовой модуляцией (Pulse Code Modulation, PCM, ИКМ), он формирует цифровой канал DS0.

Уплотненный канал называется «носитель T» (T-carrier). Канал T1 мультиплексирует 24 канала формата DS0 (64 Кбит/с), формируя кадры из 24 отсчетов каждого канала каждые 125 мкс, добавляя один управляющий бит на кадр для синхронизации, с получением формата DS1 и суммарной скорости 1,544 Мбит/с (рисунок 33).

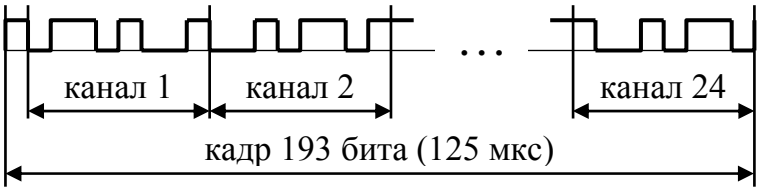


Рисунок 33 — Кадр канала T1

Носитель T2 мультиплексирует 4 T1, T3 мультиплексирует 7 T2, T4 мультиплексирует 6 T3, с получением скоростей передачи данных для DS2 6,312 Мбит/с, для DS3 44,736 Мбит/с, для DS3 274,176 Мбит/с.

В Европе аналогичный стандарт называется E1. Он объединяет 32 канала со скоростью передачи данных 32×64 Кбит/с = 2048 Кбит/с. От него образуются носители E2, E3, E4 и E5, которые мультиплексируют по 4 канала предыдущего уровня. Существуют также носители J.

3.7.3. Кодовое разделение каналов

Мультиплексирование с кодовым разделением (CDM, Code Division Multiplexing, а также CDMA, Code Division Multiple Access) распределяет узкополосый сигнал по широкому диапазону частот. В CDMA битовый интервал разбивается на *m* коротких периодов, называемых чипами (chip), и каждому каналу присваивается m-битный код (*m* принимается равным 64 или 128), так называемая элементарная последовательность.

Канал посылает бит в виде своей элементарной последовательности, при этом нулевой бит получается ее инвертированием (дополнением).

Все каналы одновременно посылают свои элементарные последовательности или их дополнения, при этом формируется спектр, который в m раз шире спектра отдельного канала, и является шумоподобным сигналом (ШПС). Выделение сигнала конкретного канала обеспечивается за счет ортогональности элементарных последовательностей и дополнений. Две последовательности ортогональны, если их скалярное произведение равно нулю: $A \cdot B = \Sigma(A_i B_i) = 0$, $i = 1 \ldots m$, A и B — векторы последовательностей. Заметим, что $A \cdot A = 1$, а $A \cdot a = 0$, если a — дополнение A. Сигнал выделяется вычислением скалярного произведения принятого ШПС и элементарной последовательности канала.

Пусть A, B, C — элементарные последовательности каналов.

Если канал C передает бит 1, то

$(A + B + C) \cdot C = A \cdot C + B \cdot C + C \cdot C = 0 + 0 + 1 = 1$.

Если канал C передает бит 0, то

$(A + B + c) \cdot C = A \cdot C + B \cdot C + c \cdot C = 0 + 0 + 0 = 0$.

Для простоты здесь опущено много деталей. Заметим, что данный метод обладает также хорошей защищенностью каналов. Он используется, например, в радиосвязи, в сотовой и спутниковой связи.

3.7.4. SONET/SDH

В оптических сетях используется уплотнение SONET (Synchronous Optical Network, синхронная оптическая сеть), разработанное исследовательской группой региональных телефонных компаний Bellcore совместно с ITU, а также рекомендации ITU под названием SDH (Synchronous Digital Hierarchy, синхронная цифровая иерархия). SONET использует мультиплексирование с разделением времени, при этом кадр состоит из 6480 бит, передаваемых за 125 мкс. Основной канал SONET называется STS-1 (Synchronous Transport Signal, синхронный транспортный сигнал), со скоростью 51,84 Мбит/с, и все магистрали SONET кратны ему.

Первые 2 байта содержат синхропоследовательность, так как кадры следуют без промежутков. Затем идет 25 байт служебной информации, за которой в произвольном месте кадра могут располагаться данные пользователя, называемые SPE (Synchronous Payload Envelope). Этот контейнер передает произвольные данные со скоростью 50,112 Мбит/с.

Носитель называется OC (optical carrier). STS-1 соответствует носитель OC-1. Существуют STS-n и OC-n, где n равно 3, 9, 12, 48, 192 и 768, с соответствующими скоростями STS-n, SPE и пользователя. В SDH нет носителя, соответствующего OC-1, первый носитель OC-3, ему соответствует канал STM-1, а OC-768 соответствует STM-256. Максимальная скорость передачи данных в этой иерархии достигает почти 40 Гбит/с.

4. Канальный уровень

Задачей канального уровня (уровня передачи данных) является организация доставки данных по физическому каналу. Данные в физическом канале могут искажаться, а скорость распространения данных отлична от нуля, поэтому специфичные для этого уровня функции — это а) обработка ошибок передачи и б) управление потоком для исключения затопления медленных приемников быстрыми передатчиками. Канальный уровень предоставляет сетевому уровню строго очерченный интерфейс. Пакеты, поступающие из сетевого уровня, инкапсулируются в кадры, при этом к пакетам добавляется заголовок и хвост, содержащий последовательность контроля кадра FCS (frame check sequence). На приемной стороне пакет извлекается из кадра и передается сетевому уровню. Управление кадрами выполняет протокол, заголовок которого приписывается к пакету (рисунок 34).

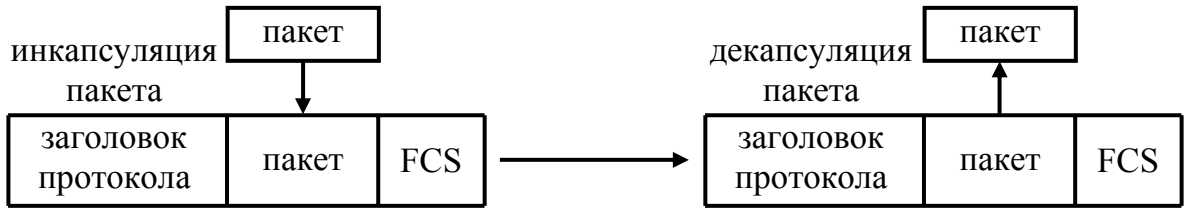


Рисунок 34 — Инкапсуляция пакета в кадр

Сервисы, предоставляемые уровнем передачи данных, зависят от применяемой технологии (и протокола). В общем случае это:

- сервис без подтверждений и без установления соединения;
- сервис с подтверждениями, без установления соединения;
- сервис с подтверждениями и с установлением соединения.

Сервис без подтверждений и без установления соединения предполагает передачу кадров приемнику, при которой доставка кадра никак не удостоверяется и не гарантируется. Если кадр по каким-то причинам потерялся, то это проблема более высокого уровня, а не канального.

Сервис с подтверждениями подразумевает, что приемник на каждый полученный кадр посылает ответный кадр, содержащий подтверждение. Если в течение заданного времени передатчик не получает подтверждения, он посылает кадр повторно. При этом кадры могут дублироваться, а сеть, возможно, загружается дополнительными кадрами.

Установление соединения подразумевает начальный диалог передатчика и приемника, в ходе которого они договариваются об условиях передачи. При этом каждый посылаемый кадр нумеруется, а протокол передачи гарантирует а) доставку ровно одной копии кадра, и б) правильный порядок передачи пакетов сетевому уровню.

4.1. Формирование границ кадров

Физический уровень — это поток бит. Биты могут теряться и (или) искажаться. Одна из причин, по которой данные разбиваются на пакеты и кадры, заключается в том, что выгоднее контролировать небольшой объем данных, чем трафик целиком, так как получение неправильного пакета или кадра влечет за собой их повторную отправку.

Для передатчика нет никакой проблемы в формировании кадров, он может просто посылать кадры в физический уровень один за другим. Но на приемной стороне биты кадра должны быть проверены, а для этого приемник должен понимать, когда заканчивается один кадр и начинается другой. Эту проблему мы называем формированием кадра. Может показаться, что кадры можно разделить, вставляя в последовательность бит некоторые пустые интервалы. Однако физический канал не гарантирует какие-либо временные интервалы кроме тех, что заданы кодированием, а пустые интервалы снижают производительность канала.

4.1.1. Подсчет символов

Длина кадра всегда известна и ее можно записать в одно из полей заголовка, при этом записывается количество байт, а не бит. Приемная сторона, причитав это поле, узнаёт количество данных, и начинает их подсчет. Проблема заключается в том, что биты могут теряться и искажаться. Искажение может привести к чтению неправильного значения в поле заголовка, а потеря бит — к неправильному определению правой границы, и тогда начало следующего кадра окажется концом текущего. Кроме того, при потере синхронизации невозможно определить начало следующего кадра. Поэтому подсчет символов, если используется, то только в сочетании с другими методами.

4.1.2. Сигнальные байты и символьное заполнение

В начало и конец каждого кадра можно вставлять специальный байт, называемый флагом. При потере синхронизации приемник начинает искать этот специальный байт, чтобы найти границу кадра, поскольку два соседних флага разделяют два соседних кадра (рисунок 35).

Рисунок 35 — Границы двух соседних кадров

Если кадр содержит байт, равный флагу, в последовательность байт вставляется специальный escape-символ ESC, и байт флага заменяется

двумя байтами, ESC и FLAG. Кадр может также содержать символ ESC. В этом случае вместо символа ESC вставляется два символа ESC. Если в кадре встречаются байты ESC FLAG, вставляется ESC, ESC, ESC, FLAG. Этот прием называется символьным заполнением.

4.1.3. Сигнальные биты и битовое заполнение

В этом случае в начало и конец кадра вставляется предопределенная последовательность бит, обычно байт 01111110, который является тем же флагом. Чтобы эту последовательность не спутать с последовательностью бит данных, используется битовое заполнение.

Если в потоке передаваемых бит встретится пять подряд идущих единиц, передатчик автоматически вставляет нулевой бит перед шестой единицей. Приемник, обнаружив нулевой бит после пяти единиц, автоматически удаляет его. При потере синхронизации приемник ищет в потоке предопределенную последовательность, поскольку она никогда не встречается в потоке данных. Заметим, что этот метод не привязан к размерности передаваемых данных, к байтам, например.

4.1.4. Запрещенные сигналы

Запрещенные сигналы часто используются для определения границ передаваемых данных не только в сетях. Например, в жестких дисках запрещенный сигнал используется для определения начала сектора, а в протоколе интерфейса I2C — для определения границ сеанса.

Запрещенный сигнал часто появляется в середине битовой последовательности. Так, в манчестерском коде в середине такта присутствует переход от низкого уровня к высокому, или наоборот. Тогда отсутствие такого перехода обозначается как специальный бит J или K (в зависимости от уровня сигнала), а комбинация битов J и K с другими битами может служить признаком границы кадра. Признаком границы может также служить избыточная комбинация в таком коде, как 4B/5B.

4.2. Контроль ошибок

Контроль ошибок применяется на физическом, канальном сетевом и транспортном уровнях, так как безошибочная передача данных является задачей, которую нельзя решить в каком-то одном месте. Для контроля ошибок используются избыточные биты. Однако контрольные биты не отличаются от бит данных и они также могут искажаться и пропадать. В одном случае искажаются одиночные биты вследствие помех (шумов), в других искажаются целые блоки из-за временного нарушения в канале.

Есть две стратегии обработки ошибок. Если ошибки являются редкостью, как, например, в оптоволоконных линиях, применяется код с обнаружением ошибок (error-detecting code), потому что в этом случае дешевле заново передать испорченные данные. Если линия ненадежная, как, например, беспроводной канал, применяется код с исправлением ошибок (error-correcting code), так как вероятность получения повторной ошибки высока. Заметим, что для коррекции ошибок требуется больше избыточной информации. Методы обнаружения и коррекции ошибок исследует целый раздел прикладной математики, это сложная математическая задача, поэтому в здесь приводятся только базовые понятия.

4.2.1. Биты четности

Общий принцип обнаружения ошибки заключается в вычислении каким-либо образом контрольной суммы, в которую включаются все передаваемые биты, и эта сумма приписывается к концу передаваемого сообщения. Достоинство этого подхода заключается в простоте реализации. Каждый посылаемый бит записывается в сумму непосредственно во время его передачи или приема. По окончании подсчета на приемной стороне принятая и подсчитанная сумма сравниваются. Часто вместо этого используется сумма, которая включается в подсчет и дает нулевой результат при отсутствии ошибок. Обычный размер контрольной суммы — 16 бит, а используемая операция — это сложение по модулю 16.

Простейшая контрольная сумма называется битом четности. При этом к группе бит приписывается дополнительный бит так, чтобы число единиц в коде было четным или нечетным. Например, слово 10010011 преобразуется в 100100110 для случая четности единиц. Очевидно, что возможность обнаружения ошибки невелика, но нечетное количество ошибок будет обнаружено. Биты четности можно также приписывать к прямоугольным блокам, и контролировать четность в столбцах.

4.2.2. Коды Хэмминга

Пусть блок данных, например, кадр, содержит m информационных бит и r контрольных, а $n = m + r$ — полная длина, называемая кодовым словом. Это обозначается как код (n, m). Тогда кодовая норма (code rate) — это избыточная часть кодового слова, равная m / n. Для зашумленных каналов норма может быть равна 1/2, а для надежных близка к единице. Количество бит, отличающих два кодовых слова, называют кодовым расстоянием d. Если, например, слова равны 1001 и 1100, $d = 2$. Складывая эти слова по модулю два, получим слово, число единиц в котором равно d.

Если известны все допустимые кодовые слова, то можно найти минимальное кодовое расстояние d_{min}. Из возможных 2^n слов только 2^m допустимые, так как только они несут информацию, а другие комбинации избыточны, и позволяют обнаруживать и исправлять ошибки. Для обнаружения d ошибок требуется код с $d_{min} = d+1$, а для исправления d ошибок требуется код с $d_{min} = 2d+1$.

Пусть допустимые коды равны 000000, 000111, 111000 и 111111, где $m = 2$, $n = 6$, $r = 4$, $d_{min} = 3$. Код обнаруживает двойные ошибки и исправляет одиночные. Пусть принята комбинация 000011. Ожидая не более одной ошибки, очевидно, что передавалось слово 000111. Если принята комбинация 000100, то могло быть передано либо 000000, либо 000111.

Для исправления одиночной ошибки можно создать код, в котором каждому из 2^m допустимых сообщений соответствует n недопустимых кодовых слов с $d = 1$, и каждой из допустимых комбинаций всего соответствует $n+1$ комбинаций. Тогда $(n+1)2^m \le 2^n$, или $(m+r+1) \le 2^r$. Это условие описывает нижний предел требуемого числа контрольных бит.

Тогда единичный бит исправляется при помощи двух контрольных. Представляя ноль комбинацией 000, а единицу комбинацией 111, принимаемые кодовые комбинации 100, 010 и 001 представляют нулевой бит, а комбинации 110, 101 и 011 — единичный, при этом не требуются сравнения принятого кода с допускаемыми комбинациями.

В кодах Хэмминга биты, соответствующие степеням двойки, являются контрольными. При длине кодового слова 7 контрольными будут биты 1, 2 и 4 (при нумерации от единицы). Контрольный бит вычисляет сумму группы бит, включая самого себя. Бит данных в позиции k входит в суммы тех контрольных бит, что составляют бит k. Например, бит 7 входит в суммы контрольных бит 1, 2 и 4. Пусть четыре бита данных равны 1100, тогда передаваемое кодовое слово $S_1S_2 1 S_4 100$, в котором контрольные суммы S_i равны:

$S_1 = 0_1+b_3+b_5+b_7 = 0$, $S_2 = 0_2+b_3+b_6+b_7 = 1$, $S_4 = 0_4+b_5+b_6+b_7 = 1$,

и передаваемый код равен 0111**0**0.

Пусть получен код 0111**1**0. Снова вычислим контрольные суммы:

$S_1 = b_1+b_3+b_5+b_7 = 0$, $S_2 = b_2+b_3+b_6+b_7 = 1$, $S_4 = b_4+b_5+b_6+b_7 = 1$,

тогда *синдром ошибки* $S_4S_2S_1 = 110_2 = 6_{10}$, то есть искажен шестой бит.

4.2.3. Коды в конечных полях

Коды в конечных полях (в полях Галуа) используют представление входных строк в виде многочленов, коэффициентами которых являются биты или символы. Арифметические действия с многочленами выполняются как сложение по модулю 2, при этом перенос при сложении и заем при вычитании не производится.

В качестве примера рассмотрим циклический избыточный код CRC (Cyclic Redundancy Code). Для вычисления CRC сначала выбирается образующий многочлен $G(x)$, в котором старший и младший бит равны 1. CRC вычисляется так, чтобы многочлен кадра с учетом CRC делился на $G(x)$ без остатка, и тогда ненулевой результат обозначает ошибку.

Примерный алгоритм вычислений. Пусть $M(x)$ — многочлен кадра.

1. Добавить в конец кадра r нулевых бит, r — степень $G(x)$.
2. Полученный многочлен $x^r M(x)$ разделить на $G(x)$.
3. Остаток от деления вычесть из $x^r M(x)$: $T(x) = x^r M(x) / G(x)$.

Передаваемый кадр $T(x)$.

Пусть биты данных кадра 11010111, а $G(x) = x^4+x+1$, $r = 4$.

Сначала приписываем 4 нуля и получаем $x^r M(x) = 11010111\mathbf{0000}$.

Делим $x^r M(x) = 11010111\mathbf{0000}$ на $G(x) = 10011$:

110101110000 / 10011 = 010011110000
010011110000 / 10011 = 000000110000
000000110000 / 10011 = 000000010110
000000010110 / 10011 = 000000000101

Получили остаток 101 длиной не более r бит, передаваемый кадр:

$T(x) = 11010111\mathbf{0000} – 101 = 11010111\mathbf{0101}$.

На приемной стороне делим $T(x) = 11010111\mathbf{0101}$ на $G(x) = 10011$:

110101110101 / 10011 = 010011110101
010011110101 / 10011 = 000000110101
000000110101 / 10011 = 000000010011
000000010011 / 10011 = 000000000000

Остаток равен нулю, передача считается успешной.

Если некоторые из бит испорчены, то вместо $T(x)$ на приемной стороне будет получен многочлен $T(x) + E(x)$, и единичные биты $E(x)$ соответствуют инвертированным битам $T(x)$. Если в $E(x)$ k единичных бит, значит произошло k единичных ошибок. Приемник делит $T(x) + E(x)$ на $G(x)$, и так как $T(x) / G(x) = 0$, результат равен $E(x) / G(x)$. Следовательно, не будут обнаружены ошибки, кратные $G(x)$. Выбор подходящего $G(x)$, таким образом, влияет на обнаружение ошибок. Важно также, что полиномиальный код с r контрольными битами обнаруживает пакеты ошибок длиной не более r.

Аналогичным образом получаются коды Рида-Соломона, только коэффициентами многочленов являются октеты. В сетях используется код (255, 223), исправляющий до 16 и обнаруживающий до 32 ошибочных символов. Максимальная длина кода 2^{s-1}, s — размер символа.

Эти коды очень эффективны, используются не только в сетях, но и в устройствах хранения. Вычисление ошибочных символов здесь не описывается из-за сложности алгоритма.

4.3. Простые протоколы передачи данных

Чтобы понять, как происходит передача кадров, разберем несколько возможных алгоритмов взаимодействия передатчика и приемника.

4.3.1. Утопический симплексный протокол

Сначала будем исходить из предположения, что физический уровень работает безошибочно, а сетевой уровень готов предоставить очередной пакет и получить его в любой момент. Если в физическом канале не возникает ошибок, то и их контроль не требуется (рисунок 36).

Передатчик	Приемник
1. пакет ← L3	1. ждать события (кадр)
2. кадр.пакет = пакет	2. кадр ← L1
3. кадр → L1	3. кадр.пакет → L3
4. перейти к 1	4. перейти к 1

Рисунок 36 — Утопический симплексный протокол

Передача ведется в одну сторону, то есть канал симплексный. Передатчик и приемник находятся в бесконечном цикле, ожидая в одном случае пакет, поставляемый сетевым уровнем, а в другом — кадр, приходящий по физическому каналу. Действия передатчика и приемника описываются в терминах процедур 1-3.

4.3.2. Симплексный протокол с ожиданием

Теперь предположим, что сетевой уровень не всегда готов принять приходящий пакет. Тогда скорость передатчика можно ограничить при помощи пустого кадра, разрешающего передачу (рисунок 37).

Передатчик	Приемник
1. пакет ← L3	1. ждать события (кадр)
2. кадр.пакет = пакет	2. кадр ← L1
3. кадр → L1	3. кадр.пакет → L3
4. ждать события (кадр)	4. пустой-кадр → L1
5. перейти к 1	5. перейти к 1

Рисунок 37 — Симплексный протокол с ожиданием

Приемник посылает пустой кадр только после того, как сетевой уровень примет пакет. Приемник, отослав кадр, ожидает получение пустого кадра от приемника, после чего ожидает пакет сетевого уровня. Таким образом, скорость потока данных синхронизируется с сетевым уровнем.

4.3.3. Симплексный протокол с подтверждением

Реальный физический канал может передавать кадры с ошибками, поэтому требуется контроль ошибок при помощи контрольной суммы. При этом возможна маловероятная ситуация, когда сумма совпадет для ошибочного кадра, и тогда пакет, доставленный сетевому уровню будет неверным. С этим ничего не сделать, верхние уровни должны обнаружить ошибку при помощи своих протоколов.

Когда кадр прибывает на сторону приемника, контрольная сумма может совпадать иди не совпадать. Если сумма совпадает, считаем кадр принятым, и посылаем передатчику положительное подтверждение. Такой протокол называется PAR (positive acknowledgement with retransmission) или ARQ (automatic repeat request). Если передатчик не получает подтверждения, он посылает кадр повторно, выждав некоторое время, для чего требуется таймер (рисунок 38).

Передатчик	Приемник
m = 0 *m* = 0	
пакет ← L3	кадр-п
1. кадр.пакет = пакет	1. ждать (кадр, ошибка)
2. кадр.seq = m	2. если (кадр)
3. кадр → L1	1. кадр ← L1
4. запустить таймер	2. Если (кадр.seq = m)
5. ждать (кадр, ошибка, таймер)	1. кадр.пакет → L3
6. если (кадр)	2. инвертировать m
1. кадр ← L1	3. кадр-п.ack = кадр.ack
2. если (кадр.ack = m)	4. кадр-п → L1
1. пакет ← L3	3. перейти к 1
2. инвертировать m	
7. перейти к 1	

Рисунок 38 — Симплексный протокол с подтверждением

Требуется еще нумерация кадров, чтобы было понятно, какой кадр или подтверждение какого кадра получено. Для этого вводится номер кадра *m*, который может быть 0 или 1, так неопределенность возможна только в двух соседних кадрах, поле seq для номера кадра и поле ack для номера подтверждаемого кадра. Передатчик сначала передает кадр *m*, запускает таймер и ждет события. Если приемник получил кадр *m*, он посылает кадр подтверждения *m*, которое может не потеряться. Если у передатчика сработал таймер или подтверждение пришло с ошибкой, он посылает кадр *m* повторно. Кадр 1 — *m* посылается только если есть подтверждение кадра *m*.

4.4. Протоколы скользящего окна

Дуплексный протокол можно реализовать при помощи двух линий передачи, но, вообще говоря, пропускная способность канала одинакова в обоих направлениях. Поэтому представляется выгодным пересылать кадры одновременно в двух направлениях. При этом можно посылать подтверждения с попутными кадрами (если они есть). В каждый момент времени протокол скользящего окна работает с набором порядковых номеров, соответствующих кадрам, которые разрешено посылать или принимать, не ожидая подтверждения.

4.4.1. Однобитовое скользящее окно

На рисунке 39 приведен протокол однобитового скользящего окна.

```
seq = 0,  ack = 0
кадр.пакет ← L3
кадр.seq = seq
кадр.ack = 1 – ack
кадр → L1
запустить таймер
1. ждать (кадр, ошибка, таймер)
2. если (кадр) -- кадр получен вовремя и без ошибок (иначе 3)
    1. кадр ← L1
    2. если (кадр.seq = ack)
        1. кадр.пакет → L3
        2. ack = 1 – ack -- инверсия номера ожидаемого кадра
    3. если (кадр.ack = seq)
        1. кадр.пакет ← L3
        2. seq = 1 – seq -- инверсия номера отправляемого кадра
3. кадр.seq = seq
4. кадр.ack = 1 – ack
5. кадр → L1
6. запустить таймер
7. перейти к 1
```

Рисунок 39 — Протокол однобитового скользящего окна

Протокол предполагает, что сетевой уровень систематически поставляет пакеты для передачи, так как эти пакеты несут подтверждения другой стороне. Фактически протокол в одну сторону работает точно так же, как симплексный протокол с подтверждениями. Поскольку для номера кадра отведен один бит, протокол называется однобитовым.

4.4.2. Протокол с возвратом на *n*

Сеть будет работать быстрее, если окна пересылки и получения расширить. В этом случае несколько кадров могут быть посланы до того, как придет подтверждение предыдущих кадров. Новому пакету сетевого уровня дается наибольший номер и верхняя граница окна увеличивается на 1. При получении подтверждения на 1 увеличивается нижняя граница окна. Так как кадры могут теряться и повреждаться, отправитель должен иметь буферы для их временного хранения. На рисунке 40 показана конвейеризация кадров, если окно приемника равно 1.

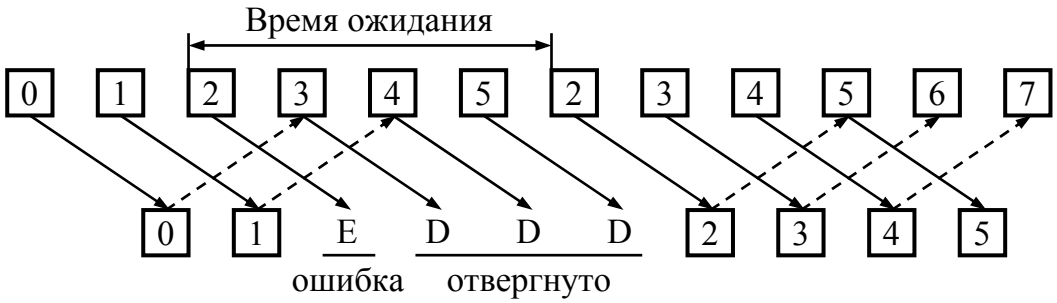


Рисунок 40 — Протокол с возвратом на *n*

Приемник, получив поврежденный кадр 2, игнорирует последующие кадры, пока на получит кадр 2, чтобы обеспечить правильную последовательность кадров для сетевого уровня. Передатчик посылает кадр 2 повторно тогда, когда истекает время ожидания.

Увеличить эффективность сети можно, расширив окно приемника. В этом случае приемник записывает кадры, отвергнутые в предыдущем примере, в свои буферы, чтобы выстроить правильную последовательность кадров перед их передачей сетевому уровню (рисунок 41).

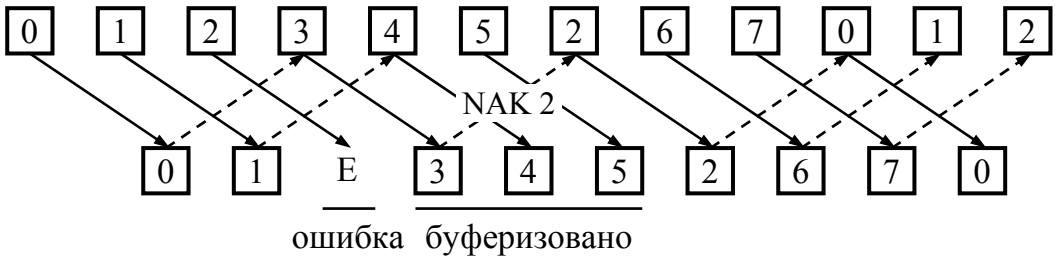


Рисунок 41 — Отрицательное подтверждение

Чтобы ускорить доставку поврежденного кадра, протокол посылает отрицательное подтверждение NAK (negative acknowledgement). Этот протокол использует также *кумулятивное* подтверждение: если получено подтверждение с номером *n*, то предыдущие кадры также считаются подтвержденными, если их подтверждения не были получены ранее.

4.4.3. Протокол с выборочным повтором

Стратегия, при которой кадры буферизуются приемником для сборки правильной последовательности, называется выборочным повтором. Окно отправителя растет от нуля до некоторого предела, окно получателя фиксировано, и размер больше единицы. Получатель имеет буфер для каждого кадра, номер которого находится в пределах окна, и может получать кадры в произвольном порядке до их передачи сетевому уровню. Если для номера кадра отводится n бит, отправитель может отсылать до 2^n–1 кадров, а размер окна получателя не может превышать 2^{n-1}. В противном случае возможно совпадение номеров старых и новых кадров.

Поскольку встречного потока кадров может и не быть, получатель передает пустой кадр подтверждения по истечении некоторого времени ожидания, для чего используется вспомогательный таймер.

4.5. Протоколы двухточечных соединений

4.5.1. Протокол HDLC

Высокоуровневый протокол управления каналом HDLC (high-level data link protocol) первоначально был разработан для терминальных систем, впоследствии был использован для двухточечных соединений. Это бит-ориентированный протокол с битовым заполнением (рисунок 42).

8	8	8	от 0	16	8
01111110	address	control	payload (data)	checksum	01111110

Рисунок 42 — Формат кадра бит-ориентированного протокола

В поле address ранее записывался номер терминала. В двухточечных сетях оно иногда используется для различения команд и ответов. Поле control предназначено для служебных данных (рисунок 43).

0	Seq		P/F	Next	*а*
1	0	Type	P/F	Next	*б*
1	1	Type	P/F	Modifier	*в*

Рисунок 43 — Управляющее поле протокола HDLC

Кадры делятся на информационные, супервизорные и ненумерованные (рисунок 43, а, б и в соответственно). В поле Seq содержит номер кадра, поле Next — номер первого не принятого кадра (или попутное подтверждение). Поле P/F (Poll/Final) используется терминалами.

Поле Type обозначает тип супервизорного кадра: 0 (receive ready) — кадр подтверждения, 1 (reject) — кадр отрицательного подтверждения, 2 (receive not ready) — приостановку передачи, при этом подтверждается прием кадров до Next–1, 3 — выборочный отказ (selective reject).

Ненумерованные кадры применяются для служебных команд, при этом поле Modifier расширяет поле Type до 32 значений.

HDLC использует скользящее окно с порядковым номером кадра размером 3 бита, поэтому в сети может находиться не более семи неподтвержденных кадров.

4.5.2. Протокол PPP

Протокол двухточечного соединения PPP (point to point protocol) является улучшенным вариантом более простого протокола для последовательной линии SLIP (serial line internet protocol). Протокол выполняет обнаружение ошибок, а также поддерживает протоколы, разрешающие аутентификацию. Это байт-ориентированный протокол (рисунок 44).

8	8	8	8 или 16	от 0	16 или 32	0 или 8
01111110	11111111	00000011	protocol	payload	checksum	01111110

Рисунок 44 — Кадр PPP для ненумерованного режима

Разделителем кадров является флаг 0x7E. В поле данных он заменяется байтами 0x7D и 0x5E, и тогда флаг встречается только на границе кадра. Кроме того, между кадрами вставляется один байт флага. Поле полезной нагрузки по умолчанию не превышает 1500 байт.

Адрес 0xFF обозначает, что все станции принимают кадр. Протокол обычно используется в ненумерованном режиме.

Поле protocol определяет тип передаваемого пакета. Номера, начинающиеся с бита 0, отведены для протокола IP и других протоколов сетевого уровня. С бита 1 начинаются коды, обозначающие конфигурационный протокол. Протокол PPP обеспечивает 3 набора методов:

1. Метод формирования кадров и обнаружения ошибок.

2. Протокол управления каналом LCP (link control protocol). Позволяет установить канал связи, договориться о параметрах.

3. Сетевой протокол управления NCP (network control protocol). Позволяет договориться о параметрах сетевого уровня.

При установлении соединения протокол посылает сначала серию пакетов LCP, а затем серию пакетов NCP, и происходит настройка. При этом поле protocol часто сокращается до 1 байта, а поля адреса и управления могут вообще отсутствовать, так как они являются константами.

5. Подуровень управления доступом к среде

Основная проблема широковещательной сети заключается в разрешении спора о том, кому предоставить канал, если желающих передать кадр несколько. Протоколы, определяющие порядок передачи кадров в такой сети, относятся к подуровню канального уровня, обозначаемому MAC (medium access control, управление доступом к сети), а уровень передачи описывает стандарт LLC (logical link control), который находится в иерархии выше уровня MAC, и все процессы уровня MAC происходят до передачи данных на уровне LLC.

Широковещательные каналы называют каналами с множественным (multi-access) доступом или каналами с произвольным (random) доступом. Как уже упоминалось, в ЛВС в подавляющем большинстве случаев используется широковещательная сеть Ethernet. Беспроводные линии связи также представляют собой общий канал.

5.1. Метод доступа CSMA/CD

Множественный доступ с контролем несущей CSMA (carrier-sense multiple access) обозначает прослушивание канала перед тем, как начать передачу кадра. При этом обычным является столкновение кадров, называемое коллизией, когда два станции одновременно, или почти одновременно, начинают передачу. При обнаружении коллизии обе станции завершают передачу, и выдерживают некоторый случайный интервал. Если после этого станция сразу начинает повторную передачу, метод называется настойчивым. Если перед повторной передачей сначала снова прослушивается канал, метод называется ненастойчивым.

В сети Ethernet используется метод CSMA/CD (CSMA with collision detection, CSMA с обнаружением коллизий). Он отличается тем, что при обнаружении коллизии станция немедленно прекращает передачу, освобождая канал, что несколько улучшает производительность сети. Этот метод является настойчивым с вероятностью 1. Обнаружение коллизии осуществляет передатчик (сетевой адаптер), сравнивая то, что передает, с тем, что «слышит» в линии.

При обнаружении коллизии станция выдерживает интервал времени в соответствии с экспоненциальным двоичным алгоритмом выдержки (binary exponential backoff). Задержка исчисляется в интервалах, равных времени двойного оборота сигнала. Для максимального размера сети в 2,5 км это время составляет 51,2 мкс, или 512 битовых интервалов BT (bit time). Число интервалов выбирается случайно из диапазона $[0, 2^N]$, где N — номер попытки. После 10 попыток всегда выдерживается 1023 интервала. После 16 попыток передатчик сигнализирует ошибку.

5.2. Предоставление доступа в беспроводных сетях

В беспроводных сетях метод CSMA не подходит, так как проблема заключается в интерференции на стороне приемника, а не передатчика.

Рассмотрим работу беспроводных станций, каждая из которых находится в зоне приема только соседней станции (рисунок 45).

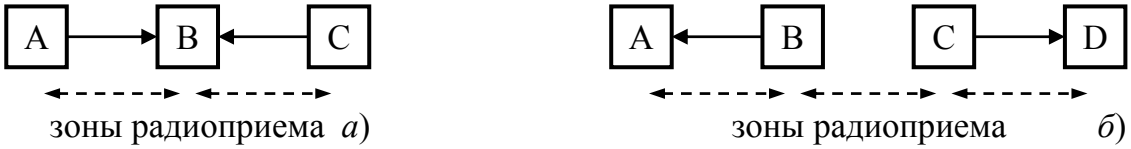


Рисунок 45 — Беспроводная локальная сеть

В случае *а*) станции A и C одновременно начинают передачу станции B, канал к которой с обеих сторон свободен. Однако в приемнике B происходит наложение сигналов и возникает коллизия. Эта ситуация называется проблемой скрытой станции (hidden terminal problem).

В случае *б*) станция B передает кадр станции A, а станция C желает передать кадр станции D. Станция C «слышит» передачу в зоне приема, и ошибочно «думает», что канал занят.

Для решения таких проблем беспроводных сетей одним из первых был разработан протокол MACA (multiple access with collision avoidance, множественный доступ с предотвращением коллизий).

Станция A начинает с того, что посылает станции B кадр RTS (request to send, запрос на передачу). Этот кадр размером 30 байт содержит в том числе длину кадра, который последует за ним. Станция B отвечает кадром CTS (clear to send, разрешение передачи), который копирует длину кадра из RTS. Любая станция, слыша передачу кадра RTS или CTS, должна сохранять молчание до окончания передачи информационного кадра, длина которого извлекается из служебного кадра. Этот протокол был улучшен, и стал называться MACAW (MACA for wireless).

5.3. Сети Ethernet

Классический Ethernet — это общая шина на коаксиальном кабеле, со скоростью передачи 3-10 Мбит/с, в котором коллизии разрешаются методом CSMA/CD. Эту технологию вытеснил коммутируемый Ethernet (switched Ethernet) на топологии «иерархическая звезда» с коммутаторами, со скоростями 100 Мбит/с (Fast Ethernet), 1 Гбит/с (Gigabit Ethernet) и выше, использующий кабели на витой паре и оптоволокно. Коллизии в коммутируемом Ethernet встречаются редко, а безопасность выше. Тем не менее, изучение принципов работы классического Ethernet позволяет лучше понять, как работают ЛВС.

5.3.1. Кадры Ethernet

В классическом Ethernet все параметры подобраны так, чтобы обеспечить обнаружение коллизии за определенное время. На рисунке 46 показаны форматы кадров DIX и IEEE 802.3 (всего есть 4 формата).

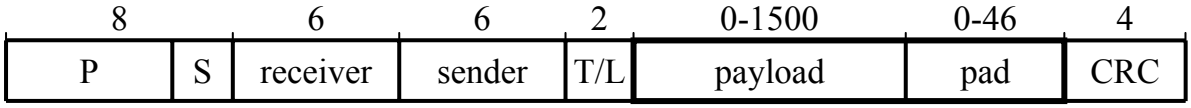


Рисунок 46 — Формат кадра Ethernet

Между кадрами выдерживается технологический интервал в 9,6 мкс. Он необходим, чтобы никакая станция не монополизировала доступ. Перед кадром передается преамбула P из семи байт 10101010 и восьмого байта 10101011. Преамбула синхронизирует передатчик и приемник, использующих манчестерский код. Если имеется ввиду стандарт IEEE 802.3, последний байт S называется start of frame.

Затем следуют два MAC-адреса, сначала адрес машины получателя, затем адрес машины отправителя, длина каждого адреса 6 байт. Последнее поле заголовка T/L длиной 2 байта — это либо тип протокола, как предусматривает стандарт DIX, либо длина полезной нагрузки payload, как предусматривает стандарт IEEE 802.3. Этот стандарт предусматривает добавление заголовка для протокола LLC, который записывается в данные (payload) перед заголовком пакета. Длина заголовка 8 байт, из них 2 байта указывают протокол (заголовок которого следует в данных непосредственно за заголовком LLC). Фактически поле используется одновременно и как тип, и как длина. Если значение в поле не превышает 0x600 (1536), то это длина данных, иначе код протокола. Например, для протокола IPv4 используется значение 0x0800. Таким образом, длина заголовка кадра равна 14 байт (без преамбулы, она не часть кадра).

Полезная нагрузка кадра (то есть передаваемый пакет) укладывается в поле payload, размер которого ограничен сверху 1500 байтами. Такое решение было принято в те далекие годы, когда память была дефицитна.

В поле данных входит также поле заполнителя pad. Оно заполняется, если длина данных меньше 46. При этом к данным приписываются произвольные байты так, чтобы получить длину 46. За полезной нагрузкой следует 4 байта контрольной суммы. Складывая длину заголовка 14 с минимальной длиной данных 46 и длиной контрольной суммы 4 байта, получаем минимальную длину кадра Ethernet — 64 байта.

Самый интересный вопрос, — почему 64 байта? Для этого есть причины, и самая важная из них — время двойного оборота.

5.3.2. Время двойного оборота

Кадры Ethernet передаются без подтверждения, и тогда отправителю нужно удостовериться в том, что либо кадр доставлен, либо произошло столкновение. Станция, обнаружившая коллизию, немедленно прерывает передачу кадра и посылает специальную последовательность из 32-х байт (jam-последовательность), которая создает в сети шумовой всплеск и, таким образом, усиливает коллизию. Если передатчик не «слышит» шумового всплеска, он считает, что кадр успешно доставлен.

Четкое распознавание коллизий является необходимым условием корректной работы сети Ethernet. Если кадр утерян, или его контрольная сумма оказалась неверна, то это будет обнаружено на одном из верхних уровней, что потребует значительно большее время для восстановления. Для надежного распознавания коллизий должно выполняться условие $T_{min} \geq 2\tau$, где T_{min} — время, требуемое для передачи кадра минимальной длины, а 2τ — время двойного оборота (рисунок 47).

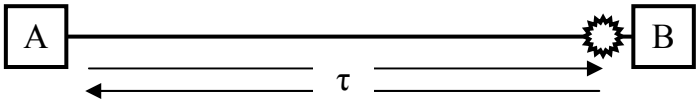


Рисунок 47 — Обнаружение коллизии

В худшем варианте столкновение происходит в тот момент, когда кадр практически достиг получателя. Станция А отправляет кадр, убедившись, что линия свободна. Кадр почти достигает станции В, и в этот момент станция В решает начать передачу, поскольку сигнал кадра еще не дошел до нее, и она справедливо считает, что линия свободна. Происходит столкновение и шумовой всплеск. Всплеск распространяется обратно, станция А принимает его и также фиксирует столкновение. Распространение сигнала от А к В занимает время τ, такое же время необходимо для распространения шумового всплеска обратно, поэтому время обнаружения коллизии называется временем двойного оборота, и обозначается PDV (path delay value, значение задержки пути).

Для обеспечения необходимой мощности сигнала на конце коаксиального кабеля его длина была выбрана равной 500 м. Время двойного оборота при этом равно примерно 4 мкс, а для пяти сегментов 20 мкс. Для передачи 72 байт кадра минимальной длины (с учетом преамбулы, так как она часть передачи) потребуется 57,6 мкс (72×8 = 576). На длине в 2500 м должно быть четыре повторителя, каждый из которых вносит задержку в распространение сигнала. Необходим также какой-то запас для того, чтобы учесть отклонения параметров кабелей и повторителей. В результате и была выбрана минимальная длина кадра в 64 байта.

5.3.3. Коммутируемый Ethernet

Сеть Ethernet на коаксиальном кабеле имеет серьезный недостаток в том, что повреждение кабеля нарушает работу почти всей сети. Поэтому очень скоро появились сети с концентраторами. Хост присоединяется к концентратору своим собственным кабелем, неисправность которого выводит из сети только один компьютер. При этом вместо коаксиального кабеля применяются кабели на основе витой пары, например, кабели существующей телефонной сети. Однако концентраторы не увеличивают емкость сети, они эквивалентны коаксиальному кабелю в смысле домена коллизий. Домен коллизий (collision domain) — это часть сети, в которой возникает конкуренция за доступ к среде передачи. В кабеле домен коллизий распределен по длине кабеля, а в концентраторе сосредоточен в одной точке (на самом деле распределен по всем присоединенным кабелям). Улучшить производительность сети Ethernet можно, если увеличить скорость передачи или сократить домен коллизий. Первый вариант требует переоборудования, во втором варианте достаточно заменить концентраторы коммутаторами.

У коммутатора каждый порт находится в своем домене коллизий. В наиболее частом случае передача по кабелю происходит в дуплексном режиме, когда одна витая пара передает кадры в одном направлении, а вторая пара — в противоположном, поэтому столкновений нет. В полудуплексном режиме домен коллизий ограничен длиной кабеля от компьютера до коммутатора, и на этом участке используется CSMA/CD, но это всего лишь две станции (компьютер и порт коммутатора).

Коммутатор позволяет хостам вести одновременную передачу, так как он передает кадры только в тот порт (соответственно, только в тот кабель), для которого кадр предназначен. Для этого каждый порт запоминает в специальной таблице, какой MAC-адрес имеет хост, соединенный с портом. При старте коммутатора таблица пуста. Как только в порт поступает кадр, в таблицу записывается MAC-адрес отправителя. Постепенно каждый порт узнает, с каким хостом он соединен. Тогда, если в некоторый порт приходит кадр с некоторым адресом получателя, из таблицы выбирается порт, соединенный с получателем и кадр направляется в этот порт и далее в хост. Поскольку в один хост одновременно может быть направлено несколько кадров, коммутатор имеет буферы для временного хранения кадров.

Коммутаторы обладают также лучшей безопасностью. Когда домен коллизий имеет протяженность, злоумышленник может подключиться к нему, прослушивать трафик, или подменять кадры. Коммутаторы могут запоминать, компьютер с каким MAC-адресом должен быть подключен к конкретному порту.

5.3.4. Высокоскоростной Ethernet

В современных локальных сетях используется Ethernet, работающий на скоростях 100 Мбит/с и 1000 Мбит/с, обозначаемый соответственно 100Base-X и 1000Base-X, где X — обозначение среды передачи.

На скорости 100 Мбит/с время передачи одного бита сокращается до 10 нс, и время двойного оборота уменьшается в 10 раз, что снижает максимальное расстояние между хостами. Проблемой является также передающая среда. Во многих офисах имелись кабельные сети на витой паре категории 3, максимальная скорость передачи которой не столь высока. С другой стороны, многие офисы могли себе позволить проложить кабели на витой паре категории 5, а оптоволоконные линии связи стали более доступны. В результате стандарт IEEE 802.3u (1995 г.) определил три варианта сред:

Технология	Среда	Длина сегмента, м
100Base-T4	4 витых пары категории 3	100
100Base-TX	2 витых пары категории 5	100
100Base-FX	2 MMF волокна	2000

В 100Base-T4 используются 4 витых пары, каждая из которых работает на частоте 25 МГц. Одна пара направлена от хоста, друга к хосту, две пары меняют направление в зависимости от потока данных. Вместо манчестерского кодирования используется сложный код 8B/6T.

В 100Base-TX используются две витые пары, работающие на частоте 125 МГц, с кодированием 4B/5B поверх MLT3, с полным дуплексом. В 100Base-FX используются два оптоволокна с кодированием 4B/5B поверх NRZI, также с полным дуплексом.

Метод доступа CSMA/CD используется только тогда, когда сеть построена на концентраторах. Оптоволоконные сети образуют каналы типа точка-точка, в которых CSMA/CD не требуется.

Широкое применение гигабитного Ethernet связывают с появлением стандарта IEEE 802.ab (1999 г.). Он вводит технологию 1000Base-T, предполагающую использование четырех витых пар категории 5, работающих на частоте 125 МГц, со скоростью 250 Мбит/с одновременно в обоих направлениях. Для кодирования используется пятиуровневая фазоамплитудная модуляция (PAM5), переносящая два бита за бод.

Большой проблемой становится маленький кадр. Приходится сокращать максимальную длину кабеля в 100 раз, если в сети есть концентраторы. Было предложено аппаратно вставлять в кадр поле, расширяющее кадр до 512 байт, снижая при этом эффективность сети. Другое улучшение заключается в пакетной передаче кадров, когда несколько кадров заполняют поле данных одного кадра.

6. Сетевой уровень

Сетевой уровень самый нижний из уровней модели OSI, занимающийся доставкой пакетов от отправителя до получателя. На этом уровне пакеты совершают прыжки между маршрутизаторами от маршрутизатора подсети отправителя до маршрутизатора подсети получателя. Так как между отправителем и получателем обычно существует несколько возможных путей доставки, основной задачей сетевого уровня является *маршрутизация*, то есть выбор маршрута передвижения пакета.

Сообщение транспортного уровня разбивается на пакеты. Очередной пакет отправителя поступает на ближайший к нему маршрутизатор. Это может быть маршрутизатор локальной сети или маршрутизатор провайдера, с которым отправитель в этом случае связан соединением типа точка-точка. Пакет принимается целиком, при необходимости обрабатывается (модифицируется) и проверяется. Такой механизм называется коммутацией пакетов с ожиданием (store and forward).

Для дальнейшего перемещения пакета маршрутизатор должен обладать информацией о топологии подсети или подсетей, с которыми он имеет прямое соединение. Для этой цели маршрутизаторы поддерживают локальную базу данных, называемую *таблицей маршрутизации*.

Канальный уровень доставляет пакеты либо до хоста, либо до ближайшего маршрутизатора, если пакет требуется доставить в другую подсеть. Сетевой уровень доставляет пакет в любую точку любой подсети, доступной на физическом уровне. Для этой цели на сетевом уровне должен быть единый способ адресации узлов сетей, как хостов, так и маршрутизаторов. Кроме того, сервисы сетевого уровня не должны зависеть от технологии доставки пакетов на канальном уровне и технологии маршрутизатора.

6.1. Сервисы сетевого уровня

Будем рассматривать прохождение пакетов по сети, изображенной на рисунке 48. Маршрутизатор подсети отправителя обозначен как A, а маршрутизатор подсети получателя обозначен как F. Предварительно сообщение транспортного уровня разбито на четыре пакета.

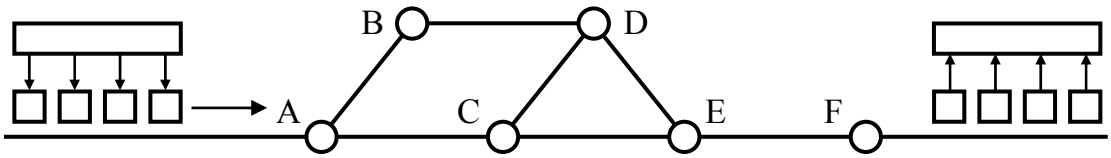


Рисунок 48 — Прохождение сообщения по сети

Сервис без установления соединения посылает пакеты в сеть независимо друг от друга, без предварительной настройки. Пакеты в этом случае называют дейтаграммами.

Маршрутизаторы A-F имеют записи о маршрутизаторах, с которыми они имеют соединение. Каждая запись содержит поле адресата и поле линии, по которой адресат доступен. Для примера на рисунке 48 эти записи могут иметь следующий абстрактный вид:

A		C		E	
A	–	A	A	A	C
B	B	B	A	B	D
C	C	C	–	C	C
D	B	D	D	D	D
E	C	E	E	E	–
F	C	F	E	F	F

Когда пакет с адресом назначения F прибывает на маршрутизатор A, маршрутизатор по своей таблице определяет, что пакет следует направить маршрутизатору C, соседнему с A. Следующий маршрутизатор, C, определяет, что следующим соседним маршрутизатором является E.

Другой пакет не обязательно будет направлен тем же маршрутом. Маршрутизаторы периодически обновляют свои таблицы, обмениваясь информацией о текущей топологии сети друг с другом. Это может изменить запись адресата F маршрутизатора A, и пакет отправится по другому маршруту. Примером сервиса без установления соединения является протокол IP.

Сервис с установлением соединения сначала прокладывает маршрут движения пакетов. Такое соединение называется виртуальным каналом, а сеть — сетью виртуального канала. При этом пакеты отправителя содержат метку виртуального канала, а маршрутизаторы используют эту метку для определения маршрута:

A				C				E			
H1	1	C	1	A	1	E	1	C	1	F	1
H2	1	C	2	A	2	E	2	C	2	F	2

Когда пакет от хоста H1 с меткой 1 поступает на маршрутизатор A, он направляется на маршрутизатор C с меткой 1, и так далее.

Если другой хост, H2, устанавливает соединение, он также может выбрать метку 1. Возникает проблема меток, и тогда маршрутизатор A присваивает новую метку виртуального каналу хоста H2. Этот процесс называется коммутацией меток (label switching). Примером сервиса, ориентированного на соединение, является MPLS (multi protocol label switching). При этом IP-пакеты получают MPLS-заголовок, содержащий 20-битную метку виртуального канала.

6.2. Алгоритмы маршрутизации

Маршрутизаторы выполняют две разные функции. Первая отвечает за пересылку пакетов (forwarding) и реализуется при помощи таблицы маршрутизации. С другой стороны, маршрутизаторы должны заполнять и с некоторой периодичностью обновлять таблицы маршрутизации. Эта функция реализуется алгоритмами маршрутизации.

Алгоритм маршрутизации должен быть корректным, простым, надежным, устойчивым, справедливым и эффективным. Корректность и простота алгоритма должна быть очевидна. Надежность алгоритма проявляется в его способности справляться с отказами аппаратуры и изменениями топологии сети. Устойчивость проявляется в способности сходиться к фиксированному набору путей за достаточно быстрое время и оставаться в состоянии равновесия.

Справедливость и эффективность алгоритма проявляется в нахождении компромисса при распределении трафика между разными потоками данных. При этом оптимизируются некоторые параметры передачи, такие, как среднее время задержки, пропускная способность сети, количество пересылок (прыжков).

Все алгоритмы маршрутизации могут быть поделены на два класса: неадаптивные и адаптивные. *Неадаптивные алгоритмы* не учитывают при выборе маршрута топологию сети и ее текущее состояние. Выбор маршрута между каждой парой станций производится заранее, и загружается в маршрутизаторы во время загрузки сети. Такой способ называется статической маршрутизацией (static routing).

Адаптивные алгоритмы периодически изменяют маршруты на основании получаемых маршрутизаторами сведений о топологии сети, состоянии линий, их загруженности. Эти алгоритмы называют также динамическими алгоритмами маршрутизации (dynamic routing algorithms). Они отличаются источниками получения информации, моментами изменения маршрутов, и критериями выбора оптимального маршрута.

6.2.1. Алгоритм нахождения кратчайшего пути

Перед вычислением кратчайшего пути (shortest path), нужно определиться с тем, как измерять расстояние между маршрутизаторами. Для этого вводится понятие метрики пути. В одних случаях в качестве метрики используется число транзитных участков, и тогда каждое ребро графа имеет метрику, равную единице. В других случаях учитывается максимальная пропускная способность линий, и тогда метрика обратно пропорциональна этому значению, а кратчайшим путем оказывается самый быстрый.

В качестве метрики можно принимать среднее время задержки, измеряемое специальными тестовыми пакетами, посылаемыми через определенные промежутки времени. В общем случае метрика является функцией расстояния, пропускной способности, средней загруженности сети, величины задержки и других факторов.

Здесь мы рассмотрим алгоритм SPF (shortest path first), предложенный Э. Дейкстрой в 1959 году. В качестве примера будем рассматривать взвешенный ненаправленный граф, описывающий сеть с топологией, показанной на рисунке 49.

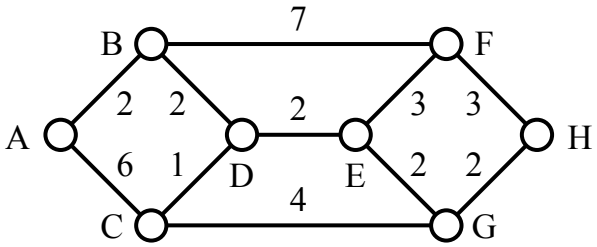


Рисунок 49 — Граф топологии примерной сети

Необходимо вычислить кратчайшее расстояние от узла A до узла H.

С каждым узлом связывается расстояние до него от начального узла и узел, через который это расстояние вычислено. Это записывается как пара (расстояние, узел). Узел, кратчайшее расстояние до которого определено, помечается как постоянный, другие узлы временные.

Сначала постоянным помечается начальный узел. От него вычисляются отметки соседних временных узлов B(2, A) и C(6, A). Далее из всех временных узлов выбирается узел с наименьшим расстоянием, этот узел становится постоянным. Последовательность вычислений следующая:

Узел отсчета	Отметка	Отметка
A	B(2, A)	C(6, A)
B	D(4, B)	F(9, B)
D	E(6, D)	C(5, D)
C	G(9, C)	
E	F(9, B)	G(8, E)
G	H(10, G)	

Заметим, что если для некоторого временного узла получена новая отметка с меньшим расстоянием, она заменяет собой старую отметку. Так произошло с отметками узлов C и G.

После того, как получена отметка с наименьшим расстоянием до целевого узла H, обратным проходом по отметкам вычисляется кратчайший путь: H(10, G) — G(8, E) — E(6, D) — D(4, B) — B(2, A). Таким образом, кратчайший путь от A к H лежит через узлы B — D — E — G.

6.2.2. Заливка

Заливка (flooding) — это алгоритм обмена локальной информацией между маршрутизаторами. Он заключается в рассылке пакетов по всем исходящим линиям, исключая ту, по которой он пришел. Если не предпринимать специальных мер, алгоритм порождает множество дублирующих пакетов в сетях с избыточными связями (петлями).

Чтобы исключить наводнение дублирующих пакетов, в каждый пакет помещается счетчик транзитных участков, называемый иногда временем жизни пакета TTL (time to live). При прохождении через маршрутизатор время жизни пакета уменьшается на 1, а по достижении нуля пакет уничтожается.

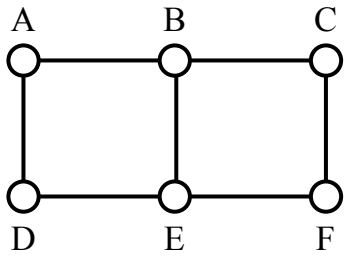
Достоинством алгоритма является надежность — любой пакет будет доставлен в любой узел сети. Алгоритм используется также для широковещательной рассылки.

6.2.3. Маршрутизация по вектору расстояний

Одним из первых алгоритмов динамической маршрутизации является маршрутизация по вектору расстояний (distance vector routing). Этот алгоритм использовался в сети ARPANET, известен как алгоритм Беллмана-Форда, в сети Интернет известен как алгоритм RIP.

При этом маршрутизаторы обмениваются друг с другом таблицами (векторами), содержащими кратчайшие пути к возможным адресатам и используемые для этого соединения. Для каждого маршрутизатора вектор содержит номер линии для данного получателя и метрику линии. В качестве метрики используется либо число транзитных участков, либо среднее время задержки, измеряемое специальным пакетом ECHO.

Пусть сеть определяется графом, изображенным на рисунке 50.



от A	от C	от E	от B
A 0	A 6	A 4	A 4 A
B 2	B 4	B 2	B 0 –
C 6	C 0	C 4	C 2 C
D 4	D 6	D 2	D
E 4	E 4	E 0	E 4 E
F 8	F 2	F 2	F

Рисунок 50 — Примерная сеть

Пусть маршрутизатор B получил от соседей A, C и E три вектора, показанные на рисунке 50 справа. В то же время маршрутизатор измерил задержку до соседей и получил метрики: BA = 4, BC = 2, BE = 4.

Он заносит измеренные значения в свою таблицу маршрутизации, показанную на рисунке 50 в самой правой части. Далее ему нужно вычислить метрики до маршрутизаторов D и F. Для этого он рассматривает таблицы, полученные от соседей. От A до D метрика равна 4. Следовательно, от B до D через A метрика составит 4 + 4 = 8. Соответственно, от B до D через C метрика составит 6 + 2 = 8, а от B до D через E метрика равна 2 + 4 = 6. Выбирая из полученных значений наименьшее, маршрутизатор B записывает в таблицу маршрутизации для D метрику 6, а выходную линию E. Таким же образом рассчитывается метрика для F. Через A метрика составит 8 + 4 = 12, через C 2 + 2 = 4, через E 2 + 4 = 6. Следовательно, для F метрика равна 4, выходная линия C.

Недостатком данного алгоритма является проблема счета до бесконечности, из-за которой алгоритм долго сходится. Он быстро реагирует на улучшение состояния линий, но медленно на их исчезновение. Пусть три маршрутизатора соединены в ряд: A—B—C, расстояние B—A равно 1, C—A равно 2. Пусть маршрутизатор A отключился, и B не может измерить расстояние до него. Однако B видит, что от C к A есть путь длиной 2, и он обновляет свою таблицу, записывая расстояние B—A, равное 3. Тогда C записывает себе расстояние C—A, равное 4, B исправляет свое расстояние на 5 и т.д. Процесс завершается, когда будет достигнуто максимально возможное расстояние плюс 1.

6.2.4. Маршрутизация с учетом состояний линий

Из-за проблемы счета до бесконечности алгоритм маршрутизации по вектору расстояний в сети ARPANET был заменен алгоритмом маршрутизации с учетом состояний линий (link state routing). Этот алгоритм используется протоколами маршрутизации IS-IS и OSPF.

В основе алгоритма лежат 5 требований к маршрутизатору. Каждый маршрутизатор должен:

1) обнаруживать своих соседей, узнавать их адреса;
2) определять метрику линии связи с каждым из соседей;
3) создавать пакеты с собранной информацией;
4) посылать пакет всем соседям и получать пакеты от них;
5) вычислять кратчайшие пути ко всем маршрутизаторам.

Маршрутизатор обнаруживает своих соседей, посылая по всем линиям пакет HELLO. Ответ соседнего маршрутизатора должен содержать уникальный идентификатор маршрутизатора. В случае, если несколько маршрутизаторов соединены широковещательным сегментом, формируется фиктивный маршрутизатор, роль которого исполняет один из них.

Стоимость или метрика линии связи может быть задана оператором или определена автоматически. Часто при этом используются значения,

обратно пропорциональные пропускной способности линии связи, например, значение 1 для 1 Гбит/с Ethernet и 10 для 100 Мбит/с Ethernet. Метрика может также определяться при помощи пакетов ECHO.

На основании полученной информации маршрутизатор создает пакеты состояний линий. Пакет содержит идентификатор маршрутизатора отправителя, порядковый номер, возраст и список соседей с указанием метрики линии, например: A | 1 | 255 | B:4 | D:6. Пакеты создаются либо периодически, либо при наступлении какого-либо события.

Для рассылки пакетов используется алгоритм заливки. Номер пакета используется для определения дубликатов, для чего маршрутизаторы записывают пары (идентификатор, номер). Кроме того, поле возраста уменьшается либо каждым маршрутизатором, либо каждые *t* секунд.

Пакеты состояния некоторое время сохраняются в маршрутизаторе на случай изменения топологии. Каждый пакет требуется переслать далее и подтвердить получение, поэтому для каждого пакета маршрутизатор хранит флаги отсылки и флаги подтверждения.

Маршрутизатор, получая пакеты состояний от соседних маршрутизаторов, формирует граф топологии сети, вычисление кратчайших маршрутов в котором выполняет алгоритм Дейкстры.

6.2.5. Иерархическая маршрутизация

Рост сети ведет к увеличению таблиц маршрутизации, времени их обработки и размеров служебных пакетов. Поэтому в сетях большого размера маршрутизация осуществляется иерархически.

В этом случае маршрутизаторы вычисляют кратчайшие пути только в пределах ограниченных по размеру регионов, а другие регионы представлены в таблицах маршрутизации одним из маршрутизаторов другого региона, называемого пограничным.

Недостатком иерархической маршрутизации является возможное удлинение маршрутов в удаленные регионы.

6.3. Борьба с перегрузкой

Перегрузка (congestion) возникает, когда очередь маршрутизатора к одной из его выходных линий заполняется. Вновь прибывающие пакеты при этом отбрасываются, что ведет к их повторной отправке и увеличению нагрузки. Причинами перегрузки являются низкая пропускная способность линий и недостаточно производительные маршрутизаторы.

Наличие перегрузки означает, что нагрузка на сеть превышает возможности ее ресурсов. Поэтому для устранения возможности перегрузки есть два подхода: снизить нагрузку и (или) добавить ресурсы.

6.3.1. Резервирование ресурсов

Добавление ресурсов достигается резервированием дополнительных линий связи и оборудования в рамках обеспечения отказоустойчивости системы. Этот процесс называется обеспечением (provisioning), и предполагает краткосрочное и долгосрочное прогнозирование нагрузки.

Краткосрочные прогнозы используются для включения в работу дополнительного оборудования и линий связи в заранее рассчитанные моменты, например, когда активность пользователей сети возрастает в течение суток. Долгосрочные прогнозы используются в случаях, когда нагрузка на сеть меняется постепенно, например, при расширении сети.

6.3.2. Маршрутизация с учетом состояния трафика

Метрики, учитывающие загруженность линий, позволяют перевести трафик в другие маршруты. При этом постоянная часть метрики учитывает топологию сети, а переменная — измеренную нагрузку и среднее время ожидания в очереди. На ранних этапах развития сети Интернет такой подход имел место. Однако это может привести к колебаниям в сети, в которой есть несколько маршрутов между двумя регионами. Одним из решений этой проблемы является распределение трафика между несколькими путями. Другое решение заключается в замедлении перемещения трафика между разными путями так, чтобы колебания сходились к устойчивому состоянию. В современных сетях Интернет учет состояния трафика не используется. Вместо этого перегрузку регулируют за счет изменения входных данных. Это называется управлением трафика (traffic engineering).

6.3.3. Управление доступом

В сетях виртуальных каналов новые соединения могут быть отклонены, если это может привести к перегрузке сети. Это называют управлением доступом (admission control). Проблема заключается в том, что трафик сложно описать так, чтобы можно было точно рассчитать нагрузку на сеть.

6.3.4. Регулирование трафика

Для регулирования трафика (traffic throttling) в сетях используется обратная связь. При этом получатель просит отправителя снизить скорость передачи данных. Этот режим работы называется предотвращением перегрузки (congestion avoidance).

Первый подход заключается в явном уведомлении отправителя при помощи сдерживающих пакетов. (choke packet). Получив такой пакет, отправитель уменьшает трафик, например, на 50%.

Другой подход заключается в передаче уведомления о перегрузке в специальном поле обычного информационного пакета. Этот метод называется явным уведомлением о перегрузке (ECN, explicit congestion notification).

6.3.5. Сброс нагрузки

Сбросом нагрузки называется игнорирование маршрутизаторами пакетов, которые они не в состоянии обработать. Для выбора отбрасываемых пакетов используют две стратегии: винная (старое вино лучше нового) и молочная (свежее молоко лучше вчерашнего). Первая стратегия лучше подходит при передаче файла данных. Вторая стратегия подойдет при передаче потового видео. Маршрутизаторы могут также помечать пакеты как важные.

Одним из способов борьбы с перегрузкой является случайное раннее обнаружение (RED, random early detect). В этом случае маршрутизаторы постоянно вычисляют среднюю длину своих очередей. При достижении некоторого порога часть пакетов удаляется случайным образом. Это позволяет протоколам транспортного уровня немедленно обнаружить утерю пакета безо всяких уведомлений о перегрузке, и снизить трафик.

6.4. Качество обслуживания

Каждый поток пакетов может быть описан следующими характеристиками: требуемая пропускная способность, время задержки, флуктуация трафика и допустимые потери. Эти характеристики формируют то, что называют *качеством обслуживания* (QoS, quality of service). Для разных видов трафика требования к этим характеристикам могут значительно различаться, как примерно показано в следующей таблице.

Трафик	Полоса	Задержка	Флуктуации	Потери
Электронная почта	0	0	0	2
Передача файлов	1	0	0	2
Web-доступ	1	1	0	1
Удаленный доступ	0	1	1	1
Аудио по заказу	0	0	2	0
Видео по заказу	2	0	2	0
Телефония	0	2	2	0
Видеоконференции	2	2	2	0

Флуктуацией (jitter) называется колебание времени задержки пакета.

Очевидно, что для передачи файлов задержки и флуктуации не столь важны, как наличие потерь, а для видеоконференций, наоборот, задержки и флуктуации снижают качество, в то время как потери части пакетов не оказывают заметного влияния.

6.4.1. Формирование трафика

Формирование трафика (traffic shaping) — это способ регулирования средней скорости и равномерности потока. Это имеет значение в основном для данных, передаваемых в реальном времени.

Ранее был рассмотрен протокол скользящего окна, регулирующий скорость передачи. Есть еще два более общих алгоритма, формирующих равномерный поток данных: алгоритм дырявого ведра (leaky bucket) и алгоритм маркерного ведра (token bucket).

Дырявое ведро с небольшой дырочкой в днище формирует поток с постоянной скоростью независимо от скорости, с которой вода поступает в ведро. При переполнении ведра вода выливается через край. Реализация этого алгоритма передает пакеты в сеть с определенной скоростью. Если достигнут определенный объем пакетов, они либо ставятся в очередь (что более характерно для формирования трафика средствами операционной системы), либо отвергаются (что более характерно для сетевых интерфейсов).

Маркерное ведро можно представить как ведро, которое начинает выливать воду через край при его заполнении. Иначе говоря, ведро должно накопить некоторый объем (некоторое количество маркеров), прежде, чем появится исходящий поток. Равномерность исходящего потока достигается постоянной скоростью поступления маркеров. С маркером связывается либо пакет, либо некоторое количество байт. Первый вариант более приемлем, когда пакеты имеют одинаковую длину, а второй — при разной длине пакетов.

6.4.2. Диспетчеризация пакетов

Диспетчеризация пакетов устанавливает очередность отправки пакетов по исходящим линиям.

Обычное обслуживание предполагает, что к каждой из исходящих линий есть своя очередь поступающих пакетов. При этом более агрессивные потоки данных получают преимущество, так как в очереди оказывается большее число пакетов этих потоков.

Алгоритм *справедливого обслуживания* (fair-queuing) предполагает, что каждый из потоков данных к одной исходящей линии формирует

собственную очередь, и данные циклически поочередно отправляются из каждой очереди. При этом приоритет получают потоки с более длинными пакетами. Этот алгоритм можно улучшить, если учитывать не количество пакетов, а количество байт.

При *взвешенном справедливом обслуживании* (WFQ, weighted fair queuing) учитывается вес потока, что дает возможность посылать большее количество байт потокам, для которых установлен больший вес.

При приоритетном обслуживании каждому пакету присваивается некоторый приоритет, и высокоприоритетные пакеты отправляются в первую очередь. Недостатком такого обслуживания является неопределенное время ожидания обслуживания низкоприоритетных потоков.

6.5. Объединение сетей

Существует множество разнородных сетей, отличающихся технологиями и протоколами, и эти сети нужно объединять, чтобы дать пользователям возможность общаться независимо от их местоположения. При этом приходится учитывать следующие аспекты отличия сетей:

- сервис (ориентирован на соединение или без соединения),
- способ адресации,
- возможность широковещания и многоадресной рассылки,
- размер пакета,
- качество обслуживания,
- надежность,
- безопасность,
- параметры,
- тарификация.

Некоторые из этих аспектов, такие, как адресация или широковещание, не вызывают особых затруднений при переходе от одной сети к другой, в то время как другие, например, качество обслуживания или безопасность, представляют собой сложную задачу.

Идею создания общего слоя, сглаживающего различия сетей, выдвинули Винтон Серф (Vinton Cerf) и Роберт Кан (Robert Kahn), в 1974 году разработавшие протоколы TCP/IP, основу сети Интернет. За это изобретение они были удостоены в 2004 году премии Алана Тьюринга.

IP использует универсальный формат пакетов, распознаваемый любыми маршрутизаторами, и передаваемый практически любой сетью. На границе двух сетей располагается мульти-протокольный маршрутизатор. Он должен либо преобразовывать протоколы, либо обеспечивать соединение на уровне протокола более высокого уровня, и ни один из вариантов не отвечает всем требованиям сетей. Поэтому соединение сетей является чрезвычайно сложной задачей.

6.5.1. Туннелирование

Одним из способов соединения разнородных сетей является туннелирование. Оно применяется, когда между двумя одинаковыми сетями находится сеть другого типа. При этом пакет первой сети инкапсулируется в пакет второй сети и пересылается по второй сети с использованием ее протокола. На приемной стороне пакет извлекается и передается получателю. Так, например, пакет протокола IPv6 может быть вложен в пакет протокола IPv4. В результате появляется новая сеть, которая как бы накладывается на старую, ее называют *оверлейной* (overlay). Туннелирование применяется также в виртуальных частных сетях VPN (virtual private network), применяемых для обеспечения безопасности.

6.5.2. Маршрутизация в объединенных сетях

В пределах каждой сети для маршрутизации используется *внутридоменный* (intradomain) или *внутренний шлюзовый протокол* (interior gateway protocol). Это может быть протокол RIP или OSPF.

Для маршрутизации между сетями используется *междоменный* (interdomain) или *внешний шлюзовый протокол* (exterior gateway protocol). Внутридоменные протоколы сетей могут быть разными, но междоменный протокол должен быть общим. В сетях Интернет в качестве междоменного протокола используется пограничный межсетевой протокол BGP (border gateway protocol).

Так как все сети управляются независимо, их называют автономными системами, AS (autonomous system). Таким образом, в объединенных сетях используется двухуровневый алгоритм маршрутизации.

6.5.3. Фрагментация пакетов

Сети и каналы имеют ограничения на размер пакетов. Например, в сети Ethernet размер пакета равен 1500 байт, в сетях 802.11 — 2272 байта, а протокол IP позволяет пересылать пакет в 65515 байт. Ограничения на размер пакета накладывают аппаратура, операционная система, протокол, желание соответствовать какому-либо стандарту, желание снизить количество повторно пересылаемых пакетов и другие. Максимальный размер пакета, пересылаемого сетью, обозначают MTU (maximum transmission unit).

Отправитель обычно не знает маршрут следования пакета, поэтому не может определить путевое значение MTU (path MTU). Альтернативным решением является разрешение маршрутизаторам разбивать пакеты на фрагменты, и посылать их как отдельные пакеты.

В современных сетях используется поиск путевого значения MTU. При отправке IP-пакета в его заголовке устанавливается признак запрещения разбиения пакета на фрагменты. Если маршрутизатор не может переслать пакет, он сообщает отправителю об ошибке и удаляет пакет. Отправитель, используя информацию от маршрутизатора, перераспределяет данные так, чтобы получить пакет необходимой длины. Понятно, что данный механизм увеличивает задержку.

Тем не менее, фрагментация между отправителем и получателем все равно необходима, но современной тенденцией является ее вынесение из сети на хосты.

6.6. IP-адресация

IP-адресация используется в сети Интернет и в IP-сетях. IP-адрес представляет собой 32-х битное целое число, записываемое в виде 4-х групп. Этот адрес должен быть уникален для каждого сетевого интерфейса в пределах сети, либо сети Интернет, либо IP-сети. Узел сети имеет отдельный сетевой интерфейс для каждой сети, с которой он соединен. Обычно хосты имеют один сетевой интерфейс и подключены к локальной сети. Маршрутизаторы соединены с несколькими сетями и имеют несколько сетевых интерфейсов.

6.6.1. Бесклассовая междоменная маршрутизация

Бесклассовая междоменная маршрутизация CIDR (classless interdomain routing) использует иерархическую организацию IP-адреса в виде адреса подсети и адреса хоста. Старшая часть адреса задает сеть и называется префиксом, а младшая часть — хост этой подсети.

Обозначим число бит старшей части N (network), число бит младшей части H (host). Очевидно, что H = 32 – N. Сетевая часть адреса одинакова для всех хостов сети, и определяет непрерывный блок адресов, используемых в данной сети. Наименьший адрес в блоке является адресом сети. Размер блока задается через размер сетевой части адреса N, и равен 2^H или 2^{32-N}.

Таким образом, сеть может быть задана записью A.B.C.D/N.

Маской сети называется 32-х битное целое число, в котором старшие N разрядов равны 1, а младшие H разрядов равны 0. Операция И над адресом и маской выделяет адрес сети.

Пусть задана сеть 1.10.96.64/26. Запишем все части адреса и маски в виде двоичных чисел:

00000001.00001010.01100000.01000000 — адрес сети

11111111.11111111.11111111.11000000 — маска

Для этой сети H = 32 – 26 = 6, соответственно, в конце маски 6 нулей, и полное число адресов в сети равно $2^6 = 64$. В этом непрерывном блоке адресов только 63 адреса могут быть использованы сетевыми интерфейсами, а начальный адрес блока — это адрес сети. Тогда минимальный адрес интерфейса равен 1.10.96.65, максимальный 1.10.96.127.

Бесклассовая междоменная маршрутизация позволяет сократить размер таблиц маршрутизации. Пусть три подсети 1.1.0.0/21, 1.1.0.8/22 и 1.1.0.16/20 соединены с маршрутизатором в Москве, который имеет соединение с маршрутизатором в Челябинске. Этот последний вместо трех записей о подсетях в Москве может иметь одну запись, содержащую агрегированный адрес 1.1.0.0/19, префикс из 19 бит которого покрывает собой любой адрес любой из подсетей в Москве.

6.6.2. Полноклассовая адресация

До 1993 года IP-адреса делили сети на 5 категорий, обозначаемых буквами от A до E. Сети категорий A, B и C определяют сети большого, среднего и малого размера, сети категорий D и E имеют специальное назначение. В следующей таблице *n* обозначает часть адреса, используемая как адрес сети, *h* — часть адреса для адресации хоста, *a* — адрес группы для групповой рассылки, *u* — зарезервировано.

A:	0*nnnnnnn.hhhhhhhh.hhhhhhhh.hhhhhhhh*	1.0.0.0…127.255.255.255
B:	10*nnnnnn.nnnnnnnn.hhhhhhhh.hhhhhhhh*	128.0.0.0…191.255.255.255
C:	110*nnnnn.nnnnnnnn.nnnnnnnn.hhhhhhhh*	192.0.0.0…223.255.255.255
D:	1110*aaaa.aaaaaaaa.aaaaaaaa.aaaaaaaa*	224.0.0.0…239.255.255.255
E:	1111*uuuu.uuuuuuuu.uuuuuuuu.uuuuuuuu*	240.0.0.0…255.255.255.255

Развитие сетей показало несостоятельность данного способа адресации, принятого на заре развития сетей. Однако вторая половина адресов этого распределения используются и в настоящее время.

Часть IP-адресов предназначены для специальных целей.

0.0.0.0	этот хост (эта сеть)
0…0*host*	хост этой сети
255.255.255.255	широковещание в текущей сети
*network*1…1	широковещание в удаленной сети
127.*x.y.z*	обратная петля

Адрес, состоящий из нулей обозначает этот хост или эту сеть (во время загрузки). Адрес, сетевая часть которого обнулена, представляет хост в текущей сети. Адрес из одних единиц является широковещательным в текущей сети. Если единицами заменен адрес хоста, адрес является широковещательным в удаленной сети. Наконец, адрес, начинающийся с байта 127, используется при отладке. Пакеты, посылаемые на это адрес, являются пакетами, приходящими на данный хост.

6.6.3. Протокол IP версии 4

Задача протокола заключается в обеспечении доставки пакета независимо от того, в какой сети находится получатель. Для этого заголовок IP-пакета содержит информацию, которой должно быть достаточно для пересылки. Заголовок состоит из обязательной и необязательной частей. Размер заголовка измеряется 32-х битными словами. Обязательная часть содержит 5 слов, то есть 20 байт, необязательная часть может содержать до 10 слов, то есть до 40 байт, максимальный размер заголовка 60 байт.

Структура обязательной части заголовка показана на рисунке 51.

Рисунок 51 — Обязательный заголовок IP-дейтаграммы IPv4

Биты передаются слева направо, сначала старший бит. Заметим, что современные процессоры используют обратную нумерацию бит.

В поле версии указывается число 4. Поле IHL (internet header length) указывает фактический размер заголовка в 32-х битных словах. Минимальное значение равно 5, максимальное 15.

Поле типа службы первоначально было предназначено для задания приоритета (3 бита) и предпочитаемой характеристики обслуживания — задержка, пропускная способность или надежность (3 бита). Сейчас поле называется Differentiated Services Code Point, DSCP (дифференциальное обслуживание), первые 6 бит задают класс обслуживания, 2 последних бита используются для явных уведомлений о перегрузке (ECN).

Поле полной длины указывает полный размер пакета (заголовок и данные) в октетах (байтах). Диапазон значений 20…65535.

Идентификатор содержит номер пакета, если пакет фрагментирован.

Поле флагов состоит из 3 бит. Первый бит всегда 0. Второй бит, DF (don’t fragment), указывает, что пакет не должен фрагментироваться. Третий бит MF, (more fragments), указывает, что данный пакет содержит не последний фрагмент, если используется фрагментация.

Смещение фрагмента задается в октетах, и показывает положение фрагмента в пакете. Длины фрагментов должны быть кратны восьми, за исключением последнего фрагмента. Максимальное число фрагментов в дейтаграмме составляет $2^{13} = 8192$.

Время жизни задает число маршрутизаторов, которое пакет может пройти. Каждый маршрутизатор уменьшает значение поля на 1. Когда значение достигает нуля, пакет отбрасывается, отправителю отправляется сообщение ICMP time exceeded, тип 11, код 0.

В поле протокол указывается протокол верхнего уровня в соответствии со списком протоколов транспортного уровня (IANA protocol numbers). Так, протокол ICMP обозначается числом 1, TCP — 6, UDP — 17.

Контрольная сумма вычисляется в соответствии с RFC 1071. При проверке сумма включается в подсчет и результат должен быть нулевой. Контрольная сумма каждым маршрутизатором вычисляется заново.

Необязательные поля заголовка редко используются маршрутизаторами, но могут быть использованы для следующих целей:

- безопасность (уровень секретности дейтаграммы);
- строгая маршрутизация (заданный путь следования);
- свободная маршрутизация (обязательные маршрутизаторы);
- запоминание маршрута;
- временные отметки.

6.6.4. Трансляция сетевого адреса

В настоящее время свободных IP-адресов IPv4 практически нет, в то время как число хостов, подключаемых к сетям Интернет, не стремится к уменьшению. Дефицит IP-адресов IPv4 привел к появлению методов более эффективного их использования.

Один из методов заключается в том, что провайдеру Интернет выделяется некоторое количество адресов. Учитывая, что не всем клиентам одновременно требуется доступ к Интернет, провайдер динамически выделяет адреса из своего пула активным клиентам, а по окончании их активности возвращает адреса обратно в пул. Это позволяет провайдеру иметь больше клиентов, чем число имеющихся у него адресов Интернет.

В некоторых случаях этот метод эффективен, однако есть организации, в которых подключения к Интернет должны быть постоянными.

Другим широко распространенным методом является трансляция сетевого адреса, NAT (network address translation, RFC 3022). Всё адресное пространство делится на публичные (белые) адреса, которые только и могут использоваться в сетях Интернет, и частные (серые) адреса, которые никогда не должны появляться в сетях Интернет. Для частных адресов выделено три диапазона адресов, которые могут многократно встречаться в разных сетях, не имеющих прямого доступа к Интернет:

10.0.0.0/8
172.16.0.0/12
192.168.0.0/16

В этом случае организация (провайдер) имеет ограниченное число публичных адресов, например, один, и выход в сети Интернет из частной сети происходит через этот публичный адрес. Например, хост с частным адресом 192.168.1.1 может послать пакет хосту Интернет с адресом 185.31.160.133, заменяя свой адрес адресом 198.16.20.24, который есть у организации. При этом возникает проблема ответного пакета. Если запрос к хосту Интернет сделан от имени одного публичного адреса, каким образом ответный пакет поступит к хосту частной сети?

Для решения этой проблемы между частной сетью и сетями Интернет устанавливается так называемое натирующее устройство, которое выполняет трансляцию частных адресов публичные и обратно. Обычно натирующим устройством является маршрутизатор.

В трансляции используется тот факт, что когда процесс одного хоста связывается через Интернет с процессом другого хоста, процессы хостов нумеруются, и номер процесса называется портом. Это совершенно необходимо потому, что хост может иметь несколько процессов, связанных с Интернет. Например, сервер Интернет может обслуживать одновременно web-запросы, электронную почту и файловый доступ. Номера портов для наиболее известных протоколов закреплены IANA под номерами 0…1023 и называются общеизвестными, номера 1024…49151 называются зарегистрированными и используются для менее распространенных протоколов, номера 49152…65535 называются динамическими и могут быть использованы для разных целей.

Собственно трансляция выполняется следующим образом. Порт отправителя в пакете, выходящем в сеть Интернет, заменяется индексом таблицы, в которой запоминаются реальный порт процесса-источника и IP-адрес хоста отправителя. Когда приходит ответный пакет, содержащий замененный порт отправителя, этот последний используется для поиска в таблице настоящего порта и IP-адреса.

Номера портов указываются в полях пакета TCP, и это самый большой недостаток NAT. Во-первых, трансляция адресов возможна только при использовании протокола TCP, который обычно используется для пересылки пакетов в Интернет. Во-вторых, протокол TCP относится к вышестоящему уровню стека протоколов, а это нарушает принцип независимости протоколов разных уровней. В третьих, нарушается принцип сквозного общения сервисов одного уровня. Входящий пакет не может быть принят частной сетью с натирующим устройством до тех пор, пока на будет отправлен исходящий пакет.

Несмотря на недостатки NAT, этот механизм используется повсеместно. Необходимость в трансляции адресов отпадет, если все узлы сетей перейдут на адресацию IPv6, но это, видимо, случится не скоро.

6.6.5. Протокол IP версии 6

Проблема нехватки IP-адресов версии 4 назрела давно. Уже в 1990 году проблемная группа проектирования Интернет IETF начала работу над новой версией протокола IP. Основные его цели следующие:

1) обеспечение принципиально большего числа адресов;
2) уменьшение размера таблиц маршрутизации;
3) упрощение протокола для ускорения работы маршрутизаторов;
4) обеспечение безопасности;
5) управление качеством обслуживания;
6) упрощение многоадресной рассылки;
7) возможность изменения положения хоста не изменяя его адрес;
8) возможность дальнейшего развития протокола;
9) сосуществование нового и старого протоколов.

В результате обсуждений нескольких вариантов был разработан протокол SIPP (simple Internet protocol plus), обозначаемый IPv6. Обязательная часть заголовка протокола приведена на следующем рисунке.

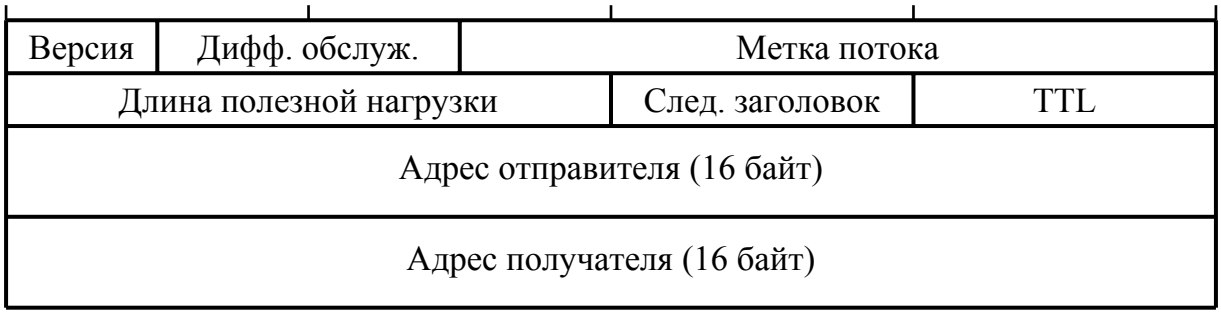


Рисунок 52 — Обязательная часть заголовка протокола IPv6

Поле версии содержит 6 или 4. Поле дифференциального обслуживания используется для обеспечения качества обслуживания. Поле метки потока указывает на определенные свойства потока и требования к обработке, при этом устанавливается соединение между отправителем и получателем. Длина полезной нагрузки не включает 40 байт заголовка, поэтому полная длина пакет составляет 65535 байт вместо 65515.

Дополнительные заголовки, если есть, образуют связанную цепочку в том смысле, что каждый заголовок имеет поле следующего заголовка, а последний из заголовков указывает в этом поле протокол пакета, то есть UDP или TCP.

Адрес IPv6 состоит из 128 бит, и записывается в виде 8 групп, разделяемых двоеточием. Если группа состоит из нулей, их можно не писать, равно как и лидирующие нули. Адрес IPv4 записывается с двумя лидирующими двоеточиями. Адреса IPv6 могут быть одноадресными (unicast), многоадресными (multicast) и групповыми (anycast).

6.7. Управляющие протоколы Интернет

6.7.1. ICMP

Протокол управляющих сообщений Интернет (Internet control message protocol) используется маршрутизаторами и тестирующими пакетами (командами ping и tracert). Пакеты ICMP инкапсулируется в пакет IP, они никогда не подтверждаются и не отправляются повторно.

Структура пакета ICMP показана на следующем рисунке.

Рисунок 53 — Структура пакета ICMP

Основные типы сообщений следующие.

0 (эхо-ответ, echo reply).

3 (адресат недоступен, destination unreachable), 16 кодов.

5 (перенаправление, redirect), 4 кода.

8 (эхо-запрос, echo).

11 (время жизни пакета истекло, time exceeded), 2 кода.

12 (неверный параметр, parameter problem), 3 кода.

13 (запрос метки времени, timestamp request).

14 (ответ с меткой времени, timestamp reply).

Сообщения 13 и 14 используются для измерения производительности сети.

6.7.2. ARP и InARP

Протокол разрешения адресов ARP (address resolution protocol) используется для определения канального MAC-адреса, соответствующего сетевому IP-адресу. Когда IP-пакет прибывает в сеть Ethernet, для формирования кадра требуется MAC-адрес хоста вместо IP-адреса пакета. Для этого всем хостам локальной сети посылается широковещательный кадр, содержащий пакет ARP с запросом, имеющим вид:

«У кого адрес A.B.C.D? Ответьте X.Y.Z.W»

Хост, IP-адрес которого совпадает с A.B.C.D, отправляет хосту с адресом X.Y.Z.W кадр с ответным пакетом ARP, содержащим MAC-адрес. Чтобы избежать определения MAC-адреса X.Y.Z.W, в пакет ARP включаются четыре адреса, два сетевых и два канальных. Канальные адреса используются для непосредственного ответа в сети Ethernet. Сетевые адреса используются для прохождения через маршрутизатор.

Структура пакет ARP приведена на следующем рисунке.

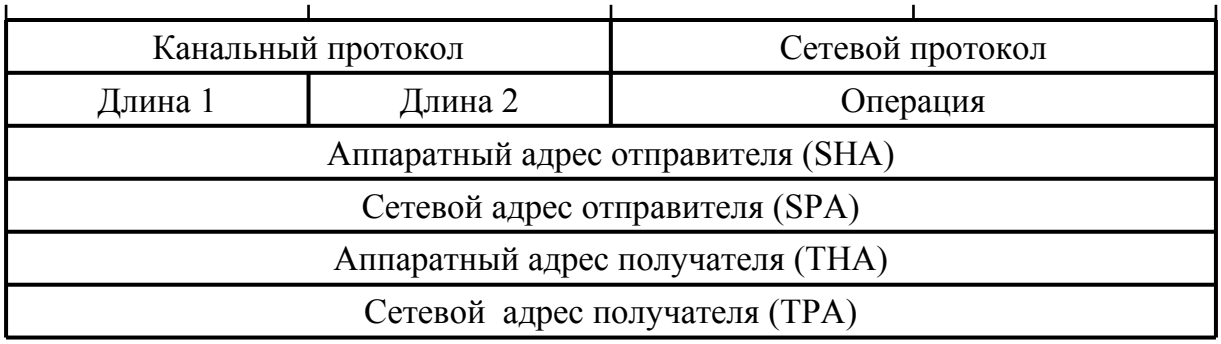


Рисунок 54 — Пакет протокола ARP

Канальный протокол Ethernet обозначается номером 0x0001. Сетевой протокол Для IPv4 обозначается номером 0x0800. В полях длины указывается размер соответствующего адреса в байтах. Операция 1 обозначает запрос, операция 2 — ответ.

Протокол InARP используется для определения сетевого адреса по известному канальному адресу (выполняет обратное разрешение).

Чтобы сократить число запросов ARP, значения сохраняются в кэше хоста или маршрутизатора, однако они хранятся там несколько минут.

6.7.3. DHCP

Ранее обратное разрешение адреса выполнялось протоколом RARP (reverse ARP). Запрос RARP имеет вид: «мой канальный адрес a:b:c:d. Знает ли кто мой сетевой адрес?» Для этой цели в сегменте локальной сети должен быть RARP-сервер. Альтернативный загрузочный протокол BOOTP использует UDP-пакеты, пересылаемые маршрутизатором в другие сети. Этот протокол сообщает станции не только IP-адрес, но и IP-адрес шлюза, IP-адрес DNS-сервера, маску подсети и другие сетевые параметры. В дальнейшем протокол BOOTP был заменен протоколом DHCP (dynamic host configuration protocol).

Каждая сеть имеет DHP-сервер, отвечающий за настройки. При загрузке хоста IP-адрес неизвестен. Хост посылает широковещательный запрос DHCP DISCOVER. Этот запрос поступает на DHCP-сервер независимо от того, в каком сегменте сети находится хост. Сервер выделяет хосту IP-адрес и посылает ответный пакет DHCP OFFER.

DHCP-сервер выделяет IP-адреса одним из следующих способов.

1. Адреса назначаются вручную администратором.

2. Адреса выделяются из заданного пула адресов.

3. Адреса сдаются в аренду на некоторый срок (leasing). По истечении срока хост должен продлить срок использования адреса.