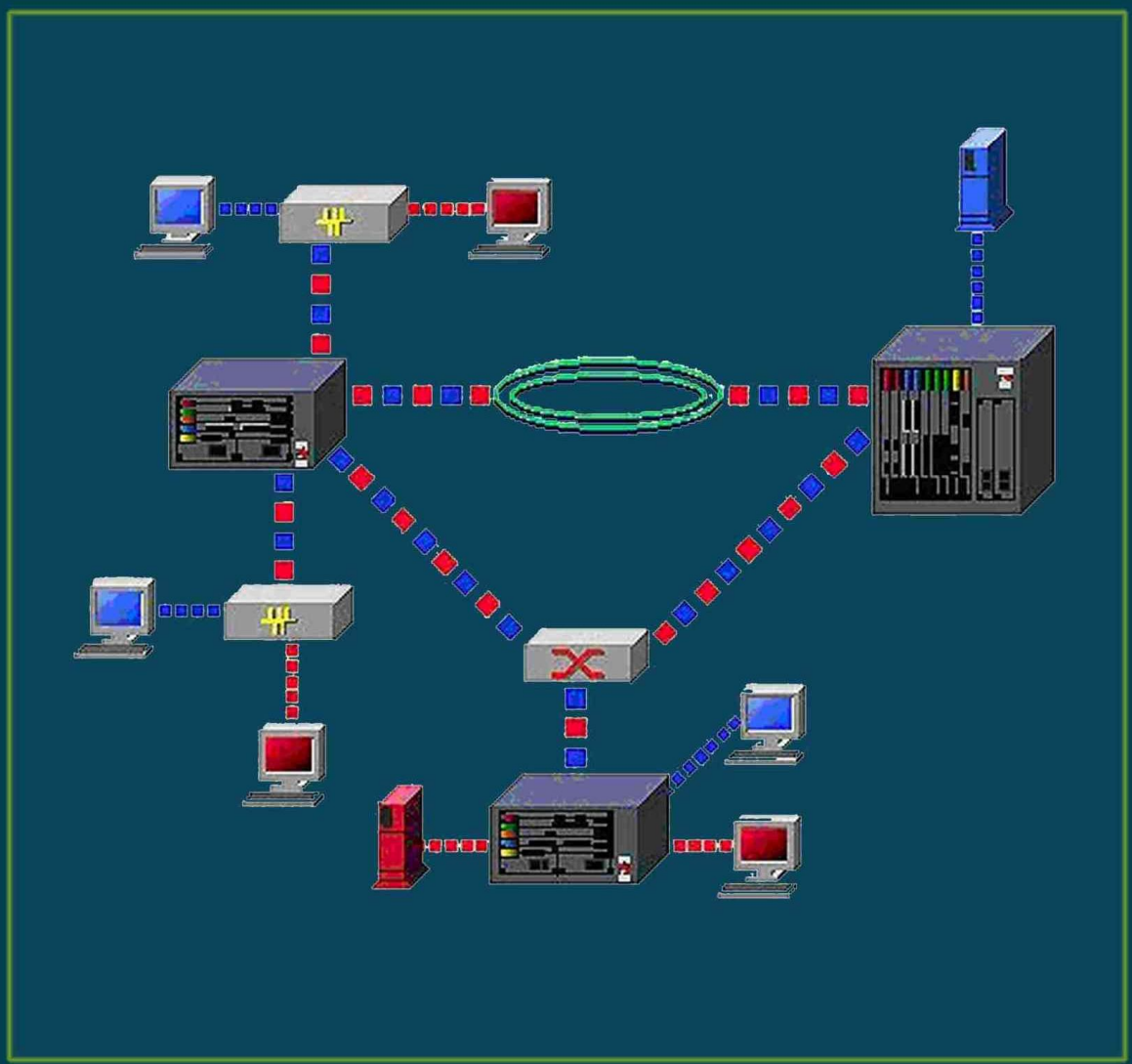
В. ЧЕРНЕГА, Б. ПЛАТТНЕР

КОМПЬЮТЕРНЫЕ СЕТИ



В. Чернега, Б. Платтнер

КОМПЬЮТЕРНЫЕ СЕТИ

Рекомендовано Министерством образования и науки Украины
в качестве учебного пособия
для студентов высших учебных заведений,
обучающихся по направлению "Компьютерные науки"

Севастополь 2006

ББК 32.973.202 я73
Ч 46
УДК 681.324 (075)

Рецензенты:
Сапожников Н.Е., проф., д-р техн. наук, Севастопольский национальный университет ядерной энергии и промышленности, зав. кафедрой компьютеризированных систем.
Кулаков Ю.А. проф., д-р техн. наук, Национальный технический университет Украины (КПИ), кафедра вычислительной техники.
Терещенко А.А. проф., д-р техн. наук, Севастопольский филиал Европейского университета, зав. кафедрой информационных систем.

Научный редактор: Маригодов В.К., д-р техн. наук, проф.

Чернега В., Платтнер Б.

Ч 46 **Комп'ютерні мережі**: Учбовий посібник для вищих навчальних закладів / В. Чернега, Б. Платтнер.- Севастополь: Вид-во СевНТУ, 2006.- 500 с.

ISBN 966-7473-98-8
Підручник призначений для студентів технічних університетів, що навчаються за напрямком "Комп'ютерні науки". В ньому викладені основні теоретичні положення архітектури і технології сучасних локальних та глобальних комп'ютерних мереж, а також способи забезпечення їх захисту від несанкціонованого доступу.
Підручник також може бути використаний студентами, що навчаються за напрямками "Комп'ютерна інженерія" та "Комп'ютеризовані системи, автоматика і управління".

Чернега В., Платтнер Б.

Ч 46 **Компьютерные сети**: Учебное пособие для вузов / В.Чернега, Б. Платтнер - Севастополь: Изд-во СевНТУ, 2006.- 500 с.

ISBN 966-7473-98-8
Учебник предназначен для студентов технических университетов, обучающихся по направлению "Компьютерные науки". В нем изложены основные теоретические положения архитектуры и технологии современных локальных и глобальных компьютерных сетей, а также способы их защиты от несанкционированного доступа.
Учебник также может быть использован студентами, которые обучаются по направлениям "Компьютерная инженерия" и "Компьютеризированные системы, автоматика и управление".

Решение МОН Украины о присвоении грифа "Рекомендовано в качестве учебного пособия для студентов высших учебных заведений" № 1-4/18-Г-1050 от 9.11.2006.

ISBN 966-7473-98-8 ББК 32.973.202 я73

СПИСОК СОКРАЩЕНИЙ

АКД	Аппаратура канала данных
АЛ	Абонентская линия
АМ	Амплитудная модуляция
АТС	Автоматическая телефонная станция
БИС	Большая интегральная схема
БК	Балансный контур
ВК	Виртуальный канал
ВОЛС	Волоконно-оптическая линия связи
ВОС	Взаимодействие открытых систем
ВРК	Временное разделение каналов
ДПФ	Дискретное преобразование Фурье
ДС	Дифференциальная система
ИКМ	Импульсно-кодовая модуляция
КАМ	Квадратурная амплитудная модуляция
КК	Коммутация каналов
ЛС	Линия связи
МККТТ	Международный консультативный комитет по телефонии и телеграфии
М	Маршрутизатор
ОДПФ	Обратное дискретное преобразование Фурье
ОЗУ	Оперативное запоминающее устройство
ОКС	Общеканальная сигнализация
ООД	Оконечное оборудование данных
ОФМ	Относительно-фазовая модуляция
ПК	Персональный компьютер
ПСП	Псевдослучайная последовательность
ПШК	Первичный широкополосный канал
РОС	Решающая обратная связь
РС	Рабочая станция
СбПД	Сборщик пакетов данных
СКС	Структурированная кабельная система
СЛ	Соединительная линия
СОВ	Система обнаружения вторжений
С1	Стык линейный (канальный)
С2	Стык последовательный
ТА	Телефонный аппарат
ТФОП	Коммутируемая телефонная сеть общего пользования
ТЧ	Тональная частота
УАВ	Устройство автоматического вызова
УЗО	Устройство защиты от ошибок
УКК	Узел коммутации каналов
УКП	Узел коммутации пакетов
УПС	Устройство преобразования сигналов
ФАПЧ	Фазовая автоподстройка частоты
ФЛ	Физическая линия
ФМ	Фазовая модуляция
ФРМ	Фазоразностная (относительная) модуляция

ЧМ	Частотная модуляция	
ЧРК	Частотное разделение каналов	
ЦСП	Первичная цифровая ступень преобразования	
AAL	ATM Adaptive Level	Уровень адаптации сети ATM
ACK	Acknowledge	Положительное подтверждение
ADSL	Asymmetric Digital Subscribe Line	Асимметричная цифровая абонентская линия
ANSI	American National Standards Institute	Американский институт национальных стандартов
ARP	Address Resolution Protocol	Протокол разрешения адресов
ARQ	Automatic Repetition Query	Автоматический переспрос
AS	Autonomus System	Автономная система
ASCII	American Standard Code for Information Interchange	Американский стандартный код для обмена информацией
ATM	Asynchronous Transfer Mode	Асинхронный режим передачи
BGP	Border Gateway Protocol	Протокол внешней маршрутизации
BPSK	Binary Phase Shift Keying	Двоичная фазовая манипуляция
BT	Bit Time	Длительность двоичного элемента
CCK	Complementary Code Keying	Комплементарные кодовые последовательности
CRC	Cyclic Redundancy Check	Контроль с использованием циклического избыточного кода
CTS	Clear To Send	Готов к передаче
DBPSK	Differential Binary Phase Shift Keying	Дифференциальная двоичная ФМ
DCE	Data Circuit-Terminating Equipment	Оконечное оборудование данных
DSL	Digital Subscribe Line	Цифровая абонентская линия
DSSS	Direct Sequence Spread Spectrum	Расширение спектра прямой последовательностью
DNS	Domain Names Service	Служба доменных имен
DTE	Data Terminal Equipment	Оконечное оборудование данных
DQPSK	Differential Quadrature Phase Shift Keying	Дифференциальная квадратурная фазовая манипуляция
DWDM	Dense Wave Division Multiplexing	Плотное волновое мультиплексирование
EOT	End of Text	Конец текста
ES-ES	End System – End System	Протокол маршрутизации стека OSI
FHSS	Frequency Hopping Spread Spectrum	Расширение спектра скачкообразным изменением частоты
FR	Frame relay	Сеть ретрансляции кадров
FDM	Frequency Division Multiplexing	Мультиплексирование с частотным разделением
FSK	Frequency Shift Keying	Частотная модуляция
FTP	File Transfer Protocol	Протокол передачи файлов
EGP	Exterior Gateway Protocol	Протокол внешней маршрутизации
HDLC	High Level Data-link Control	Высокоуровневое управление линией

HTML	Hyper Text Markup Language	Язык гипертекстовой разметки
HTTP	Hyper Text Transfer Protocol	Протокол гипертекстовой передачи
IEEE	Institute of Electrical and Electronic Engineers	Международный институт инженеров по электротехнике и электронике
ISDN	Integrated Services Digital Network	Цифровая сеть интегрального обслуживания
ICMP	Internet Control Message Protocol	Протокол передачи управляющих сообщений
IGRP	Interior Gateway Routing Protocol	Протокол внутренней межшлюзовой маршрутизации
IGP	Interior Gateway Protocol	Протокол внутренней маршрутизации
IP	Internet Protocol	Межсетевой протокол
IP v6	Internet Protocol Version 6	Межсетевой протокол 6-й версии
IPG	Inter Packet Gap	Межпакетный зазор
IS-IS	Intermediate System	Протокол маршрутизации стека OSI
ISN	Initial Sequence Number	Начальный порядковый номер
ITU	International Telecommunication Unit	Международной телекоммуникационный союз
LAN	Local Area Network	Локальная сеть
LAP	Link Access Procedure	Процедура доступа к каналу
LAPB	Link Access Procedure Balanced	Сбалансированная процедура доступа к каналу
LAP-M	Link Access Protocol for Modems	Протокол доступа к каналу для модемов (V-42)
LAWN	Local Area Wireless Network	Беспроводная локальная сеть
LLC	Logical Link Control	Логическое управление каналом (FDDI; 802.2)
MAC	Medium Access Control	Управление доступом к среде (Ethernet, FDDI, Token Ring)
MAN	Metropolitan Area Network	Региональная или муниципальная сеть
MAU	Multistation Access Unit	Блок доступа к большому числу станций
MBONE	Multicast Backbone	Канал группового доступа
MDI	Medium Dependent Interface	Интерфейс, зависимый от типа среды
MFSK	M-ary Frequency Shift Keying	Многопозиционная ЧМ
MIME	Multipurpose Internet Mail Extensions	Протокол многоцелевых расширений электронной почты
MSL	Maximum Segment Lifetime	Максимальное время жизни сегмента
MSS	Maximum Segment Size	Максимальный размер сегмента (TCP)
MTU	Maximum Transfer Unit	Максимальная длина пересылаемого блока данных
NAK	Negative AcKnowledgement	Отрицательное подтверждение/запрос
NIC	Network Information Center	Международный информационный центр
NT	Network Terminator	Сетевой терминатор
OFDM	Orthogonal Frequency Division Multiplexing	Частотное разделение с ортогональными сигналами
OSPF	Open Shortest Path First	Выбор первого кратчайшего пути
PVC	Permanent Virtual Circuit	Постоянный виртуальный канал

PDH	Plesiochronous Digital Hierarchy	Плезиохронная цифровая иерархия
PDV	Path Delay Value	Время двойного оборота
POP	Post Office Protocol	Почтовый офисный протокол
PPP	Point-to-Point Protocol	Протокол двухточечной связи
QAM	Quadrature Amplitude Modulation	Квадратурная амплитудная модуляция
QoS	Quality of Service	Качество обслуживания
QPSK	Quadrature Phase Shift Keying	Квадратурная фазовая модуляция
RAM	Random Access Memory	Память с произвольным доступом
RARP	Reverse ARP	Обратный протокол разрешения адресов
RFC	Request For Comment	Стандарты Интернета
RIP	Routing Information Protocol	Протокол маршрутной информации
RS-232	Recommended Standard 232	Последовательный интерфейс 232
RTS	Request To Send	Запрос передачи
SDH	Synchronous Digital Hierarchy	Цифровая синхронная иерархия
SDSL	Symmetric Digital Subscribe Line	Симметричная DSL
SLIP	Serial Line Internet Protocol	Канальный последовательный протокол
SMTP	Simple Mail Transfer Protocol	Простой протокол электронной почты
SNA	Systems Network Architecture	Сетевая архитектура систем (IBM)
SNMP	Simple Network Management Protocol	Простой протокол управления сетью
SOH	Start of Header	Начало заголовка
STM	Synchronous Transport Module	Синхронный транспортный модуль
STP	Shielded Twisted Pair	Экранированная витая пара
STX	Start of Text	Начало текста
SVC	Switched Virtual Circuit	Коммутируемый виртуальный канал
TA	Terminal Adaptor	Терминальный адаптер
TCM	Trellis Coded Modulation	Треллис-кодирование
TCP/IP	Transport Control Protocol / Internet Protocol	Транспортный протокол управления передачей/ межсетевой протокол
TDM	Time Division Multiplexing	Временное разделение каналов
TE	Terminal Equipment	Терминальное оборудование
TFTP	Trivial FTP	Упрощенный протокол FTP
TOS	Type of Service	Тип обслуживания
UDP	User Datagram Protocol	Протокол пользовательских дейтаграмм
URL	Uniform Resource Locator	Унифицированный указатель ресурсов
UTC	Coordinated Universal Time	Универсальное согласованное время
VC	Virtual Circuit	Виртуальный канал
VCI	Virtual Circuit Identifier	Идентификатор виртуального канала
VP	Virtual Path	Виртуальный путь
VPI	Virtual Path Identifier	Идентификатор виртуального пути
WAN	Wide Area Networks	Глобальная сеть
WDM	Wawe Division Multiplexing	Волновое мультиплексирование
Wi-Fi	Wireless Fidelity	Торговая марка беспроводного оборудования локальных сетей
WWW	World Wide Web	Всемирная гипермедиа информационная служба

ПРЕДИСЛОВИЕ

Компьютерные сети являются одним из важнейшим средств реализации современных информационных технологий. Благодаря им возможно осуществлять распределенную обработку данных, получать доступ к различным хранилищам информации, осуществлять резервирование и покупку билетов на транспортные средства, покупку товаров, производить размещение заказов на различных фирмах и осуществлять финансовые операции и т.п.

В связи с этим курс "Компьютерные сети" является одной из важнейших профилирующих дисциплин в процессе подготовки бакалавров по направлению "Компьютерные науки" и инженеров по специальности "Информационные управляющие системы и технологии". Базовыми дисциплинами, обеспечивающими курс "Компьютерные сети", являются: "Кодирование и защита информации", "Элементы информационных систем", "Операционные системы", "Архитектура компьютеров" и "Системы передачи данных". Поэтому перед началом изучения излагаемого курса необходимо повторить основные теоретические положения, освещаемые в названных дисциплинах.

Книга написана на основе материалов лекций, читаемых авторами в Севастопольском национальном техническом университете (СевНТУ) и Федеральной высшей технической школе г. Цюриха (Швейцария), а также практических и лабораторных занятий, проводимых в этих вузах.

Теоретический материал, представленный в книге, условно делится на шесть частей: общие принципы построения компьютерных сетей и теоретические основы передачи сигналов и данных в компьютерных сетях; принципы построения локальных сетей; особенности объединения локальных и региональных сетей и функционирования объединенной сети Интернет; принципы построения глобальных сетей; основы безопасности компьютерных сетей. В книге подробно рассмотрены топологии компьютерных сетей, способы адресации и управления в сетях, протоколы обмена информацией на всех уровнях сети.

Большое внимание уделяется вопросам построения локальных сетей, методам доступа к среде, а также схемотехнической и программной реализации различных подсистем. Достаточно подробно рассмотрены новые техно-

логии транспортировки сообщений в глобальных сетях, в частности, ISDN-, SDH-, FR- и ATM-сети.

Значительная часть учебника посвящена принципам построения и функционирования глобальной сети Интернет, способам адресации и структуризации сетей, методам доступа к хранилищам файлов, сервисным службам сети Интернет, функционированию электронной почты. В книгу введены подразделы, не входящие в традиционные учебники по компьютерным сетям, в частности по архитектуре беспроводных компьютерных сетей, систем модуляции и доступа к среде, а также раздел по безопасности компьютерных сетей.

При представлении учебного материала использован системно-ориентированный подход при котором упор делается на изложение основных теоретических положений материала разделов. Затем теоретические принципы иллюстрируются примерами их конкретной реализации. Такой подход соответствует международным рекомендациям “Computing Cirricula 2001: Computer Science” по преподаванию информатики в университетах и способствует живучести учебника вне зависимости от неизбежных изменений в технических и программных средствах. Построение учебника призвано также облегчить преподавателям различных направлений изложение материала при наличии ограничений на количество часов, регламентируемых учебным планом на изучение данной дисциплины.

Авторы старались применять минимальное количество аббревиатур, столь характерных в литературе по компьютерным сетям, которые существенно затрудняют первоначальное знакомство с изучаемым материалом. Многие специальные понятия сопровождаются написанием их на английском языке, что позволяет избежать неоднозначности толкования неустановившихся в русскоязычной литературе терминов, а также закрепления в памяти студентов интернациональных терминов, применяемых в компьютерных технологиях.

Учебник не претендует на исчерпывающую полноту изложения материала по компьютерным сетям. В частности, в нем не рассмотрены вопросы диагностики и оптимизации сетей, мало уделено внимания программной реализации сетевых задач. Предполагается, что студенты самостоятельно будут постоянно углублять знания в области компьютерных сетей в процессе выполнения лабораторного практикума и курсового проекта по данной дисциплине, а также в процессе самостоятельной подготовки и трудовой деятельности.

Особенностью построения учебника является наличие расширенных выводов по каждому разделу, в которых в концентрированном виде освещены основные положения и принципы соответствующей темы дисциплины. Это позволит студентам в процессе подготовки к экзамену при повторной

проработке материала учебника ограничиться прочтением и осмыслением только расширенных выводов.

Список литературы содержит перечень книг и других материалов, использованных при написании учебника и рекомендуемых для самостоятельного углубленного изучения дисциплины.

В заключение авторы благодарят ректорат СевНТУ за помощь при подготовке рукописи и руководство Федеральной высшей технической школы г. Цюриха (ETHZ) за спонсирование издания книги, а также рецензентов – коллектив кафедры компьютеризированных систем Севастопольского национального университета ядерной энергии и промышленности, зав. кафедрой д-р техн. наук проф. Сапожникова Н.Е., заведующего кафедрой вычислительной техники Киевского национального технического университета Украины (КПИ) д-р техн. наук проф. Луцкого Г.М. и д-ра техн. наук, профессора этой же кафедры Кулакова Ю.А., заведующего кафедрой информационных систем Европейского университета д-ра техн. наук, проф. Терещенко В.А., эксперта научно-методической комиссии по направлению "Компьютерные науки", д-ра техн. наук, профессора ИПСА НТУУ (КПИ) Зайченко Ю.П. за доброжелательную критику и полезные замечания, которые способствовали улучшению книги.

Особую благодарность авторы выражают д-ру техн. наук, профессору В.К. Маригодову за его большой труд по редактированию рукописи, а также профессору кафедры кибернетики и вычислительной техники СевНТУ, д-ру техн. наук Апраксину Ю.К. и канд. техн. наук, доценту кафедры информационных систем СевНТУ Кротову К.В. за внимательное первое чтение рукописи и полезные замечания.

Отзывы и предложения просьба направлять по адресу: Украина, 99053, г. Севастополь-53, Студгородок, Севастопольский национальный технический университет, НМЦ.

Раздел 1

ОБЩИЕ ПРИНЦИПЫ ПОСТРОЕНИЯ КОМПЬЮТЕРНЫХ СЕТЕЙ

Компьютерная сеть и Интернет – это синонимы или различные понятия? С какой целью сети делят на уровни? Что означает "стек протоколов"? Почему модели реальных сетей отличаются от эталонной модели? Как осуществляется коммутация у временных коммутаторов, если никакого соединения линий не происходит? Каким образом сетевые узлы знают оптимальные пути передачи пакетов, если ситуация в сети постоянно изменяется. Какой смысл использовать протоколы, не гарантирующие надежной передачи пакетов? Что такое "перегрузка сети" и как с ней бороться? Какие функции выполняет сеть сигнализации? На эти и другие вопросы Вы найдете ответ, изучив материалы данного раздела.

1.1. Обобщенная структура и топология компьютерных сетей

1.1.1. Структура компьютерных сетей и их классификация

Компьютерная сеть представляет собой совокупность взаимосвязанных технических средств и программного обеспечения, предназначенных для распределенной обработки данных, а также для обмена и передачи данных между любыми пользователями (абонентами) сети. В состав технических средств входят множество персональных компьютеров и серверов, а также узлов коммутации и распределения информации, соединенных между собой каналами передачи данных. Информационный поток данных, передаваемых между компьютерами сети, называется *сетевым трафиком* или просто **трафиком** (*traffic*). Программное обеспечение включает сетевые операционные системы, управляющие работой компьютеров в сети, а также пакеты программ, обеспечивающих передачу и установление соединений между пользователями сети, маршрутизацию и оптимальное распределение потоков сообщений между узлами, передачу управляющих сообщений между узлами и конечными пользователями, контроль и учет функционирования сети. В общем виде компьютерная сеть может быть представлена в виде совокупности *оконечных пунктов* (ОП) и промежуточных *узлов коммутации* (УК), соединенных между собой *каналами* и *линиями связи* (рисунок 1.1).

Оконечным пунктом (узлом) компьютерной сети обычно является персональный компьютер с установленной сетевой операционной системой и снабженный устройством передачи данных. В принципе к сети могут быть

подключены и другие типы источников и потребителей цифровой информации. Компьютер оконечного пункта объединенных сетей в литературных источниках часто называют также **хостом** (*Host*), а большую (главную) ЭВМ сети – **мэйнфреймом** (*Mainframe*). В локальных компьютерных сетях оконечный пункт обычно называют "**Рабочая станция**". Совокупность двух узлов сети, соединенных каналом передачи данных, получил название **звено данных** (*Data Link*).

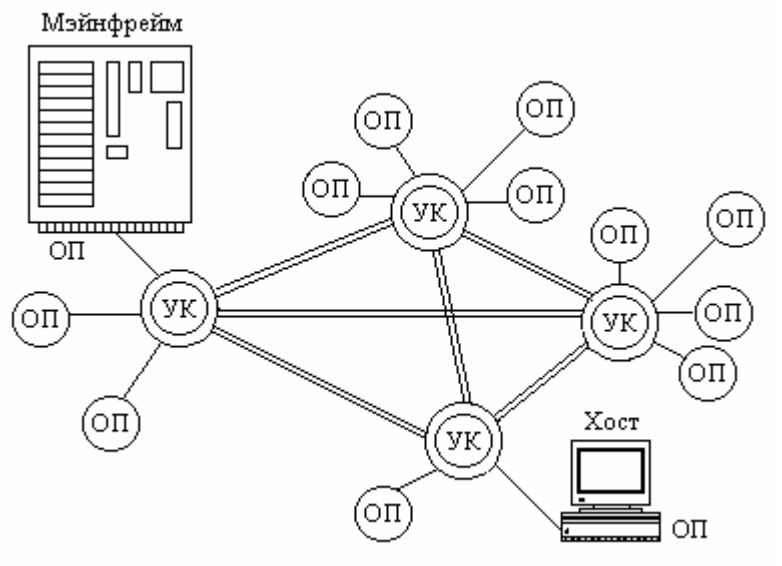


Рисунок 1.1 – Обобщенная структурная схема компьютерной сети

Узлы коммутации позволяют осуществить обмен информацией между любыми оконечными пунктами сети за счет установления между ними связи на время передачи данных путем соединения (коммутации) участков линий сети в единый тракт. После завершения обмена данными эти же участки линий могут быть использованы для организации тракта обмена между другими ОП. Таким образом, наличие в сети узлов коммутации позволяет более эффективно использовать дорогостоящие линии и линейное оборудование сети путем их максимальной загрузки.

Другим вариантом построения единой компьютерной сети является объединение независимых сетей, так называемых *подсетей (Subnet*) отдельных регионов или организаций с помощью узлов коммутации, роль которых выполняют маршрутизаторы (М) или шлюзы (рисунок 1.2). Объединенная сеть в англоязычной литературе получила название *Internetwork* или сокращенно ***Internet***.

Компьютерные сети (КС) классифицируют по назначению, составу оборудования, программному обеспечению и функциональным возможно-

стям, по пространственному расположению и способу установления соединения, а также по ряду других признаков.

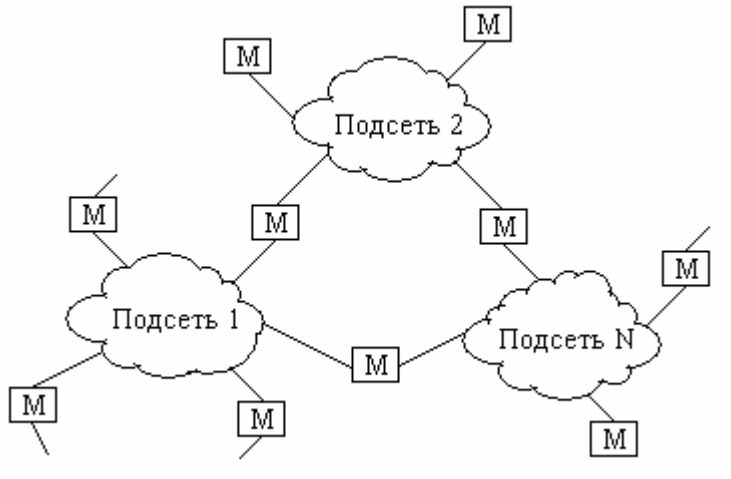


Рисунок 1.2 – Структура объединенной компьютерной сети Интернет

По функциональному назначению различают вычислительные, информационные, информационно-вычислительные и информационно-управляющие сети.

Вычислительные сети предназначены главным образом для решения задач пользователей с распределением ресурсов между компьютерами сети. **Информационные** сети ориентированы в основном для предоставления информационного обслуживания по запросам пользователей. **Информационно-вычислительные** сети объединяют функции вычислительных и информационных сетей. **Информационно-управляющие** сети осуществляют сбор оперативной информации, осуществляют ее обработку и принимают решение по управлению объектами или процессами, распределенными в пространстве. В настоящее время большинство сетей являются информационно-вычислительными.

По назначению различают компьютерные сети общего пользования (универсальные), обслуживающие круг разнообразных пользователей и специализированные сети. К последним следует отнести сети управления производством и учрежденческие.

По типу используемых компьютеров различают однородные (**гомогенные**) сети, содержащие программно-совместимые компьютеры, и разнородные (**гетерогенные**).

По расположению в пространстве различают локальные и глобальные сети. **Локальные сети** (***LAN****–Local Area Networks*) ограничены территорией одного помещения, цеха или предприятия. Характерной особенностью локальных сетей является использование в качестве среды передачи сигналов

высококачественных электрических, оптических или иных линий связи, причем длина линии связи не превышает нескольких километров. Передача данных в *LAN* **осуществляется со скоростью** от 10 Мбит/с до нескольких тысяч Мбит/с. Локальные сети чаще всего являются однородными. В настоящее время на практике наиболее широко используются следующие локальные сети: ***Ethernet***, ***Fast-*** и ***Gigabit Ethernet***, ***Token Ring*** и ***FDDI*** *(Fiber Distributed Data Interface).*

Глобальные сети (***WAN*** – *Wide Area Networks*) расположены на большой территории (населенный пункт, область, государство). К характерным особенностям таких сетей относится использование для передачи данных каналов связи общего пользования (телефонных, первичных широкополосных и каналов более высокого порядка, цифровых каналов связи различных порядков и т.д.). Скорости передачи данных в глобальных сетях сравнительно невысокие и лежат в пределах 56 … 2000 кбит/с. Кроме этого, каналы глобальных сетей характеризуются относительно высоким уровнем помех, что требует применения специальных мер защиты от ошибок. Глобальные сети в принципе являются разнородными. Примерами глобальных сетей являются: компьютерная сеть ***SNA*** *(Systems Network Architecture)* для передачи информации в сетях фирмы *IBM*; цифровая сеть с интерацией услуг ***ISDN*** *(Integrated Services Digital Network);* цифровая сеть с коммутацией пакетов ***X*.25**; сеть с ретрансляцией пакетов ***Frame Relay*;** объединенная мировая компьютерная сеть ***Internet*** и ряд других.

В последнее время в отдельный вид выделяют *городские* и *корпоративные сети*. Городские сети (***MAN*** – *Metropolian Area Networks*) предоставляют сетевые услуги на территории крупных городов. Они объединяют локальные сети различных организаций города, а также обеспечивают соединения с глобальными сетями. Городские сети для внутренних пересылок данных используют цифровые магистральные каналы связи на основе волоконно-оптических линий со скоростью передачи от 45 Мбит/с и выше. Городские сети – преимущественно разнородные. *Корпоративные сети* являются сетями масштаба предприятия, объединяющие подсети отдельных подразделений организации, расположенных территориально в разных частях населенного пункта, страны или континента. Для передачи информации между подсетями используются линии и каналы связи, применяемые как в локальных, так и глобальных сетях. В общем случае корпоративная сеть имеет гетерогенный характер.

По способу установления соединений между взаимодействующими оконечными пунктами различают сети с постоянным включением каналов связи (некоммутируемые сети), сети с коммутацией каналов и сети с коммутацией сообщений и пакетов. В **сетях с коммутацией каналов** пользователи соединяются сквозными физическими или логическими каналами только

на время обмена информацией. В **сетях с коммутацией сообщений** передача информации осуществляется без предварительного соединения взаимодействующих узлов. В этих сетях сообщение от отправителя поступает на узел коммутации сообщений, где запоминается (ставится на очередь) и передается по указанному адресу в соответствии с категорией срочности. Если необходимые участки сети заняты, то сообщения хранятся на узлах до освобождения канала связи или очередного узла. Под **сообщением** понимается логически завершенная последовательность данных – запрос на передачу файла, ответ на этот запрос, содержащий весь файл и т.п. Сообщения могут иметь произвольную длину – от нескольких байт до многих мегабайт. **Сеть с коммутацией пакетов** является разновидностью сети с коммутацией отрезков сообщений (пакетов), длина которых составляет от десятков до нескольких тысяч байтов. **Пакет** представляет собой последовательность байтов, состоящей из заголовка с управляющей информацией и данных, передаваемой через сеть как минимальная независимая единица сообщения.

В зависимости от *способа взаимодействия* различают сети с **независимыми** (равноправными) **сторонами** (*Peer-to-Peer* сети) и сети, взаимодействующие по модели "**клиент-сервер**". В первом типе каждая из сторон может выступать в роли ведущей, начинающей работу путем отправки инициирующего сообщения или запроса на обслуживание. В сетях с использованием модели "клиент-сервер" активной, запрашивающей стороной является клиент, а сервер постоянно находится в состоянии ожидания запроса. Он выполняет информационную услугу только в ответ на запрос клиента.

По функциям *управления сетевыми ресурсами* компьютерные сети делят на централизованные и децентрализованные. В **централизованных** сетях управление всеми сетевыми ресурсами осуществляет один из ее узлов (сервер). Для **децентрализованных** сетей характерно автономное распределение ресурсов, при котором каждый из узлов, используя информацию о состоянии сети, самостоятельно определяет возможность доступа к ее ресурсам.

В зависимости от *прав собственности* на сети последние могут быть **сетями общего пользования** (*public*) или **частными** (*private*). Так к сетям общего пользования относятся телефонные сети **ТфОП** (PSTN – *Public Switched Telephone Network*) и сети передачи данных (PSDN – *Public Switched Data Network*).

В последнее время появился термин **интеллектуальная сеть** *(intelligent network)*. Под этим термином понимается коммуникационная сеть, которая кроме передачи данных, предоставляет *дополнительный информационный сервис*. Наряду с традиционными компонентами (узлы коммутации, мультиплексоры, центры управления) интеллектуальная сеть содержит сервисные центры, базы данных, узлы создания услуг, позволяющие

оказывать пользователям всевозможные информационные услуги, перечень которых постоянно расширяется. Интеллектуальная сеть является некоторой надстройкой, обеспечивающей применение интеллектуальных технологий для обработки запросов пользователями услуг на получение дополнительного сервиса.

По роду деятельности в сети различают **оператора сети** (*Network Operator*) и **поставщика сетевых услуг** (*Service Provider*). Оператором сети является компания, которая поддерживает сеть в рабочем состоянии. Поставщиком услуг (*провайдером*) называют компанию, оказывающую платные услуги абонентам сети. В ряде случаев владелец, оператор и провайдер могут относиться к одной компании.

1.1.2. Топология компьютерных сетей

Каждая сеть имеет свою **топологию**, т.е. схему пространственного расположения узлов и связей между ними. Наиболее характерные топологические структуры компьютерных сетей изображены на рисунке 1.3.

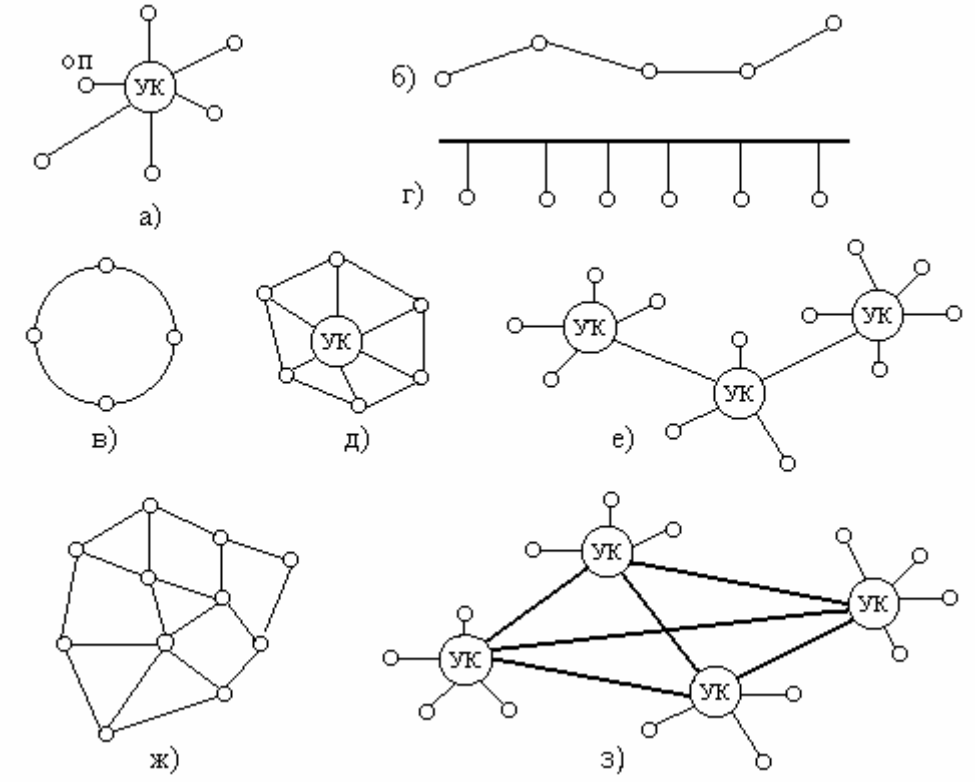


Рисунок 1.3 – Топология компьютерных сетей

К ним относятся: а) лучевая (радиальная или звездная); б) линейная; в) кольцевая; г) шинная; д) радиально-кольцевая; е) радиально-узловая; ж) сеточная; з) многосвязная.

Использование на практике той или иной топологии сети определяется расположением компьютеров и серверов (узлов коммутации) в пространстве, с учетом максимальной надежности сети и минимальных аппаратных затрат. Так, например, для предприятий управления магистральными газопроводами, линиями передачи электрической энергии или железнодорожной связи характерна линейная или радиально-узловая топология. Для локальных сетей наиболее целесообразной являются шинная, лучевая или кольцевая топологии. Для глобальных сетей характерна многосвязная или сеточная топология.

В зависимости от способа соединения звеньев сети между собой, различают **двухточечные** (рисунок 1.4,а) и **многоточечные** (рисунок 1.4,б) соединения.

Рисунок 1.4 – Двухточечное а) и многоточечное б) соединения звеньев сети

В двухточечных соединениях (*Point to Point*) информация от источника поступает на один приемник, а в многоточечных к линии передачи подключен ряд приемных устройств. Причем, информация в многоточечных подключениях может передаваться одновременно всем приемникам - **широковещательная** передача (*broadcasting*), части приемников – **групповая** передача (*multicasting*), либо любому приемнику по выбору – **адресная** передача (*unicasting*). Как видно из рисунка 1.3, для компьютерных сетей, за исключением шинной топологии, характерно использование двухточечных соединений.

В процессе функционирования компьютерной сети кроме реализации функций обмена информацией необходимо также осуществлять управление сетью. Основные функции управления и организации сети сводятся к следующему:

- установлению необходимых физических и логических соединений между взаимодействующими компьютерами;
- решению задач, связанных с адресацией и выбором пути распространения (маршрутизацией) передаваемых сообщений;

- контролю и исправлению ошибок при передаче данных по линиям и каналам связи, сжатию и защите информации;
- управлению взаимодействующими пользовательскими программами;
- управлению программами из состава математического обеспечения сети, реализующими различные виды информационных и вычислительных услуг;
- обеспечению конфигурации сети и состава ее технических и частично программных средств без нарушения функционирования сети в целом;
- обеспечению защиты сети от проникновения злоумышленников и нарушения ее функционирования.

Взаимодействие отдельных участков и компонентов компьютерной сети осуществляется по определенным правилам, которые называют **протоколами**.

1.1.3. Эталонная модель взаимодействия открытых сетей

Компьютерная сеть (КС) представляет собой сложную систему, элементами которой являются разнообразные аппаратные и программные средства. Для согласования взаимодействия ЭВМ, каналов связи, аппаратуры передачи данных (**АПД**), мостов и маршрутизаторов и пр. необходимы правила взаимодействия этих средств на различных уровнях. При этом правила взаимодействия различных уровней должны быть взаимонезависимыми.

На основе опыта разработки и эксплуатации компьютерных сетей в различных странах международной организацией по стандартизации МОС (англ. обозначение ***ISO*** – *International Standard Organization*), была разработана **Эталонная модель *взаимодействия открытых систем*** (ВОС), принятая в качестве международного стандарта (***OSI*** – *Open Systems Interconnection*). Суть эталонной модели ВОС заключается в том, что она унифицированным образом описывает принципы взаимодействия разнообразных сетевых систем друг с другом.

Термин "открытые" относится к системам, удовлетворяющим требованиям стандарта МОС по взаимосвязи, т.е. если две системы используют один и тот же стандарт, то они "открыты" друг для друга. Реальная открытая КС представляется для пользователя, взаимодействующего с ней, единым стандартным образом, который не зависит от аппаратных особенностей ЭВМ, языков программирования, типов операционных систем и т.д.

В соответствии с этой моделью сеть ЭВМ делится на ряд функциональных системных слоев – ***уровней***. Каждый уровень состоит из объектов, выполняет определенную логическую функцию и обеспечивает определен-

ный перечень услуг (сервис) для расположенного над ним уровня. Разбивка сложной системы на уровни позволяет разделить ее на ряд модулей, определить и стандартизировать функции каждого из модулей и интерфейсов между ними. Это приводит к упрощению проектирования системы в целом за счет возможности разработки и реализации каждого из модулей параллельно независимыми организациями, а модификация отдельных модулей может осуществляться без изменения остальной части системы.

МОС рекомендовала к использованию *семиуровневую* иерархию взаимодействия (рисунок 1.5). Взаимосвязь одноименных уровней компьютерной сети определяется стандартными для всей сети правилами. Объекты, выполняющие функции уровней, реализуются программным, программно-аппаратным или аппаратным способом. Как правило, чем ниже уровень (ближе к физической среде передачи), тем больше доля аппаратной части в его реализации.

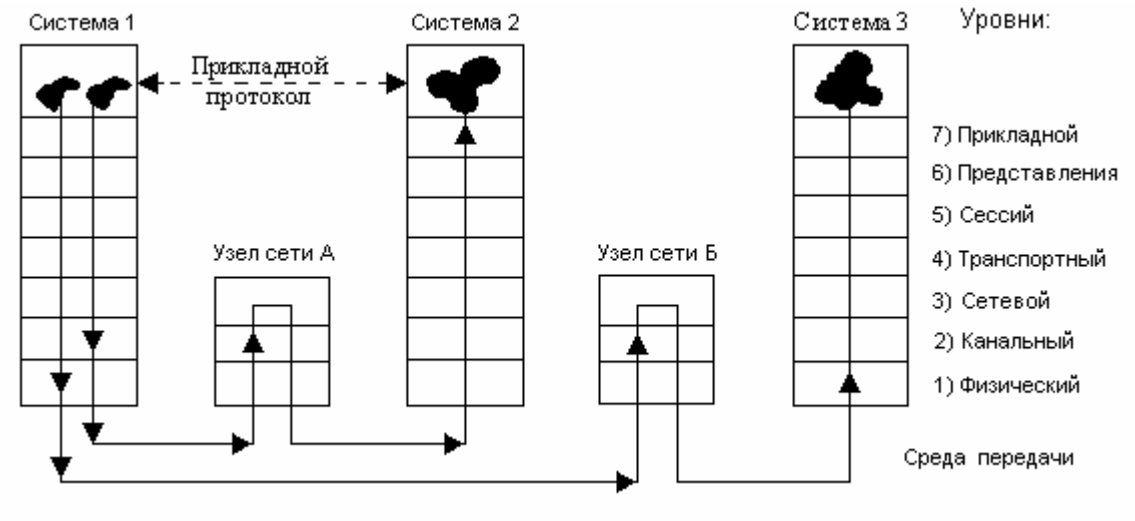


Рисунок 1.5 – Структура эталонной модели ВОС

Обмен данными между уровнями осуществляется информационными пакетами определенного формата. В каждом пакете наряду с данными содержится управляющая информация, размещенная в заголовках. При движении пакета сверху вниз по уровням (передача в сеть) каждый уровень добавляет к пакету свой заголовок. При движении пакета снизу вверх (прием из сети) каждый уровень обрабатывает пакет согласно управляющей информации в заголовке, добавленным к пакету соответствующим уровнем передающей стороны. Таким образом, одинаковые уровни на разных системах общаются между собой по определенным правилам. Совокупность процедур и правил взаимодействия объектов одноименных уровней называется

протоколом. Вложенный набор уровней образует набор протоколов, получивших название "**стек протоколов**". Правила взаимодействия смежных уровней одной и той же системы определяют **межуровневой интерфейс**.

Программа, реализующая функцию того или иного протокола, называется *модулем*, например, "транспортный модуль", "сетевой модуль" и т.д. Программные модули выполняют функции сервера и клиента. В роли сервера модуль ожидает запроса от выше- или нижележащего уровней на выполнение определенной услуги (сервиса). По завершению обработки очередного запроса модуль сам становится клиентом, посылая запрос на один из смежных уровней.

Границы между уровнями устанавливаются таким образом, чтобы взаимодействие между смежными слоями было минимальным и изменения, проводимые в пределах одного слоя, не требовали перестройки смежных уровней. В этой универсальной модели нет никакой привязки к конкретной аппаратуре используемых компьютеров, к аппаратуре соединяющих их сетей, к типу программного обеспечения, то есть модель *OSI* имеет в виду некую абстрактную систему.

Три верхних уровня – *прикладной, представления и сеансовый* – образуют *i*-й ***процесс*** и отображают в компьютерной сети пользователя и его задачу.

Четыре нижних уровня – *транспортный, сетевой, канальный и физический* – образуют ***транспортную сеть*** и обеспечивают собственно передачу данных.

Пользователям компьютерных сетей и прикладным программистам обеспечивается доступ только к самому верхнему – **прикладному** уровню, отвечающему за доступ приложений в сеть. Задачами этого уровня является управление сетью, перенос файлов, обмен почтовыми сообщениями и ряд других.

Уровень **представления** отвечает за возможность диалога между приложениями на разных машинах. Этот уровень обеспечивает преобразование данных (кодирование, компрессия и т.п.) прикладного уровня в поток информации для транспортного уровня. Протоколы уровня представления обычно являются составной частью функций трех верхних уровней модели.

На **сеансовом** уровне перечень услуг сводится к установлению сеансового соединения между двумя *приложениями*, распознаванию имен и обмену данными, выдаче сообщений об исключительных ситуациях, завершению сеансового соединения. Протоколы сеансового уровня также являются составной частью функций трех верхних уровней модели.

Транспортный уровень характеризуется рядом услуг, основными из которых являются: установление и разъединение транспортных соединений; обеспечение "***прозрачной***" передачи *пакетов* данных с *любым содержани-*

ем, форматом и способом кодирования; обеспечение заданного качества сервиса (пропускная способность, транзитная задержка, коэффициент необнаруженных ошибок и вероятность отказов).

Сетевой уровень в числе основных услуг осуществляет сетевое соединение и идентификацию конечных точек сетевых соединений; маршрутизацию сообщений и управление информационными потоками, обеспечение правильной последовательности доставляемых блоков информации.

На **канальном** уровне предоставляются следующие услуги: формирование блоков (кадров) данных и их передача; идентификация оконечных пунктов передачи данных канальных соединений; синхронизация кадров и защита от ошибок.

Физический уровень обеспечивает такие виды услуг как: установление и идентификацию физических соединений, организацию передачи последовательностей битов по физической среде, битовую синхронизацию, оповещение об окончании связи.

Услуги различных уровней определяются с помощью протоколов эталонной модели взаимодействия открытых систем.

Следует заметить, что хотя модель OSI предложена достаточно давно, однако протоколы, основанные на ней, применяются редко. Это объясняется тем, что, во-первых, эталонная модель появилась хронологически позже других моделей; во-вторых, эталонная модель характеризуется не всегда оправданной сложностью своих протоколов; и, в-третьих, существованием на момент появления модели OSI, хотя и не соответствующим строго эталонной модели, уже хорошо зарекомендовавшим себя стеком протоколов TCP/IP.

1.2. Коммуникационные протоколы

1.2.1. Общая характеристика протоколов

Протоколом называют совокупность семантических и синтаксических правил, которые определяют поведение процессов, систем и устройств или их частей, выполняющих конкретные логически взаимосвязанные функции при передаче данных (правила обмена сигналами и сообщениями между устройствами или процессами). При описании протокола принято выделять его логическую и процедурную характеристики. *Логическая характеристика* протокола – структура (формат) и содержание (семантика) сообщений – задается перечислением типов и значений сообщений. *Процедурной характеристикой* называют правила выполнения действий, предписанных протоколом взаимодействия. Такая характеристика может быть

представлена в различных формах: операторными схемами алгоритмов, моделями автоматов, сетями Петри и др.

Таким образом, логика организации компьютерной сети в наибольшей степени определяется протоколами, устанавливающими как тип и структуру сообщений, так и процедуры их обработки – реакцию на входящие сообщения и генерацию собственных сообщений. Существует большое разнообразие протоколов, различающихся областью применения, назначением, способом передачи управляющих сигналов и другими признаками.

По применению **(уровню)** различают:

- протоколы физического уровня – модемные протоколы V.21…V.92;
- канального уровня – SLIP, PPP;
- сетевого уровня – шлюзовые протоколы – IP, RIP, IPX;
- транспортного уровня; МККТТ X.224, TCP.
- протоколы прикладного уровня - передачи почтовых сообщений SMPT (*Simple Mail Transfer Protocol*); протокол для доступа к удаленному компьютеру TELNET; протокол передачи файлов FTP (*File Transfer Protocol).*

По **назначению:**

- протоколы установления соединения - V.8;
- модемной связи – V.21…34; V.90…92;
- факсимильной связи – T-30;
- коррекции ошибок – V.42;
- сжатия информации – V.42 bis, ,V.44, MNP5-6;
- передачи файлов – Xmodem, Zmodem, Kermit;
- эмуляции терминалов – Telnet;
- управления сетью – SNMP (*Simple Network Management Protocol*).

По **способу передачи управляющих сигналов**:

- байт-ориентированные – BSC;
- бит-ориентированные – HDLC.

По **области использования:**

- протоколы глобальных сетей – X.25, LAP, AAL.1-5;
- протоколы локальные сетей – NetBIOS, IPX, SPX..

Исторически сложилось так, что каждая организация, создававшая свою сеть, разрабатывала для нее стек протоколов. Слово "стек" отображает разделение сети на уровни, при котором протоколы верхних уровней располагаются над протоколами нижних. В настоящее время в компьютерных сетях используется большое количество стеков коммуникационных протоколов. Наиболее широко используются следующие стеки: TCP/IP, IPX/SPX, NetBIOS/SMB, и OSI. Все эти стеки, кроме SNA, на канальном и физиче-

ском уровне используют одни и те же стандартизированные протоколы локальных сетей Ethernet, Token Ring, FDDI. Однако на верхних уровнях различные компьютерные сети применяют стеки своих протоколов. Следует отметить, что эти протоколы часто не соответствуют уровням, рекомендуемым моделью OSI. Это поясняется тем, что модель OSI появилась позже, как результат обобщения используемых стеков.

Основным требованием к сетевому администратору являются доскональное знание разнообразных протоколов. Он должен уметь анализировать характерные поля передаваемых пакетов, обнаруживать причины отказа сети или резкое снижение ее производительности, производить оптимизацию сети и т.п.

Когда протокол настроен и функционирует оптимально, а взаимодействие его с другими протоколами осуществляется бесконфликтно, то вся сеть обеспечивает бесперебойную и своевременную доставку сообщений.

Для контроля функционирования сети разработаны специальные анализаторы протоколов, с помощью которых можно контролировать установление сетевых соединений, последовательность открытия сеанса связи, механизмы рассылки сообщений, межсетевую адресацию и маршрутизацию в сети.

1.2.2. Стек протоколов эталонной сети

Стек протоколов *OSI* представляет собой набор конкретных спецификаций сетевых протоколов (таблица 1.1), в отличие от модели *OSI*, которая является только концептуальной схемой взаимодействия открытых систем. Стек полностью соответствует модели OSI и включает спецификации протоколов для всех семи уровней. На нижних уровнях стек OSI поддерживает локальные сети Ethernet, Token Ring и FDDI, протоколы глобальных сетей ISDN, X.25 и ATM. Достаточно широко применяются протоколы прикладного уровня OSI: протокол передачи файлов FTAM (*File Transfer, Access, and Management*); справочной службы X.500; электронной почты X.400. Протоколы же сетевого, транспортного и сеансового уровней распространены мало. Наиболее применимы из них протоколы маршрутизации ES-ES (*End System - End System*) и IS-IS (*Intermediate System - Intermediate System*). По причине большой сложности алгоритмов протоколов, которая объясняется попыткой разработчиков сделать протоколы универсальными, на их реализацию расходуются значительные вычислительные ресурсы процессора. В связи с этим они применимы преимущественно в сетях с мощными компьютерами и суперкомпьютерами.

Таблица 1.1 – Уровни и стеки протоколов

<table>
<tr><td>Уровни</td><td colspan="4">Стеки протоколов</td></tr>
<tr><td>Модель OSI</td><td>OSI</td><td>TCP/IP</td><td>Novell</td><td>IBM/ Microsoft</td></tr>
<tr><td>Прикладной</td><td>X.400
X.500
FTAM</td><td rowspan="2">Telnet
FTP
SNMP
WWW</td><td rowspan="3">NCP
SAP</td><td rowspan="2">SMB</td></tr>
<tr><td>Представительный</td><td>Представительный OSI</td></tr>
<tr><td>Сеансовый</td><td>Сеансовый OSI</td><td rowspan="2">TCP
UDP</td><td rowspan="2">NetBEUI
NetBIOS</td></tr>
<tr><td>Транспортный</td><td>Транспортный OSI</td><td>SPX</td></tr>
<tr><td>Сетевой</td><td>ES-ES
IS-IS</td><td>IP
ICMP; IGMP
RIP; OSPF</td><td>IPX
RIP
NLSP</td><td></td></tr>
<tr><td>Канальный</td><td colspan="4">Ethernet, Token Ring, FDDI, Fast Ethernet, X.25, ATM, SLIP, PPP и др.</td></tr>
<tr><td>Физический</td><td colspan="4">Витые пары, коаксиальный кабель, волоконно-оптические линии, инфракрасное и радиочастотное излучение</td></tr>
</table>

1.2.3. Стек протоколов TCP/IP

В настоящее время группа протоколов TCP/IP является наиболее распространенным стеком транспортных протоколов компьютерных сетей. На этих протоколах построена всемирная сеть Интернет. Они были разработаны в середине 70-х годов XX-го столетия для связи экспериментальной сети Министерства обороны США ARPAnet с другими сетями. Протоколы позволяют осуществлять обмен между разнородными сетями.

Основными протоколами стека являются протоколы сетевого **IP** (*Internet Protocol*) и транспортного **TCP** (*Transmission Control Protocol*) уровней. Протокол IP обеспечивает негарантированную доставку пакетов между отдельными сетями составной сети, а TCP гарантирует безошибочную передачу пакетов между узлами сети.

Стек *TCP/IP* содержит протоколы четырех уровней (таблица 1.1): прикладной, транспортный, сетевой и канальный. Каждый уровень выполняет собственную функциональную нагрузку по решению основной задачи – обеспечения высокопроизводительной и надежной работы составной сети, части которой могут использовать различные сетевые технологии. Название

уровней по терминологии разработчиков *TCP/IP* несколько отличалось от названия уровней модели *OSI*, так как стек *TCP/IP* появился раньше стандарта *OSI*. В настоящее время для обозначения уровней стека *TCP/IP* преимущественно используется терминология модели *OSI*.

1. **Канальный уровень** (*link layer*). Разработчики назвали его уровнем сетевого интерфейса. Протоколы этого уровня должны обеспечивать интеграцию в составную сеть других сетей, какую бы внутреннюю технологию передачи эти сети не использовали. Он включает в себя драйвер устройства, находящийся в операционной системе, и соответствующую сетевую интерфейсную плату компьютера. Вместе они обеспечивают аппаратную поддержку физического соединения с сетью (с кабелем или с другой используемой средой передачи). На этом уровне стек протоколов *TCP/IP* поддерживает все широко используемые стандарты для локальных сетей *Ethernet, Token Ring* и *FDDI*, для глобальных – протоколы работы на аналоговых коммутируемых и выделенных линиях *SLIP, PPP*, а также протоколы цифровых сетей *X*.25 и *ISDN*.

2. **Сетевой уровень** (*network layer*), называемый ранее уровнем межсетевого взаимодействия, отвечает за передачу пакетов по сети. Маршрутизация пакетов осуществляется на этом уровне. Сетевой уровень обеспечивают: межсетевой протокол *IP* (*Internet Protocol*); протокол управления сообщениями *ICMP* (*Internet Control Message Protocol*) и протокол управления группами *IGMP* (*Internet Group Management Protocol*). На этом уровне функционируют протоколы RIP и OSPF, обеспечивающие маршрутизацию пакетов между узлами.

3. **Транспортный уровень** (*transport layer*) отвечает за передачу потока данных между двумя компьютерами и обеспечивает работу прикладного уровня, который находится выше. В семействе протоколов *TCP/IP* существует два транспортных протокола: *TCP* (*Transmission Control Protocol*) и *UDP* (*User Datagram Protocol*). *TCP* осуществляет надежную передачу данных между двумя компьютерами. Он выполняет деление данных, передаваемых от одного приложения к другому, на пакеты подходящего для сетевого уровня размера, подтверждение принятых пакетов, установку таймаутов, в течение которых должно прийти подтверждение на пакет, и так далее. Так как надежность передачи данных гарантируется на транспортном уровне, то на прикладном уровне эти функции не выполняются.

Протокол *UDP* предоставляет более простой сервис для прикладного уровня. Он просто отсылает пакеты, которые называются **дейтаграммами** (*datagram*) от одного компьютера к другому. При этом нет никакой гарантии, что дейтаграмма дойдет до пункта назначения. За надежность передачи данных, при использовании датаграмм, отвечает прикладной уровень. Для

каждого транспортного протокола существуют различные приложения, которые их используют.

4. **Прикладной уровень** (*application layer*) определяет детали каждого конкретного приложения. Существует несколько распространенных приложений TCP/IP, которые присутствуют практически в каждой реализации. В состав стека *TCP/IP* входят распространенные протоколы прикладного уровня, в частности протокол передачи файлов *FTP (File Transfer Protocol),* протокол эмуляции терминала *Telnet*, протокол пересылки почтовых сообщений *SMTP (Simple Mail Transfer Protocol)*, гипертекстовые сервисы службы *WWW* и др.

Первоначально протоколы стека *TCP/IP* использовались только в сетях *Internet*. Для реализации мощных функциональных возможностей протоколов требуются значительные вычислительные ресурсы. Поэтому на начальном этапе развития локальных сетей протоколы верхних уровней стека *TCP/IP* не находили в них широкого применения. С увеличением производительности микропроцессоров персональных компьютеров появилась возможность реализации стека *TCP/IP* и в локальных сетях. В настоящее время стек *TCP/IP* повсеместно используется в локальных, корпоративных и территориальных сетях. Широкое распространение стека протоколов TCP/IP обусловлено рядом его преимуществ, в частности следующими:

1) способностью делить (*фрагментировать*) большие пакеты на более мелкие части; это позволяет обеспечить совместимость функционирования разнородных сетей, имеющих различные максимальные длины пакетов.

2) гибкость адресации, дающую возможность относительно просто включать в объединенную сеть сети других технологий.

3) уменьшенным количеством широковещательных рассылок, при которых сетевой узел рассылает сообщение всем узлам подсети; это свойство особенно важно при передаче данных по низкоскоростным каналам связи, характерным для многих глобальных сетей.

1.2.4. Стеки протоколов Novell и IBM/Microsoft

Фирмой Novell для сетевой операционной системы NetWare в 80-х годах прошлого века был разработан стек протоколов **IPX/SPX**. Популярность стека связана с возможностью эффективной работы его в локальных сетях небольшого размера. Важным преимуществом протоколов было малое потребление сетевых ресурсов компьютера, что позволяло им реализовывать сети, построенные на компьютерах, управляемых операционной системой DOS. Стек протоколов реализован не только в операционной системе

NetWare, но и в ряде широко используемых сетевых ОС (*Sun Solaris, Microsoft Windows NT* и др.)

Протокол **IPX** (*Internetwork Packet Exchange*) обеспечивает возможность обмена пакетами данных (на уровне дейтаграмм) программам, запущенным на рабочих станциях. Относится к протоколам сетевого уровня.

SPX (*Sequence Packet eXchange*) и его усовершенствованная модификация SPX II выполняет функции, относящиеся к транспортному уровню эталонной 7-уровневой модели. Это протокол гарантирует доставку пакета и в случае потери или ошибки пакет пересылается повторно. Пакеты SPX вкладываются в пакеты IPX. SPX-протокол не посылает следующий пакет до тех пор, пока не получит подтверждение получения предшествующего. Прикладной уровень стека IPX/SPX реализуют протоколы NCP и SAP. Протокол NCP (*NetWare Core Protocol*) поддерживает все основные службы операционной системы Novell NetWare – службу обмена файлами, службу печати и т.д. Протокол SAP (*Service Advertising Protocol*) выполняет вспомогательную роль, позволяя резко сократить административные затраты по конфигурации клиентского программного обеспечения.

На сетевом уровне стек IPX/SPX использует протоколы маршрутизации RIP и NLSP (*NetWare Link Servis Protocol*), а на физическом и канальном уровнях – протоколы сетей Ethernet, Token Ring, FDDI и др.

В продуктах компаний IBM и Microsoft широко используется стек протоколов NetBIOS/SMB. Протокол **NetBIOS** (*Network Basic Input/Output System*) был создан для работы в локальных сетях для персональных ЭВМ типа IBM/PC в качестве интерфейса, независящего от фирмы-производителя. Протокол предназначен для использования группой ЭВМ, потребляет минимум системных ресурсов и реализует функции сеансового и транспортного уровней.

Через некоторое время была разработана улучшенная версия протокола NETBIOS - **NetBEUI** (*NetBios Extended User Interface*). Протокол выполняет много сетевых задач, относящихся к сетевому, транспортному и сеансовому уровням. Среди ограничений NetBEUI следует назвать отсутствие внутренней маршрутизации. По этой причине NetBEUI рекомендуется для локальных сетей. Некоторые ограничения NetBEUI устранены в последующей модификации этого протокола NBF (*NetBEUI Frame*). Протокол SMB (*Server Message Block*) выполняет функции сеансового, представительского и прикладного уровней. На его основе реализуется файловая служба, а также служба печати и передачи сообщений между приложениями.

На физическом и канальном уровнях стек NetBIOS/SMB использует протоколы сетей Ethernet, Token Ring, FDDI и др.

1.3. Способы коммутации в компьютерных сетях

В традиционных телефонных сетях, в которых передача ведется в аналоговой форме, коммутация состоит в физическом соединении в узлах коммутации необходимых участков линий и каналов связи и образовании сквозного тракта передачи сигналов от источника до получателя. В цифровых системах передачи, кроме прямого физического соединения, которое также может использоваться в узлах, возможно применение коммутации без организации физических соединений путем записи и считывания данных, относящихся к некоторому каналу, через определенную зону памяти.

Различают следующие способы коммутации в компьютерных сетях:

- *коммутация каналов* – создается сквозной тракт передачи путем последовательного соединения физических линий или каналов на время сеанса связи; после разъединения тракт распадается на отдельные составные части, которые могут использоваться для образования других каналов;
- *коммутация сообщений* – передача данных выполняется без установления сквозного соединения между взаимодействующими абонентами. Данные от абонента вначале передаются на ближайший узел коммутации, к которому он подсоединен и заносятся в запоминающее устройство узла. По мере освобождения каналов в направлении передачи и наличии свободной памяти в соседнем узле коммутации сообщение передается на следующий узел, занимая канал только на период времени передачи данных между смежными узлами;
- *коммутация пакетов* – осуществляется аналогично процедуре коммутации сообщений, но сообщение разделяется на более короткие фрагменты – пакеты. Пакет является самостоятельной адресуемой частью сообщения, передаваемой по сети независимо от других пакетов.

В цифровых системах с коммутацией каналов возможно комбинирование методов коммутации, использующих физические соединения (*пространственная коммутация*), с методами, базирующимися на применении памяти (*временная коммутация*). Задачей пространственной коммутации является перенос данных из одной электрической цепи (канала) в другую, а задачей временной – изменение временной позиции расположения битов или байтов данных в кадрах (пакетах).

1.3.1. Пространственная коммутация

В компьютерных сетях с коммутацией каналов (КК) при установлении соединения образуется сквозное физическое соединение между взаимодей-

ствующими абонентами сети. При этом сквозной канал составляется узлом коммутации каналов из отдельных участков сети и, как правило, устанавливается только на время сеанса связи. После завершения обмена данными канал разбирается и его составные части могут быть предоставлены другим пользователям. Типичным примером сети с коммутацией каналов является городская коммутируемая телефонная сеть общего пользования (рисунок 1.6), которая в настоящее время все еще используется для передачи данных в компьютерных сетях. Телефонный аппарат ТА подсоединяется к узлу коммутации каналов (УКК) посредством индивидуальной абонентской линии связи (АЛ). Узлы коммутации соединены между собой пучками соединительных линий (СЛ), которые применяются для коллективного использования. На рисунке 1.6 показан пример упрощенной схемы соединения $ТА_k$ и $ТА_m$, проложенного через четыре УКК.

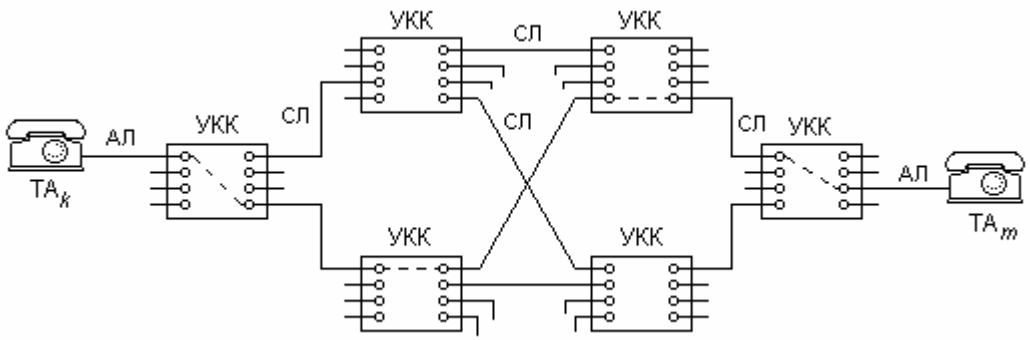


Рисунок 1.6 – Схема городской телефонной сети с коммутацией каналов

В телефонной сети с КК установление соединения осуществляется следующим образом. Абонент, инициирующий процесс обмена данными, посылает на свой узел коммутации запрос на установление соединения. Узел коммутации, при наличии свободных ресурсов (коммутирующих элементов и линий связи), выдает в ответ сигнал приглашения к набору номера посылкой непрерывного сигнала частотой 800 Гц. После чего на узел коммутации передается адрес (*номер*) вызываемого абонента.

По полученному адресу аппаратура коммутации узла производит соединение одного или нескольких участков сети в единый коммутируемый канал и извещает посылкой сигнала вызова удаленного абонента (периодические посылки частотой 425 Гц длительностью 1 с и интервалом 4 с). После подключения абонента к каналу узел коммутации предоставляет канал связи абонентам для обмена информацией. В случае отсутствия канала в нужном направлении или занятости вызываемого абонента узел коммутации извещает об отказе вызывающего абонента посылкой коротких периодических сигналов частотой 425 Гц длительностью 0,3…0,4 с.

Схема узла коммутации каналов показана на рисунке 1.7. Окончания цепей каналов связи, подходящих к узлу коммутации, через устройства сопряжения с каналами (УСК) подключаются к кроссовому устройству (сокращенно – *кросс*).

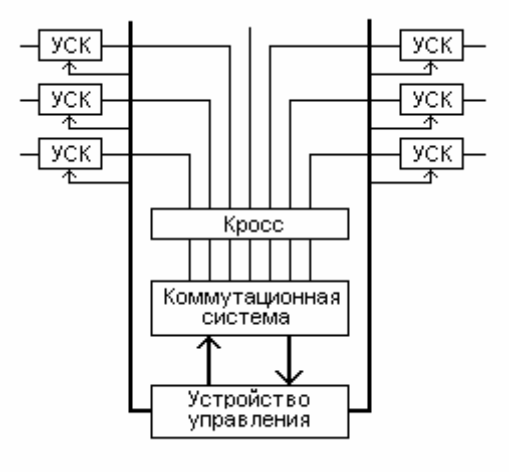


Рисунок 1.7 – Схема узла коммутации каналов

Кросс конструктивно представляет собой панель (либо щит с несколькими панелями или целые стойки с группой панелей) с контактами. На этих контактах с помощью пайки, подключения "под винт" или специальными перемычками (дужками, шнурами со штепселями и т.п.) осуществляется долговременное соединение (*кроссирование*) между отдельными частями канала, блоками каналообразующей аппаратуры либо каналов и каналообразующей аппаратуры с коммутационной системой. До недавнего времени кроссирование осуществлялось только вручную. Сейчас появились устройства автоматического кроссирования с дистанционным управлением.

В коммутационной системе осуществляется кратковременное соединение отдельных участков каналов между собой для организации сквозного тракта передачи сигналов. Коммутационная система строится на базе устройств переключения с механическими контактами (электромагнитные реле, декадно-шаговые искатели, многократные координатные соединители, герметизированные контакты – герконы) либо на основе бесконтактных ключей (КМОП транзисторы). В настоящее время коммутационные системы выпускаются только на основе герконов ("квазиэлектронные" коммутационные станции) или бесконтактных ключей (электронные станции). Управление замыканием и размыканием отдельных цепей и всем узлом коммутации в целом осуществляется устройством управления, функции которого выполняет специализированная ЭВМ.

Устройства переключения коммутационной системы конструктивно объединены в унифицированные модули – блоки переключателей имеющие *M* входов и *N* выходов. Унифицированный модуль сокращенно называют коммутатором. Такое решение позволяет легко наращивать емкость узла коммутации без существенного изменения его схемы.

Коммутатор размера *M*×*N* представляет собой матрицу, в которой *M* канальных входов подключены к горизонтальным шинам, а *N* выходов –к вертикальным (рисунок 1.8,а). В узлах матрицы располагаются коммутирующие ключи (рисунок 1.8,б), которые под действием управляющих сигналов соединяют горизонтальную и вертикальную шины. Причем, в каждом столбце матрицы разрешается замыкать только один ключевой элемент. Точка соединения линий матрицы называется **точкой коммутации**. Если M<N , то коммутатор может обеспечить соединение каждого входа с не менее чем одним выходом; в противном случае коммутатор называется *блокирующим*, т.е. не обеспечивающим соединения любого входа с одним из выходов. В реальных системах обычно применяются коммутаторы с равным числом входов и выходов, т.е. матрицы размером *N*×*N*. На рисунке 1.8,в показано условное обозначение коммутатора размером *M*×*N*.

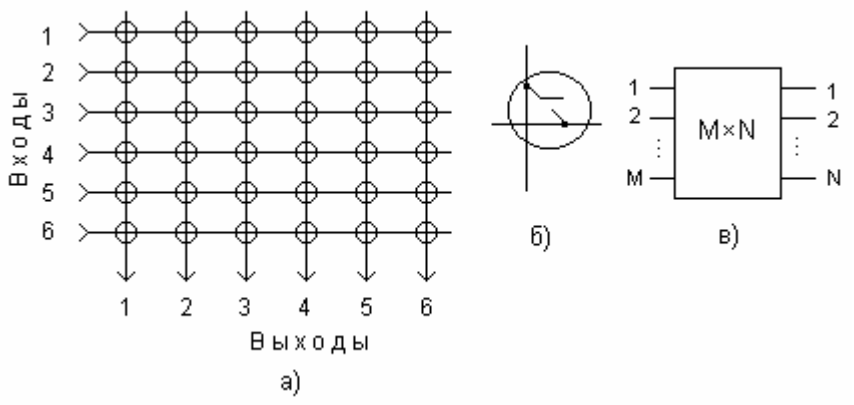


Рисунок 1.8 – Матрица пространственного коммутатора

Основной недостаток такого типа коммутаторов – квадратичный рост сложности схемы, т.е. число коммутирующих элементов для квадратной коммутационной матрицы равно N^2. Для устранения этого недостатка применяют многоступенчатые коммутаторы. Принцип построения многоступенчатого коммутатора заключается в том, что матричный коммутатор разбивают на части, которые соединяют промежуточным дополнительным коммутатором.

Рассмотрим пример построения трехступенчатого коммутатора для коммутации *N* линий. На первой ступени используется *N/n* групп коммутаторов с числом входов *n*, размером *n*×*k*. На второй ступени применяются *k*

коммутаторов размером $(N/n)\times(N/n)$ каждый. Выходы i-го коммутатора первой ступени соединяются соответственно с i-ми входами каждого из коммутаторов второй ступени. Третья ступень содержит такое же количество коммутаторов, как и первая ступень, однако, их размерность обратная размерности первой ступени, т.е. $k \times n$. Выходы i-го коммутатора второй ступени соединяются с i-ми входами соответствующих коммутаторов третьей ступени. Схема трехкаскадного коммутатора при N=12 и n=4 и k=2 показана на рисунке 1.9.

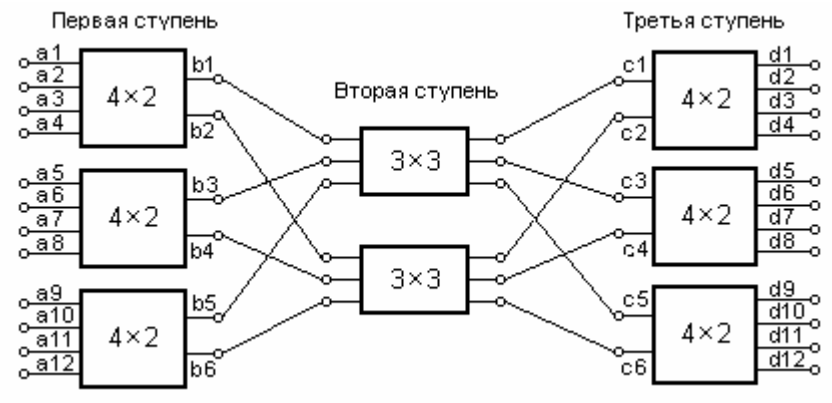


Рисунок 1.9 – Схема трехступенчатого пространственного коммутатора

Сравним сложности одно- и трехступенчатого коммутаторов. Первый каскад содержит $(N/n)\times nk = kN$ точек коммутации. Второй каскад имеет $k(N/n)^2$ точек коммутации. Третий каскад по сложности такой же, как и первый. Таким образом, необходимое количество коммутационных элементов составит $2kN+k(N/n)^2$.

Так при N =1000, n =50 и k =10 для реализации трехступенчатого коммутатора потребуется всего 24000 коммутационных элементов, вместо 1000000 для одноступенчатого. Для коммутатора, изображенного на рисунке 1.9, выигрыш составит 78 коммутирующих элемента. Очевидно, что чем больше размерность коммутатора, тем выше выигрыш.

Конечно, многоступенчатые коммутаторы нуждаются в более сложной схеме управления, так как для установления одного соединения требуется замкнуть несколько коммутирующих элементов. Другим недостатком многоступенчатых коммутаторов является возможность блокировки некоторых линий, при которой не возможно установить необходимые соединения, если требуемые коммутирующие элементы промежуточной ступени уже использованы для других соединений. В однокаскадных коммутаторах блокировка отсутствует. Для исключения блокировок в многокаскадных коммутаторах следует соблюдать соотношение k=2n-1.

В системах передачи цифровых данных при пространственной коммутации нет необходимости физически соединять цепи входящих каналов с исходящими. Достаточно, чтобы биты из входящего канала попали в соединяемый исходящий канал. Пространственный коммутатор цифровых данных строится на основе буфера памяти, размещенного в оперативном запоминающем устройстве ОЗУ (рисунок 1.10).

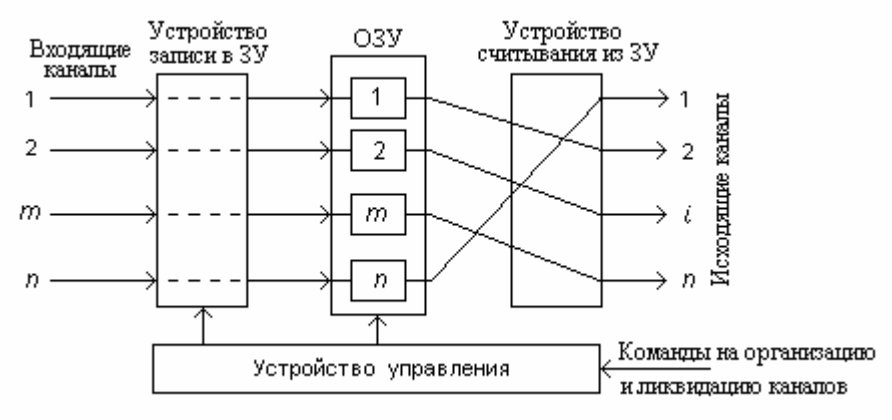


Рисунок 1.10 – Схема пространственной коммутации цифровых каналов

Запись в его ячейки осуществляется устройством записи в процессе последовательного опроса входящих каналов. Коммутация осуществляется благодаря считыванию данных из нужных ячеек памяти ОЗУ на исходящие каналы. При этом происходит задержка выходных данных на время одного цикла "запись-чтение". На рисунке 1.10 показано состояние узла, при котором произведена следующая коммутация каналов: $1 \rightarrow 2$; $2 \rightarrow i$; $m \rightarrow n$; $n \rightarrow 1$.

Такие коммутаторы являются полностью цифровыми и могут создаваться на основе больших интегральных схем (БИС). Поэтому их реализация намного проще и дешевле, чем пространственных коммутаторов для аналоговых сигналов.

1.3.2. Временная коммутация

В магистральных цифровых системах многоканальной связи передача данных между узлами коммутации происходит в форме периодической последовательности циклов (**кадров**) фиксированной длины, подразделенных на временные интервалы (*тайм-слоты*). На каждом временном интервале размещается один или несколько битов данных соответствующего канала. Задача узла коммутации состоит в том, чтобы данные из одного канала по-

пали в другой. В связи с тем, что номер канала определяется номером временной позиции, для осуществления коммутации узел должен производить перестановку битов, поступающих на определенных временных интервалах входящих кадров, на требуемые временные интервалы исходящих кадров. Правила перестановки определяются в процессе организации канала на основе анализа адреса получателя.

На рисунке 1.11 показан упрощенный случай, когда все кадры содержат по три временных интервала (тайм-слота), а узел производит коммутацию первого канала с третьим, второго с первым и третьего со вторым.

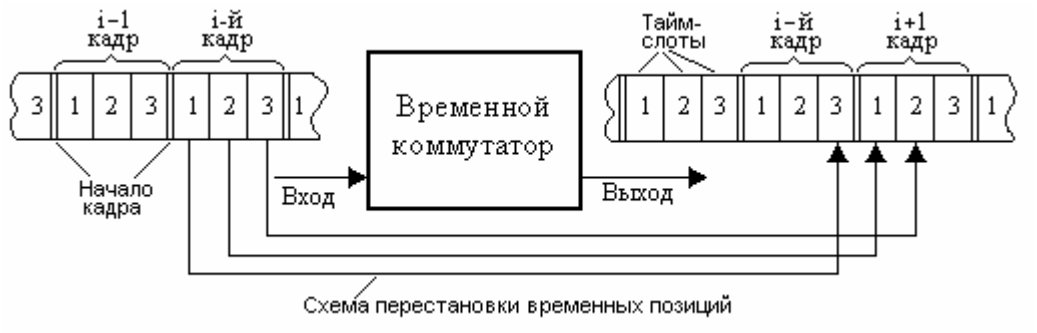


Рисунок 1.11 – Схема временной коммутации

Узел коммутации выделяет с помощью мультиплексора из i-го кадра входного потока данные, расположенные на первом временном интервале, задерживает их и выдает в выходной поток в момент наступления третьего тайм-слота i-го кадра. Если же необходимо скоммутировать входящий канал, номер которого равен или больше номера требуемого исходящего канала, то коммутатор вынужден размещать данные такого канала в последующем, (i+1)-м кадре, по той причине, что i-й интервал времени уже истек.

Временные коммутаторы могут быть реализованы с использованием ЗУ с произвольным доступом. Существуют две основные схемы построения временных коммутаторов: с последовательной записью и считыванием с произвольным доступом и записью с произвольным доступом и последовательным считыванием. Один из вариантов структурной схемы временного коммутатора показан на рисунке 1.12. Такие коммутаторы также являются полностью цифровыми и могут создаваться в виде нескольких сверхбольших БИС.

Основным ограничением применения временных коммутаторов является требуемое быстродействие коммутации. Например, если время доступа к памяти составляет 0,05 мкс и время передачи одного кадра 125 мкс (цифровая телефония), то с учетом того, что на каждый временной интервал необходимо организовать два обращения к памяти (запись и чтение), максимальное число коммутируемых каналов равно 125 / (0,05+0,05)=1250.

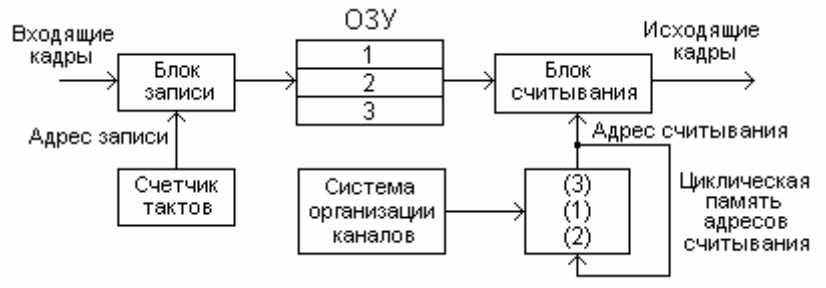


Рисунок 1.12 – Временной коммутатор с последовательной записью и считыванием с произвольным доступом

Если число коммутируемых сигналов имеет большую величину, необходимо либо увеличивать быстродействие памяти, либо переходить к более сложным схемам коммутации, в которых совмещается принцип временной и пространственной коммутации.

1.3.3. Коммутация сообщений и пакетов

В сетях с коммутацией сообщений передача данных осуществляется без установления сквозного соединения между взаимодействующими абонентами. Данные от абонента вначале передаются на ближайший узел коммутации, к которому он подсоединен, и заносятся в запоминающее устройство узла. По мере освобождения каналов в направлении передачи и наличии свободной памяти в соседнем узле коммутации, сообщение передается на следующий узел, занимая канал только на период времени передачи данных между смежными узлами. Такая процедура повторяется на каждом узле, через который проходит сообщение, до тех пор, пока сообщение не дойдет до адресата. Поэтому, даже при отсутствии свободных ресурсов, сети с коммутацией сообщений работают без отказов. Это является одним из основных преимуществ сетей с коммутацией сообщений. К преимуществу таких сетей относится также более высокая эффективность использования каналов за счет исключения повторных вызовов при отказах и более высокая надежность доставки сообщений за счет передачи данных по обходным направлениям сети при выходе из строя или перегрузки основного тракта.

Главный недостаток сетей с коммутацией сообщений – наличие задержек при доставке информации, причем задержка является случайной. Кроме того, при передаче больших сообщений повышается вероятность появления в них ошибок, что приводит к необходимости повторной передачи всего сообщения и, соответственно, к снижению эффективной скорости доставки информации.

Сети с коммутацией сообщений применяются в основном в интегральных сетях передачи данных общегосударственного масштаба. Для построения компьютерных сетей используется **принцип коммутации пакетов**, который является разновидностью коммутации сообщений. В компьютерных сетях данные содержатся чаще всего в виде файлов, которые имеют относительно большие размеры. Такие сети не могут нормально функционировать, если в них передается весь информационный блок (файл) целиком. Во-первых, информационный блок заполняет канал и связывает работу всей сети, т.е. препятствует взаимодействию остальных абонентов. Во-вторых, возникновение ошибок при передаче крупных блоков приводит к повторной передаче всего блока, что существенно снижает эффективную скорость обмена информацией.

Пакет представляет собой **короткое сообщение** длиной до нескольких тысяч байтов. Он является *самостоятельной адресуемой частью сообщения*, которая передается по сети независимо от других пакетов. Причем канал связи занимается только на время передачи пакета. При разбивке данных на пакеты сетевая операционная система добавляет к каждому пакету специальную управляющую информацию, которая обеспечивает:

- передачу исходных данных небольшими блоками;
- сборку данных в нужном порядке перед выдачей их потребителю;
- коррекцию ошибок в пакетах.

Обязательными компонентами для всех типов пакетов являются следующие:

- адрес источника, идентифицирующий компьютер-отправитель;
- адрес места назначения, идентифицирующий компьютер-получатель;
- инструкции сетевым компонентам о маршруте прохождения данных;
- информация узлу коммутации или получателю о том, как объединять передаваемые пакеты, чтобы получить данные в исходном виде;
- передаваемые данные от источника;
- избыточную информацию для защиты от ошибок.

Сеть пакетной коммутации (рисунок 1.13) состоит их нескольких узлов коммутации пакетов (УКп), к которым подключены персональные компьютеры (ПК) абонентов сети или серверы. УКп строятся на основе специализированных ЭВМ, соединенных между собой высокоскоростными аналоговыми или цифровыми линиями и каналами связи.

Процесс формирования пакета начинается на прикладном уровне эталонной модели взаимодействия открытых сетей, там, где формируются данные. Сообщение, подлежащее передачи по сети, проходит сверху вниз все семь уровней. На каждом уровне компьютера-отправителя к блоку данных добавляется информация, предназначенная для соответствующего уровня компьютера-получателя.

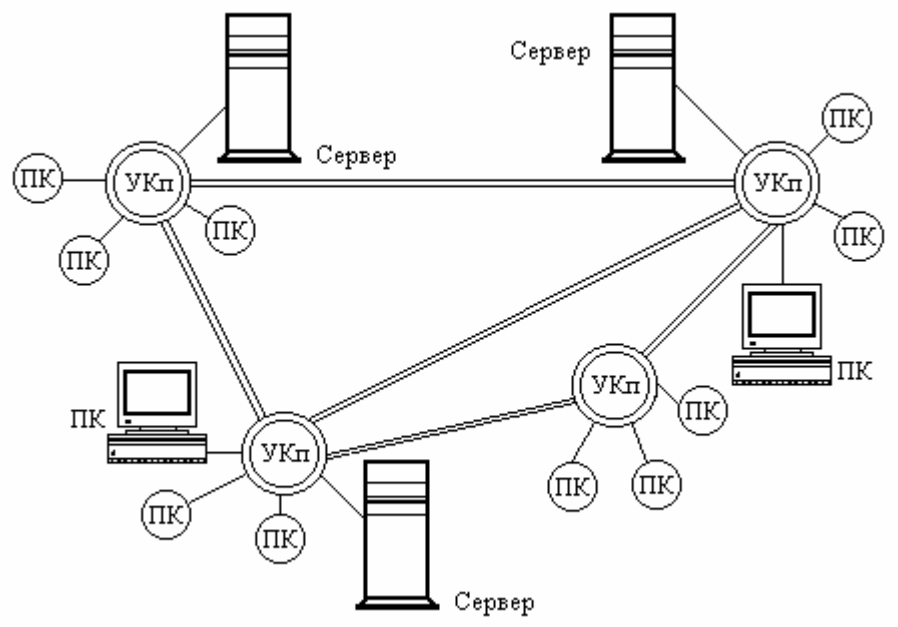


Рисунок 1.13 – Структура сети с коммутацией пакетов

В УКп реализуются три нижних уровня эталонной модели, на которых соответственно используются три типа блоков данных: *последовательность битов, кадр* и *пакет*. На верхних уровнях сообщение рассматривается как некоторый информационный блок, который на сетевом уровне упаковывается в пакет. Затем пакеты передаются на канальный уровень, где они разбиваются на блоки и к ним добавляется служебная информация, необходимая для управления передачей на канальном уровне. Блок на канальном уровне получил название "**кадр**". На физическом уровне кадр представляется последовательностью битов, выдаваемых в форме физических сигналов в дискретный канал. При приеме информации происходит обратное преобразование последовательности битов в байты, из которых формируется кадр. На канальном уровне содержимое служебных полей кадра используется для выполнения процедур канального уровня, а содержимое поля данных, являющееся сетевым пакетом, передается на сетевой уровень.

Управляющее поле пакета формирует сетевой процесс в данном узле коммутации. Затем пакет вновь преобразуется УКп в кадр, содержащий обновленные адреса и соответствующие значения управляющих полей. Сформированный таким образом кадр данных передается на физический уровень и отправляется на следующий узел коммутации или на оконечный узел, в который включен компьютер-получатель.

Существует два способа передачи пакетов по сети: **дейтаграммный** (датаграммный) и с использованием **виртуальных каналов**. *Дейтагамма* представляет собой *однопакетное* сообщение, содержащее всю служебную

информацию, позволяющую ей самостоятельно перемещаться по сети к получателю. Причем пакеты одного и того же сообщения могут передаваться по разным маршрутам, независимо друг от друга. Поэтому различные пакеты могут прибывать в пункт назначения в последовательности, отличной от той, в которой они входили в сеть. В некоторых сетях пакетной коммутации оконечные узлы производят восстановление начального порядка поступления сообщения в сеть. В других задача сборки и сортировки пакетов возлагается на компьютер-получатель. В первом случае на узлах коммутации требуется наличие более сложных протоколов для восстановления потерянных или искаженных пакетов, обнаружения пакетов-дубликатов, при этом требуется большая емкость памяти. Если поток пакетов, поступающих на УКп, превышает допустимый, т.е. пакеты не могут быть переданы на оконечный пункт вследствие отсутствия свободных каналов и памяти на УКп, то часть пакетов теряется. Это явление называют *перегрузкой сети*. В связи с этим дейтаграммный способ передачи не гарантирует стопроцентную доставку пакетов.

Виртуальное соединение (**виртуальный канал**) представляет собой логическое двухточечное соединение между абонентами отправителя и получателя. При виртуальном соединении перед передачей основной информации на узел пакетной коммутации посылается служебный пакет, запрашивающий ресурсы памяти для передачи заданного сообщения. В случае получения отказа из-за отсутствия на узле свободных ресурсов пакет с оконечного пункта не посылается и тем самым не загружается сеть. Виртуальное соединение является аналогом соединения в сети с коммутацией каналов, за исключением того, что в нем образуется логическое, а не физическое соединение. Таким образом, при передаче сообщений в виртуальных соединениях все пакеты следуют по одному и тому же заранее установленному маршруту. Для формирования необходимого маршрута и его оптимизации используются различные методы маршрутизации сообщений.

Чтобы узел смог отличать дейтаграммный способ от передачи на основе виртуальных соединений в пакете, принадлежащей сети на основе виртуальных соединений, располагается идентификатор виртуального канала. Управление потоком в сети ведется как узлами коммутации, так и оконечным оборудованием данных. Первые проверяют загруженность сети, а вторые собственно управляют потоком данных. Для контроля над перегрузкой сети используются специальные биты уведомления о перегрузке.

1.4. Адресация и маршрутизация потоков в сетях

1.4.1. Виды адресации

Для доставки сообщений в компьютерной сети необходимо указывать адрес получателя. Адреса в компьютерных сетях подразделяются на **локальные, глобальные** (межсетевые) и **символьные**.

Под *локальными* понимают такой тип адреса, который используется для доставки данных в пределах только одной сети. Так, например, если подсетью объединенной сети является локальная сеть, то локальный адрес носит название ***MAC*-адрес** (от слова *Media Access Control* – управление доступом к среде). *MAC*-адреса присваиваются производителями сетевого оборудования сетевым адаптерам и сетевым интерфейсам маршрутизаторов. В связи с этим локальные адреса иногда называют аппаратными или физическими.

Глобальные адреса предназначены для использования в объединенной сети. Адресное пространство глобальных сетей обычно подразделяется на отдельные области – **домены**. Такое решение упрощает адресацию компьютеров и позволяет использовать в различных доменах одни и те же значения адресов. Адреса действительны внутри только того адресного пространства (адресного домена), которому они принадлежат и являются однозначными.

В зависимости от количества составных частей адреса их подразделяют на **одно-** и **многоступенчатые**. *Локальные адреса* состоят из одной части, т.е. являются одноступенчатыми. Примером многоступенчатой адресации является *ISO*-адресация сетевого уровня. В такой системе адрес содержит следующие составные части:

- идентификатор первичного домена *IDP* (*Initial Domain Part*) – задает адресное пространство самой верхней ступени иерархии адресов;
- идентификатор авторизации и формата – *AFI* (*Authority and Format ID*) – указывает какая организация управляет адресами доменов (*ITU, ISO*);
- иниацилизатор домена *IDSP* (*Initial Domain Specific Part*) – задает адреса домена;
- указатель адреса поддомена *DSP* (*Domain Specific Part*).

Межсетевые IP-адреса широко используемой сети Интернет являются двухступенчатыми. Первая ступень содержит **имя сети** (подсети), а вторая – **имя компьютера** (хоста) в данной сети. Эти адреса представляют собой 32-битовые идентификаторы. Для удобства представления IP-адресов для пользователя применяется их цифровое написание, при котором адрес записывается, как десятичное представление 4-х байт (октетов), разделенных точ-

ками, например: 192.171.153.60.

Символьные адреса представляют собой последовательность некоторых символов для обозначения месторасположения устройства или абонента в сети. Для обеспечения однозначности имен абонентов различных доменов в общем пространстве адресов, дополнительно к внутридоменному адресу абонента, добавляется обозначение домена, к которому относится данный абонент. Такие имена часто используются для символического обозначения адресов и облегчают запоминание их пользователями. Например, имя *petrenko.is.sevntu* обозначает адрес нахождения пользователя, т.е. показывает, что студент Петренко учится на кафедре информационных систем Севастопольского национального технического университета.

Сетевые адреса подразделяются также на **групповые** и **широковещательные**. В случае групповой адресации (*Multicast-Address*) можно опрашивать (адресовать) несколько оконечных пунктов одновременно. Особым видом группового адреса является широковещательный адрес (*Broadcast-Address*). При указании такого адреса сообщение поступает всем компьютерам сети. На практике для задания широковещательного адреса часто используется последовательность, состоящая из всех единиц.

1.4.2. Способы маршрутизации сообщений

В процессе соединения между двумя абонентами необходимо указать маршрут в сети, т. е. определить те узлы сети, через которые будет проходить соединение. Процесс выбора маршрута прохождения сообщения называется *маршрутизацией*. Цель маршрутизации - доставка пакетов по назначению с максимальной эффективностью. Чаще всего эффективность выражена взвешенной суммой времен доставки сообщений при ограничении на вероятность доставки их получателю. Маршрутизация сводится к определению направлений движения пакетов в узлах (маршрутизаторах). Выбор одного из возможных в узле направлений зависит от текущей топологии сети, которая может меняться хотя бы из-за временного выхода некоторых узлов из строя, длин очередей в узлах коммутации, интенсивности входных потоков и т.п. Алгоритмы маршрутизации включают следующие типовые процедуры:

- измерение и оценивание параметров сети;
- принятие решения о рассылке служебной информации;
- расчет таблиц маршрутизации;
- реализация принятых маршрутных решений.

Соединения должны организовываться таким образом, чтобы, по возможности, они были наиболее короткими, но вместе с тем соблюдалась рав-

номерность загрузки каналов и узлов, т.е. при организации канала следует обходить те участки сети, которые в данный момент перегружены. Для идеальной маршрутизации необходимо учитывать всю информацию о состоянии сети, а также прогнозировать будущую загрузку на некоторый интервал времени. Реализация этого потребовала бы сбора в едином центре всей информации о сети (загрузка трактов, отказы узлов и каналов и так далее).

Сбор информации в единый центр приводит к дополнительной загрузки сети передачей служебной информацией. Объем этих данных увеличивается с ростом размерности сети, вследствие чего в больших сетях возникают задержки. Все это приводит к тому, что задача выбора маршрута в большой сети становится очень сложной и для ее решения применяются специальные методы маршрутизации.

Классификацию видов маршрутизации проводят по различным признакам, в частности по степени централизации:

- **распределенная**, при которой каждый узел сети самостоятельно принимает решение о выборе маршрута;
- **централизованная**, если маршрут определяется центром управления сети и сообщается всем узлам, которые находятся на данном маршруте;
- **смешанная**, когда маршрут вычисляется в узлах коммутации на основе рекомендаций центра управления.

По используемой информации для выбора пути различают следующую маршрутизацию:

- без учета информации о сети, являющуюся простейшим способом и осуществляемую по всем направлениям, либо по случайно выбранному пути;
- с учетом локальной информации, при которой применяется только та информация, которая имеется на узле;
- с учетом глобальной информации, в случае которой принимается во внимание состояния соседних узлов сети.

В компьютерных сетях применяются различные способы маршрутизации: волновой, с фиксированными и альтернативными путями, альтернативный.

Волновая (лавинная) маршрутизация (*flooding*). При этом способе осуществляется децентрализованная маршрутизация без учета какой-либо информации о сети. Суть его состоит в следующем. Поступивший в узел пакет передается по всем выходным направлениям (*широковещательная передача*), за исключением узла, с которого поступил пакет. Если в узел поступает пакет, который уже проходил по нему, то этот пакет стирается. Чтобы размножающиеся пакеты не перегружали сеть, удаляются некоторые пакеты, тайм-аут которых истек, либо прошедших через количество узлов, превышающее некоторое заданное число. Существует усовершенствованная

версия широковещательной маршрутизации, называемая селективной широковещательной рассылкой. По этому способу рассылка производится не по всем возможным направлениям, а только по тем, которые предположительно ведут в правильную сторону.

Основным достоинством способа является высокая надежность (использование в специальных сетях). Недостаток – сильная загрузка сети, усложнение узлов. Разработаны модификации способа, при которых ограничивается зона распространения "волны" определенной областью сети. Способ волновой маршрутизации достаточно широко используется в локальных сетях при передаче широковещательных сообщений.

Маршрутизация с фиксированными путями подразделяется на одномаршрутную и на маршрутизацию с альтернативными путями. Последняя относится к *распределенному* способу маршрутизации, при котором используется только информация о *топологии* сети. Текущее состояние сети не учитывается. Соединение между узлами при одномаршрутном способе осуществляется всегда по одному и тому же пути. Любые изменения в маршрутные таблицы вносит только администратор сети. Основное достоинство способа – его простота, однако, при отказах отдельных направлений часть соединений при этом способе вообще не возможна. К недостатку такой маршрутизации следует отнести также игнорирование фактической загрузкой трактов. Поэтому способ с фиксированными путями используется преимущественно на сетях со стабильной нагрузкой.

Маршрутизация с альтернативными путями относится к *распределенному* способу маршрутизации *с учетом локальной* информации о топологии сети. Для каждого адресата существует несколько путей. При работе сети для всех возможных маршрутов *вычисляется* определенная доля *трафика* (информационного потока), которую следует по нему передавать. При этом способе учитывается только топология сети, но не используются сведения о состоянии сети или данного узла. Администратор сам решает, по какому пути (путям) следует передавать пакеты.

Адаптивная маршрутизация учитывает динамическое состояние узла или сети, т.е. маршрут адаптируется к состоянию сети. Способы адаптивной маршрутизации разделяют на *локальные* (учет только состояния собственного узла) и *глобальные*, в которых учитывается состояние всей сети или большей ее части. Протоколы, построенные на основе адаптивной маршрутизации, позволяют всем узлам собирать информацию о топологии сети и оперативно обрабатывать все изменения ее конфигурации.

Для определения необходимого маршрута на узлах размещаются *адресные таблицы*, содержащие полный набор идентификаторов (*адресов*), опознаваемых на данном узле коммутации. В адресной части каждого пакета имеется идентификатор получателя или *набор адресов узлов*, через кото-

рые должно проходить сообщение. Специальные программы маршрутизации анализируют адресные части пакетов и на основании таблиц и дополнительной информации о состоянии сети преобразуют их в направления передачи. При этом на многих узлах осуществляется определение *оптимального* маршрута. Критерием оптимизации чаще всего выступает время доставки пакетов в сети.

В качестве математического аппарата оптимизации используется теория графов, потоков и сетей. Оптимизация маршрута сводится к выбору кратчайшего пути в графе с минимальной очередью пакетов.

1.4.3. Алгоритмы маршрутизации в компьютерных сетях

В отличие от классических телефонных сетей и интегральных цифровых сетей передачи данных, компьютерные сети имеют специфические особенности транспортировки сообщений, в частности следующие:

- многие соединения проходят через очень большое число узлов коммутации;
- линии и узлы обладают недостаточно высокой надежностью;
- характер передаваемых сообщений в течение нескольких минут может существенно измениться;
- малое количество альтернативных путей передачи.

При рассмотрении наиболее широко используемых алгоритмов маршрутизации будем исходить из того, что каждому из узлов сети известны адреса соседних узлов, а для всех линий связи, соединяющих соседние узлы, определена их метрика. В качестве **метрики** связей часто используется "**стоимость**" или "**расстояние**". Обобщенное понятие стоимости учитывает как фактическую стоимость использования линии, так и ряд других параметров (расстояние, пропускную способность, задержку передачи и т.д.).

Рассматриваемые алгоритмы относятся к распределенным, в которых узлам удается так выстроить глобальную маршрутную информацию, что определяется маршрут с минимальной стоимостью, хотя каждый узел обменивается сообщениями только со своими соседями. Для этого в общем требуется большое число шагов (*хопов*), пока сеть не установится в стабильное состояние.

При выполнении алгоритма маршрутизации узел должен получать информацию от соседних узлов, выполняющих такой же алгоритм маршрутизации, о сетях, которые могут быть достижимы при передаче данных через каждый соседний узел. Накапливая полученную информацию, каждый узел может определить направление – маршрут передачи данных для каждой из достижимых сетей. В случае, если таких маршрутов оказалось не-

сколько, алгоритм маршрутизации может предусматривать использование специального критерия для выбора лучшего из них.

В зависимости от способа, используемого для обеспечения обмена информацией о маршрутах в сети между узлами при выполнении алгоритма маршрутизации, различают два типа протоколов маршрутизации:

- **дистанционно-векторные** (*distant vector*);
- **оценки состояния линий** (*link state*).

Дистанционно-векторные протоколы передают информацию о маршрутах периодически через установленные интервалы времени.

Протоколы оценки состояния линий предусматривают передачу информации о маршрутах в момент первоначального включения или при возникновении изменений в существующей структуре информационных связей.

Алгоритм Беллмана-Форда относится к дистанционно-векторным алгоритмам **DVA** *(Distance Vector Algorithms*). Исходным положением в этом алгоритме является то, что каждому узлу известно расположение и возможности узлов сети, но неизвестны кратчайшие пути к ним. Под кратчайшим путем подразумевается путь с минимальной стоимостью. На каждом узле имеется вектор расстояний, представляющий собой список с записями вида: "*Получатель; Стоимость*". *Стоимость* обозначает при этом текущее значение суммы стоимостей доставки сообщения по кратчайшему пути к соответствующему получателю. В качестве начального значения каждый узел устанавливает такие стоимости доставки сообщений к несмежным узлам, которые заведомо выше самых высоких ожидаемых затрат (устанавливается бесконечно большое значение).

Через установленное время узлы сети периодически рассылают служебные пакеты *update*, содержащие текущие значения векторов стоимостей, всем своим непосредственным соседям. На основании этой информации каждый узел сети определяет для любого возможного получателя пути с минимальными затратами. Это производится суммированием стоимостей доставки сообщений соседям с соответствующими стоимостями доставки от соседа к получателю, которые сообщил его соседний узел.

Для формального описания алгоритма Беллмана-Форда введем вначале следующие обозначения: s – исходный узел, от которого будет находиться оптимальный путь до некоторого узла k; $w(i,j)$ – стоимость пути от узла i до узла j, которая при начальной инициализации устанавливается $w(i,i)=0$, а $w(i,j)=\infty$, если эти узлы не являются смежными и $w(i,j) \geq 0$, если узлы i и j непосредственно соединены между собой; h – максимальное количество участков (*хопов*) в пути на текущем этапе алгоритма; $L(k)$ – стоимость оптимального пути от узла s до узла k, содержащего не более h участков.

Алгоритм сводится к следующим действиям.

1. Начальная инициализация

$$L_0(k) = \infty \text{ для всех } k \neq s;\ L_h(s)=0 \text{ для всех } h=0;$$

2. Модификация

Для всех последующих $h \geq 0$; для каждого $k \neq s$ вычисляем

$$L_{h+1}(k) = \min_j [L_h(j) + w(i,k)] .$$

Соединяем k узел с предшествующим узлом j, который дает минимальную стоимость, и удаляем соединение k с другим предшествующим узлом, созданное на предыдущей итерации. Путь от s до k заканчивается участком линии от j до k.

На рисунке 1.14 показан пример, иллюстрирующий действие алгоритма для сети с топологией, изображенной на рисунке 1.14,а.

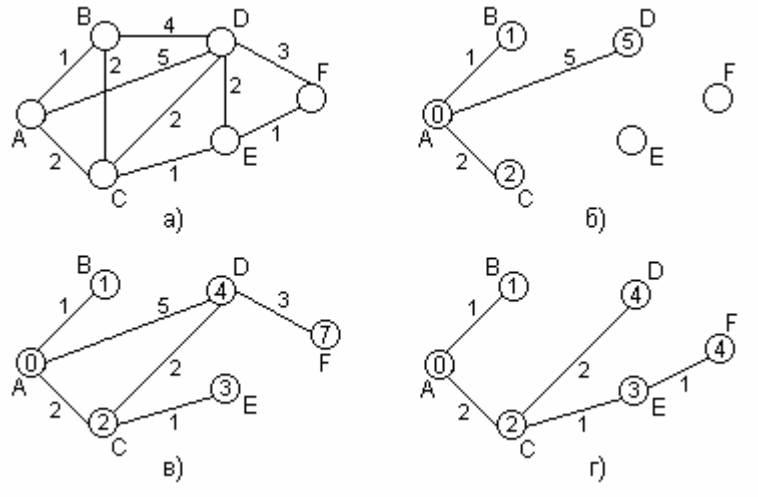


Рисунок 1.14 – Иллюстрация построения маршрутов минимальной стоимости на основании алгоритма Беллмана-Форда

Сеть моделируется графом, в котором узлы соответствуют маршрутизаторам, а ребра – линиям связи. Веса ребер – оценки стоимостей доставки сообщений между узлами. Построение оптимальных маршрутов от узла A до остальных узлов сети для трех этапов (h=3) изображены соответственно на рисунке 1.14,б-г.

Здесь цифрами на ребрах графа отмечены стоимости линий связи $w(i,j)$ между узлами, а цифрами внутри кружков – стоимости $L_h(s)$ соответствующих путей.

Использование алгоритма в представленной упрощенной форме (без учета состояния линий) может привести к проблемам в сети, в случае если произойдет нарушение (отказ) одного из соединений, или в сеть введены новые каналы связи. Одним из недостатков, присущих алгоритму Беллмана-

Форда, является возможность возникновения *петель* (шлейфов) маршрутизации, в результате чего один и тот же пакет циркулирует определенное время между двумя или несколькими узлами. Вторым недостатком алгоритма является несоответствие таблиц маршрутизации узлов мгновенному состоянию сети (большое время переходного процесса).

Для иллюстрации процесса образования петель рассмотрим фрагмент сети, состоящий из четырех узлов-маршрутизаторов A, B, C и D (рисунок 1.15), соединенный с сетью N с помощью маршрутизатора A. Пусть, как показано на рисунке, все связи между узлами имеют метрику (стоимость)1.

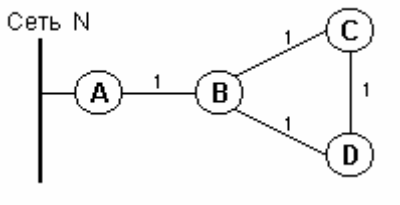


Рисунок 1.15 – Иллюстрация процесса зацикливания маршрутов

В исходном состоянии, когда все каналы передачи данных функционируют нормально, маршруты из узлов *D* и *C* к сети *N* проходят через маршрутизатор B и имеют метрику 2. Если в некоторый момент времени канал, связывающий маршрутизаторы A и B, выходит из строя, то маршрутизатор *B* в этом случае перестает принимать от маршрутизатора A модифицирующую информацию о состоянии сети *N*.

По истечении установленного интервала времени маршрутизатор B определяет сеть *N* в качестве недостижимой и исключает её из своих массивов векторов состояний. Однако из-за того, что эти массивы передаются в сети асинхронно, вполне возможно, что вскоре после этого маршрутизатор C до поступления информации от B о недостижимости узла A, получит сначала модифицирующий массив от маршрутизатора D. Последний пока ещё считает, что маршрут из B до сети *N* существует. Получив такую информацию, маршрутизатор C включит в свою таблицу маршрутизации новый маршрут до сети *N* через маршрутизатор D с метрикой 3 (C-D-B-A) и сообщает о нем соседним узлам B и D.

После того, как истечет время существования исходного пути в маршрутизаторе D о достижимости узла A, эта ситуация повторится совершенно аналогичным образом. В результате маршрутизатор D скорректирует свою таблицу маршрутизации и внесет в неё путь до сети *N* через шлюз C с метрикой 4 (D-C-D-B-A). Подобная ситуация будет таким образом циклически возобновляться снова и снова. Обратите внимание на то, что процесс зацикливания маршрутов возможен только за счет наличия разницы во вре-

мени прихода информации о состоянии сети. Для борьбы с зацикливанием ограничивают количество допустимых циклов (на практике 15…20).

В процессе маршрутизации возможно возникновение ситуации, когда периодическое обновление может быть потеряно в сети из-за возникновения краткосрочной перегрузки или временной неработоспособности канала передачи данных. Для того, чтобы в этой ситуации маршруты не были ошибочно удалены из таблицы, каждому маршруту ставится в соответствие специальный счетчик времени, который называется *timeout – timer*. В тот момент времени, когда данный путь включается в таблицу маршрутизации, или когда для него приходит очередное обновление значение счетчика *timeout – timer,* устанавливается равным $T_{to\ max}$ и этот счетчик начинает обратный отсчет времени. В том случае, если счетчик *timeout – timer* какого либо маршрута достигнет значения 0, этот путь должен быть исключен из числа активных маршрутов. В реальных сетях $T_{to\ max}$ инициируют на 180 с.

Маршрутизация с учетом состояния линий. Недостатки, присущие дистанционно-векторной маршрутизации, вызваны тем, что узлы имеют слишком мало информации о топологии всей сети. При учете состояния линий связи (*Link State Routing* - ***LSR*)** узлы сети располагают информацией о топологии всей сети и о стоимости связей между ними. Все узлы сети используют один и тот же алгоритм для определения первого кратчайшего пути **SPF** (*Shortest Path First*).

В начале функционирования сети каждый узел формирует группу пакетов состояния линий ***LSA*** (*Link State Advertisements*). *LSA*-пакет содержит идентификаторы собственного и соседнего узлов, а также стоимость связей между ними. На следующем шаге пакеты состояния линий рассылаются широковещательно *всем* другим узлам сети. Узлы, получившие *LSA*-пакеты от всех маршрутизаторов сети, параллельно друг с другом создают топологическую базу данных, содержащую все *LSA*-сообщения. На основе полученной информации узлы рассчитывают пути с минимальными стоимостями. При этом маршрутизатор рассчитывает топологию кратчайших путей в виде SPF-дерева, помещая себя в корень. Всякий раз, когда *LSA*-пакет вызывает изменение в базе данных состояния каналов, алгоритм учета состояний линий пересчитывает пути и обновляет таблицу маршрутизации.

Алгоритм SPF гарантирует правильное функционирование сети при нарушениях связи или выходе из строя отдельных маршрутизаторов и исключает возможность двукратной передачи пакета по одному и тому же пути. К алгоритмам, использующим знания о топологии всей сети и учитывающим стоимости связей между всеми ее узлами, относится алгоритм **Дийкстры** (*Dijkstra's algorithm*). С помощью этого алгоритма находятся кратчайшие маршруты от данного узла-источника до всех остальных узлов сети.

На основании алгоритма Дийкстры определяются оптимальные пути от заданного узла-источника до всех остальных узлов сети в процессе перебора маршрутов в порядке увеличения их длин. Построение путей происходит поэтапно. На *l*-м шаге находятся *l* путей с минимальной стоимостью к *l* узлам.

Для формального определения алгоритма введем следующие обозначения: N_y – множество узлов сети; *s*– узел-источник; T_y– множество узлов, уже обработанных алгоритмом; *w(i,j)* – стоимость пути от узла *i* до узла *j,* которая при начальной инициализации устанавливается *w(i,i)=0* или $w(i,j)=\infty$, если эти узлы не являются смежными и $w(i,j) \geq 0$, если узлы *i* и *j* непосредственно соединены между собой; *L(k)* – стоимость оптимального пути от узла *s* до узла *k*.

Алгоритм содержит 3 шага. Шаги 2 и 3 циклически повторятся до тех пор, пока для всех вершин сети не будут определены оптимальные пути, т.е. пока множество T_y на совпадет с множеством N_y.

1. Произвести начальную инициализацию.

$T_y = \{s\}$, это означает, что в множество исследованных узлов включен только исходный узел-источник;

$L(k) = w(s,k)$ для $k \neq s$, т.е. стоимости начальных путей к соседним узлам состоят пока что только из узла-источника.

2. Найти следующий узел.

Осуществляется поиск следующего узла *x* графа сети, не входящего в множество T_y и имеющего путь с минимальной стоимостью от этого узла до узла-источника *s*. Производится включение найденного узла в множество T_y и соединение его с узлом множества T_y, т.е. выполняется операция

$$L(x) = \min_{j \notin T_y} L(j).$$

3. Модифицировать путь с минимальной стоимостью. Обновление пути осуществляется на основании вычислений по следующей формуле:

$$L(k) = \min [L(k), L(x) + w(x,k)] \quad \text{для всех} \quad k \notin T_y .$$

Шаги 2–3 повторяются до тех пор, пока все узлы сети не будут включены в множество T_y.

На каждой итерации к множеству T_y добавляется новый узел *x*, а значение *L(x)* на этот момент времени представляет собой минимальную стоимость пути от узла-источника *s* до узла *x*. Путь с наименьшей стоимостью должен проходить только через узлы, уже входящие в множество T_y. Следует заметить, что на первой итерации узел *x* должен быть связан с источником напрямую. Второй, добавляемый в множество T_y узел, должен быть свя-

зан напрямую либо с узлом-источником, либо с узлом, уже добавленным к множеству T_y. На последующих шагах значение путей с наименьшей стоимостью обновляется только для узлов, еще не включенных в множество T_y.

На рисунке 1.15 приведена иллюстрация этапов процесса построения кратчайших путей от узла A к остальным узлам сети, изображенной на рисунке 1.15,а.

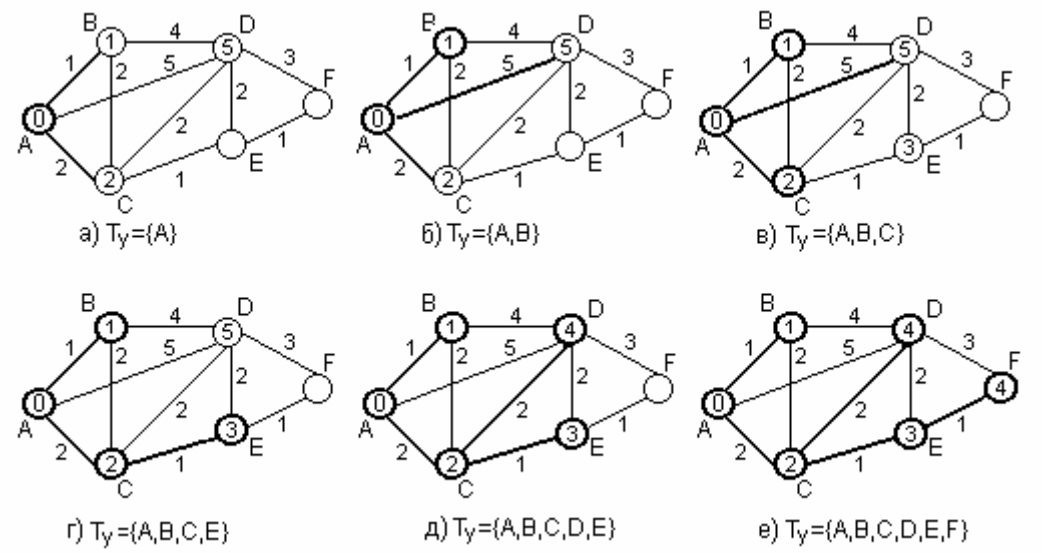


Рисунок 1.15 – Иллюстрация работы алгоритма Дийкстры

В настоящее время наиболее широко используемые протоколы маршрутизации в сети Internet – это **RIP** (*Routing Information Protocol*) и его усовершенствованная версия RIP 2, а также **OSPF** (*Open Shortest Path First*). Протокол RIP основан на *алгоритме Беллмана-Форда* и применяется преимущественно на нижних уровнях иерархии. Хотя алгоритм Беллмана-Форда сходится медленно, однако для сетей сравнительно небольших масштабов он вполне приемлем.

В больших сетях лучше себя зарекомендовал алгоритм OSPF, основанный на использовании в каждом маршрутизаторе информации о состоянии всей сети. OSPF – алгоритм динамической маршрутизации, в котором информация о любом изменении в сети рассылается лавинообразно. При этом каждый маршрутизатор считает себя исходной точкой и строит оптимальный путь до всех известных ему сетей. В основе OSPF лежит алгоритм Дийкстры – поиска кратчайшего пути в графах.

1.5. Управление потоками и сигнализация в коммуникационных сетях

1.5.1. Перегрузки в сетях

Цель процедуры управления потоками в сети – *исключение* явления *перегрузки* сети. Под перегрузкой понимается снижение производительности сети при чрезмерном повышении интенсивности входного потока. Наиболее серьезно сказываются последствия перегрузки для сетей, работающих в дейтаграммном режиме. Возникновение этого явления связано либо с недостаточной пропускной способностью узла или сети в целом, либо с неэффективным распределением сетевых ресурсов (каналов, памяти, оборудования) между отдельными абонентскими потоками.

На сетевом узле имеется множество портов ввода-вывода для подключения оконечных пользователей и несколько портов, связывающих его с другими узлами сети. Для каждого порта выделяется по два буфера памяти, один для приема поступающих пакетов, а другой – для хранения пакетов, ожидающих отправки (рисунок 1.17). Входные и выходные буферы имеют ограниченную емкость.

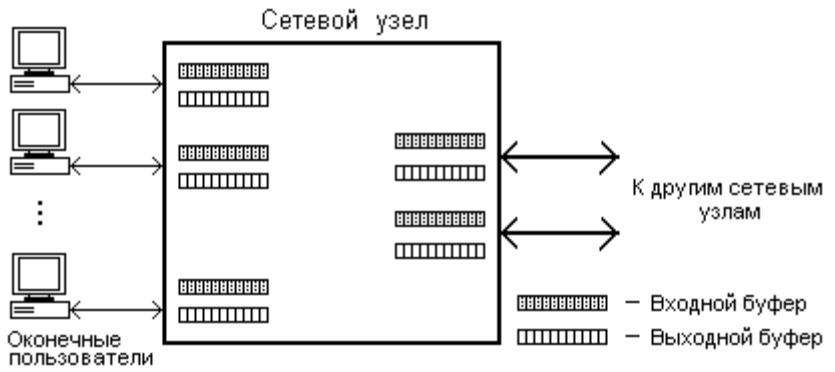


Рисунок 1.17 – Схема связей сетевого узла

Пакеты, ожидающие отправки, образуют очередь. Основная *причина* возникновения перегрузок в сети – это превышение скорости поступления информации над скоростью ее вывода, что при ограниченной емкости запоминающих устройств приводит к переполнению буферов хранения в узлах связи и при определенных условиях – к **блокировке** (*отказу функционирования*) некоторых узлов или всей сети в целом. По причине перегрузки сети происходит существенное ухудшение основных ее характеристик – времени передачи и общей пропускной способности, а в наиболее неблагоприятном случае – к отказу узла или сети. Возникновение перегрузки объясняется

следующим образом. В нормальных условиях работы сети время передачи пакета через сеть определяется в основном временем передачи по каналу связи между источником и адресатом. Время ожидания в очередях на обработку и передачу в узлах незначительно по сравнению со временем передачи по каналам. Однако по мере роста числа пакетов, циркулирующих в сети, при возникновении перегрузки *время ожидания в очередях возрастает*, и это приводит к увеличению задержки и росту числа пакетов, находящихся в сети.

В некоторых узлах может возникнуть *переполнение буферных устройств*, а это обуславливает многократное повторение передачи одних и тех же пакетов к перегруженному узлу, который не в состоянии их принять из-за переполнения памяти. В результате может возникнуть блокировка отдельных групп узлов или всей сети в целом, которая в определенных условиях не снимается сама по себе при уменьшении входного потока. Причиной блокировки является полная загрузка буферных устройств, например, пары узлов, которые имеют информацию один для другого. Каждый из этих узлов не может передать хотя бы один пакет другому узлу, так как последний не принимает их из-за отсутствия места в буферной памяти. Чтобы разблокировать такую систему, необходимо, например, стереть хотя бы один пакет в памяти одного из узлов, чтобы появилась возможность начать взаимную передачу.

Другой причиной блокировки может *служить перегрузка памяти выходных узлов* связи, используемой для сборки пакетов. Необходимость сборки вызвана тем, что в сетях с коммутацией пакетов длинные сообщения во входных узлах разбиваются на короткие пакеты, которые перемещаются по сети в направлении пункта назначения независимо друг от друга. На оконечном узле из последовательности пакетов собирается исходное сообщение, выдаваемое получателю в целом виде. Для сборки каждого сообщения в выходном узле резервируется определенный объем памяти, освобождаемый лишь после получения всех необходимых пакетов и выдачи собранного сообщения получателю. Так как количество сообщений, которые могут одновременно собираться в узле, ограничено, то не исключается ситуация, когда узел оказывается переполнен пакетами. Возникшая блокировка узлов получила название **"***блокировка сборки"*.

На основании рассмотренных ситуаций можно сделать вывод, что при отсутствии эффективных мер ограничения нагрузки происходит уменьшение производительности всей сети в целом или ее отдельных участков. Перегрузка – это ключевая проблема, которую необходимо решать на этапе проектирования компьютерных сетей с коммутацией пакетов, ретрансляцией кадров, а также объединенных сетей.

В сетях с *виртуальным соединением* проблема контроля входной нагрузки является не такой острой, так как для отдельного потока резервируется не только маршрут, но и сетевые ресурсы, включая буферную память на узлах коммутации и у потребителя, а также пропускную способность канала связи.

При выявлении перегрузки узла последний обычно посылает специальное служебное уведомление источнику, сигнализирующее о перегрузке. Получив пакет с подобной индикацией, отправитель снижает интенсивность выходного потока данных. Обнаружение явления перегрузки в сети может производиться и по косвенным признакам, т.е. неявным способом. При этом используется тот факт, что при наличии перегрузок в сети увеличивается время задержки доставки пакетов, а также возрастает количество пропавших (отброшенных) пакетов. Если отправитель в состоянии обнаружить возрастание времени доставки сообщения и явление сброса пакетов, то он может самостоятельно принять решение о снижении скорости передачи. Таким образом, за борьбу с перегрузкой на основе *неявных признаков* отвечает оконечный узел, а от сетевых узлов не требуется никаких действий.

1.5.2. Методы защиты от перегрузок

Для защиты компьютерных сетей от перегрузок необходимо осуществлять контроль перегрузок как между узлами сети так и между отправителем и получателем. Контроль перегрузок *между узлами* должен обеспечивать согласование пропускных способностей узлов, чтобы в процессе передачи не возникали такие потоки пакетов, с которыми не справляются узлы. На уровне *"отправитель-получатель"* необходимо следить за тем, чтобы информация не накапливалась во входных буферах компьютера-получателя, а достаточно быстро использовалась в процессах обработки, выполняемых в прикладных программах. Такое положение возможно лишь при условии, что процесс передачи данных будет согласован с процессами обработки на передающей и принимающих ЭВМ.

Методы борьбы с перегрузкой подразделяются на *локальные* и *глобальные*. К локальным относятся способы, в которых действия каждого узла сети осуществляются с учетом информации, имеющейся на данном узле или получаемой от соседних узлов. В глобальных методах используется информация о состоянии всей сети в целом.

В **локальных методах** широко применяются *резервирование буферов*, посылка *сдерживающих пакетов*, *адаптивная маршрутизация,* а в ряде случаев также *неявная сигнализация* о перегрузке.

Использование резервирования входных буферов анализа поступающих данных уменьшает вероятность блокировки узлов, поскольку в резервных буферах могут осуществляться функции анализа поступающих сообщений при полностью загруженных буферах, используемых для передачи сообщений. Так, например, в ряде сетей в каждом узле связи *резервируются* буферы на *один пакет* для каждой *выходной* линии, и на *два* пакета для каждой *входной.* Эти буферы не применяются для организации очередей на входах и выходах узла, а служат лишь для того, чтобы в момент перегрузок, когда все буферы, задействованные для организации очередей, заняты, была бы возможность осуществить анализ входящих управляющих пакетов и передачу служебных сообщений своим соседям.

Другим способом резервирования буферных устройств является *запрет* полной загрузки буферов для организации очередей. После того, как буферные устройства заполнены до определенного уровня, узел не принимает обычных пакетов данных, однако резервная часть буферов может использоваться для передачи специальных сообщений.

Сдерживающий пакет представляет собой служебное сообщение, формируемое на перегруженном узле и передаваемое обратно узлу-источнику для ограничения интенсивности потока данных. Получив такое сообщение, источник обязан снижать скорость генерации пакетов до тех пор, пока он не перестанет получать команды сдерживания.

Адаптивная маршрутизация – это средство выбора маршрута с учетом состояния сети в данный момент времени. Выбор соседнего узла, которому передается пакет, осуществляется с учетом оценки ожидаемого времени передачи к пункту назначения. Если на наиболее коротком пути некоторые узлы перегружены, то ожидаемое время передачи может оказаться больше, чем при передаче по более длинному, но менее загруженному пути.

Анализ локальных методов контроля перегрузки показывает, что они могут облегчить условия работы сети, однако без глобальных методов устранить возможность перегрузки сети нельзя.

Глобальные методы используют централизованное наблюдение за информацией о состоянии сети. Она поступает в узел контроля, который и осуществляет управление потоками, предупреждая перегрузки. Кроме этого в глобальных методах проводится межоконечный контроль, ограничивающий количество пакетов, находящихся в сети для каждой пары оконечных пунктов.

Централизованный контроль в идеальном случае может обеспечить оптимальное управление потоками в сети, поскольку в едином центре имеется вся информация, необходимая для оптимальных решений. Однако в реальных условиях из-за огромного объема вычислений, которые должны производиться для поиска оптимальных решений, а также больших задер-

жек поступления управляющей информации по сравнению со скоростью изменения состояния сети, централизованный метод контроля на практике оказывается малоэффективным.

Наиболее действенным для борьбы с перегрузками является *межоконечный контроль*. Он может использоваться на уровне контроля потоков между начальным и конечным узлами сети или между серверами. Суть его состоит в том, что новое сообщение может начать поступать в сеть только при условии, что на оконечном пункте зарезервированы возможности для его приема, то есть организуется виртуальный канал.

Одним из методов управления потоками в дейтаграммных сетях является метод дифференцированного сброса. Суть его состоит в том, что *транзитные* пакеты в каждом узле коммутации занимают буферное пространство и обслуживаются с более высоким *приоритетом*, чем входящие пакеты от абонента узла. Это связано с тем, что сброс транзитного потока нанесет больший ущерб суммарной производительности сети, так как на эти потоки уже были затрачены ресурсы сети. Пакеты абонентов при перегрузке сбрасываются. В другой модификации этого метода приоритет транзитных потоков возрастает с числом пройденных переприемов.

1.5.3. Способы управления потоками данных в сетях

Чтобы предотвратить слишком быстрое поступление потока байтов, и не "переполнить" терминал, компьютер, узел или другое устройство применяется процедура управления потоком данных (*Flow Control*). **Управление потоком данных** позволяет останавливать поток блоков до тех пор, пока узел не будет готов к приему следующих байтов. В процессе реализации процедуры управления потоком данных получателем посылается блок остановки потока в направлении, противоположным тому потоку байтов, который надо остановить.

Различают локальное и сетевое управление. Под локальным управлением понимается процедура взаимодействия двух устройств, находящихся в непосредственной близости (например, модем и компьютер). Сетевое управление потоком предполагает процедуру взаимодействия сетевых узлов между собой или узла с компьютером.

Для управления потоком данных на локальном уровне могут использоваться два варианта протокола: аппаратный и программный. Аппаратный протокол управления потоком (*Hardware Flow Control*) использует специальный сигнал, который позволяет остановить передачу данных, если приемник не готов к их приему. Например, для передачи и приема такого сигнала в последовательном интерфейсе RS-232C предусмотрены цепи запрос

передачи **RTS** (*Request To Send*) и готовность к приему **CTS** (*Clear To Send*). Передатчик выдает очередной байт только при включенном состоянии линии CTS. Байт, который уже начал передаваться, задержать сигналом CTS невозможно, чем гарантируется целостность посылки. Аппаратный протокол обеспечивает самую быструю реакцию на состояние приемника.

Программный протокол управления потоком XON/XOFF предполагает наличие двунаправленного канала передачи данных. Принцип функционирования поясняется рисунком 1.18. Если устройство, принимающее данные, обнаруживает причины, по которым оно не может их дальше принимать, оно по обратному последовательному каналу посылает байт-символ XOFF (код символа 13h).

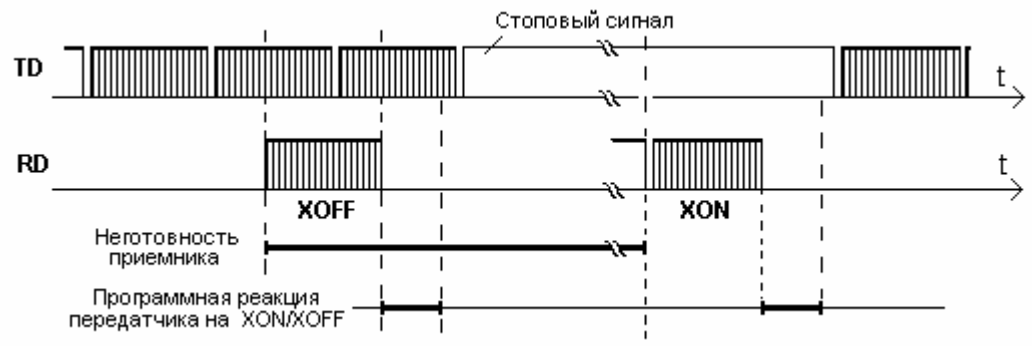


Рисунок 1.18 – Программное управление потоком XON/XOFF

Противоположное устройство, приняв этот символ, приостанавливает передачу. Далее, когда принимающее устройство снова становится готовым к приему данных, оно выдает символ XON (11h), приняв который противоположное устройство возобновляет передачу. Время реакции передатчика на изменение состояния приемника по сравнению с аппаратным протоколом увеличивается, по крайней мере, на время передачи символа (XON или XOFF) плюс время реакции программы передатчика на прием символа. Из этого следует, что данные без потерь могут приниматься только приемником, имеющим дополнительный буфер принимаемых данных и заблаговременно сигнализирующим о неготовности (пока еще в буфере имеется свободное место).

Преимущество программного протокола при непосредственном соединении устройств заключается в отсутствии необходимости передачи управляющих сигналов интерфейса. Кабель для двустороннего обмена может иметь только три провода (минимальный нуль-модемный кабель). Недостатком, кроме требования наличия буфера и большего времени реакции (снижающего и общую производительность канала из-за ожидания прохож-

дения сигнала XON), является сложность реализации полнодуплексного режима обмена. В этом случае из потока принимаемых данных должны выделяться (и обрабатываться) символы управления потоком, что ограничивает набор передаваемых символов.

На **сетевом уровне** простейшей формой управления потоком данных является *передача блоков с остановками*. При таком алгоритме источник передает один блок и останавливается, ожидая прихода подтверждения получателем приема текущего блока. После получения пакета подтверждения источник передает следующий пакет. Таким образом, станция-получатель может остановить поток данных, просто воздерживаясь от отправки подтверждения приема. Этот способ управления, например, используется в протоколах HDLC и X.25, которые применяют его в классическом виде — на каждый отправленный блок данных должно быть получено подтверждение. Преимуществом способа передачи с остановками является простота его реализации. Однако при таком способе степень использования канала очень низкая, так как во время ожидания подтверждения приема прямой канал связи не используется, т.е. эффективная скорость передачи данных снижается.

Более эффективным является способ непрерывной передачи группы блоков без ожидания подтверждения приема по каждому блоку. Такой алгоритм обмена данными получил название "**управление потоком со скользящим окном**" (*Sliding Window*).

Рассмотрим этот способ подробнее на примере обмена данными между станциями А и Б. Станция Б выделяет буферное пространство для приема *n* блоков, которые станция-источник А может отправить, не дожидаясь индивидуальных подтверждений приема по каждому блоку в отдельности. Для идентификации подтверждений о принятых блоках каждый из них помечается своим номером. Станция назначения Б посылает подтверждение о получении текущего блока, включающее в себя номер следующего ожидаемого блока. Это подтверждение косвенно извещает станцию-источник А о том, что станция-получатель Б готова принять следующие *n* блоков, начиная с указанного номера.

Для уменьшения задержек при начале передачи сообщения возможна посылка одного подтверждения на несколько блоков. Например, станция Б получив блоки 2, 3 и 4, отправляет подтверждение о приеме трех блоков 2, 3 и 4 лишь по получении блока 4. Таким образом, посылка подтверждения с номером последовательности 5 подтверждает получение блоков 2, 3 и 4 за один раз.

В процессе реализации алгоритма скользящего окна станция А поддерживает список номеров блоков в последовательности, которые ей разрешено посылать, а станция Б поддерживает аналогичный список для блоков,

которые она готова принять. Каждый из этих списков может рассматриваться как **окно блоков**. Поскольку под номер блока обычно выделяется поле фиксированного размера, то диапазон его допустимых значений ограничен. Например, если под номер блока отведено 3 бита, то номер может принимать значения от 0 до 7. Соответственно, и блоки будут нумероваться по модулю 8. Таким образом, для поля номера блока, состоящего из k битов, номера блоков располагаются в интервале значений от 0 до $(2k - 1)$, а блоки нумеруются по модулю 2^k.

В процессе передачи и приема блоки проходят ряд промежуточных состояний, изображенных на рисунке 1.19. Здесь же показаны направления, в которых движутся границы окон отправки и приема.

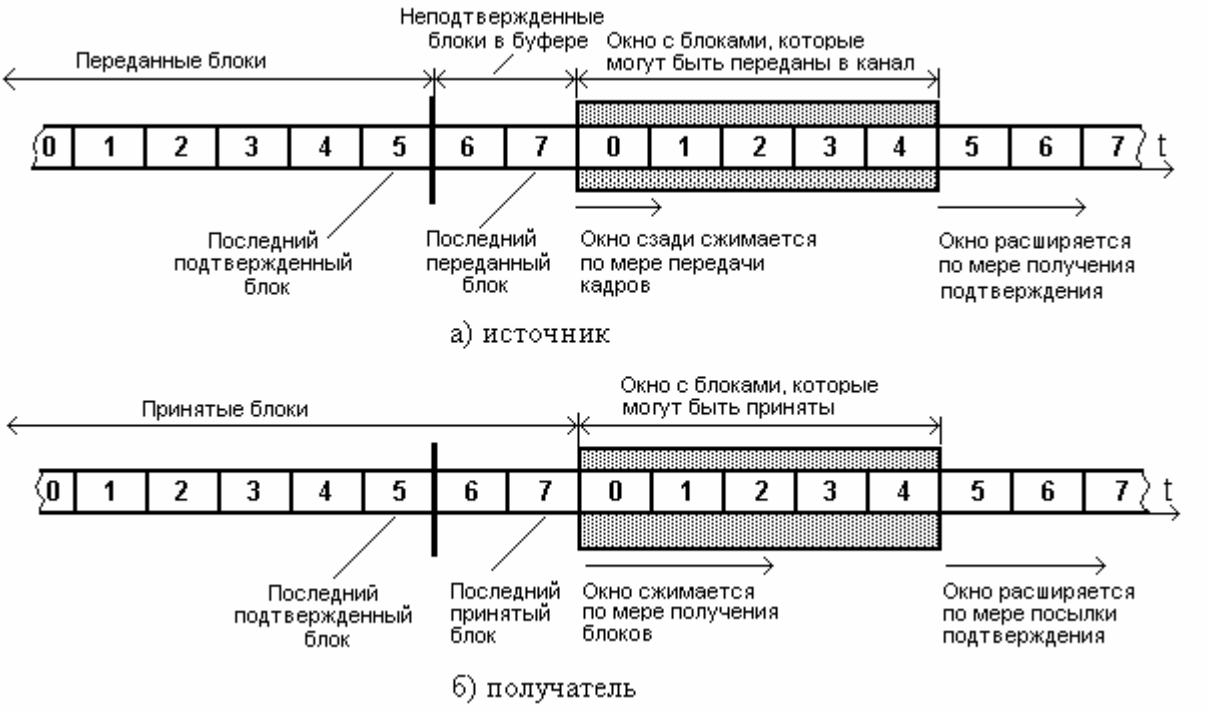


Рисунок 1.19 – Блоки в потоке данных на стороне отправителя (а) и получателя (б)

Для простоты размер поля "Номер в последовательности" принят равным 3 битам. Затененные прямоугольники указывают на блоки, которые могут быть посланы. Так, отправитель может передать пять блоков, начиная с нулевого. Каждый раз, когда блок послан, ширина затененного окна уменьшается; при очередном приеме подтверждения она увеличивается. Блоки, находящиеся между вертикальной чертой и затененным окном, уже были отправлены, но еще не подтверждены. Отправитель должен хранить копии этих блоков в своем буфере, чтобы при необходимости передать их повторно.

Рассмотрим следующий пример (рисунок 1.20), который позволяет проследить за обменом информацией. Для наглядности в нем используется трехбитное поле номера в последовательности. Максимальный размер окна в этом случае дает возможность оперировать семью блоками.

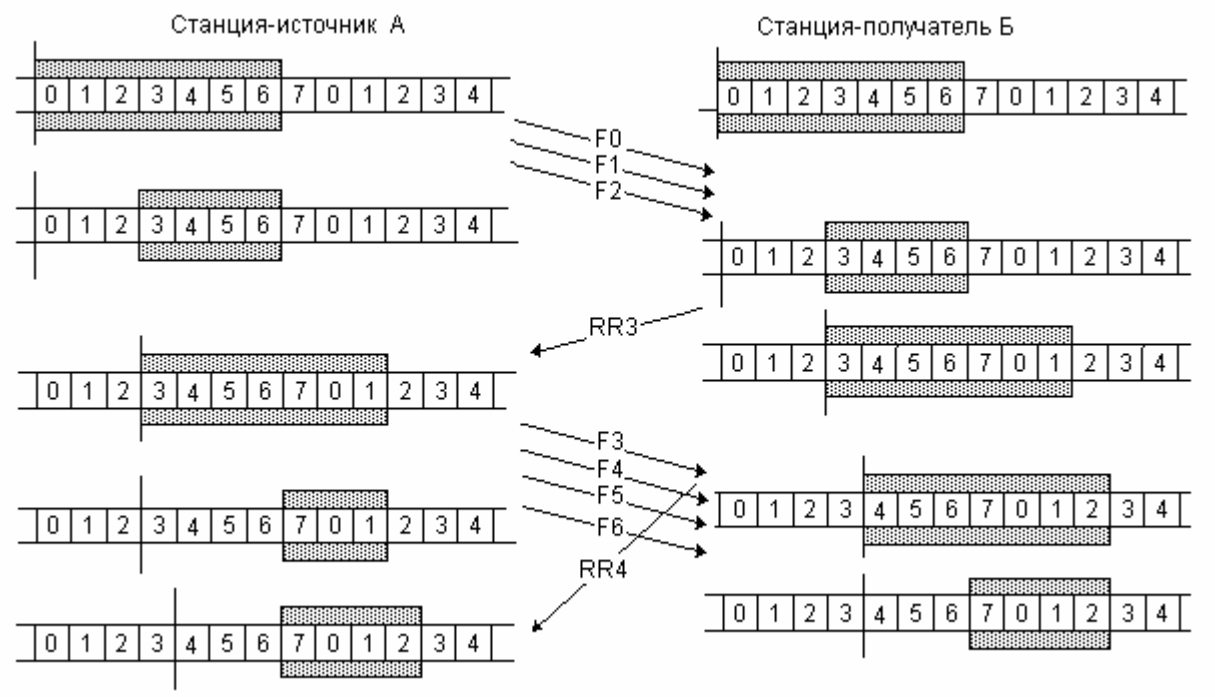


Рисунок 1.20 – Иллюстрация принципа управления потоком по способу скользящего окна

В начале работы размеры окон на станциях А и Б таковы, что станция А может передать семь блоков, начиная с F0, а станция Б — принять такое же их количество. После передачи трех блоков (F0, F1 и F2) без подтверждения станция А сокращает размер своего окна отсылки до четырех блоков и сохраняет в буферной памяти копию трех посланных блоков. Новый размер окна отсылки указывает на то, что станция А может передать четыре блока, начиная с F3.

Пусть алгоритмом установлено, что станция Б в начале работы передает подтверждение лишь после получения трех блоков, а в дальнейшей работе посылает подтверждение на каждый правильно принятый блок. На диаграмме (рисунок 1.20) видно, что получатель передает сообщение RR3 только после приема блока F3, из которого следует, что все блоки, по F2 включительно, получены и станция Б готова принять F3, а также еще шесть блоков, следующих за ним. Для станции А эта информация эквивалентна разрешению на передачу семи последующих блоков, начиная с F3.

Кроме того, станция А может очистить свою буферную память от копий первых трех блоков, как успешно принятых получателем. Теперь станция А передает F3, F4, F5 и F6, а станция Б в ответ отправляет подтверждение RR4 на получение F3 и разрешает отослать блоки от F4 до F2 (где F0…F2 относятся уже к следующей последовательности из семи блоков). Однако на момент получения станцией А этого подтверждения F4, F5 и F6 уже были посланы, следовательно, она может расширить свое окно отсылки и отправить четыре блока, начиная с F7 (правда, блоки F4…F6 пока должны оставаться в буфере).

Каждый узел постоянно отслеживает длины очередей и использование каналов связи. Если значение одной из мер превысит величину заданного порога, то генерируется специальный пакет, который посылается в адрес источника, вызвавшего перегрузку (так делается, например, в сети DECNet). Управляющие сообщения между узлами передаются через каждые 60…500 мс. Все узлы рассылают сообщения о состоянии очередей, наличии свободных направлений, зарезервированных буферах и т.п.

Для управления потоком пакетов по установленному *виртуальному* соединению широко используется техника "скользящего окна". Размер окна ограничивает число последовательно выдаваемых в сеть узлом-источником пакетов до получения от узла-получателя разрешения на выдачу очередной последовательности.

На рисунке 1.21 показана качественная зависимость производительности сети при использовании механизма ограничения нагрузки и без него. Из рисунка видно, что при превышении интенсивности потока определенной величины сеть с ограничением нагрузки обладает более высокой производительностью.

Рисунок 1.21 – Зависимость производительности сети от наличия механизма ограничения нагрузки

Способ управления передачей с использованием скользящего окна применяется также и для исправления ошибочно принятых блоков путем повторного их переспроса. Эта процедура будет рассмотрена во втором разделе учебника. Следует заметить, что управление потоком и исправление ошибок реализуется в системах передачи данных в едином механизме, регулирующем поток данных и определяющем, когда надо приостановить передачу либо повторить один или несколько ошибочно принятых блоков.

1.5.4. Качество обслуживания в сетях

В процессе обмена информацией в компьютерных сетях необходимо поддерживать требуемое качество обслуживания потребителей. Для оценки качества сервиса введен специальный показатель **QoS** (*Quality of Service*). Качество обслуживания QoS количественно оценивается рядом параметров, основными из которых являются:

- пропускная способность сети;
- задержка пакетов; степень приоритета;
- надежность;
- стоимость и др.

Часть из этих параметров имеют качественную бинарную оценку, например, "задержка" - нормальная или низкая, "пропускная способность" – нормальная или высокая, "приоритет" – обычный или повышенный и т.д. В иных случаях качественный показатель может принимать большее количество градаций, например, "приоритет" – обычный, повышенный, высокий, срочный, критический.

В зависимости от вида передаваемого трафика тот или иной параметр качества обслуживания приобретает доминирующее значение. Так в сетях передачи голосовых и видео сообщений решающую роль имеет задержка поступления пакетов, и в значительно меньшей степени – потеря части данных. Имеются приложения интерактивной графики и интерактивных вычислений, чувствительных как к задержкам, так и к потерям данных. Кроме того, разные информационные потоки обладают различными свойствами, например трафик управления сетью важнее трафика приложений, особенно в случае перегрузок или сбоев в сети.

В периоды перегрузок особенно важно, чтобы потоки данных с различными требованиями к трафику трактовались по разному и получали индивидуальное качество обслуживани QoS. Например, узел должен передавать высокоприоритетные пакеты раньше низкоприоритетных, ожидающих в той же очереди. Кроме того, узел коммутации мог бы поддерживать раз-

ную дисциплину очередей для разных показателей качества обслуживания, отдавая предпочтение более высокому показателю.

В качестве примера можно отметить, что в перспективных сетях с асинхронным режимом передачи ATM (*Asynchronous Transfer Mode*) установлено четыре уровня качества обслуживания QoS, параметром которого является скорость передачи: постоянная - CBR (*constant bit rate*), переменная - VBR (*variable bit rate*), доступная - ABR (*available bit rate*) и неопределенная - UBR (*unspecified bit rate*).

Первые два уровня качества используются обычно для передачи высокоприоритетного трафика, чувствительного к задержкам (в частности, аудио- или видеоинформации); они позволяют гарантировать определенную полосу пропускания для передаваемого трафика. ABR и UBR предназначены для менее приоритетного трафика, генерируемого, например, при объединении удаленных сегментов локальной сети. Требуемый уровень QoS определяется приложением, от которого исходит трафик. Выделение полосы пропускания в соответствии с определенной категорией QoS происходит при формировании пути от исходной точки к пункту назначения. Устройства доступа к сети ATM должны "заказывать" желаемый уровень QoS.

1.5.5. Сигнализация в коммуникационных сетях

В коммуникационных сетях производится передача не только пользовательских данных, но и большого количества служебных сообщений, обеспечивающих обмен информацией между пользователями и поддерживающих функционирование сети в заданном режиме. Формирование и передача управляющих сообщений между оконечными пользователями и узлом коммутации, либо между узлами коммутации в сетях осуществляется системой сигнализации.

Наиболее традиционными задачами сигнализации являются управление вызовом и соединением. В современных сетях системы сигнализации также обеспечивают управление производительностью сетей, передачу информации о состояниях сети и ее компонентов, о возникновении ошибок и др. В зависимости от участка сети различают следующие виды сигнализации (рисунок 1.22):

- ***абонентская*** – на участке между абонентским терминалом и коммутационной станцией (АТС);
- ***внутристанционная*** – между различными функциональными узлами и блоками внутри коммутационной станции;

- ***межстанционная*** – между различными коммутационными станциями в сети.

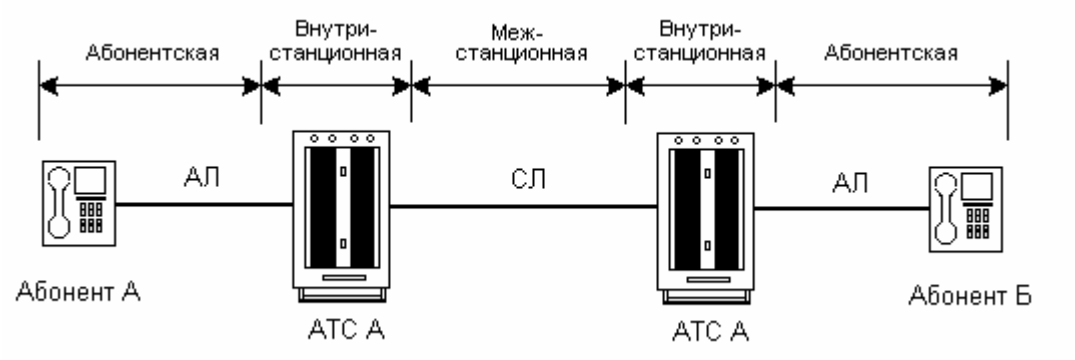


Рисунок 1.22 – Виды сигнализации в коммутируемой телефонной сети; АЛ – абонентская линия; СЛ – соединительная линия

Межстанционная сигнальная информация может передаваться различными способами, которые можно разделить на три основные класса.

1. ***Способы передачи сигналов непосредственно по информационному каналу***, называемые иногда "внутриполосными" (*in-band*), системами сигнализации. По телефонным каналам сигналы управления могут передаваться постоянным током, токами тональной частоты и др.

2. ***Сигнализация по индивидуальному выделенному сигнальному каналу (ВСК)***. В аналоговых системах передачи с частотным разделением каналов (ЧРК) передача линейных и управляющих сигналов производится методом включения/выключения передачи частоты по специально выделенному сигнальному каналу, организуемому вне разговорного спектра (*out-of-band*), как правило, на частоте 3825 Гц. Ширина полосы пропускания такого сигнального канала равна 160 Гц. Достоинством этого способа является простота передающих и приемных устройств. Для цифровых многоканальных систем для сигнализации используют один выделенный сигнальный канал, например 16-й канальный интервал в каналообразующей аппаратуре уплотнения ИКМ-30.

3. *Системы общеканальной сигнализации (ОКС)*. В системах этого класса канал передачи данных ОКС предоставляется для группы (пучка) телефонных каналов по принципу адресно-группового использования, т.е. сигналы передаются в соответствии со своими адресами и размещаются в общем буфере для использования каждым каналом когда это потребуется.

Системы сигнализации первых двух классов разработаны для применения в сетях со старыми технологиями, а системы ОКС предназначены для использования в современных и перспективных сетях, в которых и станции

и системы передачи основаны на цифровых технологиях и программном управлении.

Первая система ОКС – *система сигнализации № 6* предназначалась для передачи всех видов управляющей информации по каналам тональной частоты (ТЧ) аналоговых систем уплотнения линий на скоростях 2,4 или 4,8 кбит/с. Система обладала хорошими эксплуатационными параметрами и получила широкое распространение в мире.

По функциональному назначению сигналы, используемые в перечисленных классах сигнализации, делятся на три категории:

- ***абонентские сигналы*** – управляют каналом передачи по абонентской линии и предоставляют адресную информацию для регистрации в местной системе коммутации, а также информируют абонентов о состоянии соединения (акустические и зуммерные сигналы);
- ***линейные сигналы*** – управляют каналами передачи по каналу связи между станциями. Линейные сигналы передаются как в прямом, так и в обратном направлениях, в исходном состоянии и во время установления соединения до полного освобождения устройств. Эти сигналы отмечают основные этапы установления и завершения соединения;
- ***регистровые сигналы*** – отображают адресную информацию, используемую для маршрутизации вызовов к месту назначения (например, информация о номере вызываемого абонента, о категории и номере вызывающего абонента, сигналы категории вызова и др.).

Совокупность соответствующих сигналов и способов их передачи образуют ***абонентскую сигнализацию, линейную сигнализацию и регистровую сигнализацию***.

Адресная информация может посылаться между станциями двумя способами:

- ***"от звена к звену"***, согласно которому вся адресная информация посылается и обрабатывается на каждой станции на пути следования; например, исходящая станция А передает всю информацию на станцию Б, и ее передатчик освобождается; станция Б обрабатывает адресную информацию и посылает ее к следующей станции В и т.д.;
- ***"из конца в конец"***, когда осуществляется сквозная сигнализация. Например, станция А вызывающего абонента передает только часть информации, необходимой для маршрутизации вызова на следующей станции Б, затем часть информации передается из станции А на следующую станцию В и т.д.

Разработка и быстрое внедрение цифровых систем передачи предопределило появление ***системы сигнализации № 7*** - **SS7** (*Signaling System* 7), ориентированной на применение в цифровых сетях. Один канал SS7 со скоростью 64 кбит/с позволяет передавать сигнальную информацию для пучка,

содержащего до двух тысяч каналов ТЧ. Обладая огромным потенциалом, SS7 не только обеспечила потребности передачи сигнальной информации для существовавшего в момент ее появления уровня развития связи, но и способствовала созданию новых услуг связи.

Идея SS7 заключается в передаче всех линейных и управляющих сигналов, необходимых для функционирования множества речевых каналов, по одному каналу передачи данных. При этом все сигналы собираются в пакеты – сигнальные единицы и снабжаются заголовком, устанавливающим принадлежность каждого из сигналов определенному речевому каналу. SS7 ориентирована на использование каналов передачи данных со скоростью 64 кбит/с.

Сигнальные каналы, или так называемые звенья и пункты сигнализации (то есть коммутационные системы, использующие SS7; базы данных, принадлежащие SS7, и т. д.), образуют **сеть сигнализации**. Она строится по специфическим правилам, отличным от правил построения как телефонных, так и компьютерных сетей. По существу SS7 образует сеть передачи данных – сеть сигнализации, при этом все сигналы собираются в пакеты и снабжаются заголовком, устанавливающим принадлежность каждого из сигналов определенному каналу ТЧ.

Сети сигнализации функционируют в *связанном* и *квазисвязанном* режимах. В связанном режиме маршруты сигнальных каналов и канала сигнализации совпадают, а в квазисвязанном – могут не совпадать. Каждый режим имеет определенные достоинства и недостатки. В связанном режиме сигнализации отпадает необходимость в транзитных пунктах сигнализации, ведь отказ сигнального канала в большинстве случаев происходит одновременно с отказом группы речевых каналов. Однако при этом все направления связи должны иметь прямые сигнальные каналы, загрузка которых может быть небольшой. Поэтому применение связанного режима обычно экономически невыгодно, особенно в крупных сетях.

Квазисвязанный режим позволяет организовать сеть сигнализации более рационально, но требует создания транзитных пунктов сигнализации. К тому же возможны ситуации, когда работоспособный пучок речевых каналов нельзя использовать из-за отказа сигнального канала.

Выбор маршрутов сигнализации в сети сигнализации SS7 осуществляется в соответствии с таблицами маршрутизации. В таких таблицах указывается обычно два маршрута, один из которых, как правило используется при нормальных условиях, а второй - при отказе первого. Однако возможно использование обоих маршрутов в режиме разделения нагрузки, при этом может быть достигнуто более равномерное распределение сигнальной нагрузки по сети сигнализации.

Основными преимуществами общеканальной системы сигнализации SS7 являются следующие:

- ***скорость*** – время установления соединения не превышает одной секунды;
- ***высокая производительность*** – один канал сигнализации способен одновременно обслуживать до тысячи разговорных каналов;
- ***экономичность*** – минимальное количество оборудования на коммутационной станции;
- ***надежность*** – возможность альтернативной маршрутизации в сети сигнализации;
- ***гибкость*** – способность передачи любых данных (телефонии, цифровых сетей с интеграцией служб, сетей подвижной связи, интеллектуальных сетей и т.д.).

SS7 на данный момент является системой, обладающей огромным потенциалом. Изначально в нее были заложены большие возможности для управления другими, еще не существующими услугами связи. Сейчас общеканальная сигнализация ОКС №7 является обязательным компонентом следующих цифровых сетей связи:

- телефонной сети общего пользования (ТФОП);
- цифровой сети с интеграцией служб ISDN;
- сети связи с подвижными системами;
- интеллектуальной сети.

1.6. Выводы по разделу

1. Компьютерная сеть представляет собой распределенную в пространстве совокупность ЭВМ, линий и каналов связи, систем передачи данных, узлов коммутации и распределения информации, управляемую сетевым программным обеспечением, предназначенная для распределенной обработки данных, а также для обмена информацией между пользователями сети.

2. Существует два способа построения компьютерных сетей: на основе однородных (*гомогенных*) программно-совместимых компьютеров и узлов и интеграцию разнородных (*гетерогенных*) сетей в единую объединенную сеть – Интернет.

3. По масштабу сетей, размещения их в пространстве, способу и типу используемых каналов связи, сети подразделяются на глобальные и локальные.

4. Компьютерные сети различаются между собой топологией – схемой размещения узлов в пространстве и способом их соединения.

5. Существует стандартная эталонная модель взаимодействия открытых систем (ВОС), которая унифицированным образом описывает принципы взаимодействия разнообразных сетевых систем друг с другом. Согласно модели ВОС вся сеть делится на семь уровней: прикладной, представительный, сеансовый, транспортный, сетевой, канальный и физический.

6. Одинаковые уровни в различных системах взаимодействуют по определенным правилам (протоколам), определяющим поведение систем, устройств или их частей, выполняющих конкретные логически взаимосвязанные функции при передаче данных. Протоколы классифицируют по уровню, на котором они функционируют, назначению, способу передачи управляющих сигналов и др.

7. Различают сети с коммутацией каналов, сообщений и пакетов. Пакет представляет собой минимальную самостоятельную единицу сообщения, содержащую в заголовке всю необходимую служебную информацию, достаточную для самостоятельного перемещения по сети и доставки получателю.

8. В сетях с коммутацией каналов устанавливается физическое или логическое сквозное соединение взаимодействующих компьютеров на время сеанса связи. Различают сети с пространственной и временной коммутацией. В первом случае устанавливается физическое или логическое соединение между участками линий связи, различающихся по положению в пространстве. При временной коммутации каналов выполняется перестановка данных, находящихся на одной временной позиции (одном канале) в другую.

9. В сетях с коммутацией пакетов различают дейтаграммный способ передачи и передачу пакетов по виртуальным соединениям. Дейтаграммный способ является более простым, однако нередко нарушается очередность поступления пакетов на оконечный узел сети. Кроме этого возникает неконтролируемая задержка распространения пакетов в сети.

10. Для доставки сообщений получателю необходимо указывать его адрес, а также адрес отправителя. Различают глобальные и локальные, одно- и многоступенчатые адреса. С целью упрощения использования адреса часто представляются в виде последовательности некоторых символов.

11. В процессе соединения между двумя абонентами необходимо осуществить маршрутизацию, то есть определить те узлы сети, через которые будет проходить соединение. Различают централизованную, распределенную и смешанную маршрутизацию. Существуют различные способы маршрутизации: волновая, фиксированная, с альтернативными путями и адаптивная.

12. Для маршрутизации в объединенных сетях в основном применяются два алгоритма дистанционно-векторный алгоритм Беллмана-Форда и алгоритм маршрутизации с учетом состояния линий – алгоритм Дийкстры. Основным понятием алгоритмов является вектор расстояний, представляющий собой список с записями вида: "*Получатель, Стоимость*".

13. В процессе функционирования сети узлы через установленное время периодически рассылают служебные пакеты, содержащие текущие значения векторов стоимостей, всем своим непосредственным соседям. На основании этой информации каждый узел сети определяет по алгоритму Беллмана-Форда для любого возможного получателя пути с минимальными затратами. Это производится суммированием стоимостей доставки сообщений соседям с соответствующими стоимостями доставки от соседа к получателю, которые сообщил соседний узел. Стоимость в алгоритме Беллмана-Форда отображает количество переходов (хопов) между узлами.

14. При учете состояния линий связи узлы сети располагают информацией о топологии всей сети и о стоимости связей между ними. По алгоритму Дийкстры определяются кратчайшие пути от заданного узла-источника до всех остальных узлов сети, в процессе перебора путей в порядке увеличения их длин.

15. В процессе функционирования сети возможны ее перегрузки и, как следствие, блокировка отдельных узлов или всей сети в целом. Основная причина возникновения перегрузок – превышение скорости поступления информации над скоростью ее вывода, при ограниченной емкости запоминающих устройств. Для исключения этого явления осуществляется управление сетевыми потоками.

16. Управление потоком данных позволяет останавливать поток кадров до тех пор, пока узел не будет готов к приему следующих байтов. В процессе реализации процедуры управления потоком получателем посылается блок остановки потока в направлении, противоположному тому потоку байтов, который надо остановить.

17. На сетевом уровне простейшей формой управления потоком данных является передача блоков с остановками. Более эффективный способ – непрерывная передача группы блоков без ожидания подтверждения приема по каждому блоку. Такой алгоритм обмена данными получил название "управление потоком со скользящим окном".

18. В процессе обмена данными в сети необходимо обеспечивать заданное качество обслуживания QoS. Основными показателями QoS являются пропускная способность, задержка пакетов, степень приоритета, надежности и стоимость.

19. Процесс передачи служебной и управляющей информации в сети для обеспечения ее функционирования поддерживается системой сигнали-

зации**.** Сигнализация может осуществляться по основному каналу либо по индивидуальному выделенному сигнальному каналу. В современных компьютерных сетях передача управляющих сигналов реализуется специальными системами общеканальной сигнализации, в частности системой SS7, в которой передача всех линейных и управляющих сигналов, необходимых для функционирования множества речевых каналов, осуществляется по одному управляющему каналу передачи данных. При этом все сигналы собираются в пакеты - сигнальные единицы и снабжаются заголовком, устанавливающим принадлежность каждого из сигналов определенному речевому каналу.

20. Более подробно со стеками протоколов и особенностями настройки параметров протоколов можно ознакомиться в [2,4,6,7,9,18,24,26,32], способами коммутации в компьютерных сетях - в [16,17,18], алгоритмами маршрутизации, настройкой и программированием маршрутизаторов – в [10,11,18,24], управлением потоками и способами защиты от перегрузок – в [8,9,16].

1.7. Контрольные вопросы

1. В чем состоит коренное различие между локальными и глобальными компьютерными сетями?
2. Какая разница между хостом и рабочей станцией?
3. Из каких соображений выбирается топология в процессе проектирования компьютерной сети?
4. Назовите топологии, в которых используются двухточечные соединения, и в которых – многоточечные.
5. Что означает термин "широковещательная передача"?
6. По какой причине компьютерная сеть подразделяется на уровни? Перечислите эти уровни и назовите их функции.
7. Почему большинство известных стеков протоколов отличаются от стека эталонной модели?
8. С какой целью осуществляется коммутация каналов вместо использования прямых связей?
9. Чем отличаются между собой коммутация сообщений и пакетов?
10. С какой целью используют многоступенчатые пространственные коммутаторы?
11. Как осуществляются соединения каналов в системах с временной коммутацией?

12. Зачем во временных коммутаторах применяется запоминающее устройство и как оно влияет на максимально допустимое количество коммутируемых каналов?
13. Для решения каких задач используется управляющая информация, содержащаяся в сетевом пакете?
14. Назовите отличия виртуального соединения от физического?
15. С какой целью используются многоступенчатые адреса?
16. Что называется "маршрутизацией", и каковы критерии оптимальности маршрутизации?
17. Назовите преимущества и недостатки лавинной маршрутизации?
18. Как можно выбрать оптимальный путь при распределенной маршрутизации, если маршрутизатору в начале работы известна информация только о путях до соседних узлов?
19. В чем заключается суть, достоинства и недостатки алгоритма маршрутизации Беллмана-Форда?
20. Проиллюстрируйте процедуру поиска оптимального маршрута по алгоритму Дийкстры.
21. Назовите преимущества алгоритма маршрутизации с учетом состояния линий?
22. Какое явление в компьютерной сети называют перегрузкой и каковы способы ее предотвращения?
23. Перечислите показатели, характеризующие качество обслуживания в компьютерных сетях?
24. Каковы способы управления потоками данных в сетях и в чем состоит суть управления со скользящим окном?
25. С какой целью в сетях применяется сигнализация? Раскройте суть общеканальной сигнализации?

Раздел 2
ПЕРЕДАЧА ДАННЫХ В КОМПЬЮТЕРНЫХ СЕТЯХ

Линия, кабель и канал связи – синонимы, или принципиально разные понятия? Что же ограничивает скорость передачи по линии или каналу связи? До какой величины можно увеличивать уровень сигнала в линии? Почему компьютерные данные нельзя непосредственно передавать по телефонным каналам связи? Зачем нужна модуляция сигналов? Что такое расширение спектра и зачем оно нужно? Как обеспечить в компьютерных сетях необходимую достоверность передачи информации при наличии интенсивных помех? На эти и многие другие вопросы Вы найдете ответ, ознакомившись с данным разделом.

2.1. Линии и каналы связи

В качестве физической среды распространения сигналов в компьютерных сетях используются электрические и оптические кабели, беспроводные линии связи, а также каналы связи. *Кабелем связи* называется система изолированных пар проводников, имеющая общую оболочку для защиты от механических и климатических воздействий. В проводных кабелях в качестве среды распространения сигналов служит *медный* проводник, а в оптических – *световод*, выполненный из кварцевого стекла или полимерных материалов. Беспроводные линии связи построены на основе волн инфракрасного излучения либо радиоволн.

Зачастую существующие линии связи способны обеспечить значительно большую пропускную способность по сравнению с реальной скоростью передачи сигналов аппаратуры пользователя по таким линиям. Поэтому непосредственная передача информации по физическим линиям осуществляется только на небольшие расстояния (несколько км). При необходимости передачи сигналов на большие расстояния линии связи уплотняют, т.е. с помощью специальных многоканальных систем связи (аппаратуры уплотнения) ресурсы линии связи разделяют на несколько независимых каналов. Число таких каналов, образованных на одной линии связи, может составлять от десятков до нескольких тысяч.

Обобщенная структурная схема многоканальной системы связи (МСС) изображена на рисунке 2.1.

Рисунок 2.1 – Обобщенная структурная схема многоканальной системы связи

Оконечная передающая аппаратура предназначена для преобразования N передаваемых исходных сигналов. При этом каждый сигнал в линии связи должен отличаться от других по одному из параметров (занимаемая полоса частот, время передачи и т.д.). Совокупность таких сигналов, так называемый *групповой сигнал*, передается по линии связи (ЛС). Промежуточное оборудование служит для компенсации затухания и искажений, которые претерпевают сигналы при передаче по ЛС. Оконечная аппаратура приемной стороны осуществляет обратное преобразование группового сигнала в N исходных.

Каналом связи называется независимый тракт передачи сигналов от источника к получателю, образованный аппаратурой уплотнения на физической линии путем использования части ресурсов этой линии.

Кабельные системы являются тем связующим звеном, которое соединяет все необходимые компоненты компьютерных сетей предприятий и организаций. Рациональная организация кабельной системы здания или группы зданий является одной из важнейших задач при построении компьютерных сетей. Именно она в основном определяет надежность функционирования всех сетевых служб подразделений и организации в целом. При прокладке кабельных линий следует учитывать, что за время их эксплуатации возникнут неизбежные изменения информационных характеристик и технологий передачи сообщений, количества пользователей, аппаратного и программного обеспечения сети. Кроме этого, кабельная сеть должна позволять передачу сигналов других служб: телефонии, телевидения, пожарной сигнализации и пр. Поэтому кабельная сеть здания должна быть универсальной, обладать достаточной гибкостью, а также позволять осуществлять относительно простой доступ обслуживающему персоналу к линиям связи и распределительным устройствам. Для обеспечения таких возможностей была разработана концепция структуированных кабельных систем.

Структурированная кабельная система (**СКС**) представляет собой иерархическую кабельную систему здания или группы зданий, разделенную на структурные подсистемы. СКС состоит из набора медных и оптических кабелей, кросс-панелей, соединительных шнуров, кабельных разъемов, мо-

дульных гнезд, информационных розеток и вспомогательного оборудования. Кроссовые панели (*Cross Connect Panel*) обеспечивают соединение кабелей с портами сетевого оборудования. К кроссовым панелям подходят все кабели компьютерных сетей этажа или всего здания. Существует два типа кросс–панелей. К первому относятся панели с врезными контактами. Лезвия такого соединителя разрезают изоляцию провода при вставке, обеспечивая тем самим электрическое соединение с жилой провода и фиксацию его в контакте. Такие соединители применяются преимущественно на телефонных коммутационных узлах. Ко второму типу относятся модульные панели, разработанные специально для компьютерных сетей. Они имеют множество гнезд, объединенных в модули, для кабелей различных типов, например RJ-45, BNC, оптоволоконных кабелей и т.п. Для перекоммутации линий связи на кроссовых панелях используются стандартные гибкие электрические соединительные шнуры. Все перечисленные элементы интегрируются в единую систему и эксплуатируются согласно определенным правилам.

Структурированные системы позволяют быстро и легко изменять конфигурацию кабельной системы внутри здания и между зданиями. Для этого администратору сети достаточно перекоммутировать контакты на кросс–панелях. Это позволяет обеспечить гибкое изменение рабочих мест сотрудников и полное изменение конфигурации системы, включая замену и добавление оборудования, расширение системы.

В структурированную кабельную систему закладывается структурная избыточность, предусматривающая возникновение дополнительных рабочих мест для абонентов компьютерной сети, возможности перемещения в пределах организации оборудования и персонала. Избыточность СКС требует дополнительного количества кабеля, розеток, кросс-панелей. Однако дополнительные капитальные затраты, необходимые для создания СКС, быстро окупаются в процессе ее эксплуатации.

2.1.1. Проводные кабельные линии

Кабельные проводные линии подразделяются на *симметричные* и *несимметричные*. Симметричная линия состоит из двух совершенно одинаковых в конструктивном и электрическом отношениях изолированных медных проводников. Для улучшения симметричности двухпроводной линии ее проводники скручивают между собой. Параметры такой электрической цепи (сопротивление проводника, емкость и проводимость изоляции) симметричны относительно земли. Пара скрученных изолированных медных жил является основным элементом симметричного кабеля. Поэтому для симметричных кабелей часто применяются термины "скрученная" или "витая пара".

Витая пара проводников (*twisted pair*) представляет собой наиболее массовый способ соединения локальных компьютерных сетей ввиду ее доступности и низкой стоимости. Существует два вида витого кабеля: ***неэкранированная витая пара*** *UTP* (*Unshielded Twisted Pair*) и **экранированная витая пара** STP (*Shielded Twisted Pair*). Кабель состоит из несколько витых пар (часто 4-х), которые имеют общую защитную оболочку. Скручивание жил между собой необходимо для уменьшения электромагнитного влияния внутри многопарного кабеля. Чем выше частота сигнала, на которую рассчитан кабель, тем регулярнее и плотнее должна быть выполнена скрутка жил.

Сама медная жила может иметь различный диаметр (от 0,4 до 0,9 мм), быть сплошной (в монтажных кабелях) или в виде жгута из нескольких тонких неизолированных проволок (в гибких кабелях и шнурах для подключения оборудования и коммутации цепей).

В качестве изоляции жил применяется телефонная бумага, полиэтилен, полистирол, поливинилхлорид (ПВХ), стирофлекс, фторопласт и другие материалы. Полиэтиленовую изоляцию низкочастотных кабелей изготавливают сплошной, пористой или пористо-сплошной. Для уменьшения электрической емкости и затухания кабеля желательно изоляцию делать толще. Поэтому применяют чаще пористую изоляцию, которая позволяет снизить общий вес кабеля. Для дальнейшего улучшения значений этих параметров в высокочастотных кабелях обычно используется кордельная изоляция, у которой слой изолирующего диэлектрика имеет меньшую площадь контакта с жилой. Изолирующий материал накладывается не на саму жилу, а на предварительно наложенную **кордель** (тонкий шнур) с сохранением воздушного промежутка.

Иногда для достижения более устойчивых механических и электрических характеристик кабеля вместо парной используется четверочная скрутка (скрученные две двухпроводные цепи). Особое значение имеет шаг скрутки – расстояние, через которое повторяется взаимное расположение жил кабеля. Для минимизации переходного влияния между парами в многопарных кабелях они скручиваются с различным шагом. Пары и четверки, в свою очередь, скручиваются в так называемый **сердечник** кабеля для обеспечения однородности электрических параметров всех пар кабеля (между ними и по всей их длине).

Защита от электромагнитных влияний симметричных кабелей осуществляется за счет создания электростатического экрана вокруг витых пар и/или всего сердечника кабеля. Он может быть выполнен из фольги или в виде сплетенного из проволок "чулка". Экранированные кабели выпускаются с различным исполнением экрана: оплетка – **STP** (*Shielded Twisted Pair*), экранирование фольгой – **FTP** (*Foilled Twisted Pair*), различные варианты усиленных (двойных) – **SSTP** (*Shielded/Shielded Twisted Pair*)**, SFTP**

(*Shielded/Foilled Twisted Pair*) экранов. В последних двух типах экранируется каждая витая пара в отдельности и весь кабель в целом.

В настоящее время почти все кабели изготавливаются в оболочке из полиэтилена. Для защиты от проникновения влаги может применяться барьер из алюминиевой фольги (он же играет роль общего электростатического экрана). Вне зданий укладывают герметизированные кабели, у которых свободный объем сердечника заполнен водоотталкивающим составом (гидрофобным компаундом).

Эквивалентная схема симметричной двухпроводной линии связи изображена на рисунке 2.2.

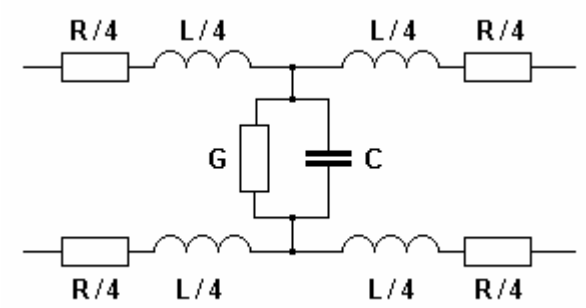


Рисунок 2.2 – Эквивалентная схема симметричной двухпроводной линии связи

Кабельная линия характеризуется первичными и вторичными параметрами. Значение первичных параметров определяется конструкцией кабеля и частотой передаваемого сигнала. Величина этих параметров зависит от длины линии связи. Для того чтобы параметры не зависели от общей длины линии, указывают нормированное *погонные* значения первичных параметров (значение параметра на единицу длины линии связи). В качестве длины линии берется 1 метр или километр. К первичным параметрам относятся сопротивление проводов отрезка линии связи R [Ом/м], их индуктивность L [Гн/м], емкость между парой проводов кабеля C [Ф/м] и проводимость изоляции G [Сим/м].

Кроме первичных параметров, проводные линии характеризуются также вторичными параметрами, к которым относятся **волновое сопротивление** $Z_в$ и **коэффициент распространения сигнала** γ, составляющими которого являются коэффициент затухания α (*Attenuation*) и коэффициент фазы β сигнала. Величина волнового сопротивления зависит от первичных параметров линии и частоты тока в ней.

Волновые параметры ЛС определяются по следующим формулам:

$$Z_в = \sqrt{(R + j\omega L)/(G + j\omega C)};$$
$$\gamma = \sqrt{(R + j\omega L)(G + j\omega C)} = \alpha + j\beta; \qquad (2.1)$$
$$\alpha \approx R/2\left(\sqrt{C/L}\right) + G/2\left(\sqrt{L/C}\right); \quad \beta \approx \omega\sqrt{LC}.$$

Волновое (характеристическое) сопротивление $Z_в$ является коэффициентом пропорциональности между волной тока и напряжения при распространении сигнала в линии. При частоте $\omega = 0$ характеристическое сопротивление цепи $Z_в = (R/G)^{1/2}$. А на достаточно высоких частотах, где справедливы соотношения $\omega L >> R$ и $\omega C >> G$, волновое сопротивление становится постоянной величиной $Z_в = (L/C)^{1/2}$, не зависящей от частоты. Поскольку $R/G >> L/C$, то модуль $Z_в$ — монотонно убывающая функция от $(R/G)^{1/2}$ при $\omega = 0$ до $(L/C)^{1/2}$ на высоких частотах.

Коэффициент затухания линии α характеризует ослабление сигнала на выходе симметричной пары длиной 1 км, нагруженной на ее волновое сопротивление. Он измеряется в дБ/км и увеличивается с ростом частоты. Затухание в стандартной витой паре при частоте сигналов 10 МГц составляет более 120 дБ/км и быстро увеличивается с частотой.

На практике коэффициент затухания измеряется как отношение мощностей или амплитуд напряжения сигнала в начале линии и точке измерения. Затухание в системах связи выражают в децибелах (дБ).

$$\alpha = 10 \lg (P_{вх} / P_x), \qquad (2.2)$$

где $P_{вх}$ и P_x — мощности сигнала на входе линии и произвольной точке *x* соответственно. Так, например, если мощность сигнала в точке *x* в 100 раз меньше входной мощности, то затухание линии равно 20 дБ.

Коэффициент фазы β определяет фазовый сдвиг гармоники сигнала определенной частоты при распространении его по кабелю. Как и коэффициент затухания α, он нормирован относительно длины 1 км, и измеряется в рад/км.

Волна напряжения и тока, приходящая к концу линии, отдает нагрузке всю энергию только в том случае, когда сопротивление нагрузки равно волновому сопротивлению линии связи. В противном случае часть энергии возвращается от конца линии к ее началу в виде отраженной волны тока и напряжения и за счет сложения с прямой волной искажает передаваемые сигналы. Поэтому условием неискаженной (и максимальной мощности) передачи сигналов является равенство сопротивления нагрузки волновому сопротивлению линии связи $Z_в = Z_н$. Согласование сопротивлений отдельных участков электрических цепей обычно выполняется посредством согласующего

трансформатора. Как видно из рисунка 2.2, эквивалентная схема линии аналогична схеме фильтра нижних частот. С возрастанием частоты сигнала затухание такого звена возрастает. Затухание линии увеличивается также с ростом температуры.

Линия, по которой в данный момент времени передаются электрические сигналы, называется активной. Активная пара, естественно, создает электромагнитное поле. Это поле может оказывать влияние на другие, находящиеся поблизости соседние пары, т.е. создавать так называемые *перекрестные помехи*. Схема влияния активной линии на соседние пары показана на рисунке 2.3.

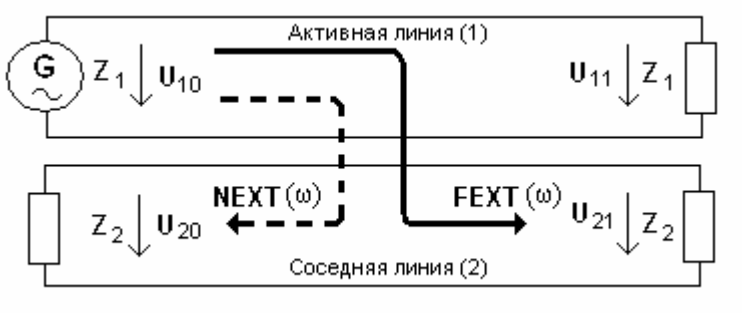


Рисунок 2.3 – Схема воздействия активной линии кабеля на соседнюю пару

Степень мешающего воздействия активной линии оценивается посредством **переходных затуханий** между парами проводов **на ближнем и дальнем концах линии (NEXT, FEXT).** Здесь параметр **NEXT (***Near End Crosstalk*) – переходное затухание измеренное на ближнем конце соседней пары, а **FEXT** (*Far End Crosstalk*) – переходное затухание, измеренное на дальнем конце соседней пары. С увеличением частоты сигнала переходные затухания уменьшаются. Количественно эти параметры оцениваются следующим образом:

$$NEXT = 20\lg (U_{10}/U_{20}) \text{ дБ}; \quad FEXT = 20\lg (U_{10}/U_{21}) \text{ дБ}. \qquad (2.3)$$

Параметр FEXT характеризует интенсивность перекрестных помех на дальнем конце линии, т. е. перекрестные помехи измеряются на другом конце по отношению к источнику сигнала. Сам по себе параметр FEXT не представляет интереса для измерений ввиду зависимости его от длины линии. Две линии на базе компонентов одной и той же категории, но разной длины, будут иметь различные значения FEXT. Поэтому для измерений был выбран параметр ELFEXT (*Equal Level Far End Crosstalk*)

$$ELFEXT = FEXT - \alpha_2. \qquad (2.4)$$

С целью уменьшения степени мешающих воздействий на соседние пары кабеля уровень сигналов передачи нормируют. Так, например, уровень сигнала передачи данных по телефонным кабельным линиям ограничивают величиной минус 13 дБ в точке нулевого измерительного уровня (напряжение в которой равно 0,775 В). Для симметричных линий введен еще ряд параметров, характеризующие их помехозащищенность.

Защищенность от помех ACR *(Attenuation to crosstalk ratio)* – это превышение сигнала над уровнем собственных шумов. Определяется разностью ARC = NEXT [дБ] – α [дБ].

Скорость распространения сигналов NVP *(Nominal Vilocity of Propagation)* – относительная скорость распространения сигналов, выражающая в процентах замедление сигналов в витой паре относительно скорости света в вакууме. Может использоваться для определения места повреждения.

Задержка прохождения сигналов *(Propagation Delay)* – представляет собой время распространения сигнала от одного конца линии до другого. Именно она является причиной ограничения длины кабельных линий для сетевых приложений.

Разброс задержек прохождения сигналов *(Skew)* – максимальная разность задержек прохождения сигнала между всеми парами. Разброс вызывается в значительной степени различным шагом скрутки каждой из пар (который делается для уменьшения взаимного влияния NEXT и FEXT) и как следствие разной электрической длиной пар. На разброс задержек в меньшей степени оказывает влияние также неоднородность параметров медных проводников и диэлектриков изоляции, обуславливающая различную скорость распространения электромагнитной волны. Введение параметра разброса задержки прохождения сигналов обусловлено тем, что некоторые локальные компьютерные сети, такие, например, как 100VG AnyLAN, 100BASE-T4, 1000BASE-T, используют для передачи сигналов одновременно все четыре пары симметричного кабеля. Если задержка прохождения сигнала в одной паре существенно отличается от задержки прохождения сигнала в другой паре, то это может привести к их рассинхронизации до такой степени, что восстановление исходного сигнала на приемной стороне будет невозможно. Задержка прохождения сигнала и разброс задержки сигнала обычно измеряются в наносекундах.

В настоящее время промышленностью выпускается 7 категорий кабеля UTP: 1-я категория – традиционный телефонный кабель, по которому можно передавать только речевые сигналы; 2 – для сигналов с частотой передачи до 1 МГц; 3 – для сигналов с частотой передачи до 16 МГц; 4 – до 20 МГц; 5 – до 100 МГц; категории 6 – до 600 и 7 – свыше 600 МГц.

Кабели категорий 2…5 состоят из 4 витых пар каждый. Волновое со-

противление всех этих кабелей равно 100 Ом. В витой паре один проводник является сигнальным, а второй используется в качестве общего провода, уравнивающего потенциалы на передающей и приемной станциях. В локальных компьютерных сетях наиболее широко используются кабели 3 и 5 категорий. Кабель третьей категории первоначально предназначался для телефонной связи. Он состоит из витых пар с 9-ю витками на метр длины. Кабель пятой категории разработан специально для компьютерных сетей. Ключевое различие между кабелями 3-й и 5-й категорий заключается в количестве витков скручивания пары проводников на единицу длины кабеля. В пятой категории количество витков на метр кабеля равно 27, что в 3 раза больше чем в кабеле третьей категории. Это позволяет существенно повысить пропускную способность линии. **Категория 7** является единственной на данный момент стандартизированной средой передачи, которая без каких либо оговорок способна обеспечивать передачу со скоростью 10 Гбит/с по линиях длиной до 100 м. В кабелях 7 категории существенно уменьшен уровень шумов. Этот фактор является очень важным, так как основным мешающим фактором для систем передачи данных, работающих со скоростью 10 Гбит/с, является тепловой шум. Уменьшение шумов в этом кабеле достигается благодаря особенностям конструкции кабеля и модульных разъемов. Пары составляются из жил диметром не менее 0,58 мм, причем каждая пара заключается в индивидуальный экран из фольги. Экранирование каждой пары по всей окружности обеспечивается и в модульном разъеме. Благодаря этим мероприятиям, для такого кабельного оборудования являются менее ощутимыми наводки, в том числе и межкабельные.

Другим видом электрического кабеля, широко применявшимся в первых локальных компьютерных сетях, является ***коаксиальный кабель*** (от латинского **co** – совместно и **axis** – ось), представляющий собой два соосных гибких металлических цилиндра, разделенных диэлектриком (рисунок 2.4).

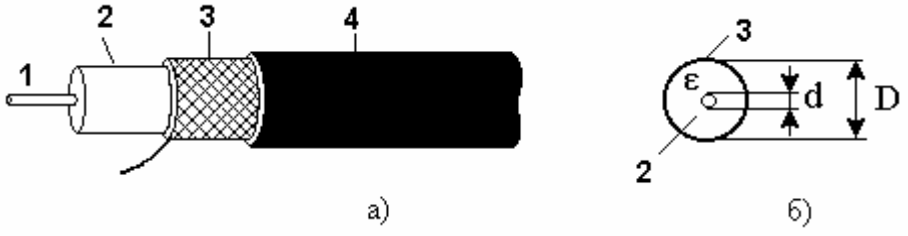


Рисунок 2.4 – Конструкция коаксиального кабеля

Здесь 1– центральный провод (жила); 2 – изолятор центрального провода; 3 – экранирующий (3) проводник (*оплетка*); 4 – внешний изолятор и защитная оболочка. Роль общего проводника в нем играет внешний цилиндрический проводник, выполняющий функцию электрического экрана. В свя-

зи с этим коаксиальные кабели обладают высокой защищенностью от внешних электромагнитных полей. Коаксиальный кабель относится к несимметричным линиям связи. Волновое сопротивление коаксиального кабеля Z_B зависит от диэлектрической проницаемости слоя изолятора ε и соотношения наружного диаметра изоляционного слоя D и диаметра жилы d (рисунок 2.4,б)

$$Z_B = 60\sqrt{\varepsilon}\ln(D/d).$$

Коаксиальный кабель бывает двух видов: тонкий и толстый. **Тонкий** (*thin*) коаксиальный кабель имеет диаметр оплетки около 5 мм. Он способен передавать сигнал на расстояние до 180 м без его заметного искажения, вызванного неравномерностью затухания на различных частотах. Тонкий коаксиальный кабель относится к группе, которая по маркировке производителей называется семейством RG–58, его волновое сопротивление равно 50 Ом. Основная отличительная особенность этого семейства – медная жила. Она может быть сплошной или состоять из нескольких переплетенных проводов. Для подключения тонкого коаксиального кабеля к компьютеру используются коаксиальные разъемы со штыковым способом фиксации, так называемые **BNC**-коннекторы (*Bayonet Nut Connector*).

Толстый (*thick*) коаксиальный кабель – относительно жесткий кабель с диаметром оплетки около 10 мм. Иногда его называют "стандартный Ethernet кабель", поскольку он был первым типом кабеля, применяемым в сетях *Ethernet*. Медная жила этого кабеля толще, чем у тонкого коаксиального кабеля. Чем толще жила у кабеля, тем меньше затухание и тем большее расстояние способен преодолеть сигнал. Толстый коаксиальный кабель может передавать сигналы в сетях Ethernet на расстояние до 500 м. Поэтому толстый коаксиальный кабель иногда используют в качестве магистрального (*backbone*) кабеля, который соединяет несколько небольших сетей, построенных на тонком коаксиальном кабеле. Для подключения к толстому коаксиальному кабелю применяют специальное устройство – **трансивер** (*transceiver*). Трансивер снабжен специальным коннектором, который назван "зуб вампира" или "пронзающий ответвитель". Этот зуб проникает через изоляционный слой и вступает в непосредственный физический контакт с проводящей жилой. Чтобы подключить трансивер к сетевому адаптеру, надо кабель трансивера подключить к коннектору AUI-порта сетевой платы. Этот соединитель известен также как коннектор DB-15 или DIX-коннектор (*Digital Intel Xerox*), в соответствии с названиями фирм-разработчиков.

Недостатком толстого кабеля является сложность его прокладки (из-за малой гибкости) и относительно высокая стоимость. Тонкий коаксиальный кабель достаточно гибок, прост в установке и сравнительно недорог. До не-

давнего времени коаксиальные кабели представляли собой наиболее массовую физическую среду передачи информации в локальных сетях. Сейчас их почти повсеместно вытеснили витые пары и волоконно-оптические кабели.

2.1.2. Оптические линии связи

Для создания оптических линий связи широко применяются **волоконно-оптические кабели** (*fiber-optic cable*). Они заметно конкурируют с некоторыми видами коаксиальных кабелей и являются основной средой передачи в высокоскоростных моноканалах локальных компьютерных сетей. В качестве физической среды распространения сигналов используются сверхпрозрачное стекловолокно или волокно, изготовленное на базе полимеров. Простейший оптический кабель состоит из светопроводящей (кварцевой или полимерной) сердцевины диаметром 2…200 мкм, окруженной тонкой пластмассовой или стеклянной пленкой со значительно меньшим коэффициентом преломления, чем в сердцевине. Этим достигается практически полное внутреннее отражение световых потоков. Снаружи кабель покрывается защитной оболочкой.

По своей конструкции волоконно-оптический кабель подобен коаксиальному. Однако вместо центральной жилы в его центре располагается сердцевина, окруженная оптической оболочкой, покрытой тонким слоем лака. Кроме этого оптический кабель содержит элементы усиления его механической прочности и внешнее покрытие. Сердцевина и оболочка изготавливается как одно целое. Оболочка имеет толщину от сотен микрометров до единиц миллиметров. Оболочка может быть покрыта дополнительно буферным слоем, который в свою очередь может быть свободным (жесткая пластиковая трубка) или плотно прилегающим. Свободный слой защищает от механических повреждений и температуры, прилегающий – только от механических повреждений. Элементы усиления выполняются из стальной проволоки, нитей *кевлара* (вид особопрочной пластмассы) и т.д. Внешнее покрытие изготавливается аналогично покрытию электрических кабелей.

Скорость передачи сигналов по оптическому кабелю составляет 2…5 Гбит/с и выше. Затухание оптического кабеля имеет величину 0,2…10 дБ/км, причем оно незначительно возрастает с ростом частоты передачи сигналов. Каждое оптоволокно передает сигналы только в одном направлении, поэтому кабель состоит из двух волокон с отдельными коннекторами. Для защиты от внешних воздействий кабель имеет общее покрытие из пластика, а для повышения прочности внутри кабеля, наряду с оптоволокном, проложены нити из кевлара.

Различают два типа оптических кабелей: **многомодовые** и **одномодо-**

вые. По словом "мода" понимают световые лучи внутри кабеля, которые имеют одинаковые углы отражения. В многомодовых кабелях распространяются несколько световых лучей, которые попадают на границу раздела оптических свойств и отражаются от нее под различными углами. Лучи, распространяющиеся в середине световода (без отражения), имеют моду нулевого порядка, так называемые аксиальномодовые лучи. Одномодовые кабели имеют настолько малый диаметр (2…10 мкм), что в нем возможно распространения лучей только нулевой моды.

На рисунке 2.5 показана схема распространения лучей в многомодовых и одномодовом волоконно-оптических кабелях при различном профиле изменения коэффициента преломления лучей в среде распространения.

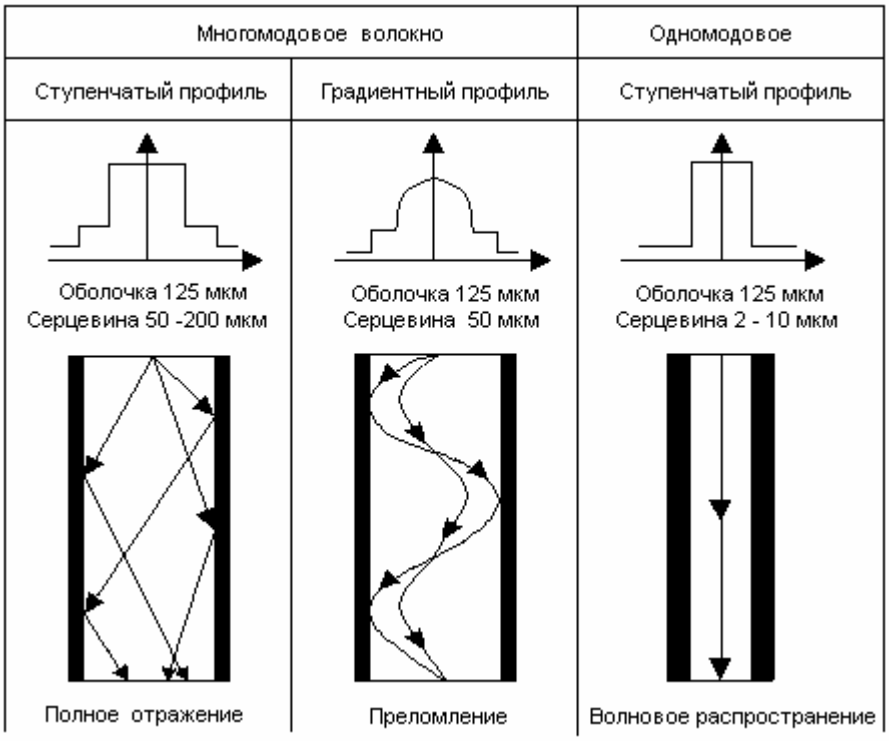


Рисунок 2.5 – Схема распространения лучей в волоконно-оптическом кабеле и зависимости коэффициента преломления лучей

Различный коэффициент преломления и его распределение внутри сердцевины достигается путем введения специальных добавок в оптическую массу в процессе производства (вытягивания) кабельной нити.

В связи с тем, что в многомодовом кабеле одновременно распространяется несколько лучей одного и того же сигнала, имеющих различное время прохождения, результирующий сигнал на выходе кабеля расширяется и происходит межсимвольная интерференция передаваемых единичных элемен-

тов, которая возрастает с увеличением длины кабеля. Это затрудняет различение и регистрацию единичных элементов. В связи с этим приходится ограничивать скорость передачи в многомодовых волоконно-оптических линиях связи. В одномодовых кабелях межсимвольная интерференция практически отсутствует. Поэтому скорость передачи в таких линиях связи выше.

Существенным преимуществом волоконно-оптического кабеля, кроме высокой пропускной способности, является независимость от внешних электромагнитных полей. Основной недостаток волоконно-оптических линий – высокая стоимость производства компонентов, а также большие затраты на их монтаж и ремонт. Для обеспечения большей пропускной способности линии связи промышленностью выпускаются оптоволоконные кабели, содержащие несколько (до 8) одномодовых волокон с малым затуханием. Разработаны и производятся кабели для распределительных сетей, которые могут содержать до 216 волокон как одномодовых, так и многомодовых.

2.1.3. Беспроводные линии связи

В практической деятельности встречаются ситуация, в которой принципиально невозможно проложить кабель для компьютерной сети (это, как правило, старинные здания, памятники архитектуры, либо очень дорогие интерьеры, где стоимость прокладки кабельной сети непомерно высока), или важна скорость развертывания сети (временные сети на выставках, семинарах и т.п.). Решение такого рода проблем возможно при использовании беспроводной среды передачи сигналов.

В настоящее время в беспроводных компьютерных сетях для передачи данных применяются виды излучений: инфракрасное (тепловое); оптическое (видимое); радиоволновое.

Инфракрасное излучение (*Infrared radiation*) представляет собой разновидность оптического излучения с длиной волны большей, чем у видимых лучей. По длине волны колебаний инфракрасное излучение подразделяется на коротковолновое (от 800 до 1400 нм), средневолновое (от 1400 до 3000 нм), длинноволновое (от 3000 до 10000 нм).

Инфракрасное (тепловое) излучение испускается всеми телами, имеющими температуру выше абсолютного нуля. Источником инфракрасного излучения в беспроводных системах передачи данных служит лазер или фотодиод. Лазер представляет собой квантовый генератор, испускающий когерентные электромагнитные волны вследствие вынужденного излучения активной среды, находящейся в оптическом резонаторе. Напомним, что когерентными называются волны, характеризующиеся одинаковой частотой и постоянством разности фаз в заданной точке пространства. В зависимости от

вида активной среды различают газовые, твердотельные и жидкостные лазеры.

В инфракрасных беспроводных сетях необходимо генерировать довольно сильный сигнал, так как на него воздействуют помехи других источников тепла. Этот способ позволяет передавать сигналы с большой скоростью, поскольку инфракрасные колебания имеет широкий диапазон частот. Инфракрасные сети способны нормально функционировать на скорости около 10 Мбит/с. Существует четыре типа инфракрасных сетей.

Сети прямой видимости. Передача возможна лишь в случае прямой видимости между передатчиком и приемником. Сети, использующие прямое излучение, строятся по схеме "точка-точка". Организация сетей, применяющих прямое излучение, требует очень точного наведения, особенно если в качестве источников излучения применяются лазеры. В инфракрасных сетях используются частоты излучения в диапазоне 100…1000 ГГц, достигаемая пропускная способность колеблется от 0,1 до 16 Мбит/с.

Сети на рассеянном инфракрасном излучении. Связь между пользователями обеспечивается за счет поступления на вход инфракрасного приемника лучей, отраженных от различных поверхностей (стен, потолка и пр.). Эффективная область распространения сигналов ограничивается расстояниями до 30 м. Сети, использующие рассеянное излучение, не предъявляют требования к точной настройке излучателей и приемников, что позволяет абоненту перемещаться в пространстве. Однако такие сети обладают меньшей пропускной способностью, которая обычно не превышает 1 Мбит/с.

Сети на отраженном инфракрасном излучении. В этих сетях инфракрасные передатчики, расположенные рядом с компьютером, передают сигналы на отражатель, устанавливаемый, как правило, на потолке. Отраженный сигнал направляется на вход приемника соответствующего компьютера.

Широкополосные инфракрасные сети. Сети такого рода позволяют вести передачу данных на очень высокой скорости, они практически не уступают кабельным сетям.

Хотя скорость и удобство использования инфракрасных сетей очень привлекательны, возникают трудности при передаче сигналов на расстояние более 30 м. К тому же такие сети подвержены помехам со стороны источников тепла (лампы накаливания, калориферы), которые имеются в большинстве организаций.

Технология, использующая видимые оптические лучи, похожа на инфракрасную тем, что требует прямой видимости между передатчиком и приемником. Если по каким–либо причинам оптический луч будет прерван, то это приведет к прекращению обмена данными. Для генерирования оптических колебаний используется полупроводниковый прибор – лазер. Качество передачи в большой степени зависит от внешних атмосферных условий

(дождь, туман, запыленность атмосферы). Использование в сетях передачи данных источника видимого света, по сравнению с инфракрасным излучением, более проблематично, так как функционирующий источник видимого света (лазер) может вызвать ожог тканей человека. Поэтому при построении сетей, использующих видимый свет, следует также решать проблемы исключения случайной травмы пользователей сети, обслуживающего персонала или случайных людей.

Отличительной особенностью **радиолиний** является распространение электромагнитных сигналов в свободном пространстве. Радиоизлучение осуществляется на частотах от сотен кГц до сотен ГГц. В сетях передачи данных нашли применение радиоволны ультракороткого (УКВ) диапазона (выше 30 МГц), которые распространяются прямолинейно и не отражаются ионосферой (как короткие волны) и не огибают встречающиеся препятствия, как длинные и средние волны. Поэтому связь в сетях передачи данных, построенных на УКВ радиосредствах, ограничена по расстоянию (до 40 км). Для преодоления этого ограничения обычно используют ретрансляторы. УКВ колебания, в свою очередь, подразделяются на метровые, дециметровые, сантиметровые, миллиметровые и субмиллиметровые волны. В последнее время в локальных вычислительных сетях начали использоваться радиоизлучения на дециметровых волнах (частота – 300…3000 МГц), сантиметровых (3…30 ГГц) и миллиметровых волнах (30…300 ГГц).

К беспроводной технологии радиосвязи, использующий диапазон частот от 2,4 до 2,4835 ГГц, относится популярная технология *Bluetooth*. Передатчики, излучающие радиоволны в этом диапазоне, разделены по мощности на три категории: 100 мВт (дальность связи 100 м), 2,5 мВт (10 м) и 1 мВт (10 см).

К системам передачи, использующим радиоизлучение в сверхвысокочастотном (СВЧ) диапазоне (3…30 ГГц), относятся микроволновые системы. Микроволновая технология помогает организовать взаимодействие между зданиями в небольших, компактных системах, например в университетских городках. Микроволновая система состоит из приемопередатчика и двух направленных антенн. Они нацелены друг на друга так, чтобы осуществить прием сигналов, передаваемых трансивером. Эти антенны часто устанавливают на вышки, чтобы преодолеть большие расстояния.

В беспроводных локальных компьютерных сетях преимущественно используется ненаправленное излучение радиоволн в СВЧ диапазоне (2… 30 ГГц). Электромагнитная энергия радиоволн распространяется во всех направлениях и может приниматься многими антеннами. Радиоволны СВЧ диапазона обладают практически идеальной отражающей способностью. Поэтому на антенну радиоадаптера компьютера приходят волны, многократно отраженные от потолка и стен помещения (рисунок 2.6). При этом имеет ме-

сто так называемое *многолучевое распространение* сигнала *S*(t). В связи с тем, что волны распространяются по различным путям, они проходят не одинаковое расстояние. Время распространения сигналов t_{pk} каждого из лучей различно. Таким образом, в месте приема происходит сложение (интерференция) колебаний электромагнитных волн, прошедших различный путь.

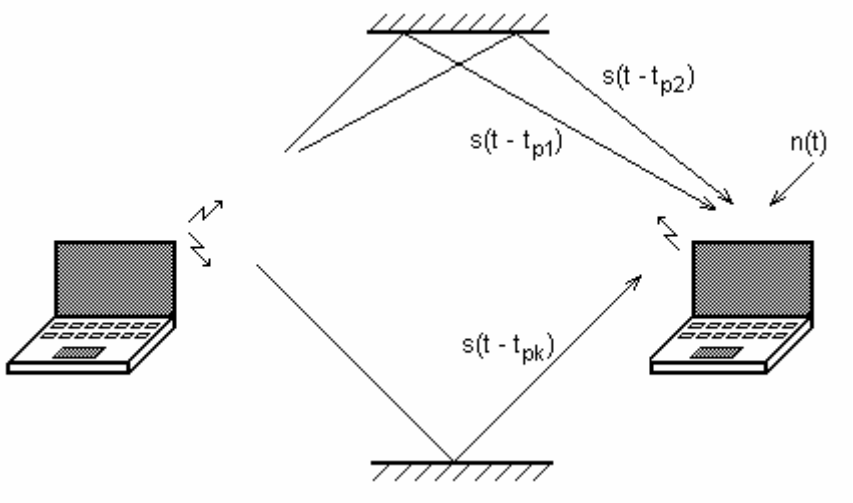


Рисунок 2.6 – Схема распространения радиосигналов в помещении

Кроме этого, приемная антенна воспринимает электромагнитные колебания других источников, суммарный эффект воздействия которых представляет собой аддитивную помеху *n*(t). Результирующее колебание в антенне можно представить в виде суммы *k* сигнальных лучей и аддитивной помехи *n*(t)

$$S_c(t) = \sum_{k=1}^{L} \mu_k s(t - t_{pk}) + n(t), \qquad (2.5)$$

где μ_k – коэффициент затухания *k*-гопути распространения; t_{pk} – время распространения *k*-голуча; *L* – общее количество лучей.

Следствием многолучевой интерференции является искажение принимаемого сигнала, так как сигнал в антенне представляет сумму отдельных колебаний с различными фазами и амплитудами. При многолучевой интерференции выделяют два крайних случая. В первом максимальная задержка между сигналами, прошедшими различный путь, не превосходит времени длительности одного единичного элемента сигнала и интерференция возникает в пределах одного передаваемого элемента. Во втором случае максимальная задержка между различными сигналами больше длительности одного единичного элемента, а в результате интерференции складываются сигна-

лы, представляющие разные символы, и возникает так называемая межсимвольная интерференция (*Inter Symbol Interference*, ISI).

2.1.4. Каналы связи, их типы и иерархии

Как уже упоминалось выше, под каналом связи понимают независимый тракт передачи сигналов от источника к получателю, образованный аппаратурой уплотнения на физической линии путем использования части ресурсов этой линии. Канал также может быть образован путем вторичного уплотнения широкополосного канала.

Для организации каналов связи на линиях первичной сети применяются аналоговые, с частотным разделением каналов (ЧРК), и цифровые, с временным разделением каналов (с ИКМ и дельта-модуляцией) многоканальные системы передачи. Процесс объединения информационных потоков от многих источников в один называется мультиплексированием. Поэтому в зарубежной литературе частотное разделение каналов имеет название *мультиплексирование с частотным разделением*, сокращенно **FDM** (*Frequency Division Multiplexing*), а временное разделение каналов – мультиплексирование с разделением времени, сокращенно **TDM** (*Time Division Multiplexing*).

Основным параметром каналов, образованных аналоговой аппаратурой уплотнения, является *эффективная полоса пропускания*, измеряемая в Гц, а основным параметром цифровых каналов – *пропускная способность*, измеряемая в бит/с.

В основе созданной в нашей стране первичной сети лежит **телефонный** канал, так как исторически сеть связи создавалась для передачи речевых (голосовых) сообщений. При создании многоканальной аппаратуры передачи, обеспечивающей организацию большого числа каналов, оказалось целесообразным наращивать емкости системы последовательной организацией и объединением групп каналов. Это позволило более эффективно использовать каналообразующее оборудование и обеспечить его широкую унификацию.

Аналоговая аппаратура уплотнения с ЧРК позволяет образовывать следующие типовые каналы:

- **канал тональной частоты** (ТЧ) с полосой пропускания то 0,3 до 3,4 кГц;
- **первичный широкополосный** канал с полосой пропускания 60…108 кГц, который состоит из 12 каналов ТЧ, перенесенных в диапазон 60…108 кГц;
- вторичный широкополосный канал, содержащий 5 первичных 12-ти канальных групп (60 каналов ТЧ), перенесенных в диапазон частот 312…552

кГц (фактическая полоса составляет 312,3…551,4 кГц);

- третичный широкополосный канал, состоящий из пяти вторичных групп (300 каналов ТЧ), перенесенных в диапазон частот 812…2044 кГц;
- четверичный широкополосный канал – из трех третичных групп (900 каналов ТЧ), перенесенных в диапазон частот 8516…12388 кГц.

Наряду с устаревшими аналоговыми системами передачи на внутризоновых и местных участках первичной сети широко применяются системы уплотнения с импульсно–кодовой модуляцией (ИКМ). В настоящее время определились следующие типы групп уплотнения:

- **основная цифровая группа** (ОЦ), соответствующая основному каналу ТЧ в аналоговых сетях, скорость передачи 64 кбит/с;
- **первичная цифровая группа**, эквивалентная тридцати двум ОЦ, со скоростью передачи 2048 кбит/с, организована на основе аппаратуры уплотнения ИКМ-30 (30 каналов информационных и 2 канала для служебных целей);
- вторичная цифровая группа, созданная на основе аппаратуры ИКМ-120, скорость передачи в групповом тракте 8448 кбит/с;
- третичная цифровая группа, образованная объединением цифровых потоков четырех вторичных групп, скорость передачи 35 Мбит/с, создана на основе аппаратуры ИКМ-480;
- четверичная цифровая группа, объединяющая цифровые потоки четырех третичных систем, образует цифровой поток со скоростью 139 Мбит/с, что обеспечивает 1920 каналов ТЧ.

Аппаратура уплотнения построена на электронных компонентах, которые, как известно, обладают свойством односторонней передачи сигналов. Поэтом канал связи в принципе передает сигналы только в одном направлении, т.е. является однонаправленным (***симплексным***). Для организации двухстороннего (***дуплексного***) канала используют два разнонаправленных симплексных канала.

Канал двустороннего действия, образованный из двух симплексных каналов, является четырехпроводным, так как передача в разных направлениях осуществляется по двум различным двухпроводным цепям (рисунок 2.7). Такой канал называется каналом однополосной четырехпроводной системы передачи (передача сигналов в обоих направлениях осуществляется в одной полосе частот).

Сигнал, приходящий по двухпроводной линии со станции А, направляется дифференциальной системой (ДС) в двухпроводную линию в направлении передачи (верхняя ветвь схемы), усиливается последовательно усилителями передачи УсПд, промежуточным УсПр и на приемной стороне приемным усилителем УсПм. Затем дифференциальная система направляет его

в двухпроводную линию станции Б и почти полностью подавляет принятый сигнал в направлении обратной двухпроводной линии (нижняя ветвь), устраняя тем самым нежелательную обратную связь.

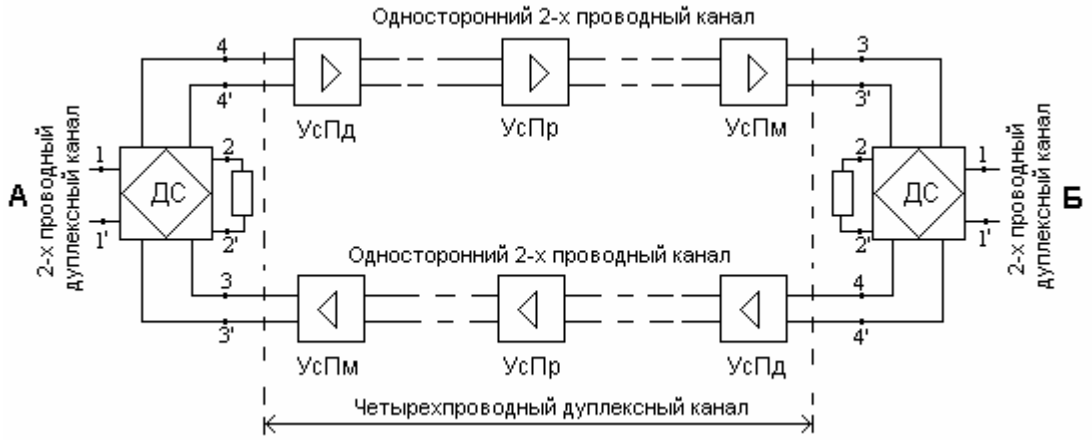


Рисунок 2.7 – Схема четырехпроводного дуплексного канала связи

Дифференциальная система представляет собой шестиполюсник, обладающий различным затуханием в направлениях передачи и приема. Направлением пропускания ДС называются пути передачи сигналов с малым затуханием, а направлением развязки (задерживания) – пути с большим затуханием. Для обеспечения гальванической развязки приемо-передающих устройств с линией (каналом) связи применяются дифференциальные системы трансформаторного типа.

Дифференциальная система (рисунок 2.8,а) представляет собой уравновешенный мост переменного тока. В одно из плеч включается двухпроводная линия связи, а три других образуются линейной $W_Л$ и балансной обмотками $W_Б$ трансформатора и сопротивлением балансного контура $Z_Б$. В диагональ моста 4-4′ включают цепь передачи, а в диагональ 3-3′– приема. Выводы 4-4′ соединяются со входом передающей части, а 3-3′ – с выходом приемника. Дифференциальная система должна вносить минимальное затухание в направлении передачи $a_{1–4}$ и приема $a_{3–1}$, а переходное затухание $a_{3–4}$ должно быть максимальным. Подбирая значение сопротивления балансного контура $Z_Б$, уравновешивают мост и таким образом обеспечивают максимальное переходное затухание $a_{3–4}$.

Обычно при изготовлении дифференциального трансформатора количество витков выполняют $W_Л = W_Б$. Тогда для уравновешивания моста сопротивление $Z_Б$ должно равняться сопротивлению 2-проводной линии связи $Z_Л$. Переходное затухание ДС в децибелах определяется выражением

$$a_{\text{пер}} = a_{4-2} = 20 \lg [(Z_Л + Z_Б) / (Z_Л - Z_Б)] + 20\lg 2.$$

При $Z_Л = Z_Б$ переходное затухание ДС равно бесконечности. Физически это объясняется тем, что ток, поступающий с приемной линии $I_{\text{прм}}$, в точке соединения линейной и балансной обмоток делится пополам, т.е. $i_Л = i_Б$. Так как обмотки катушек трансформатора $W_Л$ и $W_Б$ относительно точки втекания тока $I_{\text{пр}}$ включены встречно (на схеме точками обозначены начала обмоток), то магнитные потоки в сердечнике трансформатора, создаваемые катушками $W_Л$ и $W_Б$ равны по величине и противоположны по направлению и взаимно компенсируются. Следовательно, Э.Д.С., наводимая в выходной обмотке дифсистемы W_4 за счет входного сигнала U_{3-3} будет равна нулю.

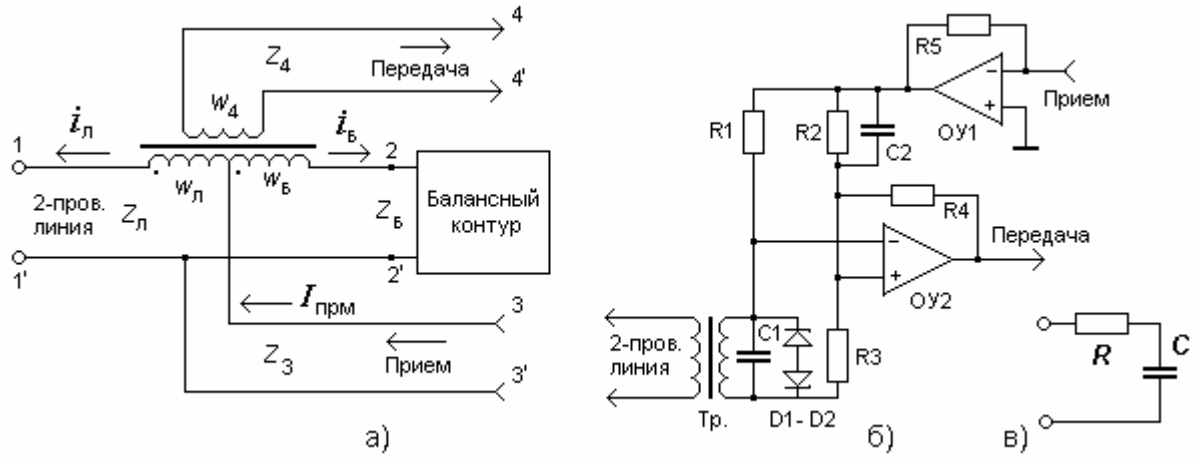


Рисунок 2.8 – Схемы дифференциальных систем

Затухание дифференциальной системы от выводов 1-1´ к выводам 2-2´ равно бесконечности, так как приходящий из двухпроводной линии ток, проходя по линейной обмотке $W_Л$ наводит в балансной обмотке $W_Б$ Э.Д.С. обратного направления, равной практически Э.Д.С. в линейной обмотке. За счет этого результирующий ток в балансном контуре близок к нулю. Поэтому мощность сигнала, приходящего по 2-проводной линии, будет распределяться поровну между Z_3 и Z_4, т. е. затухание в направлении передачи при отсутствии потерь в обмотках трансформатора определится как

$$a_{1-2} = a_{1-4} = 10 \lg (2P_л / P_л) = 10 \lg 2 = 3 \text{ дБ},$$

где $P_л$ – мощность сигнала, поступающего из линии.

В направлении приема затухание a_{3-1} в идеальном случае тоже равно 3 дБ в связи с тем, что мощность поступающего сигнала делится поровну ме-

жду двухпроводной линией и балансным контуром. В действительности, в связи с потерями в катушках трансформатора, затухания a_{1-4} и a_{3-1} несколько больше 3-х децибел.

В реальных системах подобрать величину входного сопротивления балансного контура, равную величине входного сопротивления двухпроводной линии во всем диапазоне рабочих частот практически невозможно. Поэтому в полосе тональных частот 0,3…3,4 кГц величина переходного затухания ДС не превышает 30 дБ. На практике в устройствах передачи сигналов по телефонным каналам схема балансного контура представляет собой цепочку, состоящую из последовательного включения резистора 600 Ом и конденсатора емкостью 1 мкФ (рисунок 2.8,в).

На рисунке 2.8,б показано устройство разделения направлений передачи и приема на основе мостовой схемы, реализованной на резисторах R1, R2, R3 и обмотки трансформатора Тр. Стабилитроны D1,D2 защищают входные цепи от перенапряжений в линии связи.

В высокоскоростных устройствах передачи данных для уменьшения степени воздействия эхо-сигналов на вход приемника дополнительно применяют эхокомпенсаторы. Эхокомпенсатор (ЭК) представляет собой адаптивный фильтр, включаемый между выходом передатчика Прд и входом приемника Прм параллельно дифференциальной системе, разделяющей направления передачи (рисунок 2.9).

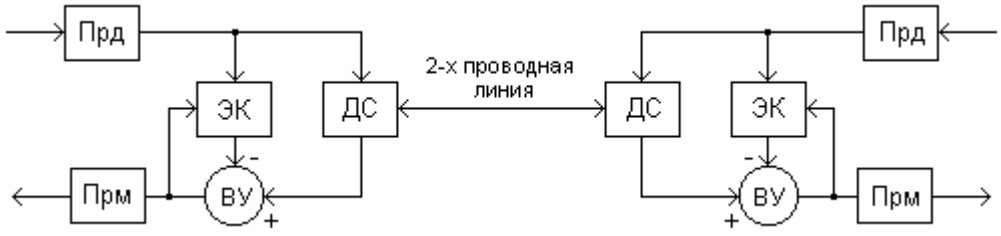


Рисунок 2.9 – Схема дуплексной передачи сигналов по

Коэффициенты фильтра ЭК настраиваются таким образом, чтобы его частотная характеристика повторяла характеристику цепи, по которой сигнал передатчика поступает на вход своего приемника. В результате на входы вычитающего устройства (ВУ) поступают близкие по форме напряжения с выхода фильтра и дифсистемы, которые взаимно компенсируются. При этом сигнал передатчика удаленной станции проходит на вход приемника без изменений.

2.2. Сигналы для передачи по физическим линиям

2.2.1. Основные параметры сигналов и требования к их характеристикам

Сигналом называется физический процесс, однозначно отображающий сообщение. Для представления цифрового сообщения в компьютере применяются импульсы напряжения постоянного тока единичного и нулевого уровней.

К временным параметрам сигнала относится длительность единичного элемента τ_0, для периодической последовательности единичных элементов – период Т и скважность $\alpha = T/\tau_0$ (рисунок 2.10). Количество единичных элементов ***B***, передаваемых в единицу времени, называется скоростью манипуляции. Эта величина получила размерность **бод**.

$$B = 1/\tau_0 . \qquad (2.6)$$

Скорость манипуляции и частота периодической последовательности со скважностью α связаны следующим соотношением: $F = 1/T = 1/\alpha\tau_0 = B/\alpha$.

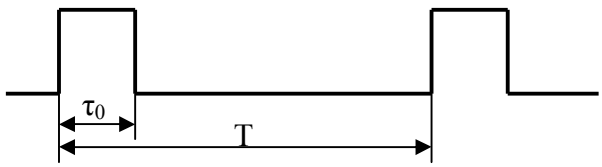


Рисунок 2.10 – Периодическая последовательность сигналов данных

Скорость передачи информации определяется количеством информации *I* (бит), передаваемых в единицу времени. Для многопозиционных равновероятных сигналов с числом значащих позиций (уровней, фаз) равном m_c, вероятность появления *i*-го сигнала $p_i = 1/m_c$. Тогда информационная скорость передачи данных

$$V = I / \tau_0 = \log_2 m_c / \tau_0 = B \log_2 m_c \quad (\text{бит/с}). \qquad (2.7)$$

В системах передачи данных периодическую последовательность единичных элементов (Е. Э.) записывают в виде $\tau_0 : (T - \tau_0)$ или в нормированной форме 1: (α – 1). На рисунке 2.11 и показаны последовательности типа "точки" 1:1 (а) и сигналы вида 1:3 (б).

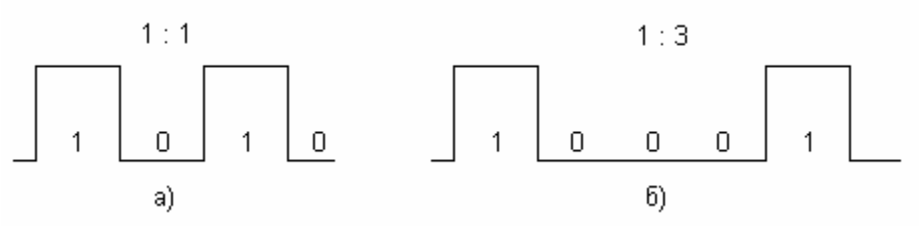


Рисунок 2.11 – Примеры периодических последовательностей различных видов

К энергетическим параметрам сигналов относятся мощность P_c и энергия E_c сигнала, определяемые соответственно по формулам:

$$P_c = U^2_{эф} / R ; \quad \text{при} \quad R=1 \text{ Ом} \qquad P_c = U^2_{эф};$$

$$E_c = \int_0^{\tau_0} U^2_{эф} dt = U^2_{эф} \tau_0 \quad \text{(на единичном сопротивлении).}$$

Спектр периодической последовательности прямоугольных импульсов типа "точки" вида 1:1 амплитудой U_0 с периодом следования T определяется на основании ряда Фурье:

$$C_k = \left| \frac{2}{T} \int_{-\tau_0/2}^{\tau_0/2} U_0 \cos k\Omega t dt \right| = \frac{2U_0}{\alpha} \left| \frac{\sin \frac{\pi k}{\alpha}}{\frac{\pi k}{\alpha}} \right|, \qquad (2.8)$$

$$C_0 = \frac{1}{T} \int_{-\tau_o/2}^{\tau_o/2} U_0 dt = \frac{U_0}{\alpha}.$$

Здесь C_0 – постоянная составляющая; C_k – амплитуда k-й гармоники; Ω – частота первой (основной) гармоники, $\Omega = 2\pi/T$.

Реальный процесс передачи данных представляет собой последовательность независимых и равновероятных кодовых комбинаций, образующих случайную последовательность взаимнонезависимых импульсов с постоянной амплитудой U_0 и одинаковой длительностью τ_0. При этом вероятности появления импульсов со значениями $+U_0$ и $-U_0$ обычно одинаковы. Такой вид последовательности называют случайным телеграфным сигналом. Для оценки спектральных составляющих случайного сигнала используется энергетический спектр $G(\omega)$, определяемый известным соотношением на основе его корреляционной функции $F(\tau)$, которая имеет вид

$$F(\tau) = U_0^2 [1 - (| \tau | / \tau_0)] .$$

Тогда

$$G(\omega) = 2\int_0^{\infty} F(\tau)\cos\omega\tau\, d\tau \;=\; U_0\tau_0\,\frac{\sin^2(\omega\tau_0/2)}{(\omega\tau_0/2)^2}\;. \qquad (2.9)$$

Исходя из вышеизложенного, можно сделать вывод, что спектр сигнала данных содержит бесконечное число гармоник, амплитуда которых с ростом частоты быстро уменьшается. Однако для возможности различения сигналов приемником, как минимум, необходимо обеспечить передачу постоянной составляющей и основной гармоники. В связи с этим минимально необходимая полоса пропускания тракта ΔF передачи должна быть $1/(\alpha\tau_0) = B/2$.

Сигналы, передаваемые по физической линии (ФЛ), должны соответствовать особенностям среды передачи и обеспечивать выполнение требований к линейному тракту передачи цифровых сигналов. Очевидно, что для обеспечения передачи сигналов по ФЛ, ширина их спектра не должна превышать полосы пропускания линии связи. Так как полоса пропускания ФЛ начинается с 0 Гц, то передача данных может осуществляться импульсами постоянного тока.

Физическая линия, как правило, является составной частью кабеля связи, состоящего из большого числа (*пучка*) пар изолированных жил медного провода. Поэтому, при выборе уровня передаваемого сигнала по физической паре необходимо следить за тем, чтобы сигналы данных не создавали переходных помех для систем, использующих другие жилы кабеля. Уровень переходных помех зависит от скорости передачи, амплитуды передаваемых сигналов, а также от величины переходного затухания между жилами кабеля связи. В связи с этим амплитуду сигнала, передаваемого со скоростью 2400 Бод и выше, ограничивают значением 3 В для кабелей с переходным затуханием 87 дБ и величиной 0,3 В – для кабелей с переходным затуханием 69,5 Дб. Кроме этого, при передаче сигналов данных по ФЛ должны также выполняться следующие условия:

- передаваемая по линии цифровая последовательность должна обеспечивать возможность выделения синхронизирующего сигнала в каждом линейном регенераторе и на приемной стороне;
- необходимо обеспечивать возможность постоянного контроля верности передачи информации в линейном тракте без перерыва связи;
- в энергетическом спектре линейного сигнала не должна содержаться постоянная составляющая, а низкочастотные составляющие должны быть незначительными; это позволяет осуществлять дистанционное питание линейных регенераторов по физическим линиям, используемым для передачи линейного сигнала, а также снизить межсимвольные помехи в регенераторе,

возникающие из-за подавления низкочастотных составляющих в спектре сигнала данных;

- спектр линейного сигнала должен быть компактным и с низким уровнем высокочастотных составляющих; сокращение полосы частот позволяет увеличить длину участка регенерации, а ослабление высокочастотных составляющих снижает переходные влияния между цепями кабеля;
- должна обеспечиваться возможность безошибочной передачи произвольного числа следующих подряд друг за другом импульсов или пробелов.

Для получения ансамбля линейных сигналов, удовлетворяющих вышеизложенным требованиям, осуществляют преобразования входной последовательности данных по определенным правилам. Эта процедура называется ***линейным кодированием***.

2.2.2. Простые сигналы для передачи данных по физическим линиям

Если между передающей и принимающей станциями проложена кабельная (физическая) линия и имеется гальваническая связь между передатчиком и приемником, то данные можно передавать однополярными или двуполярными импульсами постоянного тока прямоугольной или иной формы. Применение разнополярных (*биполярных*) импульсов является предпочтительнее в связи с более высокой помехоустойчивостью разнополярных сигналов. Разнополярные сигналы называют еще сигналами без возврата к нулю – **NRZ**-сигналами (*Non Return to Zero*), так как на протяжении длительности тактового интервала не происходит перехода (возврата) сигнала к нулю. Вид сигналов показан на диаграмме (рисунок 2.12,а).

Метод NRZ прост в реализации, обладает высокой помехоустойчивостью, однако выделить из NRZ-последовательности тактовые импульсы при определенных условиях затруднительно. Так, при передаче длинной последовательности нулей или единиц сигнал в линии не изменяется, в связи с чем устройство синхронизации по единичным элементам приемника лишено возможности определить по входной последовательности моменты смены посылок и подстроить под них фазу тактовых импульсов собственного генератора.

Другим серьезным недостатком сигналов NRZ является наличие постоянной составляющей, которая приближается к нулю только при передаче длинных последовательностей типа “точки”. В результате сигналы NRZ в чистом виде в современных системах связи практически не применяются. Однако используются его различные модификации, которые формируются путем линейного кодирования.

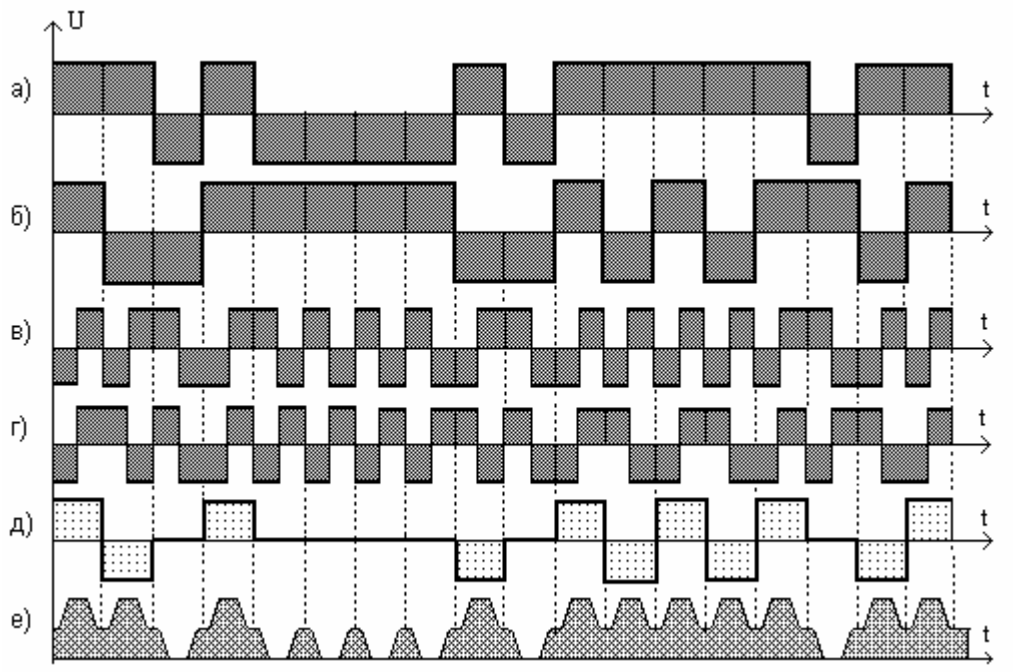


Рисунок 2.12 – Виды сигналов для передачи данных по физическим линиям: а) биполярные импульсы (NRZ); б) без возврата к нулю с инверсией при единице (NRZI); в) манчестерский код; г) дифференциальный манчестерский код; д) *AMI*-сигналы; е) квазитроичные сигналы для оптических линий

На практике чаще находит применение способ передачи без возврата к нулю с инвертированием предыдущей посылки при передаче очередной единицы данных **NRZI** (*Non Return to Zero with ones Inverted*). При поступлении на вход передатчика логического нуля в линию выдается сигнал, равный предыдущей посылке. При таком способе в случае наличия во входном потоке длинных единичных последовательностей на выходе передатчика происходит постоянное изменение полярности сигналов (рисунок 2.12,б). Способ NRZI улучшает условия синхронизации приемника и уменьшает постоянную составляющую сигнала. Поскольку код NRZI не защищен от долгих последовательностей "нулей", то это может привести к сбоям синхронизации. Поэтому перед передачей, заданную последовательность битов рекомендуется предварительно скремблировать.

Следует заметить, что для передачи сигналов по оптическим линиям используются обычные двоичные сигналы ("1" – есть свет, "0" – нет). Ширина полосы пропускания оптоволоконных линий настолько широка, что можно без опасения расширять спектр оптического сигнала путем введения дополнительных битов, обеспечивающих тактирование передаваемых сигналов и требуемую избыточность для исправления ошибок.

Весьма широко для передачи данных по ФЛ применяют **манчестерское** кодирование. При этом способе логические "0" и "1" передаются на протяжении единичного интервала двумя разнополярными импульсами. Смена полярности происходит в середине единичного интервала сигнала данных. Направление перехода определяет передаваемое двоичное значение информационного сигнала. Так, передача "1" производится биимпульсом 01, а передача "0" – 10 (рисунок 2.12,в). Очевидно, что независимо от длины последовательностей "0" или "1" в линии всегда имеет место смена полярности линейного сигнала. Недостатком такого метода является расширение спектра сигнала за счет уменьшения длительности передаваемых импульсов. А это приводит к увеличению затухания сигнала и соответственно уменьшению дальности передачи, а также к увеличению переходных помех в кабеле.

Другим вариантом передачи кодовых элементов биимпульсами является **дифференциальное манчестерское кодирование** (рисунок 2.12,г). Суть преобразования сигналов аналогична способу NRZI, т.е. "0" кодируется биимпульсом, совпадающим с предыдущим, а при "1" сигнал меняется на инверсный. Синхронизирующие свойства этого кода такие же как и у обычного манчестерского кодирования, однако, помехоустойчивость, как и у всех дифференциальных способов, выше.

Одной из модификаций метода NRZ является метод биполярного кодирования с альтернативной инверсией единицы, так называемый **AMI**-метод (*Alternative Mark Inversion*). Получаемый линейный код является *квазитроичным*, так как для передачи сигнала единицы (англ. *Mark*) используется положительный либо отрицательный импульс, а для передачи нуля (англ. *Space*) – нулевой потенциал. При этом полярность каждой новой единицы противоположна полярности предыдущей (рисунок 2.12,д). Постоянная составляющая в AMI-сигнале несколько уменьшена и частично решается проблема выделения тактовых сигналов при длительных последовательностях единиц. Однако при длинных последовательностях нулей устройство тактовой синхронизации приемника AMI-сигналов не имеет возможности выделять тактовые импульсы на основании передаваемого сигнала.

На рисунке 2.12,е изображены квазитроичные сигналы для передачи по оптическим линиям. При отсутствии передачи в линии устанавливается уровень, равный половине максимального. При передаче логической единицы уровень света становится максимальным, а при передаче логического нуля – свет в линии отсутствует. Для улучшения синхронизационных свойств на каждом единичном интервале осуществляется возврат к половинному значению уровня света.

2.2.3. Сигналы с улучшенными синхронизирующими свойствами

Для улучшения процедуры формирования на приемной стороне тактовых импульсов на основе входных информационных сигналов разработаны линейные коды вида CHDB (*Compatible High Density Binary*). После следования *n* периодов значения "0" они обеспечивают обязательную смену полярности сигнала. Так код CHDB3 (обычно называемый просто **HDB3**) предполагает, что после трех "0" в линию связи обязательно передается импульс. Для того чтобы на приемной стороне он не был воспринят как единица, применяется *нарушение* правила перехода, которое требует обязательного чередования положительных и отрицательных импульсов. Поэтому при кодировании по методу HDB3 после трех значений “0” передается импульс того же знака (так называемый V-импульс), что и у последнего импульса, представлявшего значение “1” (рисунок 2.13).

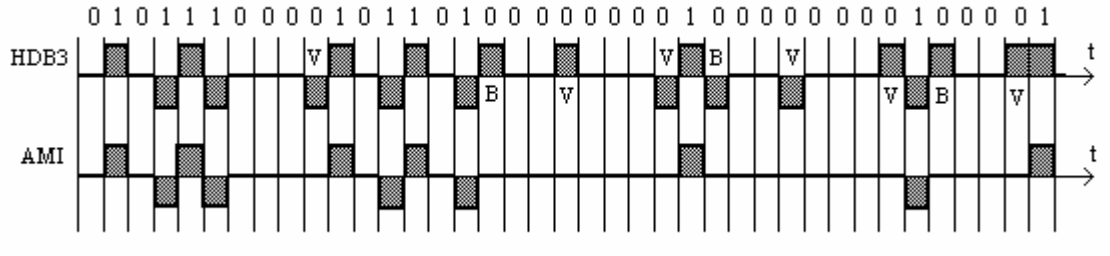


Рисунок 2.13 – Принцип формирования цифровых сигналов HDB3

Однако в связи с введением дополнительного импульса в линии возникает постоянная составляющая. Чтобы обеспечить смену полярностей следующих друг за другом дополнительно вводимых импульсов, производится замена первого нуля группы четырех "0" так называемым B-битом, полярность которого противоположна полярности предшествующего линейного импульса. Приемник декодирует группу B00V как четыре нулевых элемента. В линии связи группа B00V чередуется с последовательностью 000V. На рисунке 2.13 для сравнения показана также последовательность AMI-сигналов. Любая одиночная ошибка при использовании HDB3-сигналов либо создает новое нарушение чередования полярностей, либо уничтожает ранее введенное нарушение этого закона. В том и другом случаях возникает некомпенсированное нарушение полярностей сигнальных импульсов, что сравнительно просто обнаруживается устройствами контроля на приемной стороне.

Похожий способ линейного кодирования используется в цифровых системах передачи США, который получил название **B8ZS** (*Bipolar with 8 Zeros Substitution*), где 8 нулей кодируются последовательностью 00B0VB0V.

В цифровых системах передачи данных широко используются методы линейного кодирования, которые обозначаются в общем виде **xByB**, **xByT** или **xByQ**. Их суть состоит в том, что группа, состоящая из ***x*** битов (B–*binary*), заменяется группой ***y*** троичных (**T**-*ternary*), четверичных (**Q**-*quaternary*) или двоичных (**B**) элементов. Так, например, в локальных компьютерных сетях Fast-Ethernet 100BASE-FX и сетях FDDI применяется преобразование кодов вида **4B/5B**. При таком кодировании из 32–х возможных двоичных комбинаций выбираются только 16, в которых имеется максимально возможное число смены позиций двоичных элементов. Этим достигается более равномерное распределение спектральных составляющих сигнала, а также обеспечивается высокая частота смены его позиций, что облегчает процесс тактовой синхронизации. При высокоскоростной передаче по оптическим линиям также применяется код **8B10B**, в котором полностью устранена постоянная составляющая. Применение этого кода не только улучшает процесс синхронизации, но и исключает перегрев лазерного диода при поступлении от источника многих "единиц" подряд.

В коде **4B3T** (таблица 2.1) производится замена четырех битов двоичной последовательности комбинацией, состоящей из трех троичных (*тернарных*) элементов (+, 0 и –). В этом коде для передачи 16 двоичных комбинаций может быть использовано 3^3 =27 комбинаций из трех троичных символов. Повышение избыточности применяется для защиты от ошибок и улучшения условий синхронизации. Скорость манипуляции в линии уменьшается при этом на 25%, соответственно снижается затухание сигнала в линии связи, которое пропорционально корню квадратному из частоты передачи сигналов.

Комбинация вида 000 используется для передачи синхронизирующей информации в линейном сигнале. Из оставшихся 26 комбинаций шесть имеют цифровую сумму, равную 0, десять комбинаций имеют положительную сумму (от +1 до +3) и десять – отрицательную сумму (от –1 до –3).

Чтобы достичь минимально возможной частоты передачи (помните, что $F_П = 1/\alpha\tau_0 = B/\alpha$; здесь α – скважность, B–скорость манипуляции) и предельно уменьшить постоянную составляющую, в передаваемой последовательности сигналов для кодирования и декодирования используется не одна, а четыре кодовых таблицы (таблица 2.1), которые переключаются в зависимости от предыстории передачи. Каждой из таблиц приписывается определенный статус: S1...S4. После получения от источника очередной четырехбитовой комбинации осуществляется передача в линию тройки трехпозиционных элементов соответствующей этой комбинации и текущему статусу таблицы. Затем происходит переключение таблицы в другой статус. Переход в другой статус S_{i+1} зависит от предыдущего статуса S_i и текущей цифровой суммы троичной группы.

Таблица 2.1 – Таблица кодирования линейного кода 4B3T

4-х битовое слово	16-ричное значение	Троичное слово и следующий статус таблицы S_{i+1} (CCT)							
		Статус S1	CCT	Статус S2	CCT	Статус S3	CCT	Статус S4	CCT
0000	0h	+ 0 +	3	0 – 0	1	0 – 0	2	0 – 0	3
0001	1h	0 – +	1	0 – +	2	0 – +	3	0 – +	4
0010	2h	+ – 0	1	+ – 0	2	+ – 0	3	+ – 0	4
0011	3h	0 0 +	2	0 0 +	3	0 0 +	4	– – 0	2
0100	4h	– + 0	1	– + 0	2	– + 0	3	– + 0	4
0101	5h	0 + +	3	– 0 0	1	– 0 0	2	– 0 0	3
0110	6h	– + +	2	– – +	3	– – +	2	– – +	3
0111	7h	– 0 +	1	– 0 +	2	– 0 +	3	– 0 +	4
1000	8h	+ 0 0	2	+ 0 0	3	+ 0 0	4	0 – –	2
1001	9h	+ – +	2	+ – +	3	+ – +	4	– – –	1
1010	Ah	+ + –	2	+ + –	3	+ – –	2	+ – –	3
1011	Bh	+ 0 –	1	+ 0 –	2	+ 0 –	3	+ 0 –	4
1100	Ch	+ + +	4	– + –	1	– + –	2	– + –	3
1101	Dh	0 + 0	2	0 + 0	3	0 + 0	4	– 0 –	2
1110	Eh	0 + –	1	0 + –	2	0 + –	3	0 + –	4
1111	Fh	+ + 0	3	0 0 –	1	0 0 –	2	0 0 –	3

Так, например, если предыдущий статус был S_1 и текущая цифровая сумма последовательности равна нулю, то переключение таблицы не происходит; при текущей сумме +1 происходит переключение в S_2; при сумме +2 – в S_3 и при сумме +3 кодер переключается на таблицу со статусом S_4. Закономерность перехода из других состояний проследите по таблице 2.1 самостоятельно. Аналогично происходит переключение таблиц и на приемной стороне.

Способ MLT-3 (*Multi Level Transmission* -3) также относится к линейному кодированию типа xByT. Он похож на NRZ, но в отличие от него имеет три значения сигнала. Единице соответствует переход с одного уровня сигнала на другой, причем изменение уровня сигнала происходит последовательно с учетом предыдущего перехода. При передаче “нуля” сигнал не меняется. Он применяется, например, в локальных сетях 100BASE-TX для передачи сигналов 5-битовых комбинаций кода 4B/5B.

2.2.4. Многопозиционные сигналы

Для повышения дальности передачи сигналов по физическим линиям необходимо выбирать сигналы с меньшей шириной спектра. В современных компьютерных сетях с этой целью применяется линейное кодирование вида **2B1Q**, в котором двум бинарным элементам соответствует один четверичный сигнал, принимающий один из четырех возможных уровней напряжения: +1, –1, +3 и –3. Этот код позволяет в два раза снизить скорость манипуляции сигналов и тем самым увеличить дальность передачи без промежуточной ретрансляции примерно в 1,5 раза.

Таблица преобразования дибитов в четырехпозиционный сигнал и пример передаваемой последовательности кодом 2B1Q показаны на рисунке 2.14, а,б соответственно. В таблице преобразования приведены также значения амплитуды соответствующих четырехпозиционных сигналов.

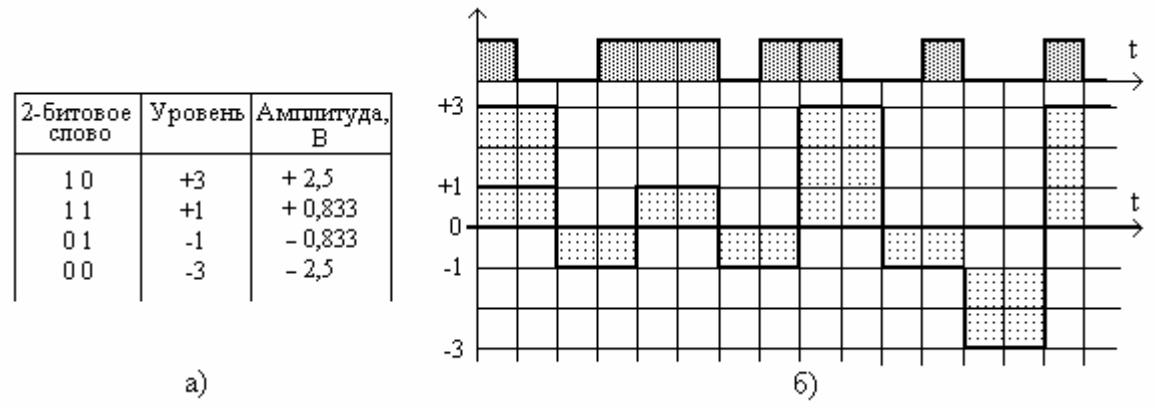


Рисунок 2.14 – Линейное кодирование сигналов данных методом 2B1Q

В компьютерных сетях Fast-Ethernet 100BASE-T2 и Gigabit-Ethernet 1000BASE-T реализована 5-ти уровневая амплитудно-импульсная модуляция (**PAM-5**) в которой используются пять значений состояний (–2, –1, 0, +1, +2). При этом уровням +2/–2 соответствует напряжение +1/–1 В, а уровням +1/–1 — напряжение +0,5/–0,5 В. Каждое состояние соответствует одной из двухбитовых комбинаций. Пятое значение уровня применяется для обнаружения ошибок.

2.3. Сигналы для передачи по каналам связи

2.3.1. Необходимость преобразования спектров сигналов

Каналы связи образуются аппаратурой уплотнения (мультиплексирования) на линиях связи путем использования части ресурсов этих линий. Отличительной особенностью каналов является их **полосовой характер**, т.е. они пропускают колебания в определенной полосе частот. Так, например, канал тональной частоты имеет полосу пропускания 3100 Гц с регламентированной средней частотой 1900 Гц, первичный широкополосный канал занимает полосу пропускания от 60 до 108 кГц. Каналы беспроводной связи компьютерных сетей используют полосу частот в несколько десятков мегагерц со средней частотой около 2,4 ГГц. В то же время основная энергия сигналов данных расположена от нулевой частоты (постоянной составляющей) до частоты равной половине скорости манипуляции, т.е. $F = 1/2\tau_0 = B/2$. Отсюда следует, что большинство каналов связи являются "непрозрачными" для сигналов данных. Для возможности передачи таких сигналов по каналу связи необходимо перенести спектр сигнала данных в полосу пропускания канала связи. Перенос (*транспонирование*) спектра осуществляется с помощью процедуры модуляции.

Модуляцией называется процесс изменения одного или нескольких параметров вспомогательного колебания по закону модулирующего колебания. В качестве вспомогательного колебания u(t) обычно используется гармоническое колебание высокой частоты, которое называют *несущей* частотой (*carrier*) вида

$$u(t) = U_m\cos(\omega_o t + \varphi_o). \qquad (2.10)$$

Параметрами несущего колебания являются амплитуда U_m, круговая частота ω_o и начальная фаза φ_o. Каждый из этих параметров можно изменять и получить соответственно амплитудную (АМ), частотную (ЧМ) и фазовую (ФМ) модуляцию. При этом данный параметр несущей имеет приращение Δ, пропорциональное модулирующему сигналу $f(t)$. Так,

при АМ $U = U_м + \Delta U\, f(t)$,
при ФМ $\varphi = \varphi_o + \Delta\varphi\, f(t)$,
при ЧМ $\omega = \omega_o + \Delta\omega\, f(t)$.

Для цифровых сигналов модулирующая функция $f(t)$ принимает значения (0,1) или (+1, –1). Можно изменять несколько параметров несущей одновременно, например, амплитуду и фазу или частоту и фазу. Соответственно по-

лучим амплитудно-фазовую (АФМ) или частотно-фазовую (ЧФМ) модуляцию.

Временные диаграммы различных видов модуляции показаны на рисунке 2.15. Роль модулирующего колебания в процессе модуляции выполняет информационный сигнал, спектр которого необходимо перенести в полосу пропускания канала.

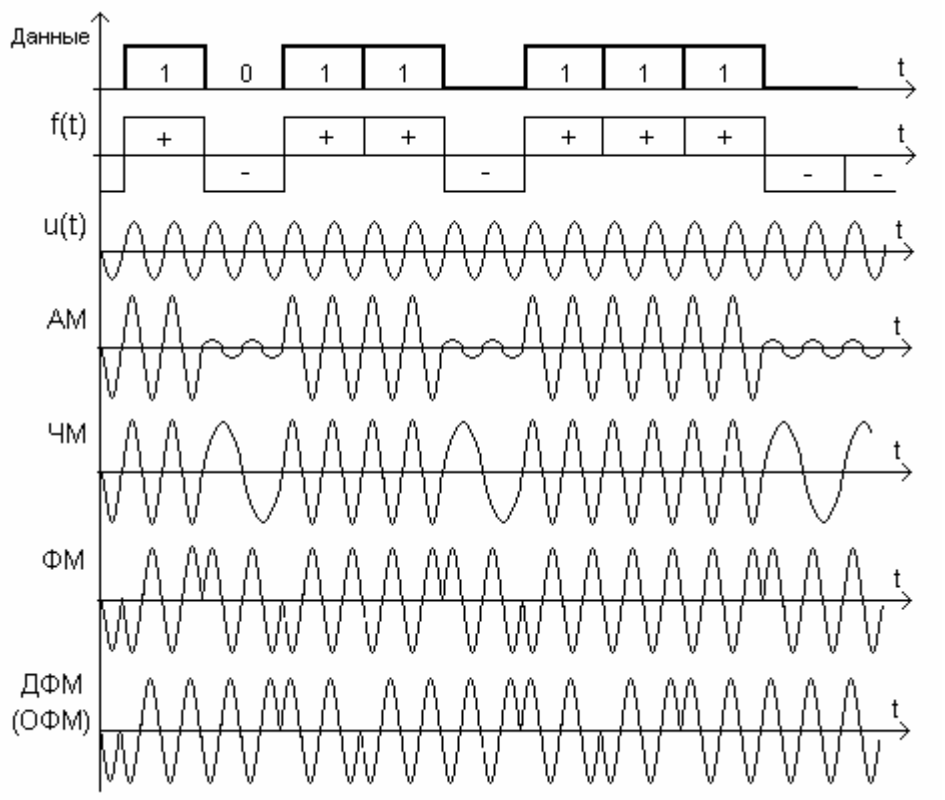


Рисунок 2.15 – Временные диаграммы различных видов манипуляции сигналов

В системах передачи данных модулирующим колебанием является последовательность дискретных импульсов. Скачкообразное изменение параметров несущего колебания называют также *манипуляцией* или *телеграфией*. Для различения сокращенных обозначений модуляции от манипуляции зачастую амплитудная, частотная и фазовая манипуляция обозначается соответственно АТ (амплитудная телеграфия), ЧТ и ФТ.

Различают абсолютную (ФМ) и относительную (ОФМ) фазовую модуляцию. ОФМ носит также название дифференциальная фазовая модуляция (ДФМ). При абсолютной двухпозиционной фазовой манипуляции (англ. обозначение **BPSK** – *Binary Phase Shift Keying*) фаза модулированного колебания при значении входного сигнала равного уровню логического "0" совпа-

дает со значением фазы опорного (*несущего*) напряжения ($\Delta\varphi=0^0$), а при поступлении "1" – меняется на противоположную ($\Delta\varphi=180^0$), т.е. фаза модулированного колебания меняется всякий раз при изменении значения входного сигнала.

В случае дифференциальной фазовой манипуляции ДФМ (англ. **DPSK** – *Differential Phase Shift Keying*), фаза текущего колебания изменяется не по отношению к опорному колебанию, а по отношению к фазе предыдущей посылки. Из временной диаграммы видно, что скачкообразное изменение фазы модулированного колебания на 180^0 происходит в случае абсолютной фазовой модуляции при каждом изменении знака модулирующего сигнала, а при относительной (дифференциальной) – каждом единичном значении сигнала данных.

2.3.2. Амплитудно-модулированные сигналы

При *амплитудной модуляции* амплитуда модулированного колебания изменяется по закону модулирующего сигнала *f(t)*. В цифровых системах передачи данных при модуляции прямоугольными посылками *f(t)*= ±1. Если амплитуду модулирующего сигнала обозначить ΔU, то амплитуда модулированного напряжения будет изменяться по закону

$$U_1 = U_m + \Delta U f(t) = U_m\left[1+\frac{\Delta U}{U_m} f(t)\right] = U_m\left[1 + m_{ам} f(t)\right], \quad (2.11)$$

где $m_{ам}$ – коэффициент модуляции ($m_{ам}= \Delta U/U_m$). Модулированный сигнал запишется так:

$$u_{ам}(t) = U_m\,[1 + m_{ам} f(t)]\cos(\omega_o t + \varphi_o). \quad (2.12)$$

Для наиболее часто применяемой стопроцентной модуляции ($m_{ам}$=1):

$$U_{ам}(t) = U_m\,[1 + f(t)]\cos(\omega_o t + \varphi_o). \quad (2.13)$$

Спектр сигнала для периодической последовательности прямоугольных посылок с единичной амплитудой определяется на основании преобразования Фурье

$$u_{ам}(t) = \frac{U_m}{\alpha}\cos(\omega_0 t + \varphi_0) +$$

$$+\frac{U_m}{\alpha}\sum_{k=1}^{\infty}\frac{\sin(\frac{\pi k}{\alpha})}{\frac{\pi k}{\alpha}}\{\sin[(\omega_0+k\Omega)t+\varphi_0]+\sin[(\omega_0-k\Omega)t+\varphi_0]\}. \quad (2.14)$$

Здесь $\Omega = 2\pi/T$– круговая частота повторения посылок; T – период следования единичных элементов; α – скважность единичных элементов, равная отношению периода к длительности единичного элемента, $\alpha = T/\tau_o$.

Энергетический спектр АМ сигнала $G(\omega)$ определяется на основании его корреляционной функции и имеет вид

$$G_{ам}(\omega)=\frac{U_m^2}{2}+\frac{U_m^2\tau_0}{2}\frac{\sin^2[(\omega-\omega_0)\tau_0/2]}{[(\omega-\omega_0)\tau_0/2]^2}. \quad (2.15)$$

Из (2.14 и 2.15) следует, что спектр амплитудно-манипулированного сигнала располагается в области частоты вспомогательного колебания и содержит несущую частоту и две боковые полосы: верхнюю и нижнюю. Форма боковых частот спектра манипулированного сигнала аналогична форме спектра модулирующих посылок. Спектр модулированного сигнала получается вдвое шире спектра сигнала данных.

2.3.3. Сигналы с фазовой модуляцией

В случае фазовой модуляции (*Phase Shift Keying*, **PSK**) при изменении модулирующего сигнала по закону *f(t)* и максимальном изменении начальной фазы на величину $\Delta\varphi$ фаза сигнала изменяется по закону:

$$\theta = \omega_o t + \varphi_o + \Delta\varphi f(t). \quad (2.15)$$

Мгновенное значение фазомодулированного напряжения имеет вид:

$$U_{фм}(t) = U_m \cos\theta = U_m \cos[\omega_o t + \varphi_o + \Delta\varphi f(t)], \quad (2.16)$$

где $\Delta\varphi$ – **девиация фазы** или, как еще ее называют, *индекс фазовой модуляции*.

При фазовой манипуляции прямоугольными периодическими посылками при $f(t) = \pm 1$ спектр сигнала вычисляется по формуле:

$$u_{ФМ}(t) = U_m\left[\cos(\omega_0 t+\varphi_0)\cos\Delta\varphi - \sin\Delta\varphi\sum_{k=1}^{\infty}\frac{\sin\frac{\pi k}{2}}{\frac{\pi k}{2}}\cos(k\Omega t)\sin(\omega_0 t+\varphi_0)\right] =$$

$$= U_m\cos\Delta\varphi\cos(\omega_0 t+\varphi_0) + \frac{1}{2}\sum_{k=1}^{\infty}\frac{U_m\sin\Delta\varphi\sin\frac{\pi k}{2}}{\frac{\pi k}{2}}\sin(\omega_0 t + k\Omega t+\varphi_0) +$$

$$+\frac{1}{2}\sum_{k=1}^{\infty}\frac{U_m\sin\Delta\varphi\sin\frac{\pi k}{2}}{\frac{\pi k}{2}}\sin(\omega_0 t - k\Omega t+\varphi_0). \qquad (2.17)$$

Таким образом, в общем случае спектр ФМ-колебания содержит несущую, симметрично от которой располагаются боковые составляющие, отстоящие на частотные интервалы, кратные частоте манипуляции. При фазовой модуляции периодической последовательностью прямоугольных посылок имеет место следующее:

- ширина спектра фазовой манипуляции равна ширине спектра амплитудной манипуляции и не зависит от индекса модуляции;
- амплитуды боковых частот ФМ-сигнала отличаются от таковых при АМ на величину $sin\Delta\varphi$.

Для увеличения информационной скорости на практике широко применяется многопозиционная фазовая модуляция с 4, 8 и 16 значениями сдвига фазы. При 4-позиционной модуляции ($m_c = 4$) последовательности битов, поступающих от источника, объединяются по два (в *дибиты*). Дибитам соответствуют разности фаз двух соседних посылок сигнала 0^0, 90^0,180^0 или 270^0. С целью улучшения условий синхронизации по единичным элементам фазы посылок смещают на 45^0, т.е. фазовые углы ФМ-сигналов могут принимать значения 45^0, 135^0, 225^0 и 315^0. Для отображения многопозиционных сигналов с ФМ широко используются фазовые диаграммы с представлением элементов сигнала в виде векторов, длина которых равна амплитуде посылки, а углы поворота – разности фаз (рисунок 2.16). Фазовые диаграммы часто называют *сигнальным созвездием*.

При 8- и 16-позиционной ФМ информационный поток разделяется на группы соответственно по три (*трибиты*) или четыре бита – (*квадрибиты*). Фазовые углы между соседними векторами в первом случае отличаются на 45^0 (рисунок 2.3.2,б), а во втором – на $22{,}5^0$.

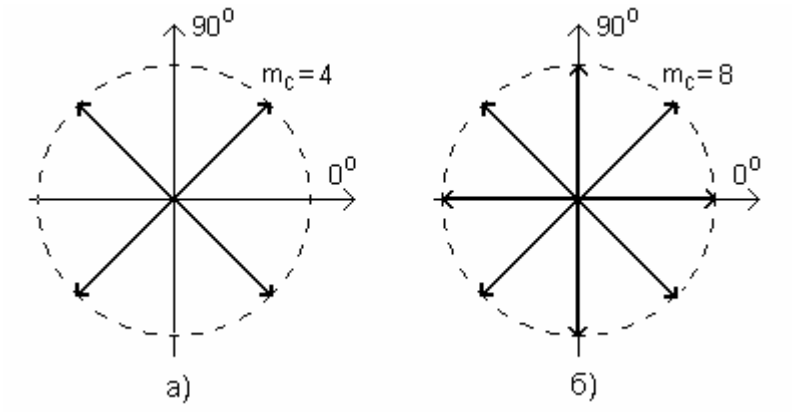


Рисунок 2.16 – Сигнальные созвездия для ФМ-4 и ФМ-8

Информационная скорость при многопозиционной передаче увеличивается в $\log m_c$ раз (см. формулу (2.7), т.е. для четырехпозиционной манипуляции скорость передачи в два раза выше скорости манипуляции, а при 16–позиционной эта скорость возрастает в четыре раза.

Формирование ФМ-сигналов со сдвигом фазы на 180^0 легко осуществляется путем инвертирования колебаний генератора несущей частоты. Для получения модулированных колебаний с числом позиций фаз больше двух используют два колебания, имеющих одинаковую частоту, но сдвинутых по фазе на 90^0, т.е. находящихся в квадратуре. В этом случае говорят о так называемой *квадратурной* фазовой модуляции (*Quadrature Phase Shift Keying*, **QPSK**).

Модуляция QPSK является частным случаем квадратурной амплитудной модуляции QAM-4, при котором информационный сигнал кодируется изменением фазы несущего колебания с шагом 90^0.

Аналитически QAM-сигнал представляется в виде

$$u_{KAM}(t)=U_m\left[A(t)\cos\omega_0 t+B(t)\sin\omega_0 t\right], \qquad (2.18)$$

где A(t) и B(t) – модулирующие сигналы в квадратурном и синфазном каналах соответственно.

В передатчике, производящем модуляцию, одна из этих составляющих синфазна колебанию генератора несущей частоты, а вторая находится в квадратуре по отношению к этому колебанию (отсюда — квадратурная модуляция). Синфазная составляющая обозначается зачастую как ***I*** (*In Phase*), а квадратурная — как ***Q*** (*Quadrature*). Входной битовый поток преобразуется в кодирующую последовательность $\{d_k\}$ так, что логическому нулю соответствует кодирующий бит +1, а логической единице – кодирующий бит –1. После этого кодирующий поток разделяется на четные и нечетные биты.

Четные биты поступают в *I*-канал, а нечетные — в *Q*-канал. Причем, длительность каждого управляющего импульса d_i и d_q в два раза больше длительности единичного элемента сигнала данных d_k.

Для цифровой фазовой манипуляции характерно, что при модулировании синфазной и квадратурной составляющей несущего колебания используется одно и то же значение величины *изменения амплитуды*. Поэтому окончания векторов модулированного колебания образуют прямоугольную сетку на фазовой плоскости действительной $Re\{U_{кам}\}$ и мнимой составляющей вектора модулированного сигнала $Im\{U_{кам}\}$. Число узлов этой сетки определяется количеством позиций результирующего сигнала, т.е. типом используемого алгоритма QAM. Схема расположения узлов на фазовой плоскости модулированного QAM-колебания представляет другую форму изображения созвездия сигналов. На рисунке 2.17 показана упрощенная структурная схема формирователя QAM-сигнала.

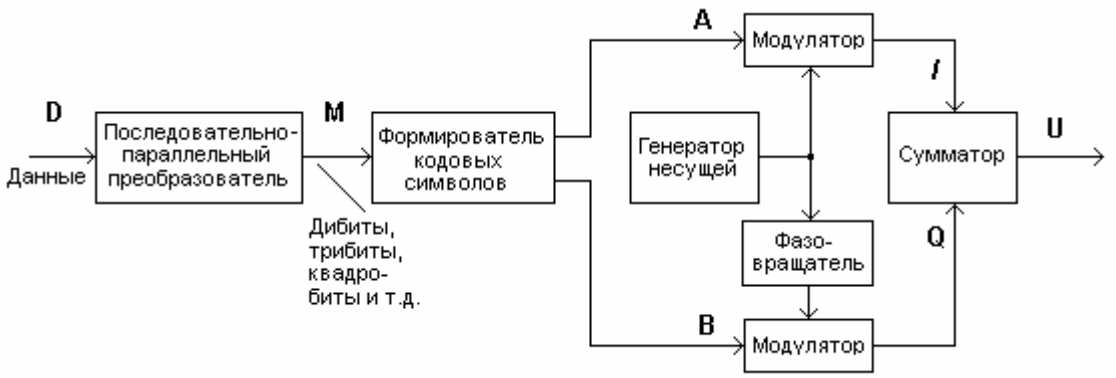


Рисунок 2.17 – Схема формирователя QAM–сигнала.

Генератор несущей частоты формирует гармонические колебания, которые подаются на модулятор синфазного канала. Эти же колебания сдвигаются фазовращателем по фазе на 90^0 и поступают на модулятор квадратурного канала. На первом этапе преобразования поток входных данных $\boldsymbol{D}\{d_0, d_1, \ldots d_k\}$, поступающих от источника сигнала, преобразуется в последовательность групп битов $\boldsymbol{M}\{m_0, m_1, \ldots m_j\}$.

Число битов в этой группе равно $\log m_c$. Формирователь кодовых символов преобразует группу битов в пару кодовых символов a_j и b_j. Так, например, для алгоритма QAM-16 стандартом установлены значения a_j и b_j, принадлежащие множеству {1, 3, –1, –3}, а для QAM-64 a_j и b_j могут принимать значения {1, 3, 5, 7, –1, –3, –5,–7}.

Величины a_j и b_j и определяют соответственно значения реальной и мнимой координаты вектора модулированного колебания. Сформированные значения $\boldsymbol{A}$ $\{a_j\}$ и $\boldsymbol{B}$ $\{b_j\}$ используются для амплитудной модуля-

ции синфазной ***I*** и квадратурной ***Q*** составляющих несущего колебания. На последнем этапе преобразования выполняется суммирование этих колебаний и формирование результирующего сигнала ***U***. Сигнальные созвездия для QAM-16 и QAM-64 изображены на рисунке 2.18.

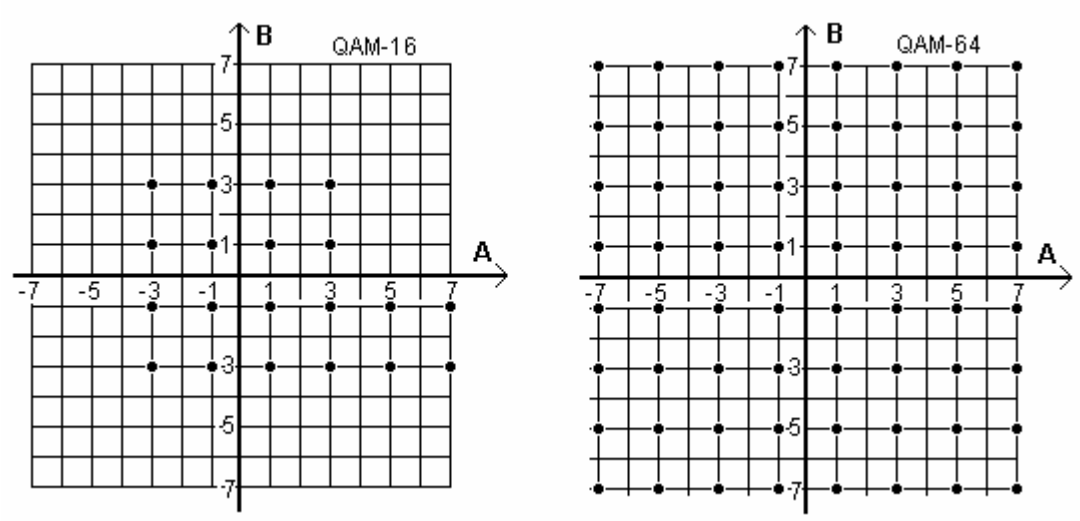


Рисунок 2.18 – Сигнальные созвездия для квадратурной амплитудной манипуляции QAM–16 и QAM–64

В случае QPSK-модуляции сигнальное созвездие состоит из четырех точек с координатами (+1, +1); (+1, –1); (–1, +1); (–1, –1). Эти четыре точки соответствуют четырем возможным дибитам.

На практике преобразование модуляционных символов в кодовые символы выполняется с применением алгоритмов Грея для помехоустойчивого кодирования данных. Так, векторам модулированного колебания, которые находятся близко один от другого на фазовой плоскости, ставятся в соответствие значения кодовых символов, которые отличаются значениями только одного бита. Например, для сигналов с координатами{1, 1} и {1, 3} приписывают кодовые символы {0, 0} и {0, 1}.

Несмотря на кажущуюся простоту метода фазовой модуляции ему присущи некоторые недостатки, связанные с трудностями технической реализации. Один из недостатков связан с тем, что в случае квадратурной фазовой модуляции при одновременной смене символов в обоих каналах модулятора (с +1, –1 на –1, +1 или с +1, +1 на –1, –1) в сигнале QPSK происходит скачок фазы на 180°. Такие скачки фазы, имеющие место и при обыкновенной двухфазной модуляции, вызывают паразитную амплитудную модуляцию огибающей сигнала. В результате этого при прохождении сигнала через приемный узкополосный фильтр возникают провалы огибающей почти до нуля. Такие изменения сигнала весьма нежелательны, поскольку приводят к

расширению его спектра и тем самим к созданию помех для других каналов.

Для того чтобы избежать этого негативного явления, прибегают к так называемой квадратурной фазовой модуляции со сдвигом (*Offset QPSK*, **OQPSK**). При таком типе модуляции формирование сигнала в квадратурной схеме происходит так же, как и в модуляторе QPSK, за исключением того, что кодирующие биты в Q-канале задерживают на длительность одного бита. Изменение фазы при таком смещении кодирующих потоков определяется лишь одним битом последовательности, а не двумя. В результате скачки фазы на 180° отсутствуют, поскольку каждый элемент последовательности, поступающий на вход модулятора синфазного или квадратурного канала, может вызвать изменение фазы на 0^0, 90^0 или 270° .

Другим, существенным недостатком фазовой модуляции является то обстоятельство, что при декодировании сигнала приемник должен определять абсолютное значение фазы сигнала, так как в фазовой модуляции информация кодируется именно абсолютным значением фазы сигнала. Для этого необходимо, чтобы приемник имел информацию об "эталонном" синфазном сигнале передатчика. Тогда путем сравнения принимаемого сигнала с эталонным можно определять абсолютный сдвиг фазы. Следовательно, необходимо каким-то способом синхронизировать сигнал передатчика с эталонным сигналом приемника (по этой причине фазовая модуляция получила название синхронной). Реализация синхронной передачи достаточно сложна, поэтому более широкое распространение нашла разновидность фазовой модуляции, называемая дифференциальной фазовой манипуляцией (*Differential Phase Shift Keying*, **DPSK**). При дифференциальной фазовой манипуляции кодирование информации происходит за счет сдвига фазы по отношению к предыдущему состоянию сигнала. Фактически приемник должен различать не абсолютное значение фазы принимаемого сигнала, а определять лишь величину изменения этой фазы, так как сообщение кодируется изменением фазы. Такой способ модуляция реализуется намного проще синхронного. Во всем остальном DPSK-модуляция не отличается от PSK-модуляции.

Другой разновидностью амплитудно-фазовой модуляции является АФМ с подавлением несущей и передачей одной боковой полосы. Такой способ в зарубежной литературе известен под названием **CAP**-модуляция (*Carrier less Amplitude modulation / Phase modulation*). Известно, что несущая частота используется при модуляции только для переноса спектра сигнала и не является информативной. Передача двух боковых полос модулированного сигнала является в информационном смысле избыточной. Поэтому передача на одной боковой позволяет более эффективно использовать мощность сигнала и полосу канала связи.

Для формирования CAP-сигналов на передающей стороне производят

специальную процедуру обработки модулированного сигнала. Суть ее состоит в том, что перед суммированием синфазной и квадратурной составляющих в модуляторе QAM-сигналов их подвергают дополнительной фильтрации посредством синфазного и квадратурного фильтров.

Однозначное демодулирование CAP-сигналов на приемной стороне происходит в результате предварительного восстановления несущего колебания. Затем приемник, функционирующий в соответствии с алгоритмом CAP, выделяет собственно переданный сигнал, используя при этом те же алгоритмы, что и приемник QAM-колебаний.

2.3.4. Сигналы с частотной манипуляцией

Частным случаем частотной модуляции ЧМ является частотная манипуляция (*Frequency Shift Keying*, **FSK**) при которой модулирующим сигналом выступает последовательность прямоугольных посылок, т. е. передача осуществляется на двух сменяющих друг друга частотах: верхней частоте $\omega_в$, соответствующей положительному модулирующему сигналу, и нижней частоте $\omega_н$, соответствующей отрицательному сигналу. Среднее арифметическое этих частот $\omega_с = (\omega_в + \omega_н)/2$ называется средней частотой, а величина $\Delta\omega = (\omega_в - \omega_н)/2$ – девиация частоты.

При *частотной модуляции* изменение модулирующего сигнала по закону *f(t)* и максимальном изменении частоты на величину $\Delta\omega$ вызывает отклонение частоты сигнала по закону

$$\omega(t) = \omega_o + \Delta\omega\, t. \tag{2.19}$$

Изменение частоты сопровождается изменением фазы сигнала, причем мгновенная фаза сигнала связана с частотой очевидной зависимостью

$$\varphi(t) = \int_0^t \omega(t)\,dt + \varphi_0, \tag{2.20}$$

следовательно,

$$\varphi(t) = \omega_0 t + \Delta\omega \int_0^t f(t)dt + \varphi_0 . \tag{2.21}$$

Таким образом, напряжение, модулированное по частоте, можно записать так:

$$u_{YM}(t) = U_m \cos\left[\omega_0 t + \Delta\omega \int_0^t f(t)dt + \varphi_0\right]. \qquad (2.22)$$

Спектр сигнала для периодической последовательности прямоугольных импульсов определяется по следующей формуле:

$$u_{YM}(t) = U_m \frac{\sin\frac{\pi m_{чм}}{2}}{\frac{\pi m_{чм}}{2}} \cos(\omega_0 t + \varphi_0) + U_m \frac{2}{\pi} \sum_{k=2,4,6}^{\infty} \frac{m_{чм} \sin\frac{\pi m_{чм}}{2}}{m_{чм}^2 - k^2} \times$$
$$\times \{\cos[(\omega_0 + k\Omega)t + \varphi_0] + \cos[(\omega_0 - k\Omega)t + \varphi_0]\} + \qquad (2.23)$$
$$+ U_m \frac{2}{\pi} \sum_{k=1,3,5}^{\infty} \frac{m\cos\frac{\pi m_{чм}}{2}}{m_{чм}^2 - k^2} \{\cos[(\omega_0 - k\Omega_0)t + \varphi_0] - \cos[(\omega_0 + k\Omega_0)t + \varphi_0]\}.$$

Здесь $m_{чм} = \Delta\omega / \Omega$ – индекс частотной модуляции; Ω – круговая частота сигналов данных; k – номер гармоники.

Отсюда следует, что: спектр частотно-манипулированного колебания состоит из несущей частоты, верхней и нижней боковых частот; четные и нечетные боковые частоты подчиняются разным законам и отличаются по фазе на 90^0; спектры боковых частот отличаются от спектра модулирующего сигнала; форма спектра зависит от индекса модуляции.

2.3.5. Способ многочастотной передачи модулированных сигналов

Способ многочастотной передачи **DMT** (*Discrete Multi Tone*) предполагает одновременную передачу QAM-сигналов в различных частотных полосах используемой линии связи. Несмотря на очень высокую сложность технической реализации этого способа, он нашел широкое применение в системах высокоскоростной передачи данных, таких как ADSL и VDSL, а также в компьютерных беспроводных сетях.

При использовании этого алгоритма модуляции весь расчетный частотный диапазон линии делится на несколько участков фиксированной ширины. Каждый из этих участков применяется для организации независимого канала передачи данных.

Перед началом передачи данных производится проверка качества линии. Передатчик, исходя из уровня помех в частотном диапазоне участка, для каждого из этих каналов выбирает подходящую модуляционную схему. На чистых каналах с малым уровнем шумов применяются алгоритмы с

большими значениями позиций сигнала m_c, например, QAM-64, а на более зашумленных участках могут быть использованы более простые алгоритмы модуляции, например QPSK. Очевидно, что реализация такого принципа регулирования скорости передачи данных позволяет наиболее точно согласовывать параметры модулированного сигнала с параметрами линии, по которой он будет передаваться. При передаче данных информация распределяется между независимыми каналами пропорционально их пропускной способности, приемник выполняет операцию демультиплексирования и восстанавливает исходный информационный поток.

При очень высоком уровне помех в определенной полосе канала он может быть полностью исключен из списка используемых каналов до восстановления его работоспособности.

2.4. Способы передачи данных в беспроводных сетях

2.4.1. Требования к сигналам для беспроводной передачи

В компьютерных беспроводных сетях с использованием радиоизлучения существуют некоторые проблемы, для решения которых необходимо выполнить ряд требований. К ним относятся следующие:

- необходимость обеспечение высокой скорости передачи данных, что требует широкой полосы пропускания канала;
- работа множества компьютеров в одном частотном диапазоне, а также использование рядом бытовых (микроволновых печей), промышленных и медицинских приборов частотного диапазона, выделенного для компьютерных радиосетей, требует обеспечения необходимой помехоустойчивости приема сигналов данных;
- свободное распространение радиоволн позволяет злоумышленникам легко перехватить сигналы и вмешаться в работу компьютерной сети, что вызывает потребность обеспечения скрытности радиопередачи.

Решение этих проблем существенно упрощается при использовании в беспроводных компьютерных сетях систем связи, базирующихся на применении сигналов с расширением спектра (*Spread Spectrum*, **SS**). Такие сигналы обладают статистическими свойствами, характерными шумовым процессам, в частности их энергетический спектр почти равномерный, а функция корреляции имеет узкий основной пик и небольшие боковые выбросы. По этой причине в отечественной литературе сигналы с расширенным спектром получили название "шумоподобные сигналы".

В процессе технологии расширения спектра исходный узкополосный

информационный сигнал при передаче преобразуется таким образом, что его спектр оказывается значительно шире спектра первоначального сигнала (рисунок 2.19), т.е. спектр сигнала как бы "размазывается" по частотному диапазону. Одновременно с расширением спектра сигнала происходит и перераспределение спектральной плотности мощности G(f) сигнала; энергия сигнала также "размазывается" по спектру.

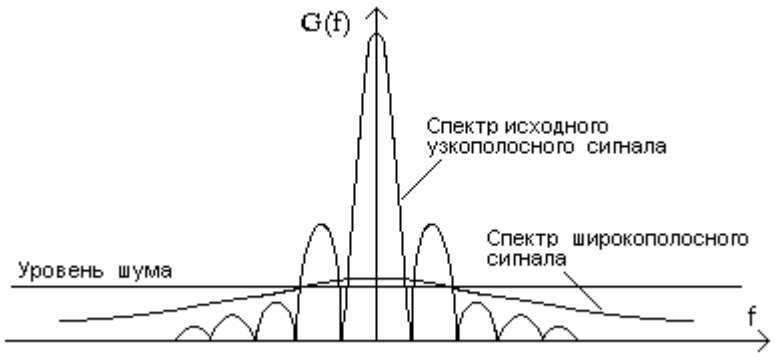


Рисунок 2.19 – Спектры исходного и широкополосного сигналов

Поскольку для обеспечения высокого КПД выходных каскадов радиопередатчика амплитуду сигналов желательно сохранять постоянной, то к настоящему времени наибольшее распространение получили методы расширения спектра сигналов, основанные на изменении в соответствии с некоторым законом их фазы, частоты или временного положения. В компьютерных беспроводных сетях используется два способа расширения спектра: скачкообразная перестройка частоты (*Frequency Hopping Spread Spectrum,* ***FHSS***) и расширение спектра способом прямой последовательности (*Direct Sequence Spread Spectrum*, **DSSS**).

2.4.2. Скачкообразная перестройка частоты

Для формирования широкополосного сигнала по способу *FHSS* осуществляется скачкообразная перестройка несущей частоты. При этом несущая частота излучаемого сигнала очень быстро изменяется от одного значения к другому по псевдослучайному закону. За счет такого переключения энергия передаваемого сигнала распределяется в широком диапазоне частот. Очевидно, что перед началом передачи источник и получатель должны согласовать между собой алгоритм изменения частоты.

В локальных беспроводных сетях передачи данных весь выделенный частотный диапазон изменения несущей разделен на ряд каналов (в Европе таких каналов 79). Смежные канальные частоты разнесены между собой на 1

МГц. Каналы нумеруются от 2 до 80. Так, например, для канала 2 стандартом установлена средняя частота 2,402 ГГц, а для канала 80 – 2, 480 ГГц. Все частоты разбиты на три набора по 26 частот в каждом. В каждом из трех наборов определены по 78 различных шаблонов переключения частот. Для установления соединения используется вызывной канал. Приемник и передатчик выбирают среди одного из трех наборов согласованный шаблон последовательности переключения частот. Очевидно, что обмен данными может происходить между абонентами сети, использующих один и тот же шаблон и синхронно переключающими с одной частоты на другую. Минимальный скачок частоты должен быть 6 МГц, а максимальная длительность генерации одной частотной посылки равна 400 мс.

На рисунке 2.20 в качестве примера изображены образцы частотно–временных последовательностей для трех одновременно работающих станций.

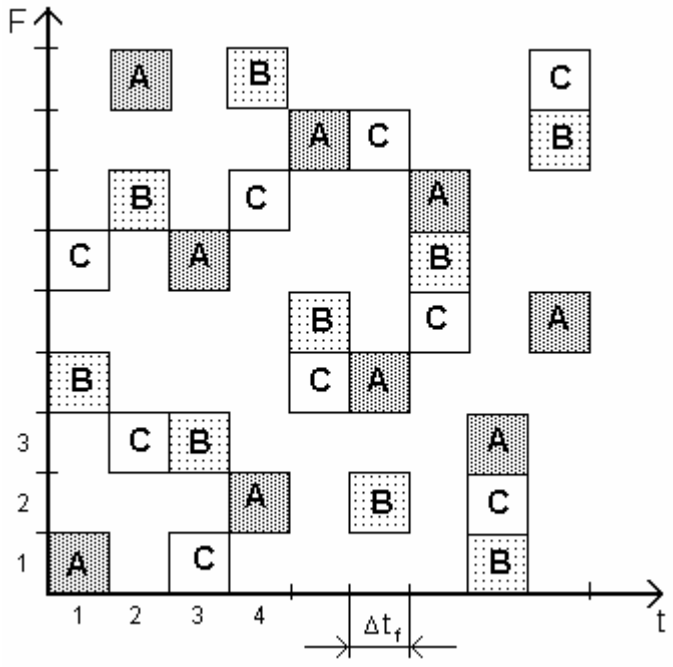


Рисунок 2.20 – Пример бесконфликтной работы трех станций в одной полосе частот способом скачкообразного изменения частоты

В идеальном случае, когда все передатчики начинают работать одновременно, в любой промежуток времени Δt_f нет никаких станций, излучающих сигналы на одних и тех же частотах. Таким образом станции могут одновременно вести передачу, абсолютно не мешая друг другу.

В реальном случае такого синхронизма не существует. Поэтому в некоторые временные интервалы возможны частотные помехи. Однако они существуют в относительно узком диапазоне частот и непродолжительное время. За счет широкополосности сигналов такие помехи не оказывают су-

щественного влияния на помехоустойчивость приема данных.

На рисунке 2.21 показана схема передающей и приемной частей системы FHSS с многопозиционной частотной манипуляцией **MFSK** (M-ary Frequency Shift Keying) с использованием m_c частотных позиций.

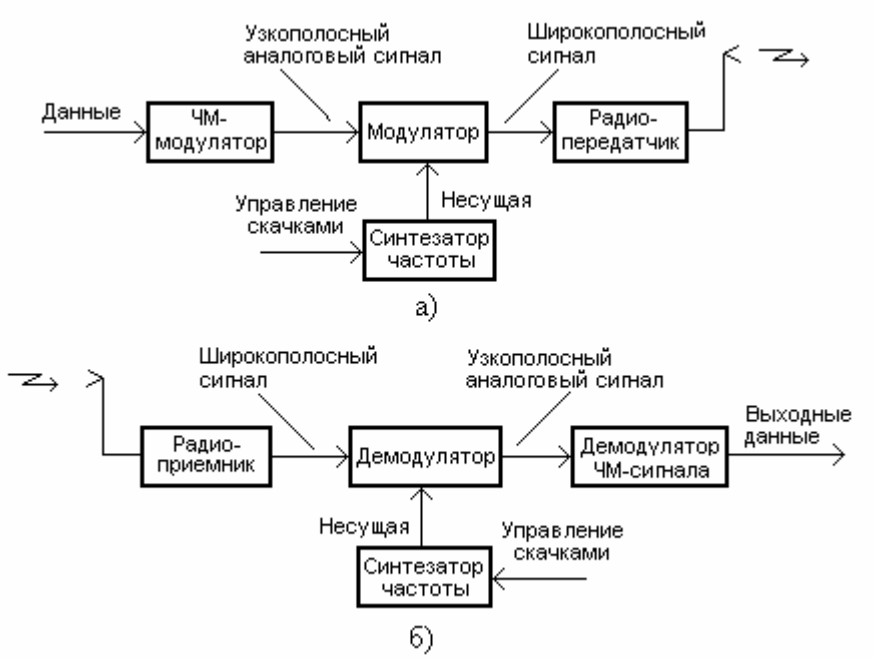


Рисунок 2.21 – Структурная схема передающей а) и приемной б) частей широкополосной системы связи со скачкообразным изменением частоты

Данные от источника объединяются по k бит (k=$\log_2 m_c$) и подаются на многопозиционный частотный манипулятор, который формирует узкополосные непрерывные ЧМ-сигналы. Затем спектр этого сигнала переносится в область частоты несущего колебания. Несущая частота формируется быстродействующим синтезатором частот, который вырабатывает гармоническое колебание с частотой, определяемой кодовой комбинацией, поступающей с генератора псевдослучайных последовательностей (ПСП). Последовательность чисел с генератора ПСП задает шаблон изменения частот при их переключении. Через интервал времени Δt_f происходит скачкообразное изменение несущей частоты. Для интервала времени Δt_f ширина полосы излучаемого сигнала будет такой же, как и в обычной схеме частотного манипулятора MFSK. В то же время при усреднении по множеству скачков спектр сигнала FH/MFSK будет занимать всю полосу расширенного спектра.

Как видно из рисунка 2.21,б, приемник повторяет все действия передатчика в обратной последовательности. Полученный сигнал демодулируется путем выделения низкочастотных компонентов частотно–модулированного сигнала, а затем в ЧМ демодуляторе осуществляется де-

тектирование информационных сигналов, отображающих переданные данные. На практике двухэтапный процесс модуляции зачастую реализуется как один этап, когда синтезатор частот в течение времени Δt_f формирует гармонику, основываясь на псевдослучайном коде номера частоты и ее смещении в соответствии с информационной последовательностью.

Различают **медленные скачки** несущей частоты и **быстрые**. При медленных скачках (*Slow Hopping*) средняя частота несущей сохраняется на протяжении нескольких информационных битов, а при быстрых скачках (*Fast Hopping*) изменение среднего значения несущей происходит несколько раз на протяжении одного информационного элемента. На рисунке 2.22 показаны графики изменения частот несущих при поступлении на вход передатчика последовательности данных вида 010110.

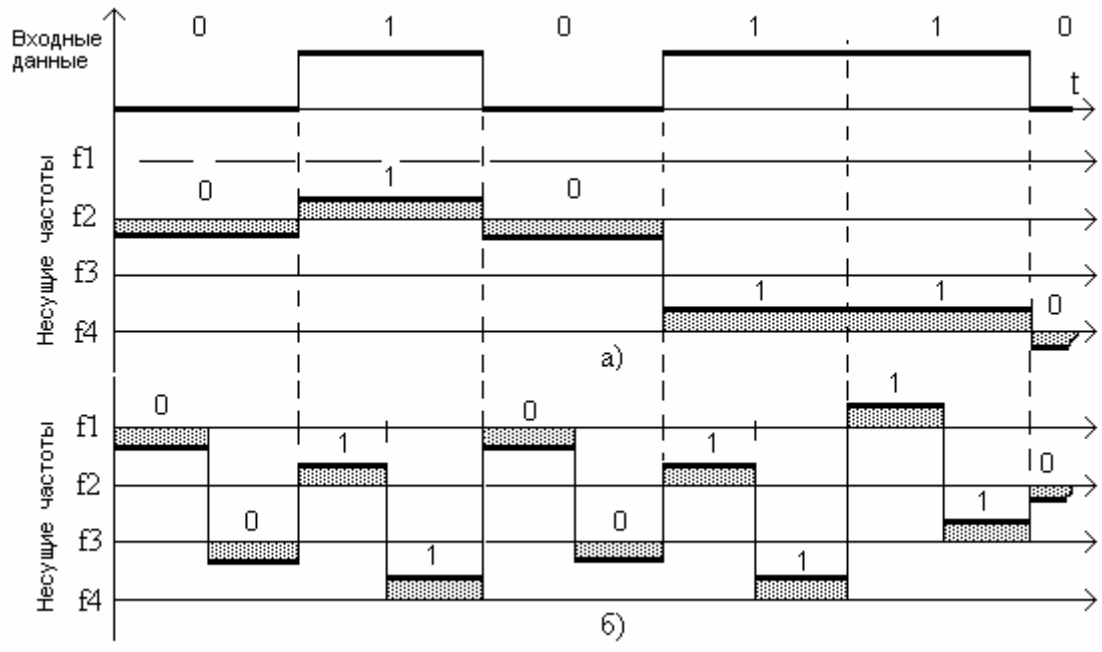


Рисунок 2.22 – Иллюстрация медленных а) и быстрых б) скачков несущей частоты

Рисунок 2.22,а иллюстрирует процесс медленных скачков. Для простоты в примере используется только четыре несущие частоты. Из рисунка видно, что на протяжении первых трех информационных элементов 010 средняя частота несущей равна f2, а трех последующих бит 110 – f4. Частоты несущих отклоняются от своих средних значений на ± Δf только в зависимости от значения входных данных, т.е. имеет место частотная модуляция несущей. На рисунке 2.22,б показан процесс быстрых скачков несущей, значение которой меняется дважды на протяжении одного единичного элемента. Изменение частот происходит в порядке f1–f3–f2–f4–f1–f2–f3–f4.

2.4.3. Расширение спектра способом прямой последовательности

При расширении спектра информационных сигналов способом прямой последовательности **DSSS** (*Direct Sequence Spread Spectrum*) единичные элементы сигналов длительностью τ_0 (рисунок 2.4.5,а) разбиваются на N бинарных элементов (*чипов*) длительностью $\tau_ч = \tau_0/N$ (рисунок 2.23,б).

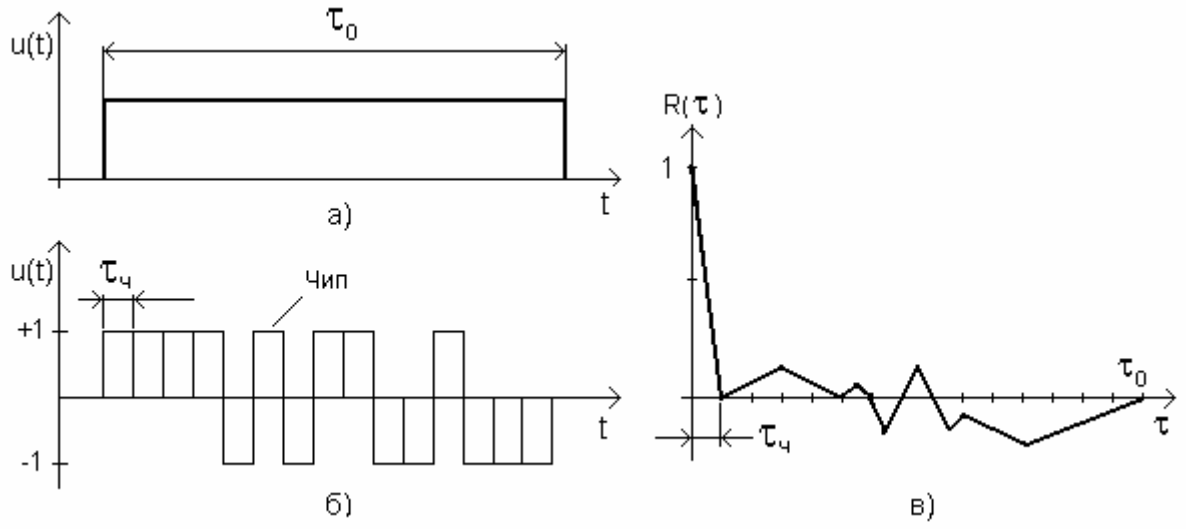


Рисунок 2.23 – Иллюстрация способа формирования шумоподобного сигнала DSSS

На практике в системах широкополосной связи используются последовательности с числом чипов от нескольких сотен до десятков тысяч бинарных элементов. В компьютерных беспроводных локальных сетях применяются сигналы с количеством чипов около 10. Разбиение информационных бит на последовательность чипов позволяет сформировать широкополосный сигнал длительностью τ_0 с шириной полосы частот в N раз больше полосы исходного сигнала, т.е. $F_{шп} \approx 1/\tau_ч = N/\tau_0$ герц.

Последовательности бинарных чипов образуют коды, которые выбираются так, чтобы обеспечить заданные шумовые свойства сигналов. В частности, функция корреляции последовательности чипов при достаточно большом N должна содержать главный максимум, сосредоточенный в области от $-\tau_ч$ до $\tau_ч$ и боковые лепестки, имеющие сравнительно низкий уровень (рисунок 2.23,в).

Функция автокорреляции $R(\tau)$ для сигнала U(t) определяется по известной формуле

$$R(\tau)=\frac{1}{2E}\int_{-\tau_0}^{\tau_0} U(t)\overset{*}{U}(t-\tau)dt\ , \qquad (2.24)$$

где Е – энергия сигнала; значок * означает комплексно-сопряженное значение сигнала.

Если подобрать чиповую последовательность, для которой функция автокорреляции имеет резко выраженный пик лишь для одного момента времени, то такой информационный сигнал возможно выделить на фоне шума и других шумоподобных сигналов. Для этого в приемнике полученный сигнал умножается на чиповую последовательность, аналогичную переданной, т.е. вычисляется автокорреляционная функция сигнала. В результате сигнал становится опять узкополосным, поэтому его фильтруют в узкой полосе частот. Любая помеха, попадающая в полосу исходного широкополосного сигнала, после умножения на чиповую последовательность, наоборот, становится широкополосной и "обрезается" фильтрами, а в узкую информационную полосу попадает лишь часть помехи, по мощности значительно меньшая, чем помеха, действующая на входе приемника. Имеется обширный класс дискретных псевдослучайных последовательностей. К ним относятся М-последовательности Хаффмена, сигналы Пэйли-Плоткина, коды Баркера и др.

Для переноса спектра широкополосного сигнала в область полосы пропускания канала шумоподобный сигнал может подвергаться всем известным способам модуляции. При амплитудной модуляции изменяется амплитуда его импульсов, при частотной модуляции элементы сигнала отличаются частотой, при фазовой – разностью фаз между двумя посылками. В компьютерных локальных беспроводных сетях используется преимущественно дифференциальная (относительная) двоичная фазовая манипуляция **DBPSK** (*Differential Binary Phase Shift Keying*) с двумя (0^0 и 180^0) или четырьмя (0^0, 90^0, 180^0 и 270^0) значениями начальных фаз. Для расширения спектра передаваемого сигнала применяются **псевдослучайные последовательности Баркера**, являющиеся одними из лучших с точки зрения корреляционных свойств.

Известны последовательности Баркера длиной от 3 до 13 элементов. Лучшей последовательностью с точки зрения автокорреляционной функции является последовательность, состоящая из 11 элементов (чипов), например, вида **11100010010**. Иногда используется ее модификация вида 10110111000. На каждый передаваемый информационный бит приходится 11 элементов последовательности Баркера. Различают прямую и инверсную последовательности Баркера. Единичные информационные биты передаются прямым кодом Баркера, а нулевые – инверсным. Инвертирование последовательно-

сти Баркера осуществляется очень просто с помощью логической операции исключающего ИЛИ.

На рисунке 2.24 изображена схема реализации передающей и приемной частей системы связи с расширением спектра при использовании последовательности Баркера и двухпозиционной относительной (дифференциальной) фазовой манипуляции со скоростью передачи 1 Мбит/с.

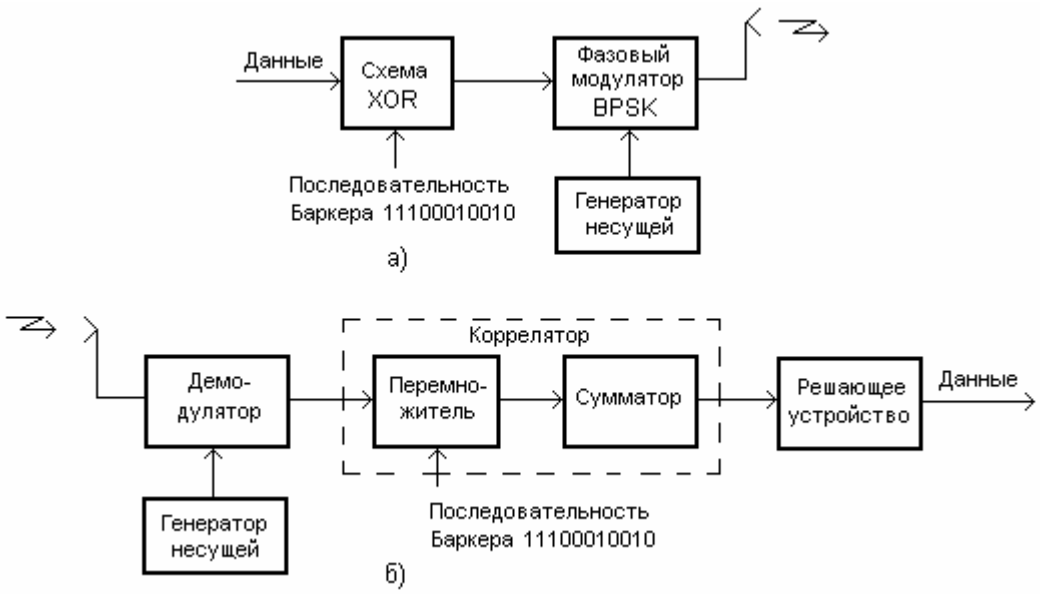


Рисунок 2.24 – Схема передающего а) и приемного б) устройств расширения спектра методом DSSS с последовательностью Баркера

Собственно говоря, относительная фазовая модуляция применяется именно к чиповой последовательности. При информационной скорости 1 Мбит/с скорость следования отдельных чипов последовательности Баркера составляет 11 Мчип/с, а ширина спектра такого сигнала – 22 МГц, так как длительность одного чипа равна 1/11 мкс. На рисунке 2.25 изображены временные диаграммы сигналов на выходах функциональных элементов передатчика. Из диаграммы видно, что фаза несущего колебания изменяется на 180^0 относительно предыдущего колебания при передаче каждого единичного чипа, и не меняется при передаче нулевых элементов последовательности Баркера. В приемнике полученный сигнал умножается на код Баркера (вычисляется корреляционная функция сигнала), в результате он становится узкополосным. Помеха, попадающая в полосу данного широкополосного сигнала, после умножения на код Баркера, наоборот, становится широкополосной. Поэтому в узкую информационную полосу попадает лишь часть помехи, по мощности примерно в 11 раз меньше, чем помеха, действующая на входе приемника.

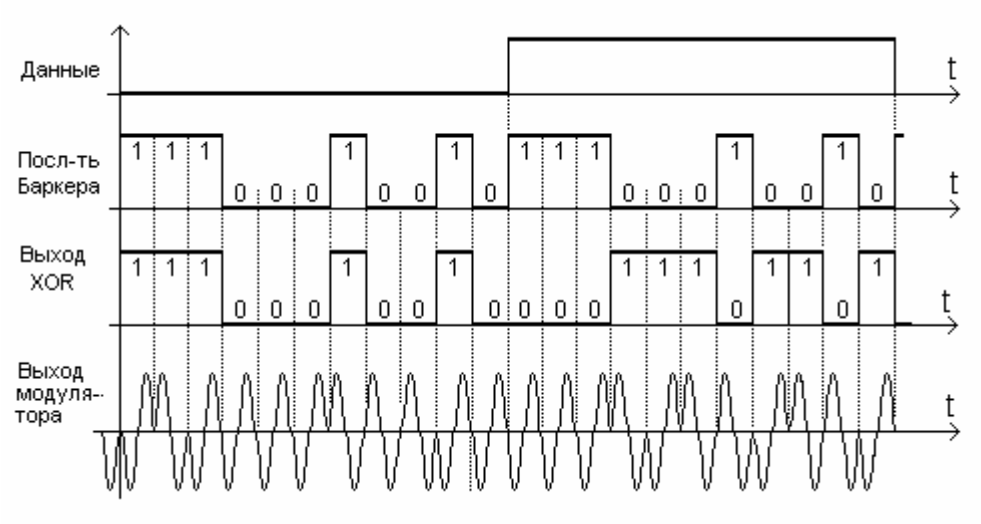


Рисунок 2.25 – Временные диаграммы сигналов передающей части системы с расширением спектра при использовании последовательности Баркера

Для передачи данных со скоростью 2 Мбит/с используется аналогичная технология DSSS с 11-чиповыми кодами Баркера, но для модуляции несущего колебания применяется относительная квадратурная фазовая модуляция **DQPSK** (*Differential Quadrature Phase Shift Keying*). При такой модуляции сдвиг фаз гармонического колебания может принимать четыре различных значения: 0^0, 90^0, 180^0 и 270^0, т.е. в одном дискретном состоянии сигнала отображается два информационных бита и тем самым в два раза повышается информационная скорость передачи.

2.4.4. Использование комплементарных кодовых последовательностей

Комплементарные кодовые последовательности (*Complementary Code Keying*, CCK) представляют собой группу бинарных элементов (чипов), состоящую из нулей и единиц. **CCK-последовательности** обладают тем свойством, что сумма их автокорреляционных функций для любого циклического сдвига, отличного от нуля, всегда равна нулю. В беспроводных компьютерных сетях применяются преимущественно CCK-последовательности, состоящие из 8 чипов, определенных на множестве комплексных элементов. Если ограничиться множеством комплексных элементов {1, –1, j, –j}, то можно сформировать восемь комплексных чисел, т.е., каждый из элементов 8-чиповой CCK-последовательности может принимать одно из следующих восьми значений: 1, –1, j, –j, (1+j), (1–j), (–1+j), (–1–j).

Основное отличие CCK-последовательностей от рассмотренных ранее кодов Баркера заключается в том, что существует не строго заданная после-

довательность, посредством которой можно кодировать либо логический нуль, либо единицу, а некоторое множество последовательностей. Учитывая, что каждый элемент 8-чиповой комбинации может принимать одно из восьми значений в зависимости от значения фазы, можно скомбинировать 8^8=16777216 вариантов последовательностей, однако не все они будут комплементарными. Из всего этого множества выбираются 64 пары ССК-последовательностей, обладающих максимальным взаимным различием. Для формирования 8 значений элементарных сигналов применяется DQPSK – дифференциальная квадратурная амплитудно-фазовая манипуляция. Использование многопозиционной манипуляции позволяет кодировать в одном передаваемом символе несколько информационных битов и тем самым повысить эффективную скорость передачи данных.

Схема устройства передачи данных в беспроводных компьютерных сетях со скоростью 11 Мбит/с, с использованием 11-чиповых комплементарных последовательностей и восьмипозиционной дифференциальной фазовой манипуляции, показана на рисунке 2.26.

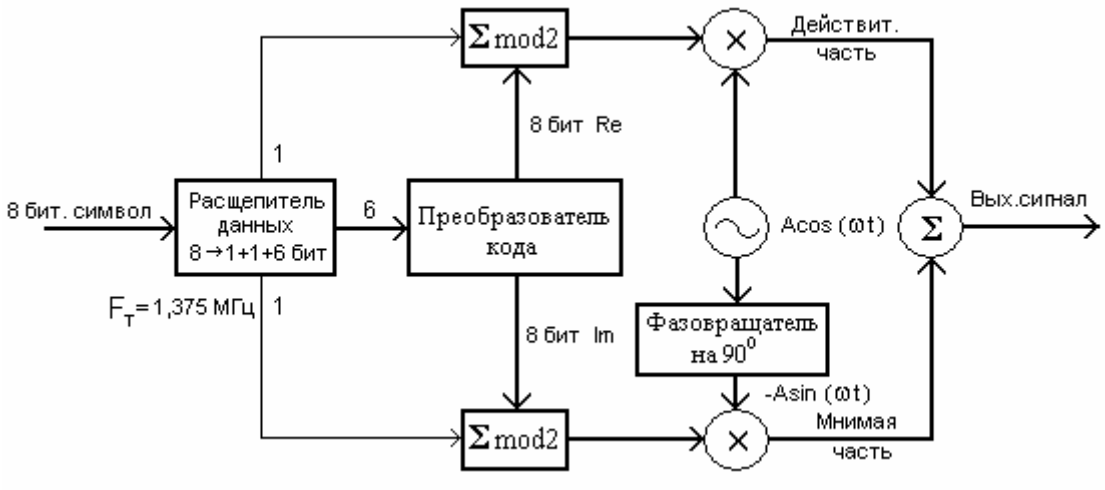


Рисунок 2.26 – Схема передатчика на основе ССК–последовательностей

Устройство производит выделение из последовательного информационного потока посредством расщепителя данных группы по 8 битов. 8-битовые группы выбираются с тактовой частотой 1,375 МГц. Такой выбор частоты объясняется тем, что информационная скорость передачи данных должна быть 11 Мбит/с.

Затем расщепитель направляет первый бит этой группы в верхнюю ветвь схемы, а второй – в нижнюю на соответствующие входы сумматоров по модулю 2. Оставшиеся 6 битов используются для формирования в преобразователе кода пары 8-чиповых ССК-последовательностей. Чипы называ-

ются комплексными, поскольку они определяют синфазную и квадратурную составляющие сигнала для DQPSK. Восемь чипов подаются поэлементно на вторые входы сумматоров по модулю 2. Суммарные сигналы (прямая или инверсная ССК-комбинация чипов) последовательно модулируют несущую с частотой манипуляции 11 МГц. Фактически каждый чип в радиоканале представляет собой сигнал несущей частоты с определенным фазовым сдвигом. Фазовые сдвиги отдельных посылок определяются по специальным формулам, которые здесь не рассматриваются.

Передача данных со скоростью 5,5 Мбит/с осуществляется по аналогичной схеме. Отличие состоит лишь в том, что вместо 8-битовой последовательности расщепитель выбирает из входного потока 4-битовые комбинации с частотой 5,5/4=1,375 МГц. Эти комбинации задают одну из четырех необходимых комплементарных пар ССК-последовательностей, что позволяет выбрать последовательности с большими взаимными расстояниями и обеспечить более высокую помехоустойчивость приема.

Для скорости передачи данных 5,5 Мбит/с используется двухпозиционная относительно-фазовая манипуляция DBPSK, и ССК-последовательность определяет не восемь, а четыре информационных бита, поэтому и скорость получается вдвое ниже. В обоих случаях скорость следования отдельных чипов равна 11 Мчип/с. Соответственно, и ширина спектра сигнала как при скорости 11 Мбит/с, так и при 5,5 Мбит/с составляет 22 МГц.

2.4.5. Мультиплексирование с разделением по ортогональным частотам

В процессе передачи радиосигналов в точке приема за счет многолучевого распространения электромагнитных колебаний происходит их интерференция, которая приводит к замираниям и межсимвольным искажениям сигналов. Одним из наиболее эффективных способов обеспечения требуемой скорости и помехоустойчивости в компьютерных сетях является многоканальная передача сигналов с частотным разделением каналов **OFDM** (*Orthogonal Frequency Division Multiplexing*). Этот способ является частным случаем способа передачи DMT, рассмотренным выше. Суть способа OFDM заключается в том, что поток передаваемых данных распределяется по множеству частотных подканалов и передача ведется параллельно на всех этих подканалах. При этом высокая скорость передачи достигается именно за счет одновременной передачи данных по всем каналам, хотя скорость передачи в отдельном подканале может быть и невысокой.

При частотном разделении каналов необходимо, чтобы ширина полосы пропускания отдельного канала была, с одной стороны, достаточно узкой

для минимизации искажения сигнала в пределах отдельного канала, а с другой — достаточно широкой для обеспечения требуемой скорости передачи. Кроме того, для экономного использования всей полосы канала, разделяемого на подканалы, желательно как можно более плотно расположить частотные подканалы, но при этом избежать межканальной интерференции, чтобы обеспечить полную независимость каналов друг от друга.

Для исключения межканальной интерференции частотные поднесущие отдельных каналов (а точнее, функции, описывающие эти сигналы) должны быть ортогональными. Хотя спектры сигналов в отдельных каналах и перекрываются, огибающая спектра любой из поднесущих F_i имеет нулевое значение для центральной частоты спектра соседней поднесущей F_{i-1} и F_{i+1} (рисунок 2.27).

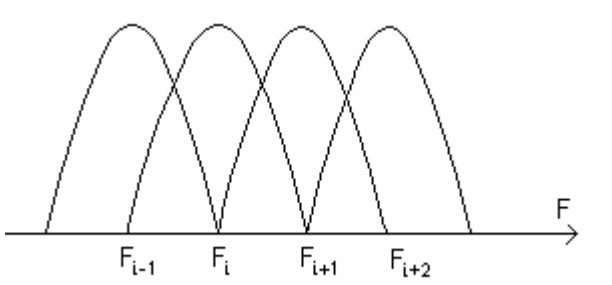


Рисунок 2.27 – Перекрывающиеся частотные каналы с ортогональными несущими

Сигналы являются ортогональными, если разнос по частоте $\Delta F=F_{i+1}-F_i = 1/\tau_0$. Здесь τ_0– длительность единичного элемента сигнала, которая определяет интервал ортогональности.

Одним из важных преимуществ способа OFDM является сочетание высокой скорости передачи с эффективным противостоянием многолучевому распространению. Следует заметить, что сама по себе технология OFDM не устраняет многолучевого распространения, но создает предпосылки для устранения эффекта межсимвольной интерференции.

Длительность тактового интервала $T_{ти}$ системы передачи выбирается несколько большей, чем интервал ортогональности τ_0. Их разность составляет защитный временной интервал $\tau_З$ между последовательно передаваемыми единичными элементами группового сигнала, вводимый для повышения защищенности систем передачи с OFDM от интерференционных помех. Защитный интервал несколько снижает эффективную скорость передачи, однако он служит защитой от возникновения межсимвольной интерференции. Во время действия защитного интервала передатчиком добавляется избыточная информация в начале передаваемого единичного элемента путем циклического повторения его окончания, которая отбрасывается при обработке сигнала в приемнике.

Наличие защитного интервала создает временные паузы между отдельными элементами, и если длительность защитного интервала превышает максимальное время задержки сигнала в результате многолучевого распространения, то межсимвольной интерференции не возникает. В компьютерных беспроводных сетях при использовании технологии OFDM длительность защитного интервала составляет одну четвертую длительности самого сигнала. При этом сам сигнал имеет длительность 3,2 мкс, а защитный интервал — 0,8 мкс. Таким образом, длительность сигнала вместе с защитным интервалом составляет 4 мкс.

Для локальных компьютерных сетей в соответствии с международным стандартом в диапазоне частот 5,2 ГГц выделено 12 неперекрывающихся каналов с одинаковой полосой пропускания 20 МГц. Каждый из этих каналов разделен на 64 подканала с полосой пропускания 20000/64=312,5 кГц. Из них для передачи собственно данных используется 48 подканалов. Четыре подканала служат для передачи опорных колебаний, а по 6 подканалов слева и справа остаются незанятыми и выполняют функции защитных полос (рисунок 2.28).

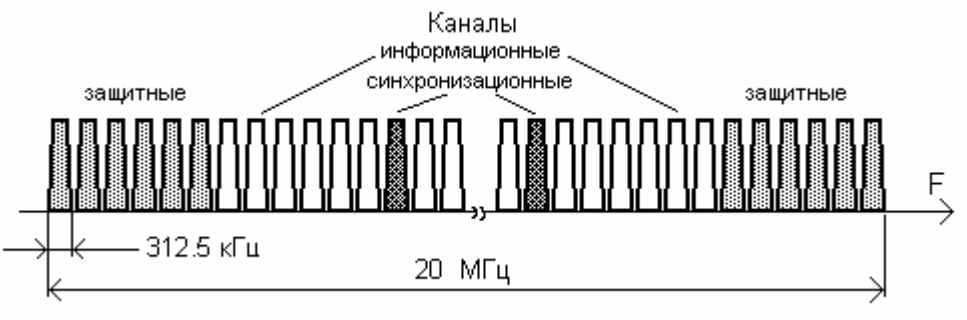


Рисунок 2.28 – Схема распределения подканалов в системе OFDM

В любом из каналов может осуществляться передача данных с результирующей скоростью 6, 9, 12, 18, 24, 36, 48 или 54 Мбит/с. Для обеспечения такого диапазона скоростей применяются различные способы фазовой и амплитудно-фазовой манипуляции. На нижнем уровне используется BPSK, которая позволяет организовать подканал со скоростью 125 кбит/с. Это обеспечивает суммарную пропускную способность канала, состоящего из 48 подканалов, 6 Мбит/с. За счет применения QPSK пропускная способность канала удваивается (12 Мбит/с). Ее можно удвоить еще в 2 раза, используя 16-уровневую квадратурную амплитудную модуляцию QAM-16. Таким образом, модуляция QAM-64 обеспечивает скорость передачи данных в канале 54 Мбит/с.

2.5. Способы передачи данных на канальном уровне

2.5.1. Асинхронная и синхронная передача

Данные в компьютере представлены в параллельном виде (байтами, словами, двойными словами и пр.). Передача сообщений на удаленные пункты с целью экономии линий связи и линейного оборудования осуществляется побитно, т.е. последовательным способом. При этом возникает проблема нахождения на приемной стороне начала передаваемой последовательности и границ байтов. Кроме того, приемник для правильной регистрации единичных элементов сигналов должен знать расположение их границ.

Существуют два способа передачи сигналов данных: *асинхронный* и *синхронный*. При асинхронном способе сообщение передается посимвольно, причем символы выдаются в канал в произвольный момент времени, по мере поступления их от компьютера (отсюда и название способа). Длина символа может содержать от 5 до 8 бит. В отсутствие передачи информации передатчик и приемник остановлены, т.е. находятся "на стопе". Для информирования приемника об этом состоянии и контроля целостности линии связи передатчик постоянно выдает в линию стоповый сигнал – напряжение, соответствующее уровню логической единицы. В случае необходимости передачи очередного байта передающее устройство сначала выдает стартовый сигнал (нулевой уровень) продолжительность которого равна длине единичного элемента τ_0, затем следуют 5…8 информационных бит, отображающих собственно передаваемый символ (рисунок 2.29). Биты символа обычно передаются, начиная с младшего значащего разряда.

Рисунок 2.29 – Формат асинхронной передачи символа

За информационными битами на 9-й временной позиции следует бит контроля четности (*паритета*). Значение бита паритета передатчик выбирает так, чтобы полное число единиц в символе, включая бит четности, было четным или нечетным, в зависимости от задания вида контроля при начальной инициализации системы передачи. Завершает последовательность стоповый элемент, минимальная длина которого при начальной инициализации

устанавливается от 1 до 2 бит, максимальная его продолжительность не регламентируется. При передаче равномерного потока байтов интервал между символами постоянен и равен длине стопового элемента.

Достоинством асинхронной передачи является ее простота. На приемной стороне можно использовать несложные схемы формирования тактовых сигналов, от которых не требуется жесткой стабильности, поскольку тактовые импульсы вырабатываются на коротком интервале времени передачи одного знака. При приеме следующего байта схема формирователя тактовых импульсов запускается заново с момента его начала (по фронту стартовой посылки).

Чтобы предотвратить ложный запуск схемы синхронизации при воздействии кратковременных импульсных помех, на приемной стороне производится опрос состояния канала передачи данных в момент перехода его в “0”. Если в середине единичного интервала (в момент времени $\tau_0/2$) состояние канала также равно “0”, то принимается решение о поступлении стартового сигнала и разрешается формирование тактовых импульсов с частотой $f_т = 1/\tau_0$. В противном случае схема генерирования тактовых импульсов запирается. С целью уменьшения величины ошибки определения момента поступления стартовой посылки частота опроса f_0 состояния канала выбирается значительно выше тактовой частоты передачи (обычно $f_0 = 16f_т$). Рисунок 2.30 иллюстрирует процесс синхронизации при использовании повышенной частоты опроса.

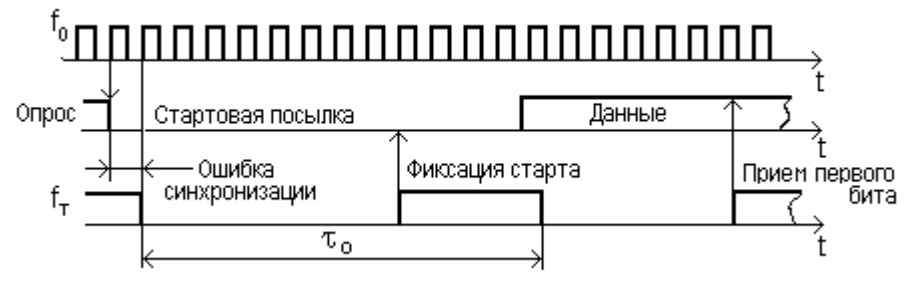


Рисунок 2.30 – Временная диаграмма процесса синхронизации старт–стопного приемника

Синхронность работы приемника с передатчиком при асинхронном способе передачи достигается сравнительно просто за счет установки и поддержания одинаковых тактовых частот на обеих сторонах. Расхождение колебаний генераторов по фазе устраняется путем запуска работы приемного распределителя по срезу стартового сигнала. Точность такого способа синхронизации сравнительно невысокая, однако достаточная при скорости передачи данных до сотни кбит/с.

Существенный недостаток асинхронной передачи – высокая избыточ-

ность (около 25%) за счет передачи с каждым символом стартового и стопового битов. Другим недостатком этого способа является невысокая скорость передачи, которая ограничивается невозможностью обеспечения требуемой помехоустойчивости приема единичных элементов при относительно низкой точности синхронизации. Более высокую эффективность передачи обеспечивает синхронный способ передачи.

При синхронной передаче символов стартовые и стоповые биты между ними отсутствуют, а передаваемые знаки могут, как и прежде, дополняться битом паритета. Все сообщение или его часть передается на протяжении фиксированного временного интервала, называемого циклом передачи. Совокупность символов, передаваемых в течение цикла, носит название *блока, кадра* или *фрейма*. Для обозначения начала блока часто используются управляющий символ **СИН** (SYN), входящий в таблицу *ASCII*, либо двоичная, так называемая ***флаговая***, комбинация вида **01111110**. В ряде случаев для повышения помехоустойчивости обнаружения начала блока символ СИН дублируется.

С целью обеспечения безошибочной передачи данных по каналу связи в текст сообщения включается служебная информация, состоящая из заголовка и управляющих символов (рисунок 2.31).

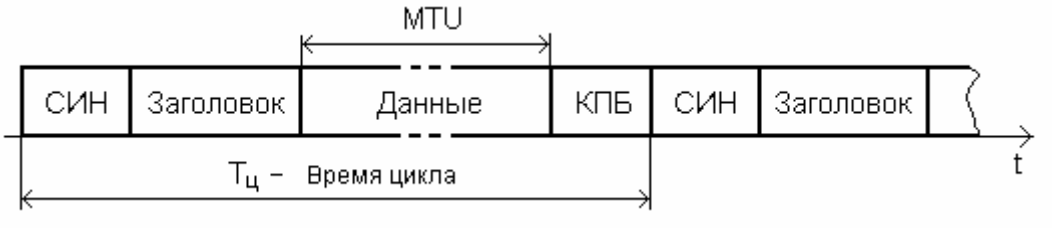


Рисунок 2.31 – Формат блока при синхронной передаче данных

В заголовке обычно указываются адреса передающей и принимающей станции, длина и порядковый номер блока, время ввода сообщения в канал, способ защиты от ошибок и др. Допускается использовать заголовок переменной длины. В таком случае в формате блока должно содержаться поле, указывающее длину заголовка. Поле "Данные" может содержать фиксированное либо произвольное число символов. Обычно в конкретных сетях определяется максимальное значение длины поля данных. Эта величина называется **максимальной единицей передачи данных** MTU (*Maximum Transfer Unit*).

В конце блока передается контрольная последовательность (КПБ), формируемая в соответствии с принятой процедурой защиты от ошибок (циклическое кодирование, контрольное суммирование и т.д.).

Для определения начала цикла приемное устройство непрерывно ана-

лизирует последовательность поступающих на его вход битов. В момент совпадения анализируемой последовательности с кодовой комбинацией СИН выключается схема поиска символа синхронизации и начинается прием сообщения. Синхронизация по битам осуществляется с помощью схемы фазовой автоподстройки частоты (ФАПЧ) путем оценки тактовой частоты передатчика на основе информационной последовательности.

Процесс приема продолжается до тех пор, пока не будет обнаружен знак окончания блока, после чего включается синхронизация, и приемник переходит в режим поиска нового символа СИН. Если передача сообщения не закончилась, то в паузах между передачей блоков для поддержания синхронизма приемника с передатчиком последний непрерывно посылает в канал синхронизирующую последовательность вида 10101… .

Управление таким сложным процессом, как передача данных в вычислительных сетях, в котором участвует многочисленная и разнообразная аппаратура, требует формализации процедур установления и разъединения соединений, передачи информации и защиты ее от ошибок, выделения и освобождения ресурсов ЭВМ и т.д. Эти задачи решаются с помощью стандартных правил, называемых *протоколами* связи, реализующих процедуры взаимодействия элементов сети при установлении–разъединении связи и передаче данных.

При синхронной передаче различают два вида протоколов: **знак-ориентированный** (*байт-ориентированный*) и **бит-ориентированный**.

2.5.2. Байт-ориентированная передача данных

При байт-ориентированной передаче управление обменом данных обеспечивают специальные **управляющие знаки** (*байты*), которые включаются в общий информационный поток. Байт-ориентированные протоколы используются преимущественно для обмена отображаемыми символами *ASCII*-кода. Они относятся к классу универсальных протоколов как для коммутируемых, так и выделенных двух- и многоточечных каналов, для пакетного и диалогового режимов работы. Однако применение их для диалогового режима не очень эффективно по причине достаточно высокой избыточности. Обмен данными при байт-ориентированной передаче осуществляется преимущественно полудуплексным способом. В двухпунктовых звеньях любая станция может быть управляющей, либо подчиненной. В многопунктовых соединениях одна станция задается в качестве управляющей, а остальные – подчиненные.

Для управления передачей данных в байт-ориентированных протоколах применяются специальные управляющие символы кода ***ASCII*** (КОИ-7).

Кроме синхронизации блоков эти символы обеспечивают переключение станций из режима управления в режим передачи текста, изменения направления передачи и т.д.

Рассмотрим некоторые наиболее часто встречающиеся символы управления. При обозначении символов в скобках приводится их международное обозначение на английском языке.

КТМ – кто там? (*ENQ-Enquiry*). Служит для установления контакта на двухпунктовых звеньях или для запроса повторной передачи ответа на блок сообщения, если принят неверный ответ, либо не было ответа. Он также указывает на конец последовательности выборки или опроса. **НЗ** – начало заголовка (*SOH – Start of Heading*). Применяется, если текст содержит заголовок. Указывает, что последующие кодовые комбинации, относящиеся к заголовку, являются служебными символами, которые используются для дальнейшей обработки сообщения, его маршрутизации и т.д. **НТ** – начало текста (*STX – Start of Text*). Информирует, что за этим символом следует текст сообщения. Если сообщение содержит заголовок, то НТ завершает заголовок и обозначает начало текста. **КБ** – конец блока (*ETB – End of Transmission Block*) служит для указания конца блока текста (в случае разбивки его на блоки). **КТ** – конец текста (*ETX – End of Text*). Обозначает конец последнего блока текста. **ДА** – подтверждение (*АСК – Acknowleadge*). Используется в качестве положительного ответа на правильность принятия сообщения, готовность абонентского пункта и др. **НЕТ** – отрицание (*NAK – Negative Acknowleadge*). Обозначает отрицательный ответ (станция на готова к приему, опрашиваемое устройство не имеет данных к передаче). **КП** – конец передачи (*EOT – End of Transmission*). Индицирует завершение процесса передачи, используется для перевода станций в управляющий режим, а также в начале передачи перед проведением опроса или выборки станции, подключенной к каналу связи. **СИН** – синхронизация (*SYN – Synchronous Idle*). Применяется для установления и поддержания синхронизации.

Кроме этого, используются комбинации, служащие для изменения значения следующих за ним знаков: **АР** – авторегистр (*DLE – Data Link Escape*). Так например, **АР1 КП** означает необходимость выполнить разъединение. Эта последовательность является признаком окончания процесса передачи, при получении которой происходит разъединение коммутируемого канала связи.

Иногда возникает необходимость исключить реагирование станции данных на кодовые комбинации, являющиеся управляющими символами, для чего используется специальный режим передачи, называемый "прозрачным". При таком режиме передаваемый текст может содержать, кроме буквенных, цифровых и графических символов, еще и управляющие комбинации. Для установления режима "прозрачной" передачи используется после-

довательность **АР1 НТ** (DLE STX). Текст, передаваемый в этом режиме, должен заканчиваться символами АР1 КБ и АР1 КТ, переводящими звено в режим обычной передачи.

Данные по каналу связи передаются в форме блоков (кадров), что позволяет использовать эффективные методы защиты от ошибок. Формат передаваемого блока изображен на рисунке 2.32. Здесь КПТ – контрольная последовательность текста, содержащая проверочные биты в соответствии с принятым методом защиты от ошибок. КПТ следует непосредственно за знаком КТ, причем символ КТ учитывается процедурой формирования КПТ.

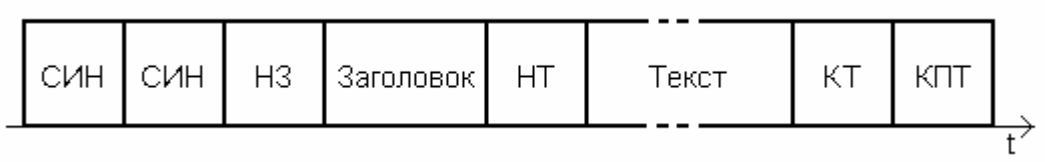


Рисунок 2.32 – Формат блока байт-ориентированного протокола

После передачи блока данных канал связи переключается для передачи в обратном направлении сигналов подтверждения или отрицания приема. В случае безошибочного приема блока получатель передает последовательность подтверждения в виде: СИН СИН ДА. При обнаружении ошибок посылается отрицательный ответ: СИН СИН НЕТ. По завершении формирования ответа вновь переключается направление передачи, и станция, инициировавшая передачу, передает следующий блок, либо повторяет предыдущий, если получен отрицательный ответ. Направление передачи по каналу автоматически изменяется каждый раз, когда передаются знаки КТМ, КТ, КБ, ДА или НЕТ.

Наиболее распространенным протоколом байт-ориентированного типа является протокол **BSC** (*Binary Synchronous Communications*) фирмы IBM. Он был разработан для реализации в системах телеобработки данных ЭВМ серии IBM-360. Это универсальный протокол для коммутируемых и выделенных двух- и многоточечных каналов, для пакетного и диалогового режимов работы. Но так как эта процедура первично была разработана для пакетной обработки, то применение ее для диалогового режима не очень эффективно.

Байт-ориентированные протоколы передачи характеризуется относительно большим объемом избыточной информации, служащей для управления передачей. Наличие управляющих символов заметно снижает эффективную пропускную способность канала связи.

2.5.3. Бит-ориентированная передача данных

Более универсальной и эффективной для компьютерных сетей является бит-ориентированная передача данных. Ее преимуществом является сравнительно малый объем избыточной служебной информации и более простое обеспечение режима "прозрачной" передачи, то есть возможность передачи текстов, содержащих произвольные кодовые комбинации. Существует ряд аналогичных стандартных процедур для управления передачей данных на канальном уровне, к которым относятся *SDLC, HDLC, ADCCP* и *LAPB*.

Первый бит-ориентированный протокол был разработан и применен фирмой IBM для систем телеобработки на базе ЭВМ типа IBM-370. Этот протокол имеет сокращенное название **SDLC** (*Synchronous Data Link Control* – управление синхронным звеном передачи данных). На базе этого протокола, с некоторыми изменениями, в международной организации по стандартизации (ISO) был разработан протокол **HDLC** (*High-level Data Link Control* – управление звеном передачи данных высокого уровня). Аналогичная процедура явилась основой протокола **LAPB** (*Balanced Link Access Procedures* – балансная процедура доступа к линии), разработанного МККТТ (рекомендация X.25) для канального уровня сетей коммутации пакетов. Бит-ориентированный протокол ADCCP разработан Американским национальным институтом стандартов.

В соответствии с протоколом HDLC передача между станциями данных осуществляется кадрами (блоками) в полудуплексном или дуплексном режимах. Формат кадра показан на рисунке 2.33.

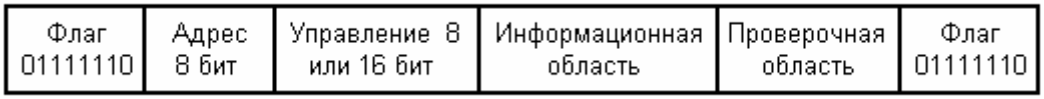


Рисунок 2.33 – Формат кадра HDLC-протокола

Во избежание ложного выделения флага внутри кадра в случае поступления от источника символа, совпадающего с флаговой комбинацией, на передающей стороне производится специальное преобразование знаков. Оно состоит в том, что после формирования кадра вся последовательность, подлежащая передаче, просматривается. Если в ней встречается подряд 5 единиц, то за ними автоматически вставляется дополнительный служебный бит "0". Эта операция называется "*бит-стаффингом*" и выполняется обычно аппаратным способом. На приемной стороне после регистрации подряд пяти "1" следующий за ним "0" исключается. Благодаря таким преобразованиям

обеспечивается передача любых кодовых комбинаций, т. е. процедура передачи становится "прозрачной".

Область адреса кадра представляет собой 8-битовую комбинацию, которая используется для выбора абонента, если передача ведется по многопунктовым линиям. Передача адреса, как и всех последующих символов кадра, производится, начиная с младшего разряда. **Область управления** обычно представляет собой 8-битовую комбинацию, хотя процедура HDLC допускает использование при расширенном формате 16-разрядной комбинации. Управляющий символ (задается программно) содержит команды и ответы для управления данными, а также определяет один из трех типов кадров:

Информационный (***I***-кадр) – является единственным кадром, который используется для переноса информации от источника;

Служебный нумерованный (супервизорный ***S***-кадр);

Служебный ненумерованный (***U***- кадр).

Структура области управления для трех типов кадров нерасширенного формата приведена в таблице 2.2. Первый бит области определяет, является ли кадр информационным (бит равен 0) или кадром команды ответа (бит равен 1). Биты 2…4 и 6…8 используются для задания порядковых номеров кадров NS на передаче и NR на приеме, которые формируются и обрабатываются на каждой станции для переданных и принятых I-кадров. В расширенном формате для передачи номеров NS и NR используются биты соответственно 2…8 и 10…16. С каждым переданным I-кадром NS увеличивается на 1, а увеличение NR на 1 происходит только при безошибочном приеме и в правильной очередности I-кадра.

Таблица 2.2 – Формат кадра бит-ориентированного протокола

Тип кадра	Порядок передачи битов в области управления в канал							
	8	7	6	5	4	3	2	1
I-кадр	NR			P/F	NS			0
S-кадр	NR			P/F	s	s	0	1
U-кадр	u	u	u	P/F	u	U	1	1

Каждая вторичная станция имеет свои собственные счетчики NS и NR по кадрам, принятым от первичной станции, которая в свою очередь ведет отдельный счет NS и NR для каждой вторичной станции звена. Нумерация принятых кадров предназначена для указания другой станции ожидаемого номера следующего кадра NR, что является подтверждением (*квитанцией*)

правильности приема станцией всех информационных кадров с номерами до NR–1.

Оставшийся 5-й бит (9-й для расширенного формата) является битом команды запрос/ответ (P/F), который при размещении в кадре команды интерпретируется как бит ответа F. Бит P/F используется для опроса (P/F=1) первичной станцией желаемой вторичной станции. Ответ может состоять из одного или нескольких кадров. Вторичная станция обычно использует бит P/F=1 для указания последнего кадра в передаваемой последовательности.

Служебные кадры не несут полезной информации, а являются кадрами команд/ответа и используются для обеспечения управления потоком данных в линии данных. Команды могут передаваться только первичной, а ответы – вторичной станциями. Нумерованные S – кадры, содержащие номер переданного NS или принятого NR кадров (Таблица 2.2), применяются первичной станцией для опроса подчиненных станций, либо для посылки квитанции (*квитирования*) I - кадров, запроса на повторную передачу или временной приостановки I - кадров. Биты S, расположенные на 3 и 4 позициях области управления, определяют команды или ответы (не более 4-х команд или 4-х ответов).

U - кадры не содержат номера и используются для образования дополнительных команд управления звеном передачи данных и дополнительных ответов и сообщений об ошибках алгоритма. 5 битов могут образовывать 32 дополнительные команды и 32 ответа.

Команды и *ответы* I- и S- кадров определяются следующим образом:

"**Передача информации**" – является признаком наличия в I-кадре информации от источника. Номер NS указывает на порядковый номер передаваемого I-кадра, а номер NR подтверждает все ранее принятые информационные кадры с номерами до NR–1 включительно;

"**Готов к приему**" – указывает, что передающая его станция готова к приему I-кадров;

"**Неприем**" – используется для запроса повторной передачи всех входящих кадров, начиная с NS;

"**Не готов к приему**" – свидетельствует о том, что станция временно не может принимать *I*-кадры;

"**Выборочный неприем**" – позволяет станции запросить повторную передачу конкретного кадра с номером NR.

Примером дополнительных команд *U*-кадра являются команды/ответы: "Установить расширенный формат", "Некорректный кадр", "Разъединение" и др.

Информационная область кадра предназначена для размещения данных, которые могут иметь любую битовую структуру. Длина информационной области ограничивается предельной длиной кадра, определяемой степе-

нью образующего полинома циклического кода, используемого для защиты передаваемого сообщения от ошибок. При необходимости информационное поле блока может быть занято для размещения дополнительных управляющих знаков. Характер информации, расположенной в информационной части кадра, определяется командой или ответом, которые содержатся в области управления этого кадра. Размер информационной области должен быть кратным байту.

Проверочная область кадра содержит контрольную последовательность, получаемую в результате деления остальных областей кадра, исключая флаговую комбинацию, на образующий полином циклического кода. Длина проверочной области в битах равна степени образующего полинома. Вид образующего полинома $X^{16}+X^{12}+X^5+1$. Он стандартизирован рекомендацией международного консультативного комитета по телефонии и телеграфии (МККТТ).

Работа протокола *HDLC* заключается в обмене *I*-, *S*- и *U*-кадрами между двумя станциями. Она состоит из трех фаз. Сначала одна из сторон занимает канал передачи данных и сигнализирует удаленной станции о запросе инициализации. На этой фазе стороны договариваются о параметрах обмена, которые они будут использовать (задание главной станции, указание 3- или 7-битовых порядковых номеров блоков). Затем наступает фаза обмена данными и управляющими пакетами для устранения ошибок и управления потоком. В третьей фазе одна из станций сигнализирует о завершении работы.

2.5.4. Связь скорости передачи сигналов с полосой пропускания

Знание спектров сигналов, используемых для передачи данных, динамики их изменения при различных видах сигналов, способов и параметров модуляции, а также переходных процессов в каналах при передаче этих сигналов, позволяет установить соотношения между скоростью передачи и требуемой шириной полосы пропускания используемого канала связи.

На практике нет необходимости (да и возможности) передавать весь спектр сигнала. Достаточно передать лишь те составляющие, в которых сосредоточена основная часть энергии (>50%). Так, например, при передаче “точек” импульсами постоянного тока, основная часть энергии содержится в двух первых компонентах спектра: постоянной составляющей и первой гармонике с частотой $f=1/2\tau_0$, где τ_0 – длительность единичного элемента. Следовательно, минимально необходимая полоса частот канала связи в этом случае равна

$$\Delta F_{min} = 1/2\tau_0 = B/2, \qquad (2.30)$$

где В – скорость модуляции, бод. Анализируя спектры других видов сигналов можно заметить, что через такую полосу всегда пройдет основная часть энергии этих сигналов. Следовательно, предельная скорость модуляции при передаче импульсами постоянного тока равна

$$B_{max} = 2\Delta F_{эф},. \qquad (2.31)$$

где $\Delta F_{эф}$ – ширина полосы пропускания канала. Эта формула называется *формулой Найквиста.*

При передаче модулированных сигналов предельная скорость модуляции снижается вдвое за счет передачи двух боковых полос, т.е.

$$\Delta F_{min} = 1/\tau_0 = B, \quad а \qquad B_{max} = \Delta F_{эф.} \qquad (2.32)$$

Эти же формулы справедливы для ЧМ и ФМ при малых индексах модуляции $m_{ЧМ,}\, m_{ФМ} < 1$. Если передача модулируемых сигналов осуществляется с одной боковой полосой, то требуемая ширина полосы канала связи уменьшится вдвое, т.е.

$$B_{max} \approx 2\Delta F_{эф}. \qquad (2.33)$$

2.6. Способы защиты от ошибок на канальном уровне

2.6.1. Общая характеристика способов защиты передаваемых данных

В процессе передачи информации по линиям и каналам связи компьютерных сетей на сигналы данных воздействуют различного рода помехи. Кроме того, они дополнительно искажаются за счет неидеальности частотных характеристик тракта передачи. В большой степени задача борьбы с помехами и искажениями в линиях и каналах связи решается в модуляторах и демодуляторах систем передачи данных путем использования оптимальных сигналов и схем приема. Однако, несмотря на это, результирующая вероятность ошибки по единичным элементам на выходе демодулятора находится в пределах $10^{-3}…10^{-5}$. Нижняя граница этого диапазона соответствует цифровым каналам, образованным по коммутируемым линиям городских телефонных сетей, и декаметровым каналам связи. Верхняя граница относится к каналам, организованным на выделенных (некоммутируемых) линиях и каналах, получаемым за счет уплотнения линий каналообразующей аппаратурой с ИКМ. В трактах передачи с использованием оптических линий вероятность ошибки по единичным элементам существенно ниже и находится в

пределах $10^{-6}…10^{-8}$. В то же время вероятность ошибки по символам (байтам) при передаче данных в компьютерных сетях общего пользования должна быть не хуже 10^{-6}, а в некоторых сетях требуется обеспечивать вероятность ошибок по символам не более 10^{-8}. Из этого следует, что перед обработкой информации в компьютере верность передачи ее на канальном уровне должна быть повышена на 3…5 порядков (от 10^{-3} по элементам до 10^{-8} по символам).

Снижение количества ошибок может быть обеспечено за счет улучшения качественных характеристик линий и каналов связи. В настоящее время происходит процесс замены магистральных проводных линий связи на оптоволоконные. Однако этот процесс очень трудоемкий, дорогостоящий и продолжительный во времени. На нижнем уровне сетей еще долго будут оставаться проводные линии связи. А каналы радиосвязи, которые все шире используются в беспроводных компьютерных сетях, практически невозможно защитить от помех.

Значительного повышения качества связи можно достичь только с помощью дополнительных методов защиты от ошибок, реализуемых специальными **устройствами защиты от ошибок** (УЗО), либо программным способом при обработке данных. Применение УЗО в виде отдельных аппаратных средств характерно для канального уровня сети, где защита от ошибок программным способом требует значительных затрат машинного времени. В настоящее время УЗО обычно интегрируется в одной БИС совместно с модулятором-демодулятором и системой синхронизации.

Методы защиты от ошибок зависят от типа каналов, применяемых для передачи данных и регламентируются соответствующими протоколами канального уровня (BSC, HDLC и др.). При симплексных каналах для защиты информации используется ***многократное повторение*** одного и того же блока (кадра) данных или ***корректирующие коды***, исправляющие ошибки. В случае дуплексных и полудуплексных каналов защита информации осуществляется с помощью ***кодов, обнаруживающих ошибки***. Блок, в котором обнаружены искаженные символы, повторяется после запроса, посылаемого по постоянно действующему каналу обратной связи (ОС) на передающую сторону.

При многократной передаче каждая кодовая комбинация (или блок) передается нечетное количество раз (обычно 3…5), а на приеме производится сравнение принятых знаков, и решение принимается "голосованием по большинству" (*мажоритарный метод*). Выбирая нужное количество повторений *К*п, можно обеспечить сколь угодно малую вероятность ошибок, но и эффективная скорость передачи при этом снижается в *К*п раз.

Поток ошибок в канале отличается большой неравномерностью; ошибки часто группируются в пакеты, разделенные интервалами, в течение

которых ошибки появляются редко. Кодовые методы исправления ошибок требуют больших аппаратурных затрат или времени, и поэтому в системах телеобработки находят относительно редкое применение (в ряде случаев используются коды Хемминга).

Большое распространение для исправления ошибок получили системы с обратной связью. Системы с ОС, в зависимости от назначения канала обратной связи, делятся на системы с автоматическим запросом повторения блока при ошибках **ARQ** (*Automatic Repeat reQuest)* и системы с *информационной обратной связью* (**ИОС**). В таких системах обратная связь используется для информирования передатчика о текущем состоянии канала передачи данных и изменения избыточности передачи в зависимости от количества и характера ошибок.

В системах с автоматическим запросом решение о необходимости повторения информационного блока вырабатывается в приемнике путем анализа его на отсутствие ошибок. Если ошибки не обнаружены и имеется свободный буфер для записи блока, то в канал ОС посылается подтверждение (*квитанция*) правильности приема, а в противном случае – запрос повторной передачи ошибочно принятого блока.

В системах с ИОС по каналу обратной связи осуществляется передача всего принятого информационного блока, который на передающей стороне сравнивается с переданным. При их совпадении в канал связи поступает следующий блок, а при обнаружении ошибок в прямой канал посылается команда "Стирание", и искаженный при передаче блок передается повторно. В более сложных системах с ИОС по обратному каналу передается не весь блок, а некоторая комбинация, отражающая характерные признаки принятого сообщения. Способ передачи данных с информационной обратной связью используются в компьютерных сетях относительно редко. Он применяется преимущественно в упрощенном виде – так называемый "эхоплекс". В соответствии с этим способом каждый символ, посылаемый компьютером на удаленный пункт, возвращается по каналу обратной связи в виде "эха".

В системах передачи данных количество повторений одного и того же блока ограничивается, и при превышении заданного числа повторений сигнализируется аварийное состояние канала связи.

Для обнаружения и исправления ошибок, возникающих при передаче данных, разработано большое количество различных кодов. Все множество кодов делится на блочные и непрерывные. В *блочных кодах* передаваемая информационная последовательность разбивается на отдельные блоки, которые кодируются и декодируются независимо друг от друга. В *непрерывных кодах*, называемых также *сверточными*, передаваемая информационная последовательность не разделяется на блоки, а проверочные элементы размещаются в определенном порядке между информационными. Процессы коди-

рования и декодирования в сверточных кодах также имеют непрерывный характер.

Блочные коды в свою очередь делятся на разделимые и неразделимые. В разделимых кодах функции элементов, входящих в блок разграничены: одни разряды являются информационными, а другие – проверочными. Во всех блоках данного кода информационные и проверочные элементы занимают одни и те же позиции. Разделимые коды обозначают как коды (n,k), где n – разрядность блока, k – число информационных разрядов в блоке. В неразделимых кодах деление на информационные и проверочные разряды отсутствует. К таким кодам относятся коды с постоянным весом.

Корректирующая способность кода зависит от кодового расстояния d, численно равного минимальному количеству элементов, которыми отличается любая кодовая комбинация от другой. В общем случае должно соблюдаться равенство

$$d=t_{\text{о}}+t_{\text{и}}+1 ,$$

где $t_{\text{о}}$ и $t_{\text{и}}$ – число обнаруживаемых и исправляемых ошибок соответственно, причем обязательно условие $t_0 \geq t_{\text{и}}$.Если код только обнаруживает ошибки, то $d=t_0+1$, а в случае только исправления – $d = 2t_{\text{и}} +1$. Количество проверочных элементов r корректирующего кода зависит от вида кода, а число информационных элементов k равно $k = n–r$. Отношение r/n называют коэффициентом избыточности корректирующего кода.

Ниже будут рассмотрены только несколько видов кодов, которые нашли наибольшее применение для защиты информации от ошибок в компьютерных сетях.

2.6.2. Передача данных с автоматическим запросом

Существует несколько разновидностей систем с автоматическим запросом повторной передачи *ARQ*: системы *ARQ* с ожиданием подтверждения; системы *ARQ* с непрерывной передачей и системы с выборочным (*селективным*) запросом. В русскоязычной литературе системы с автоматическим запросом получили название системы с решающей обратной связью. Самыми простыми и достаточно распространенными в компьютерных сетях являются системы *ARQ* с ожиданием подтверждения (*ARQ*-О). Структурная схема системы *ARQ*-О изображена на рисунке 2.34.

Кодовая n-разрядная комбинация, поступающая от источника информации (ИИ), через логическую схему ИЛИ, записывается в буферный накопитель (БН), кодируется помехоустойчивым кодом в кодере (К) и с помощью передатчика (Пд) выдается в прямой канал связи (ПКС). Затем источ-

ник информации ИИ останавливается, и передача данных прекращается до приема сигнала подтверждения **АСК** (*Acknowledgement*).

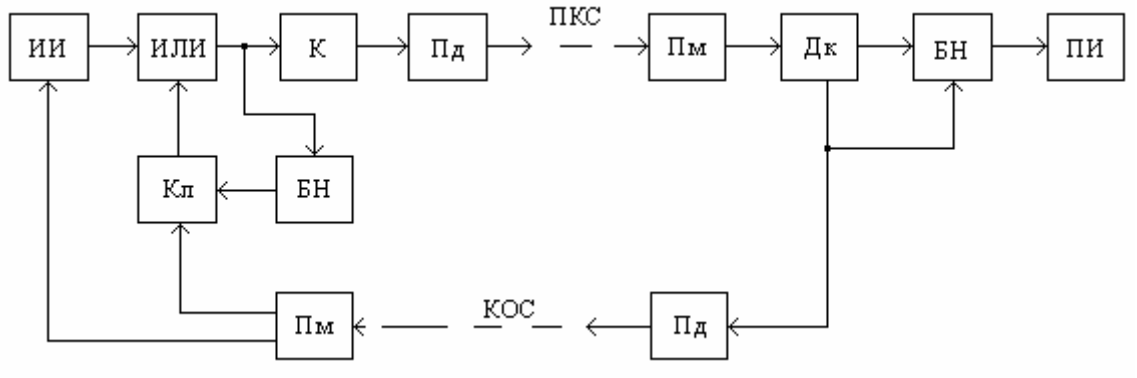


Рисунок 2.34 – Структурная схема системы передачи данных с автоматическим запросом *ARQ*-О

На удаленной станции поступающее сообщение регистрируется приемным устройством (Пм), декодируется в блоке Дк, заносится в БН и при отсутствии в нем ошибок выдается потребителю информации (ПИ), а по каналу обратной связи (КОС) передается сигнал подтверждения правильности приема АСК. При обнаружении ошибки Дк посылает через передатчик канала ОС сигнал повторного запроса искаженного блока **NAK** (*Negative Acknowledgement*), а принятая комбинация, находящаяся в БН приемника, стирается.

На передающей стороне приемник сигналов обратной связи, приняв запрос NAK, формирует управляющий сигнал, который открывает ключ (Кл), и блок, находящийся в БН передатчика, повторно поступает в ПКС. Одновременно передаваемый блок вновь запоминается в накопителе БН. В случае приема сигнала подтверждения Кл остается закрытым, и от ИИ запрашивается новый блок данных, который параллельно с кодированием и передачей в ПКС заносится в БН на место предыдущего.

При трансформации сигналов запроса NAK в подтверждение АСК под действием помех в канале ОС, возможно пропадание блока, а при обратном преобразовании АСК в NAK один и тот же блок передается дважды, т.е. имеет место так называемая "вставка" блока (кадр-дубликат). Для уменьшения вероятности выпадения или вставки используют циклическую нумерацию блоков, а на приемной стороне контролируют очередность их поступления. При нарушении очередности предыдущий блок запрашивается вновь.

Достоинством систем ARQ-О является их простота, а недостатком – потери времени на ожидание сигнала подтверждения или запроса. Такие системы целесообразно применять в полудуплексных каналах ПД, когда по-

сле завершения передачи блока устройство переключается в режим приема сигнала по каналу ОС, и время ожидания частично совпадает со временем переключения направления передачи.

Более высокую эффективную скорость передачи информации по каналам связи обеспечивают системы ARQ с непрерывной передачей (ARQ-НП). Для реализации этого алгоритма требуется полнодуплексное соединение. В таких системах закодированные помехоустойчивым кодом блоки данных поступают непрерывно в ПКС без ожидания сигнала подтверждения. Максимальное количество блоков W, которое можно передать без подтверждения их приема, называют ***шириной окна***. Одновременно идет запись информации в буферный накопитель. При обнаружении ошибки в информационном блоке приемник передает по каналу ОС сигнал запроса NAK и блокирует запись в приемный накопитель последующих $W - 1$ блоков, что предотвращает возможность нарушения очередности выдачи блоков потребителю информации.

Передающая сторона, получив сигнал запроса NAK, прекращает подачу в ПКС новых сообщений и повторяет из БН все комбинации, начиная с той, на которую поступил запрос. Номер запрашиваемого блока определяется по времени поступления сигнала NAK.

Современные канальные протоколы предусматривают семикадровые окна. Это означает, что ООД может посылать семь блоков без получения ответного подтверждения. Для предотвращения вставок и дублирования блоков по причинам, указанным выше, применяют циклическую нумерацию блоков по модулю $W + 1$.

С целью уменьшения объема информации, передаваемой при запросах, разработаны системы ARQ и адресным переспросом (ARQ-АП), отличия которых состоит в том, что по обратному каналу передается сигнал запроса с указанием номеров (адресов) ошибочно принятых блоков.

2.6.3. Блочное кодирование

В компьютерных сетях для защиты информации от ошибок из блочных кодов наиболее широко используются коды с проверкой по паритету, коды Хемминга и циклические коды.

Коды с проверкой на четность/нечетность (по *паритету*) являются одними из простых видов кодов, позволяющих обнаруживать одиночные ошибки. Они образуются добавлением к передаваемой комбинации, состоящей из k информационных элементов безизбыточного кода, одного контрольного бита так, чтобы общее количество единиц в передаваемой комбинации было четным или нечетным. Если контрольный бит выбран так, что

результат суммирования четный, то имеет место положительный паритет (*even parity*); если при добавлении контрольного бита результат будет нечетным, то отрицательный паритет (*odd parity*). В итоге общее количество битов в передаваемой комбинации $n = k + 1$. На приемной стороне производят проверку поступающей последовательности на четность (либо нечетность). При четном (нечетном) числе единиц предполагается, что ошибок нет, и потребителю выдаются k информационных битов, а контрольный элемент отбрасывается.

Вероятность обнаруживаемых ошибок на четность (нечетность) n – разрядной кодовой комбинации $P_{кк}$ зависит от длины кодовой комбинации n и вероятности ошибочного приема единичных элементов P_0

$$P_{кк} \approx C_n^2 P_0^2 (1-P_0)^{n-2},$$

где C_n^m – число сочетаний из n по m; $C_n^m = n! / (m!(n-m)!)$.

Итеративные коды являются разновидностью кодов с проверкой на четность. В англоязычной литературе их также называют прямоугольными кодами (*rectangular code*). Они характеризуется наличием двух или более систем проверок внутри каждой кодовой комбинации. Итеративный код строится следующим образом. Из элементов передаваемого блока формируется прямоугольная матрица, состоящая из M строк и N столбцов. Затем к каждой строке и каждому столбцу прибавляются биты паритета, что в результате дает матрицу размером $(M+1)\times(N+1)$

$$\begin{pmatrix} a_{11}\ a_{12}\ a_{13}\ \ldots a_{1n}\ r_1 \\ a_{21}\ a_{22}\ a_{23}\ \ldots a_{2n}\ r_2 \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ a_{71}\ a_{72}\ a_{73}\ \ldots a_{7n}\ r_7 \\ q_1\ \ q_2\ \ \ q_3\ \ \ldots q_n\ \ q_{n+1} \end{pmatrix},$$

где a_{ij} (i=1..7; j=1,2,..,n) – информационные биты; $q_1, q_2, q_3, .., q_n$ – проверочные биты знаков, образующие первую совокупность проверок; r_i (i=1..7) – контрольные элементы, которые являются суммой по модулю 2 всех элементов строки; q_{n+1} – контрольный элемент проверочных битов знаков. Каждый знак нужно передавать последовательно, начиная с первого бита, $a_{1,j}$ и кончая восьмым проверочным q_j.

Приведенный итеративный код является простейшим кодом этого класса с кодовым расстоянием d=4. Он обнаруживает ошибки кратности до 3 и все ошибки нечетной кратности, а также любой пакет ошибок длиной S+1,

где S – длина строки матрицы кода. Основным недостатком итеративных кодов, использующих проверки на четность по столбцам и строкам, является высокая избыточность – около 15%. Однако декодирование и кодирование таких кодов очень просто реализуется программными методами. При более жестких требованиях по достоверности передачи данных применяется итеративный код с тремя проверками. Причем, третья дополнительная проверка на четность осуществляется по диагоналям матрицы.

Код Хэмминга – один из наиболее эффективных кодов, позволяющий исправлять одиночные ошибки. Кодовое расстояние d=3. Этот код образуется дополнением информационной части передаваемого блока, состоящей из k битов, r проверочными элементами. При выборе длины передаваемого блока n и количества проверочных элементов r руководствуются неравенством

$$2^r \geq n+1.$$

Учитывая, что $r = n–k$, неравенство может быть представлено в виде

$$2^k \leq 2^n / (n+1) , \qquad (2.34)$$

где n и k принимают только целые значения. Неравенство является исходным для определения длины кодовой комбинации по заданному числу k.

Первый проверочный бит $П_1$ кода Хэмминга образуется суммированием по модулю 2 всех нечетных элементов блока, начиная с первого, т.е.

$$П_1=a_1 \oplus a_3 \oplus a_5 \oplus a_7 \ldots . \qquad (2.35)$$

Результат проверки $П_2$ определяет второй разряд проверочной комбинации (*синдрома ошибки*). Он вычисляется суммированием тех элементов блока, номера которых соответствуют n–разрядным двоичным числам, имеющим единицу во втором разряде,

$$П_2 = a_2 \oplus a_3 \oplus a_6 \oplus a_7 \oplus a_{10} \oplus a_{11} \ldots . \qquad (2.36)$$

Третья проверка $П_3$ охватывает разряды, номера которых соответствуют n - разрядным двоичным числам, имеющим единицу в третьем разряде.

$$П_3=a_4 \oplus a_5 \oplus a_6 \oplus a_7 \oplus a_{12} \oplus a_{13} \oplus a_{14} \oplus a_{15} \ldots . \qquad (2.37)$$

Аналогичным образом находятся разряды, охватываемые четвертой, пятой и т.д. проверками

$$П_4=a_8 \oplus a_9 \oplus a_{10} \oplus a_{11} \oplus a_{12} \oplus a_{13} \oplus a_{14} \oplus a_{15} \ldots \quad (2.38)$$

$$П_5=a_{16} \oplus a_{17} \oplus a_{18} \oplus a_{19} \oplus a_{12} \oplus a_{20} \oplus a_{21} \ldots . \quad (2.39)$$

В принципе местоположение проверочных элементов не имеет значения. Их можно размещать перед информационными символами, после них, чередуя с информационными. Если их расположить на местах, кратных степени 2, т.е. на позициях 1, 2, 4, 8 и т.д., то код двоичного числа, образованного проверочными элементами, на приемной стороне будет указывать номер разряда, в котором произошла ошибка.

Основной операцией в кодирующих и декодирующих устройствах кода Хэмминга является суммирование по модулю 2 передаваемых единичных элементов в соответствии с формулами (2.35–2.39). Таким образом, их схема мало отличается от кодеров итеративного кода. Отличие состоит только в схеме образования проверочных элементов. Для упрощения технической реализации (исключения многоразрядных параллельных сумматоров, входного накопителя) вначале в канал посылаются информационные биты, а затем – проверочные. При таком способе формирование контрольных элементов осуществляется с помощью одноразрядных последовательных сумматоров по модулю 2 одновременно с передачей информационных разрядов. Чтобы сохранить корректирующие свойства кода Хэмминга, необходимо произвести перестановку разрядов в проверочных соотношениях (2.35–2.39) с учетом изменения номеров позиций суммируемых элементов за счет вынесения в конец блока проверочных битов.

Циклические коды. Этот вид кодов находит наибольшее распространение в системах передачи данных с автоматическим запросом ARQ, что обусловлено их высокими корректирующими свойствами, сравнительно простой реализацией, низкой избыточностью. Особенно эффективны эти коды при обнаружении пакетов ошибок. Циклические коды относят к блочным систематическим кодам, где каждая комбинация кодируется самостоятельно в виде блока таким образом, что информационные k и проверочные r элементы всегда находятся на определенных местах. Для упрощения процедуры кодирования и декодирования проверочные биты размещают в конце блока. Кодирование передаваемого сообщения осуществляется умножением двоичной последовательности $G(x)$ на одночлен x^r, имеющий ту же степень, что и образующий полином $P(x)$, с добавлением к этому произведению остатка $R(x)$, полученного после деления произведения $G(x)x^r$ на образующий полином. Таким образом, передаваемое в канал связи сообщение $F(x)$ имеет вид:

$$F(x) = G(x)x^r + R(x). \quad (2.40)$$

При декодировании принимаемая последовательность $F(x)$ снова де-

лится на образующий полином *P(x)*. Полученный нулевой остаток *R(x)=0* свидетельствует об отсутствии ошибок в принятом блоке, а отличие от нуля – о наличии ошибок. Анализируя вид остатка, можно определить номера искаженных разрядов и скорректировать их.

Для построения циклических кодов в качестве образующих полиномов используются *неприводимые многочлены*. Неприводимыми называются многочлены, делимые без остатка только на самого себя и на единицу. *P(x)* может быть представлен в алгебраической форме, либо в виде двоичного или восьмеричного числа. Например, для полинома вида $x^8+x^4+x^3+x+1$ двоичная запись имеет вид *100 011 011*, а соответствующая ему восьмеричная – 433.

При выборе образующего полинома *P(x)* следует иметь в виду, что степень полинома не может быть меньше числа проверочных элементов *r*.

Для систем передачи данных общего пользования с *ARQ* и использованием циклических кодов в режиме обнаружения ошибок стандартами предусмотрено несколько видов полиномов:

$$x^8+x^4+x^3+x+1; \quad x^{16}+x^{12}+x^5+1; \quad x^{16}+x^{15}+x^{13}+x^{11}+x^5+x^3+x+1.$$

Полиномам 16-й степени соответствует проверочная область длиной 16 битов. Стандартом разрешается в обоснованных случаях использовать полиномы 24-й и 32-й степеней с длиной проверочных областей 24 и 32 бита соответственно.

Для построения кодирующего устройства циклического кода необходимо в соответствии с (2.40) выполнить две процедуры: умножить многочлен *Q(x)* на x^r и полученное произведение разделить на образующий полином *P(x)* по модулю *P(x)*. Для проведения первой операции не требуется специального устройства, так как умножение многочлена на x^r означает добавление к нему *r* нулей со стороны младшего разряда, т.е. после передачи *k* информационных элементов за ними следуют *r* проверочных. Реализация процедуры циклического кодирования программными методами требует больших затрат машинного времени. В то же время схемотехнические кодеры и декодеры циклических кодов отличаются малыми аппаратурными затратами и способностью работать в реальном времени. Поэтому в современных программируемых УЗО, сетевых картах и различных адаптерах связи циклическое кодирование и декодирование реализуется преимущественно аппаратно.

В качестве делителей полинома на полином в кодеках циклических кодов применяются устройства, построенные на основе регистров сдвига с обратными связями и сумматоров по модулю 2, причем схема делителя определяется видом образующего полинома. Количество триггеров регистра

сдвига выбирается равным степени образующего полинома *r*. Ячейка регистра для старшей степени исключается, но всегда присутствует триггер, соответствующий нулевой степени x^0. Число сумматоров по модулю 2 в регистре должно быть на единицу меньше ненулевых членов выражения *P(x)*. Сумматоры располагают перед ячейками регистра, соответствующими ненулевым членам образующего полинома. На первые входы сумматоров подаются сигналы с предыдущих ячеек регистра, а на вторые – с выхода делителя. Очевидно, нет необходимости ставить сумматор перед ячейкой x^0. На рисунке 2.35 в качестве примера изображена структурная схема кодирующего устройства циклического кода с образующим полиномом x^5+x^2+1.

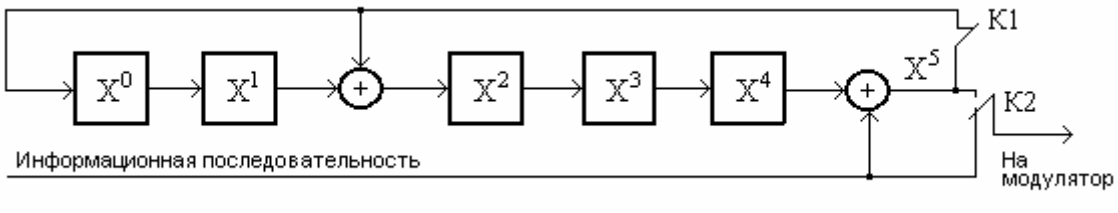


Рисунок 2.35 – Пример реализации кодирующего устройства циклического кода

Прямоугольниками на схеме обозначены ячейки памяти (триггеры), а кружками сумматоры по модулю 2. В исходном состоянии ключ К1 замкнут, а ключ К2 находится в нижнем положении. Информационная последовательность, подлежащая кодированию, подается одновременно на вход модулятора и через сумматор – в схему деления на образующий полином *P(x)*. Деление начинается с приходом первого информационного элемента и прекращается после выдачи в дискретный канал *k*-го бита. Затем схема управления УЗО переводит К2 в верхнее положение, размыкает К1, и в течение последующих *r* тактов осуществляется выдача в модулятор остатка от деления, зафиксированного триггерами делителя. Причем, при выводе остатка из делителя ввод данных от источника прекращается.

Основу декодирующих устройств циклических кодов также составляют делители многочленов на образующий полином. Признаком наличия ошибок принятой последовательности является ненулевой остаток от деления этой последовательности на полином *P(x)*. До завершения процесса деления необходимо хранить поступивший блок в буферном накопителе. После окончания цикла производится опрос делителя, и в случае ошибки принятый блок стирается. При нулевом остатке блок выводится получателю через ключевой элемент Кл, а на его место в буферный накопитель записываются следующие информационные элементы. Структурная схема декодера изображена на рисунке 2.36.

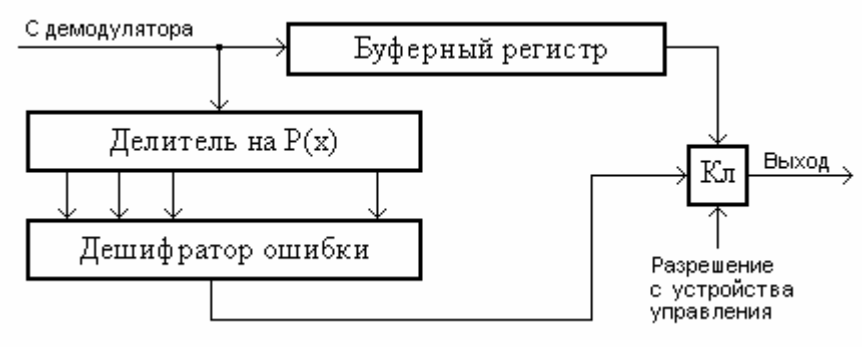


Рисунок 2.36 – Структурная схема декодера циклического кода

В процессе приема устройство управления вырабатывает управляющие импульсы таким образом, что в буферный регистр заносятся только информационные биты, а на делитель подаются все элементы, которые участвовали в процессе деления в кодере, а также проверочная последовательность *R(x)*.

2.6.4. Сверточное кодирование

При сверточном кодировании преобразование информационных последовательностей в кодовые происходит *непрерывно*. Кодер двоичного сверточного кода (СК) содержит регистр сдвигов на *K*-разрядов и сумматоры по модулю 2 для образования кодовых символов. Параметр *K* называется кодовым ограничением. На рисунок 2.37,а изображена схема кодера с *K*=3. Входы сумматоров соединены с определенными разрядами регистра. Коммутатор (Sw) на выходе устанавливает очередность посылки кодовых символов в канал. В общем случае эффективная скорость передачи кода *R=k/n*, где *k*–число информационных символов, поступающих за один такт на вход кодера, *n*–количество соответствующих им символов на выходе. В нашем случае скорость кода R=1/2 (рисунок 2.37,а). На рисунке 2.37,б показана схема кодера с эффективной скоростью R=2/3.

Сверточный кодер, как конечный автомат с памятью, описывают диаграммой состояний. Внутренними состояниями кодера считают символы S_1S_2 (рисунок 2.37,а). Кодер может находиться в 4-х состояниях S_1S_2=(00,01,10,11). Диаграмма состояний представляет собой направленный граф, который содержит все состояния и описывает возможные переходы из одного состояния в другое, а также символы выхода кодера, сопровождающие эти переходы.

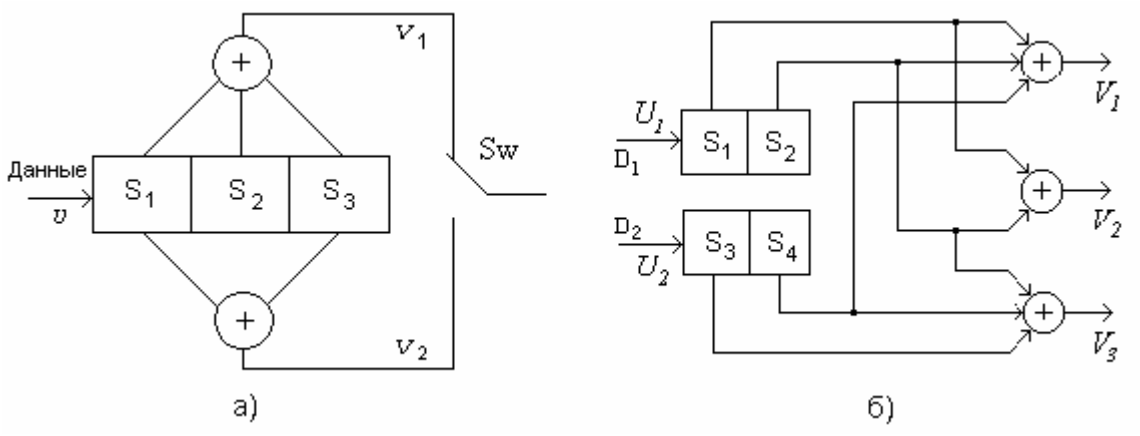


Рисунок 2.37 – Схемы кодеров сверточного кода со скоростью 1/2 (а) и 2/3 (б)

Диаграмма состояний показана на рисунке 2.38,а. В кружках указаны состояния кодера, стрелками – возможные переходы. Около стрелок показаны символы на выходе кодера V(1) V(2), соответствующие каждому переходу.

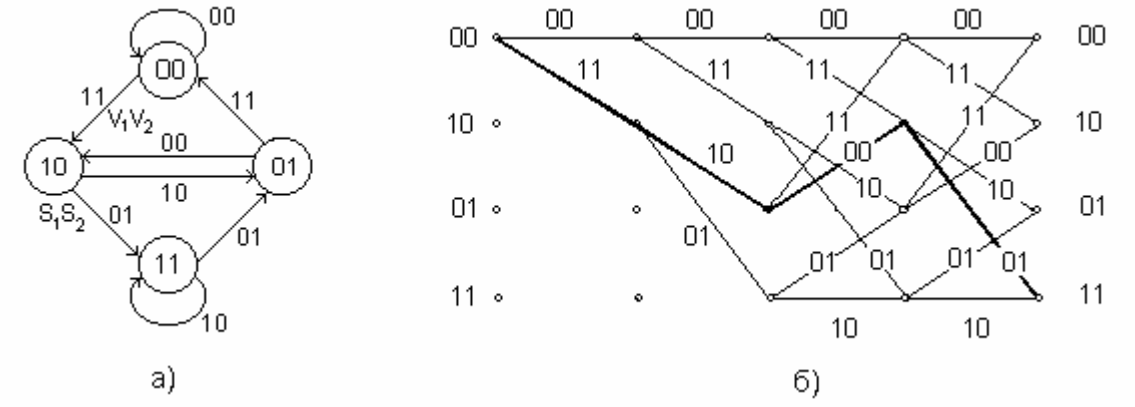


Рисунок 2.38 – Диаграмма функционирования а) и решетчатая диаграмма б) сверточного кодера

Кодер работает следующим образом. Первоначально он находится в состоянии 00 и поступление на вход символа 0 переводит его также в состояние 00. На выход кодера выдаются символы $V_1V_2 = 00$. На диаграмме этот переход обозначают петлей 00 около состояния 00. Далее, при поступлении символа 1, кодер переходит в состояние 10 и на его выходе будут символы 11. Этот переход обозначают стрелкой из состояния 00 в состояние 10, и т.д. Построение диаграммы заканчивается, когда просмотрены возможные переходы из каждого состояния во все остальные. Развертка состояний во времени образует *решетчатую диаграмму* (рисунок 2.38,б). На решетке состояния показаны узлами, а переходы – соединяющими их линиями. После

каждого перехода из одного состояния в другое происходит смещение на один шаг вправо. Решетчатая диаграмма изображает все разрешенные пути, по которым может продвигаться кодер при кодировании. Жирной линией на рисунке 2.38,б показан путь по решетке 11 10 00 01, соответствующий поступлению на вход кодера последовательности 1011. Соответственно ветвям решетчатой диаграммы по мере продвижения получаем кодовые посылки 11.10.00.01... и т.д.

На рисунке 2.39 изображена решетчатая диаграмма при поступлении на вход кодера двоичной последовательности 00111000010110001101. Путь по диаграмме отмечен жирной линией, которому соответствуют двоичные комбинации:

00.00.11.01.10.01.11.00.00.11.10.00.01.01.11.00.11.01.

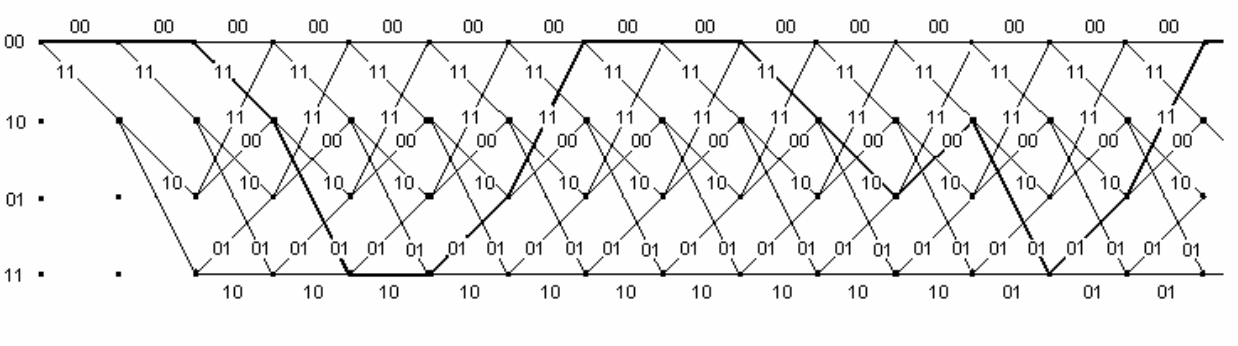


Рисунок 2.39 – Решетчатая диаграмма кодирования последовательности вида 00111000010110001101

Для декодирования сверточных кодов наиболее широко применяется *алгоритм декодирования* ***Витерби*** (алгоритм максимального правдоподобия). Алгоритм имеет ряд преимуществ перед другими, в связи с чем его широко используют для декодирования коротких сверточных кодов. Рассмотрим алгоритм декодирования на примере кода со скоростью R=l/2. Из дискретного канала поступают кодовые посылки: 00 00 11 01 10 01 11 00 00 11 10 00 01 01 11 00, которые попадают на вход декодера. Функционирование декодера, как и кодера, основано на развитии и построении решетчатой диаграммы. Процесс развития решетчатой диаграммы показан на рисунке 2.40. Здесь на диаграмме под вертикальными рядами состояний указаны шаги декодирования, слева проставлены слова-состояния, а в скобках рядом с состояниями – информационные символы, соответствующие выходной (декодированной) последовательности.

Развитие диаграммы для кодеров R=1/2 происходит всегда за 3 шага. В кружках обозначаются метрики состояний. В начальный момент времени полагаем, что декодер находится в состоянии 00 и исходная метрика МС(00)=0.

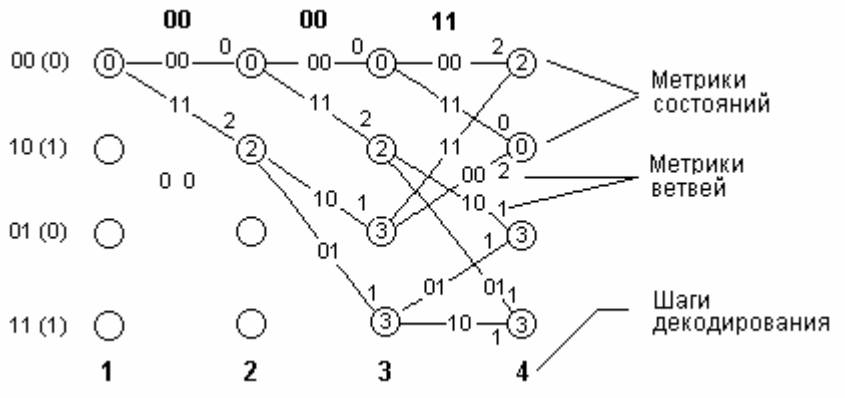


Рисунок 2.40 – Процесс развития решетчатой диаграммы при декодировании

Это показано на первом шаге декодирования. К каждому новому состоянию ведет две ветви (см. диаграмму состояний рисунок 2.38,б). В процессе развития решетчатой диаграммы к новому состоянию ведет только одна ветвь. Из состояния 00 ветвь идет к состояниям 00 и 10. Для этих ветвей необходимо вычислить *метрику ветви*. Метрика ветви (МВ) равна расстоянию Хэмминга между набором символов на выходе декодера (символы входа обозначены на рисунке 2.40 вверху диаграммы жирным шрифтом) и набором символов, соответствующих данной ветви на решетчатой диаграмме (рисунок 2.38,б).

Расстояние Хэмминга между двумя двоичными словами равно числу битов, в которых они отличаются. Для вычисления расстояния используют посимвольное сложение по модулю 2. Для ветви, ведущей в состояние 00, МВ(00)=0; для ветви, ведущей в состояние 10, МВ(10)=2. Это есть основная процедура шага декодирования, т.е обработка декодером принимаемых из канала данных в интервале между двумя соседними уровнями узлов.

Далее вычисляются метрики следующих состояний. Если других ветвей в этих состояниях нет, то метрики состояний определяются как суммы метрик входящих ветвей с метриками предыдущего состояния. Для 2-го шага декодирования метрики состояния равны:

Шаг кодирования: (2)
Метрики состояний: МС(00) = 0+0=0;
МС(10) = 2+0=2.

Затем вычисляются метрики ветвей (их уже 4) и метрики состояний третьего шага декодирования. Этот процесс виден на диаграмме (рисунок 2.40) на 3-м шаге декодирования. Вычисление осуществлялось следующим образом:

Шаги кодирования: (3) (2)

Метрики состояний:

$$MC(00) = MB(00)+MC(00) = 0+0=0;$$
$$MC(10) = MB(11)+MC(00) = 2+0=2;$$
$$MC(01) = MB(10)+MC(10) = 1+2=3;$$
$$MC(11) = MB(01)+MC(10) = 1+2=3.$$

На этом процесс развития решетчатой диаграммы заканчивается. Процесс построения решетчатой диаграммы для декодирования последовательности вида 00.00.11.01.10.01.11.00.00.11.10.00.01.01.11.00 показан на рисунке 2.41.

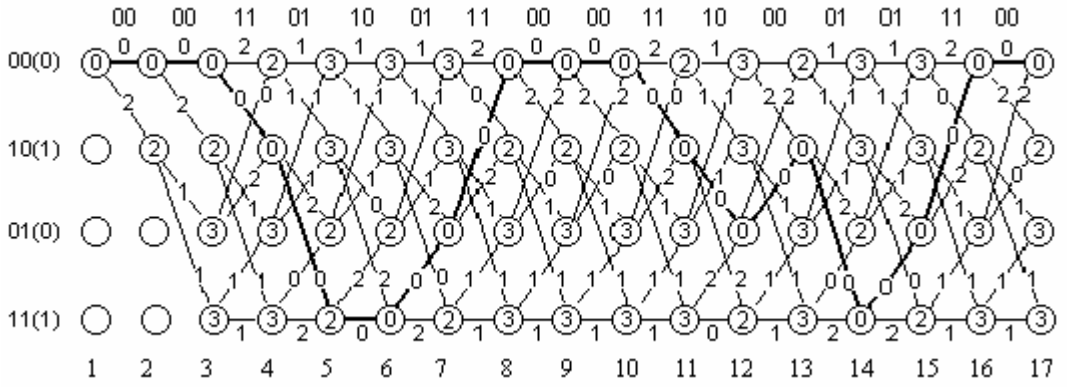


Рисунок 2.41 – Решетчатая диаграмма декодирования входной последовательности 00.00.11.01.10.01.11.00.00.11.10.00.01.01.11.00

Далее алгоритм периодически повторяет один основной шаг. В момент времени i в памяти декодера хранятся метрики состояний, вычисленных на предыдущем шаге:

$$MC(00)^{(i-1)};\ MC(10)^{(i-1)};\ MC(01)^{(i-1)};\ MC(11)^{(i-1)}.$$

По принятым кодовым посылкам от кодера производится вычисление метрик ветвей

$$MB(00)^{(i)};\ MB(11)^{(i)};\ MB(10)^{(i)};\ MB(01)^{(i)}$$

и формирование четырех новых метрик состояний

$$MC(00)^{(i)};\ MC(10)^{(i)};\ MC(01)^{(i)};\ MC(11)^{(i)}$$

по следующему правилу. К каждому новому состоянию ведут два пути (в отличие от процесса развития решетчатой диаграммы). Декодер вычисляет метрики путей как суммы метрик предыдущих состояний и метрик входя-

щих путей. Метрика пути есть сумма метрик ветвей, образующих некоторый путь на решетчатой диаграмме. Метрика данного состояния равна метрике пути (МП), который заканчивается в данном состоянии.

Производя попарное сравнение метрик путей, входящих в каждое состояние, выбирают меньшую метрику и ее считают метрикой данного состояния для последующего шага декодирования. Путь, входящий в данное состояние с меньшей метрикой, считают выжившим (на диаграмме выжившие пути показаны жирными линиями). Тонкой линией показаны пути, которые отбрасываются при попарном сравнении. В результате сравнения выбирают меньшую метрику и ее считают метрикой данного состояния для последующего шага декодирования.

Таким образом, на каждом шаге декодирования в соответствии с алгоритмом Витерби в каждом из состояний решетчатой диаграммы производятся однотипные операции:

1) сложение метрик предыдущих состояний с метриками соответствующих ветвей;

2) сравнение метрик входящих путей;

3) выбор путей с наименьшими метриками, величины которых используют как метрики состояний на последующем шаге декодирования.

Если метрики сравниваемых путей одинаковы, то выбор одного из двух путей производят произвольным образом. На каждом шаге в результате сравнения половина возможных путей отбрасывается и в дальнейшем не используется. Другая половина образует продолжения путей для следующего шага декодирования. Из каждого состояния на следующем шаге вновь появляются два варианта продолжения путей. Это обеспечивает постоянство вычислений, производимых на очередном шаге. Декодер прослеживает по кодовой решетке путь, имеющий минимальное расстояние от пути, который генерирует кодер.

Предположим, что при передаче кодового сообщения по дискретному каналу возникла одиночная ошибка, вследствие чего вместо исходной последовательности на вход декодера поступает последовательность:

00 00 11 01 10 01 11 00 00 11 10 01 **01** 01 11 00 11 01 01 00.

Пусть в 13 паре возникла ошибка: вместо исходной пары 00 получено 01.

Рассмотрим процесс построения декодером решетчатой диаграммы. На этапе развития диаграммы (первые три шага) и построения ее далее не происходит отклонений от диаграммы безошибочного декодирования (рисунок 2.41). На 13-м шаге изменяется исходная последовательность. Вследствие одиночной ошибки имеем диаграмму, построенную декодером для последовательности с одиночной ошибкой (рисунок 2.42). Как видно, путь с

наименьшей метрикой не изменился, что говорит о том, что последовательность, подаваемая на кодер, будет восстановлена.

Методика построения диаграммы полностью соответствует алгоритму Витерби. Так же выбираются пути с наименьшей метрикой, т.е. декодер прослеживает по кодовой решетке путь, имеющий минимальное расстояние от пути, который генерирует кодер. Единственное отличие диаграммы декодирования кодовой последовательности с одиночной ошибкой то, что увеличивается число выживших путей и метрики состояний принимают другие значения.

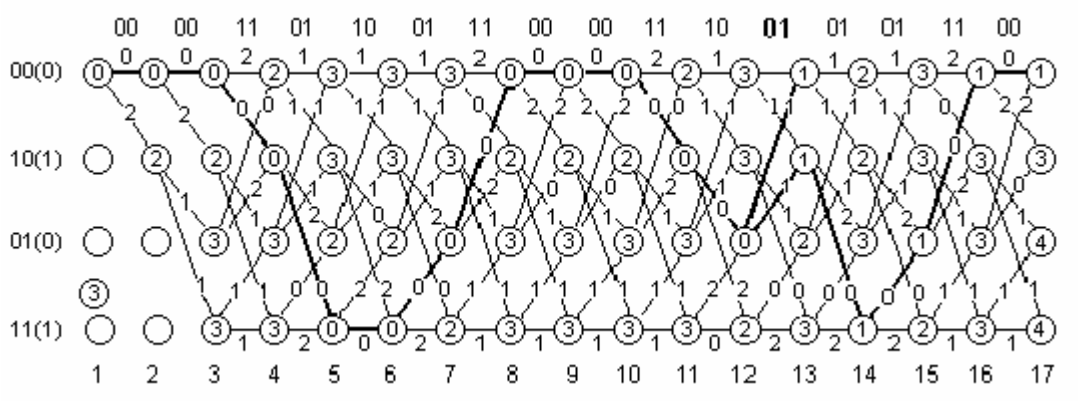


Рисунок 2.42 – Диаграмма исправления ошибки при декодировании сверточного кода

Этот способ позволяет устранять ошибки при их следовании через 4 бита, что связано с регистром сдвига кодера. Так как зависимость кодовой посылки от входной информации лежит в пределах 4-х битов, то увеличение количества следующих подряд ошибок приводит к неправильному декодированию. Сложность реализации алгоритма определяется в основном структурой решетчатой диаграммы. Поэтому его применяют для коротких сверточных кодов и скорость передачи обычно берут равной 1/2 или 2/3.

Различные варианты сверточного кодирования широко используются в телефонных и xDSL-модемах, гигабитовых локальных сетях Ethernet, в системах передачи беспроводных компьютерных сетей. В последнем случае сверточное кодирование носит название "**двоичное пакетное сверточное кодирование**" (*Packet Binary Convolutional Coding*, **PBCC**). В технологии PBCC используются сверточные кодеры на семь состояний (K = 7) со скоростью r=1/2. Метод пакетного сверточного кодирования опционально предусмотрен как альтернативный метод кодирования в протоколе 802.11b на скоростях передачи 5,5 и 11 Мбит/с.

2.7. Выводы по разделу

1. В компьютерных сетях для передачи данных используются физические линии (кабели), каналы связи и беспроводные линии. Проводные линии связи бывают симметричные и несимметричные. Для повышения степени симметричности параметров двухпроводной симметричной линии пары изолированных проводников скручиваются (свиваются) между собой. Имеется семь категорий кабеля на основе витых пар. К несимметричным линиям относится коаксиальный кабель. Оптические линии позволяют передавать данные в одном направлении по одному волокну.

2. Канал – это тракт передачи, использующий часть ресурсов линии связи (полоса частот или время использования). Каналы в свою очередь подразделяются на аналоговые и цифровые. Они образуются с помощью аппаратуры уплотнения (мультиплексирования), использующей частотное или временное разделение каналов. В широкополосных системах возможно кодовое разделение каналов.

3. Полоса пропускания физической линии начинается с нулевой частоты и зависит от длины и качества линии. Непрерывный канал связи представляет собой полосно-пропускающую систему, непрозрачную для сигналов данных. Цифровые каналы характеризуются пропускной способностью, величина которой стандартизирована и определяется типом канала.

4. Максимальная скорость передачи сигналов по линии или непрерывному каналу связи ограничивается полосой пропускания и уровнем шумов в канале (линии). Допустимый уровень сигнала в линии (канале) ограничивают из-за возникновения мешающих наводок в соседних парах кабеля, либо по причине возможной перегрузки трактов в каналообразующей аппаратуре. Максимальную пропускную способность имеют волоконно-оптические линии связи.

5. Компьютерные данные отображаются сигналами постоянного тока, в спектре которых содержится постоянная составляющая и низкочастотные компоненты. Каналы связи не могут непосредственно пропустить такие сигналы. Для согласования спектра сигналов данных с полосой пропускания каналов применяется модуляция несущего колебания.

6. Лучшим способом модуляции является фазовая, так как она обеспечивает максимальную помехозащищенность сигналов, а увеличение количества позиций ФМ сигналов не приводит к расширению их спектров.

7. В беспроводных компьютерных сетях для передачи данных с высокой скоростью и осуществления одновременной работы нескольких источников в одном частотном диапазоне, а также обеспечения скрытости передачи используются сигналы с растянутым спектром, обладающих свойст-

вами шумоподобных колебаний.

8. Расширение спектра сигналов данных производится либо за счет быстрого скачкообразного переключения по определенному алгоритму несущей частоты радиопередатчика (способ *FHSS)*, либо путем замены единичных элементов данных кодовыми последовательностями, состоящими из множества импульсов "чипов" (способ *DSSS)*. Чипы в свою очередь осуществляют фазовую модуляцию несущего колебания. В качестве кодовых последовательностей чаще всего применяются последовательности Баркера, или комплементарные последовательности типа ССК, обладающие хорошей автокорреляционной функцией.

9. Эффективным способом повышения скорости передачи по каналам с неравномерной частотной характеристикой, а также при наличии в канале межсимвольных искажений является многоканальная передача сигналов с частотным разделением каналов и с использованием ортогональных поднесущих (способ OFDM). Эффект повышения скорости достигается за счет одновременной передачи с относительно невысокой скоростью по множеству независимых каналов. Дополнительно повышение скорости происходит за счет использование в каждом из каналов многопозиционной фазовой модуляции.

10. Передача данных на канальном уровне может производиться асинхронным (старт-стопным) и синхронным способом. В первом случае осуществляется передача отдельными байтами, причем передача очередного символа может начаться в произвольный момент времени. В промежутках между передачей аппаратура находится "на стопе". Синхронная передача осуществляется непрерывно в виде отдельных блоков. В паузах между передачей полезной информации по каналу производится передача синхронизирующих блоков.

11. Для защиты информации от ошибок применяют коды с исправлением и коды с обнаружением ошибок. Во втором случае при обнаружении ошибок приемник по каналу обратной связи осуществляет повторный запрос искаженного блока. Существуют системы с ожиданием подтверждения и с непрерывной передачей группы блоков.

12. В компьютерных сетях наиболее широко применяются циклические коды с образующим полиномом не менее 16-й степени. Это позволяет обнаруживать как одиночные, так и пачки ошибок с длиной, равной степени образующего полинома. Для уменьшения потока одиночных ошибок и исключения дополнительных переспросов блоков, а также для снижения задержек при декодировании используются сверточные коды. При этом применяется эффективный алгоритм декодирования Витерби.

13. Более детально вопросы передачи данных рассмотрены в специальной литературе, в частности, параметры линий связи, их характеристики,

влияние на степень искажения сигналов – в [14,16,28,34], особенности передачи сигналов по физическим линиям и проводным каналам, спектры сигналов – в [14,16,28,30], способы передачи сигналов в беспроводных сетях, методы расширения спектра сигналов – в [13,28,31,35], способы помехоустойчивого кодирования, сжатия информации – в [8,15,28,30,33].

2.8. Контрольные вопросы

1. Каковы отличительные особенности кабеля и канала связи от линии связи?
2. Назовите первичные параметры кабельных линий. Какова связь их с вторичными параметрами?
3. Рассчитайте затухание линии связи, если амплитуды сигналов на входе и выходе равны 1В и 10мВ соответственно.
4. Определите защищенность линии от помех, если переходное затухание составляет 70 дБ, а собственное затухание 30 дБ.
5. Почему пропускная способность многомодового оптического кабеля ниже по сравнению с одномодовым?
6. В чем состоит отличие дуплексного четырехпроводного канала от дуплексного двухпроводного?
7. Поясните принцип разделения направлений передачи дифференциальной системой трансформаторного типа.
8. Почему при передаче данных по кабельным линиям в сигналах должна отсутствовать постоянная составляющая?
9. Зачем применяют манчестерское кодирование сигналов, если это приводит к нежелательному расширению их спектра?
10. Как при передаче данных по оптическим линиям можно контролировать целостность среды?
11. Почему нельзя для передачи данных непосредственно соединять последовательный порт компьютера с телефонным каналом, а при использовании в качестве среды передачи физической линии – можно?
12. В чем отличие многопозиционной ФМ-16 от квадратурной АФМ-16 модуляции при одинаковом количестве значащих позиций сигнала?
13. Почему для повышения скорости передачи данных применяют многопозиционную амплитудно-фазовую модуляцию, а не частото-фазовую или амплитудно-частотную модуляцию?
14. Почему на практике применяется преимущественно дифференциальная фазовая модуляция, а не абсолютная?
15. С какой целью в беспроводных компьютерных сетях используют сигна-

лы с расширением спектра?

16. Каким образом могут быть выделены шумоподобные сигналы на фоне шума аналогичной мощности?
17. Почему в системы с расширением спектра ввели ССК-последовательности, если последовательность Баркера обладает лучшей корреляционной функцией?
18. С какой целью в асинхронных системах передачи в качестве стопового сигнала применяют сигнал единичного уровня, хотя использование нулевого уровня для стопового сигнала было бы более экономно?
19. Каким образом избегают ложного выделения синхронизирующей комбинации из информационной последовательности, если в передаваемом сообщении может находиться аналогичная комбинация?
20. Как на практике определить, с какой максимальной скоростью можно передавать сигналы по данной линии (каналу) связи?
21. Почему для защиты информации от ошибок в компьютерных сетях применяются преимущественно коды не исправляющие, а только обнаруживающие ошибки?
22. Зачем в компьютерных системах связи применяют сверточное кодирование, если его избыточность выше, чем при блочном кодировании?
23. Как передающая станция в случае переспроса на повторную передачу блока определяет, что блок принят с ошибкой из-за помех в канале, или по причине нарушения цикловой синхронизации? В последнем случае ей следовало бы переключиться из режима передачи данных в режим синхронизации по циклам.

Раздел 3
ЛОКАЛЬНЫЕ КОМПЬЮТЕРНЫЕ СЕТИ

Каким образом, имея только один отрезок двухпроводного кабеля, обеспечить связь между всеми компьютерами локальной сети? В каких случаях локальную сеть с явной шинной топологией называют логическим кольцом? По какой причине пауза после коллизии имеет случайную длительность? Как сохраняется целостность "кольца" при обесточенном приемопередатчике одного из компьютеров сети Token Ring? Каким образом на витой паре кабеля третьей категории удается обеспечить передачу данных со скоростью 100 Мбит/с? Как функционирует беспроводная компьютерная сеть? На эти и другие вопросы Вы найдете ответ, изучив данный раздел.

3.1. Топология сетей и методы доступа к среде

3.1.1. Топология локальных компьютерных сетей

Локальные компьютерные сети (ЛКС) представляет собой такую разновидность сетей, в которой все ее компоненты, включая ЭВМ различных классов, расположены на ограниченной территории одного предприятия или учреждения и соединены через единую физическую среду. Расстояния между компьютерами локальной сети составляют от сотен метров до десятков (10…20) км. В локальных сетях сетевые компьютеры называют **рабочими станциями**. Ограниченность территории создает предпосылки для использования специфических способов передачи данных, отличных от традиционных, применяемых в глобальных сетях. Благодаря этому в ЛКС удается реализовать значительно более высокую скорость передачи (до тысяч Мбит/с) и на несколько порядков более низкую вероятность ошибок при существенно меньших затратах. Расположение локальной сети на ограниченной территории влияет также на способы административного сетевого управления, а технические характеристики ЛКС приводят к необходимости введения новых протоколов.

В качестве физической среды ЛКС наибольшее распространение получили электрические кабели типа “витая пара”, коаксиальные и волоконно-оптические кабели. В последнее время все большую популярность получают беспроводные линии связи. В дальнейшем, при описании ЛКС понятия "среда", "линия" и "канал" используются как синонимы.

Основные отличия архитектуры ЛКС от архитектуры глобальных сетей связаны с нижними тремя уровнями. Использование единой физической среды позволяет существенно упростить функции уровня маршрутизации. Нижние два уровня ЛКС имеют свою специфику, связанную с топологией сети и методами доступа к физической среде.

Различают линейную (а), звездообразную (б), кольцевую (в), шинную (г) и древовидную (д) топологию ЛКС (рисунок 3.1). Все структуры сети, кроме шинной, представляют собой двухточечные звенья. В линейной структуре сети сообщения должны пройти через несколько узлов, прежде чем они достигнут цели. Поэтому в случае повреждения одного из звеньев сообщение не может быть доставлено адресату, что является существенным недостатком такой сети.

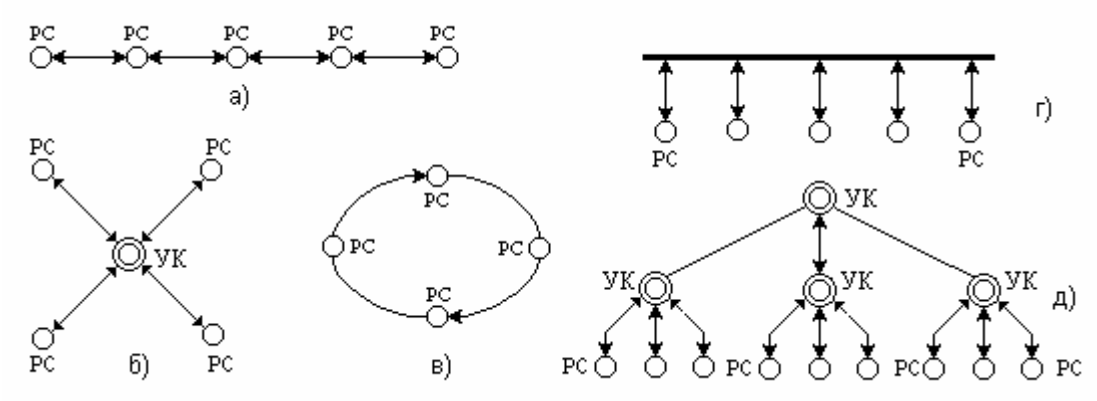


Рисунок 3.1 – Топология локальных компьютерных сетей

Топология "звезда" характеризуется наличием центрального узла коммутации (УК), к которому подключаются все остальные рабочие станции (РС). Через этот узел циркулирует весь сетевой трафик, поэтому нагрузка на узел очень высокая. Сетевое оборудование центрального узла оказывается намного сложнее, чем оборудование абонентов сети. К достоинствам "звезды" относится достаточно высокая надежность сети в целом. Так, обрыв одного сетевого кабеля или короткое замыкание в нем нарушает работу только одного компьютера, а все остальные могут продолжать работу. Положительным свойством является также наличие на каждой линии связи только одного передатчика и приемника, что заметно упрощает сетевое оборудование по сравнению с "шиной". Недостаток звездообразной структуры состоит в низкой скорости обработки информации и большой суммарной протяженности линий связи. При этом при выходе центрального узла из строя отказывает вся сеть.

Преимуществом “кольца” является возможность использования однонаправленной линии связи. На каждом участке к линии связи подключены

только один передатчик и один приемник. По этой причине нет необходимости использовать согласующие сопротивления (терминаторы). Недостатком кольцевой топологии является загрузка узлов всей той информацией, которая передается по сети.

Шинная топология является одной из простейших по способу подключения рабочих станций. В такой структуре отсутствует центральный узел, через который передается вся информация. Это увеличивает надежность сети. Однако она предполагает идентичность сетевого оборудования компьютеров, а также равноправие всех абонентов. При таком соединении компьютеры могут передавать данные только по очереди, так как линия связи одна на всех (моноканал). На концах линии связи должны устанавливаться согласующие сопротивления (терминаторы) для исключения появления отраженных волн, вызывающих искажение сигналов. Обрыв или замыкание в линии выводит из строя всю сеть. К существенным недостаткам шинной структуры также относится возможность возникновения ***сетевых коллизий***. Коллизия возникает всякий раз, когда одновременно ведут передачу две или несколько рабочих станций сети, что приводит к разрушению информации. Разработаны специальные протоколы связи, позволяющие исключить потери информации при возникновении коллизий, либо исключающие их возникновение.

Древовидная топология представляет собой иерархическую звезду. Она имеет достоинства, присущие звездной топологии. Используется для увеличения количества рабочих станций сети при ограниченном числе портов узловой станции.

3.1.2. Шина со случайным доступом

Характерной чертой многих локальных сетей является коллективное использование ресурсов среды передачи данных – линии связи, которая является ***моноканалом***. Через такую среду в заданный промежуток времени может передавать информацию только один сетевой компьютер. Поэтому возникает проблема разделения ресурсов среды передачи данных, которая решается различными методами доступа к среде. Под *доступом к среде* понимают взаимодействие рабочей станции (узла сети) со средой передачи данных для обмена информацией с другими станциями. Различают *случайные* и *детерминированные* методы доступа. Среди случайных наиболее широко используется метод множественного доступа с распознаванием несущей и обнаружением конфликтов **CSMA/CD** (*Carrier Sence Multi Access/Collision Detecting*). К детерминированным относятся методы доступа с передачей маркеров.

Метод доступа CSMA/CD. При использовании этого метода станции могут передавать сообщения, только если канал связи свободен. В случае одновременной передачи информации несколькими станциями возникает конфликтная ситуация (*коллизия*), в результате чего происходит разрушение передаваемых данных. Поэтому станции должны прекратить передачу, выждать некоторое время и продолжить ее по одной только при наличии свободного канала. Для определения занятости канала используется контроль уровня несущей в среде. Чтобы избежать повторения коллизий, время ожидания включения станций выбирается различным. Если одна из станций начала передачу, то канал оказывается занятым и все другие станции должны ждать его освобождения. На рисунке 3.2 показана диаграмма состояний, иллюстрирующая схему управления CSMA/CD.

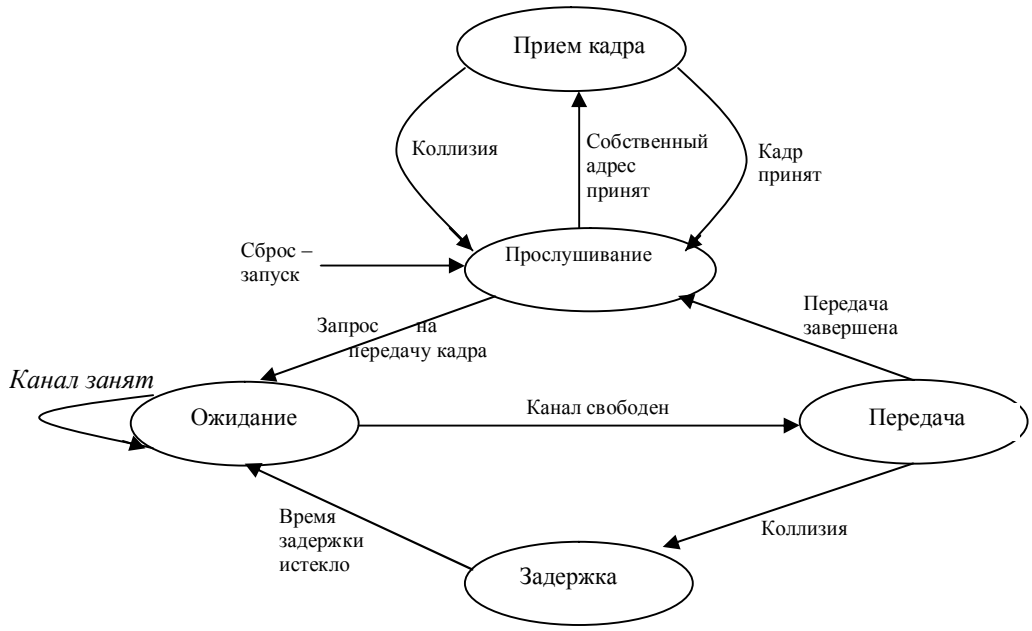


Рисунок 3.2 – Диаграмма состояний метода доступа CSMA/CD

Большую часть времени схема канального уровня находится в режиме прослушивания канала связи. В этом состоянии анализируются все кадры, передаваемые в канале. Если заголовок кадра содержит адрес назначения, совпадающий с адресом узла, то схема канального уровня переходит в состояние приема, во время которого осуществляется прием кадра.

После завершения приема кадра выдается сообщение на сетевой уровень сети, а приемник переключается в режим прослушивания. Возможно, что коллизия произойдет во время приема кадра. В этом случае прием кадра прерывается и приемник канального уровня переключается в состояние прослушивания.

Передача кадра в среду может быть произведена только по запросу сетевого уровня. Если станция во время этого запроса не находится в состоянии приема, то схема канального уровня переходит в состояние ожидания. В этом состоянии узел ждет освобождения канала и начинает передачу пакета. В случае, если передача завершается успешно (без коллизий), состояние аппаратуры канального уровня вновь изменяется на состояние прослушивания. Если же во время передачи кадра появляется конфликтная ситуация, то передача прерывается и затем, после прослушивания, возобновляется снова. Время задержки включения станции после коллизии вычисляется различными способами. Наиболее часто это время выбирается случайно.

Одним из важнейших в сетях с алгоритмом CSMA/CD является вопрос: какое максимальное время $t_{ок}$ требуется для обнаружения коллизий передающей станцией в наиболее неблагоприятном случае и какой должна быть длина пакета $N_п$, если длина сегмента кабеля равна d?

Коллизия происходит в том случае, если одна из станций А начала передачу, но сигнал, распространяющийся по линии связи с конечной скоростью, еще не успел дойти до другой станции Б, которая, обнаружив, что линия свободна, также начинает передачу (рисунок 3.3).

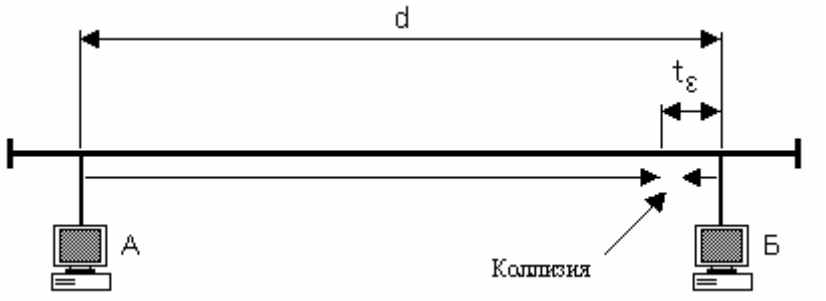


Рисунок 3.3 – Схема возникновения коллизии в самом неблагоприятном случае

Наиболее неблагоприятная ситуация имеет место в случае, когда интервал времени t_ε между началом момента передачи станцией Б и временем достижения сигналом станции А точки подключения станции Б к линии становится пренебрежимо мал, т.е. $t_\varepsilon \approx 0$.

В момент появления коллизии происходит *изменение состояния линии связи* в точке коллизии (при манчестерском кодировании изменяется постоянная составляющая напряжения в линии). Это изменение доходит до станции А через интервал времени, равный времени распространения сигнала t_p по кабелю длиной d. Следовательно, время обнаружения сигнала коллизии в сети CSMA/CD в наиболее неблагоприятном случае равно удвоенному времени распространения сигнала между двумя станциями, наиболее удален-

ными друг от друга, т.е.

$$t_{ок} = 2t_{р} = 2d/v. \quad (3.1)$$

При расчетах принимается, что скорость распространения сигнала в кабеле связи v в среднем равна 200 000 км/с. Интервал $t_{ок}$ называется **временем двойного оборота PDV** (*Path Delay Value*).

При передаче слишком короткого кадра станция может завершить его передачу до обнаружения коллизии. Отправитель посчитает, что передача прошла без коллизии и не станет пытаться повторить кадр. Для исключения такой ситуации станции, участвующие в передаче, вынуждены были бы ожидать по окончании передачи пакета до тех пор, пока не истечет время, требуемое для обнаружения коллизии в наиболее неблагоприятном случае. Очевидно, что это привело бы к простаиванию линии и, следовательно, неэффективному использованию канала. Поэтому минимальная длина кадра выбирается таким образом, чтобы при возникновении коллизии на максимальном удалении от передатчика, т.е. в месте подключения самой дальней станции, и достижении сигнала коллизии передающей станции, передача кадра не была бы завершена. Следовательно, минимальное количество битов в кадре можно определить из соотношения

$$N_{П\ мин} = B \cdot t_{ок} = 2Bd / v. \quad (3.2)$$

Максимальная длительность кадра также ограничивается. Это вызвано тем, что если станции используют очень длинные кадры, то при захвате канала одной из них другие пользователи сети вынуждены будут долго ожидать освобождения линии, что существенно увеличивает общую задержку передачи сообщений в сети и тем самым снижает качество обслуживания. Кроме этого, увеличение длины кадра приводит к повышению вероятности поражения его помехой и необходимости повторной передачи. Ограничение длины кадра вызвано также стремлением уменьшить размер буфера для промежуточного хранения информации. На практике максимальная длина кадра содержит около 1500 байт.

Из рассмотренного способа доступа к среде видно, что он имеет случайный характер. Вероятность получения станцией в монопольное пользование линии связи зависит от загруженности сети, то есть от интенсивности трафика каждого из компьютеров, подключенных к общей шине.

3.1.3. ЛКС с шиной и маркерным доступом

Данный метод доступа относится к детерминированным и характеризуется тем, что в нем право использования среды с топологией шины передается от станции (узла) к станции *организационным* способом, а не *состязательным.* Право работы с каналом реализуется посредством посылки специального кадра разрешения – **маркера**. Станция, получившая маркер (*Token*) , может начинать передачу данных, и после ее завершения пересылает маркер следующей, в порядке увеличения адресов, станции. Маркер передается по логическому кольцу и, достигнув узла с максимальным адресом, вновь поступает на станцию с наименьшим адресом. Такая процедура управления носит название **передача по логическому кольцу.**

Помимо передачи маркера схема с шиной должна решать *проблему потери маркера и реконфигурации* кольца. Потеря маркера может произойти из-за повреждения одной из станций логического кольца. В некоторый момент времени маркер приходит в поврежденный узел, но узел не пропускает его дальше, и другие станции по этой причине не получают маркер. *Реконфигурация* кольца выполняется, когда в логическое кольцо добавляется или из него удаляется один из узлов.

Большую часть времени аппаратура канального уровня находится в состоянии прослушивания. Если заголовок приходящего кадра в адресной части содержит адрес узла, то канальный уровень переходит в состояние приема кадра. При условии, что принятый кадр является кадром пакета данных, сетевой уровень информируется о приеме, а канальный уровень возвращается в состояние прослушивания.

Однако если принятый кадр оказался маркером, то это означает, что узел получает право передачи в среду. В случае наличия на узле информации, подлежащей передаче, состояние станции переходит в активный режим, при котором производится передача пакета. По окончании передачи в канал выдается маркер. Передача маркера происходит также в случае отсутствия на станции пакета данных, подлежащих отправке получателю. После передачи маркера узел снова переключается в режим прослушивания.

При потере маркера или сбое в сети все узлы переходят в состояние ожидания (бездействия). Время ожидания каждой из рабочих станций различное и выбирается пропорционально ее номеру, т.е. после отключения компьютеров первой возбудится станция с наименьшим адресом. Она формирует маркер и посылает его следующему компьютеру в сети, начиная с узла, адрес которого на 1 больше его собственного. Приход маркера активизирует вызываемый узел и он сам начинает опрос сети, что является признаком восстановления сети.

3.1.4. ЛКС с кольцевой структурой и маркерным доступом

Основное различие между данной схемой и двумя предыдущими заключается в физической топологии среды. В кольцевой среде сигналы, переданные одним из компьютеров сети, распространяются через однонаправленные двухточечные линии между станциями, которые соединяются последовательно, образуя физическое кольцо (рисунок 3.4). Во время передачи по кольцевой среде сигналы проходят через станции от приемного к передающему порту. При этом станции могут анализировать и модифицировать приходящие сигналы. Преимуществом такого решения является возможность увеличения длины соединительных линий за счет усиления и ретрансляции сигналов на узлах. Однако повреждение одной из станций или кабельного сегмента физического кольца приводит к выводу из строя всей сети. При ретрансляции сигналов узел вносит задержку, которая равна длительности единичного элемента сигнала.

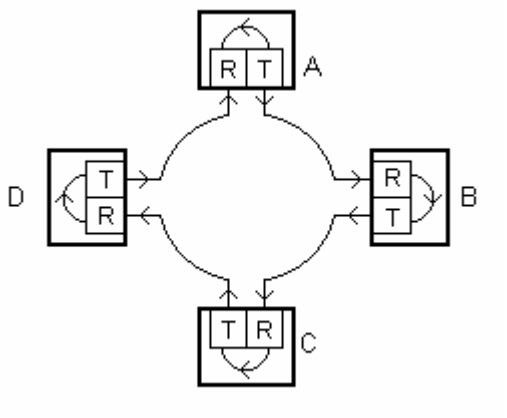


Рисунок 3.4 – Топология физического кольца

Как и в случае шинной структуры с передачей маркера, в схеме доступа к кольцевой среде в качестве маркера используется специальный укороченный кадр, у которого имеется **бит-индикатор Т** (*Token*) признака маркера. Первые два байта маркерного и информационных кадров полностью совпадают по формату. Если бит Т установлен в единицу, то кадр является маркером, в противном случае дальнейшая последовательность воспринимается как информационный кадр. Если ни у одного из узлов сети нет пакета данных для передачи, маркер непрерывно циркулирует по кольцу. Такой кадр носит название ***свободного маркера***.

Узел, в котором имеется пакет данных для передачи, должен ждать, пока он не получит свободный маркер. В момент прихода свободного маркера станция переходит в режим передачи, изменяет состояние маркера на за-

нятое (T=0) и передает маркер дальше по кольцу, добавляя к нему информационную и служебную часть кадра.

Кадр данных вместе с занятым маркером перемещается по всему кольцу. Модифицировать значение маркера снова на свободное может только тот узел, который изменил его на занятое. В каждом кадре данных содержится адрес узла назначения. Все узлы кольца, за исключением узла источника, обнаружив занятый маркер (T=0), ретранслируют кадр, а принимает его только узел назначения. Таким образом, на узле назначения принимаемый кадр фиксируется (копируется) и вместе с маркером передается далее по кольцу.

Когда занятый маркер вместе с остальной частью кадра возвращается в узел источника, состояние маркера меняется на свободное, а пакет удаляется из кольца (не передается дальше). Как только маркер становится свободным, любой узел может изменить его на занятый и начать передачу данных.

3.1.5. Общая характеристика сетей Ethernet и Token Ring

Из нескольких десятков типов систем проводных соединений в локальных компьютерных сетях лидируют два стандарта: **802.3** (*Ethernet*) и **802.5** (*Token Ring*). Спецификации 802.3 и 802.5 – стандарты, разработанные организацией IEEE (*Institute of Electrical and Electronic Engineers*). Важной чертой стандартов является их открытость, т.е. они не контролируются каким-либо разработчиком аппаратуры.

Стандарт 802.3 (*Ethernet*) был разработан в 1975 г. на основе сетевой системы, созданной фирмами Xerox и DEC. Исходная система называлась *Ethernet*. Основные причины популярности *Ethernet* заключаются в следующем:

- стандарт *Ethernet* утвержден значительно раньше *Token Ring*;
- сеть *Ethernet* обеспечивает высокую производительность при приемлемой стоимости;
- *Ethernet* является неотъемлемым компонентом локальных компьютерных сетей, поставляемых многими производителями.

Технология *Token Ring*, разработанная фирмами IBM и *Texas Instruments*, утверждена в качестве стандарта 802.5 IEEE в 1985г. (Слово *Token* означает – маркер). Стандарты семейства IEEE 802.x охватывают только два нижних уровня семиуровневой модели OSI: *физический и канальный*. Старшие уровни, начиная с сетевого, имеют общие признаки как для локальных, так и для глобальных сетей.

Существующие локальные компьютерные сети могут функционировать как в *однополосном* режиме (передача сигналов только одного источни-

ка) и в широкополосном (передача сигналов нескольких источников одновременно). Сети *Ethernet* и *Token Ring* являются однополосными. Важнейшие отличительные особенности этих сетей – методы передачи сообщений и способы обеспечения их целостности, а также способы организации кабельных соединений.

Сеть *Ethernet* относится к сетям множественного доступа с прослушиванием линии и обнаружением коллизий CSMA/CD. Стандартная скорость передачи данных по кабелям связи на настоящее время установлена 10, 100, 1000 и 10000 Мбит/с. Основной недостаток технологии *Ethernet* – наличие коллизий; и другой – повреждения кабеля приводит к выходу из строя всей сети.

Технология *Token Ring* была разработана с целью преодоления указанных недостатков. Она обеспечивает регулярность передачи сообщения для каждой станции в сети. К тому же схема кабельных соединений для *Token Ring* облегчает локализацию большой части неполадок. Стандартная скорость передачи 4 и 16 Мбит/с. Структура стандартов IEEE 802.x изображена на рисунке 3.5.

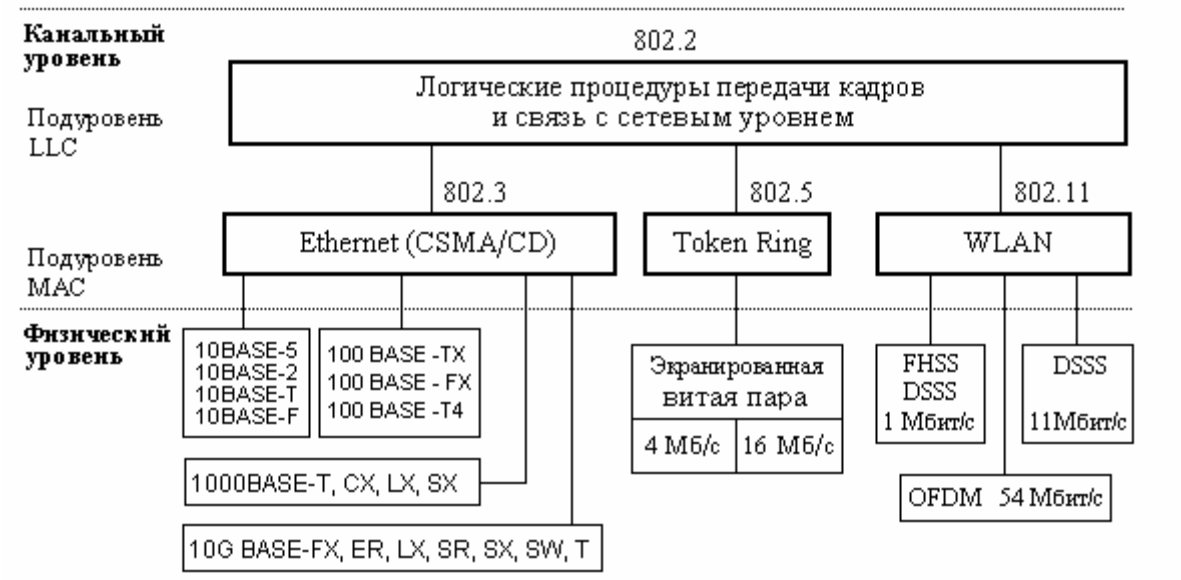


Рисунок 3.5 – Структура стандартов IEEE 802.x

Канальный уровень дополнительно подразделяется на подуровни ***управления логической передачей данных*** **LLC** (*Logical Link Control*) и ***управления доступом к физической среде*** **MAC** (*Media Access Control*). Подуровень LLC обеспечивает интерфейс протокола *Ethernet* с протоколами вышележащих уровней, например, с IP или IPX. Он регламентирует передачу кадров данных между узлами с различной степенью надежности. Через

этот подуровень сетевой протокол запрашивает у канального уровня нужную ему транспортную операцию с требуемым качеством. Введение уровня MAC вызвано наличием в локальных сетях разделяемой среды передачи сигналов, которая *попеременно* используется парой компьютеров – источником и получателем. Формат кадра LLC, изображен на рисунке 3.6. Поля кадра LLC имеют следующие назначения: **DSAP** (*Destination Service Access Point*) – адрес точки входа службы назначения; **SSAP** (*Source Service Access Point*) – адрес точки входа службы источника; **Control** – управляющее поле; **OUI** (*Organizationally Unique Identifier*) – идентификатор организации, которая контролирует коды протоколов в поле Type; **Type** – состоит из двух байт и повторяет по назначению аналогичное поле кадра *Ethernet.*

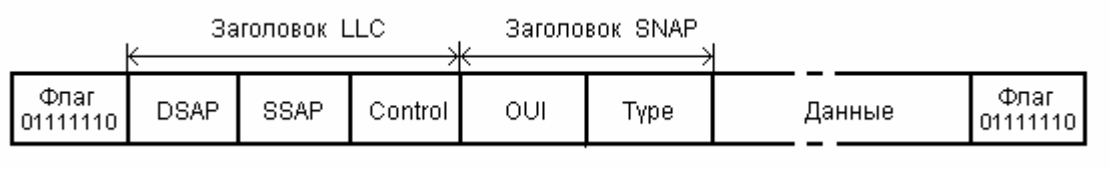


Рисунок 3.6 – Формат кадра управления логическим каналом LLC

Кадр LLC вкладывается в кадр MAC, и предоставляет возможность за счет полей DSAP и SSAP идентифицировать адрес сервисов назначения и источника соответственно. Например, при вложении в кадр LLC пакета IPX, значения как DSAP, так и SSAP должны быть равны E0h. Поле управления кадра LLC позволяет реализовать процедуры обмена данными трех типов.

1. ***Процедура типа 1*** определяет обмен данными без предварительного установления соединения и без повторной передачи кадров в случае обнаружения ошибочной ситуации, то есть является процедурой **дейтаграммного типа**. Именно этот тип процедуры и используется во всех практических реализациях *Ethernet*. Поле управления для этого типа процедур имеет значение 03, что определяет все кадры как ненумерованные.

2. ***Процедура типа 2*** определяет режим обмена **с установлением соединений**, нумерацией кадров, управлением потоком кадров и повторной передачей ошибочных кадров. В данном режиме протокол LLC аналогичен протоколу HDLC. В локальных сетях *Ethernet* этот режим используется относительно редко.

3. ***Процедура типа 3*** задает режим передачи данных без установления соединения, но **с получением подтверждения** о доставке информационного кадра адресату. Только после такого подтверждения может быть отправлен следующий информационный кадр.

С целью устранения различий в кодировках типов протоколов, сообщения которых вложены в поле данных кадров *Ethernet*, был разработан

протокол доступа к подсетям SNAP (*Subnetwork Access Protocol*), расширяющий формат кадра LLC. В случае использования расширения SNAP в поля DSAP и SSAP записывается значение AAh. Тип кадра по-прежнему равен 03. Для обозначения типа протокола, вложенного в поле данных, используются следующие 4 байта, причем байты идентификатора организации (OUI) всегда равны 00 (за исключением протокола *AppleTalk*). Последний байт (TYPE) содержит идентификатор типа протокола (например, 0800 для IP). С помощью заголовка SNAP достигается совместимость с кодами протоколов различных типов кадров *Ethernet*. Заголовок SNAP является дополнением к заголовку LLC, поэтому он допустим не только в кадрах *Ethernet*, но и в кадрах протоколов других технологий стандарта 802.

3.2. Классическая локальная сеть Ethernet

3.2.1. Принципы построения, общая характеристика

В основе наиболее часто применяемых в настоящее время локальных сетей лежит сеть Ethernet, архитектура которой разработана компанией DEC с участием *Xerox* и *Intel Corporation*. Построение и функционирование сети регламентируется стандартом IEEE 802.3. Классической версией Ethernet является стандартная сеть на скорость 10 Мбит/с, работающая в однополосном режиме. В ее состав входят разновидности сетей, обозначаемые 10BASE-5, 10BASE-2 и 10BASE-T. Здесь цифра 10 означает скорость передачи в Мбит/с, "BASE" – что сигналы передаются в основной полосе (без переноса спектра). Последние символы отображают либо вид линии (T,F), либо округленную максимальную длину кабельного сегмента (в сотнях метров). Максимальная длина сегмента ограничивается допустимым затуханием, при котором еще возможен устойчивый и достоверный прием сигналов.

Самой распространенной версией Ethernet является сеть **10BASE-T**, передача данных в которой осуществляется со скоростью 10 Мбит/с по неэкранированной витой паре. Обычно топология такой сети имеет вид звезды, лучи которой расходятся из центра кабельных соединений.

Вторым вариантом является **10BASE-2**. Применяется коаксиальный кабель для создания топологии типа "общая шина". Максимальная длина сегмента приблизительно 185 м. Поскольку используется тонкий коаксиальный кабель (d≈ 6 мм), то ее часто называют **ThinNet** (*тонкая сеть*).

10BASE-5, называется стандартной Ethernet. Использует шинный вариант сети. Применяется более толстый (d≈12 мм), коаксиальный кабель. Сеть часто называют **ThickNet** (*толстая сеть*). Толстый коаксиальный кабель допускает длину сегмента до 500 м и он лучше защищен от помех.

Имеются варианты сетей с использованием волоконно–оптических кабелей **10BASE-F**. Длина линии связи в таких сетях достигает 2 км.

Во всех сетях Ethernet применяется метод коллективного (поочередного) доступа к среде с опознаванием несущей и обнаружением коллизий CSMA/CD, характерный для топологии общей шины. Данные, подлежащие передаче по сети, помещаются в кадр, в котором указывается адрес станции назначения и отправителя. Прежде чем начать передачу станция прослушивает линию на предмет наличия в ней гармоники сигнала данных (т.н. "несущей"). В случае отсутствия несущей станция начинает передавать кадр в линию (рисунок 3.7). На рисунке показано, что кадр 1 начала передавать станция 1.

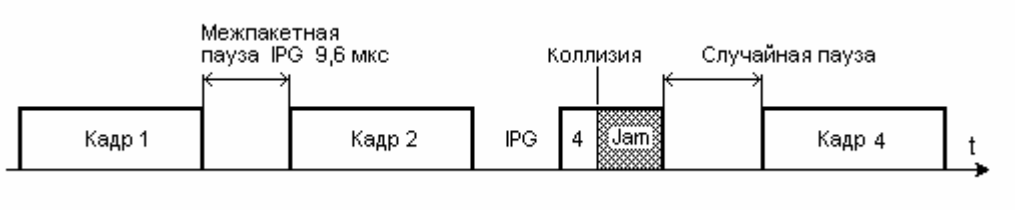


Рисунок 3.7 –Временная диаграмма передачи кадров в сети Ethernet

Передаваемый кадр поступает на приемники всех станций, подключенных к кабельному сегменту. Компьютер, сетевой адрес которого совпадает с адресом получателя, указанного в пакете, копирует его в свой буфер. По окончании передачи кадра все узлы должны выдержать технологическую **межпакетную паузу** IPG (*Inter Packet Gap*), продолжительность которой установлена 9,6 мкс. Эта пауза предназначена для завершения переходных процессов в приемопередатчиках сетевых адаптеров, а также для исключения захвата сегмента одной и той же станцией. Даже если в компьютере имеется несколько кадров для передачи и данный узел является единственным передающим, то после пересылки каждого пакета он должен сделать паузу длиной не менее чем интервал IPG.

По завершении межпакетной паузы линию занимает станция, в которой имеются данные для передачи. В случае попытки нескольких станций одновременно начать передачу возникает коллизия. Передающая станция, первая обнаружившая коллизию (на рисунке 3.7 станция 4), прекращает передачу и посылает в линию специальную 32-х битовую *последовательность индикации коллизии* **Jam** (от англ. – *затор*), которая способствует повышению надежности распознавания коллизии всеми станциями сети. Станции, обнаружившие коллизию, переключаются в состояние ожидания. Продолжительность интервала ожидания для каждого из компьютеров устанавливается случайной, по истечении которого любая из станций может попытаться захватить среду и передать кадр. Длительность случайной паузы $T_{сп}$ определяется в соответствии с выражением

$$T_{сп} = L \times 512\tau_0, \qquad (3.3)$$

где τ_0 – длительность единичного (*битового*) интервала.

Для сети Ethernet со скоростью передачи 10 Мбит/с τ_0 = 0,1 мкс, а для скорости 100 Мбит/с τ_0=0,01 мкс. L представляет собой целое число, выбранное с равной вероятностью из интервала [0, $2^{N_{пп}}$], где $N_{пп}$ – номер повторной попытки передачи данного кадра, изменяющийся с каждой попыткой от 1 до 10. После десятой попытки интервал, из которого выбирается пауза, не увеличивается. Следовательно, случайная пауза может принимать значение от 0 до 52,4 мс. Если 16 последовательных попыток передачи кадра каждый раз приводят к коллизии, то передатчик должен прекратить попытки и отбросить этот кадр.

Для предотвращения снижения пропускной способности сети Ethernet при появлении временных сбоев в функционировании сетевых интерфейсных карт применяется процедура *Jabber Control*. Она предназначена для исключения возможности возникновения ситуации, при которой одна рабочая станция или сегмент монополизирует процесс информационного обмена во всей сети. Согласно процедуре *Jabber Control* каждой станции разрешено работать со средой ограниченное время. По истечении установленного допустимого интервала активности на аппаратном уровне осуществляется прерывание процесса передачи данных и рабочая станция или сегмент сети переводятся в пассивное состояние. Возобновление процесса передачи данных такой станцией или сегментом сети невозможно до истечения установленного интервала задержки. Величины допустимого интервала активности установлены для рабочей станции от 20 до 150 мс, а для повторителя (3…7,5) мс. Время задержки повторной передачи находится в пределах (500…2500) мс для рабочей станции и (9,6…11,6) мкс для повторителя.

3.2.2. Типы кадров сети Ethernet

Кадр Ethernet обеспечивает перемещение данных по сети. В сети Ethernet используется несколько типов кадров:

- стандартный Ethernet II (разработчик DIX – фирмы *Digital-Intel-Xerox*);
- RAW 802.3 Ethernet (разработчик фирма *Novell*);
- IEEE 802/2 SNAP Ethernet (*SubNetWorkAccessProtocol*); его задача – устранить различия в кодировках типов протоколов.

Все типы кадров (рисунок 3.8) содержат преамбулу, кодовую комбинацию **СтРБ** "Стартовый разделитель блока" (*Start of Frame Delimiter*), адреса получателя и отправителя, поле типа и длины кадра, информационное поле и контрольную последовательность кадра (**КПК**).

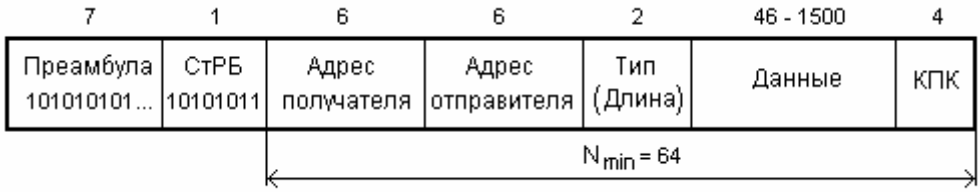


Рисунок 3.8 – Формат кадра сети Ethernet

Преамбула используется для целей синхронизации приемника по тактам и состоит из 7 байт вида 1010101010.

Байт ***СтРБ*** представлен комбинацией 10101011. Появление его означает начало кадра. Он служит для маркерной синхронизации по циклам.

Адрес получателя имеет длину 6 байт. Первый бит старшего байта адреса является признаком индивидуального (0) или группового (1) адреса. ***Адрес отправителя*** также состоит из 6 байтов, причем старший бит адреса имеет всегда нулевое значение.

Поле "***Длина/Тип***" – состоит из двух байтов и содержит сведения о длине поля данных. Для некоторых типов кадров (Ethernet II) оно определяет тип используемого протокола верхнего уровня. Если в поле записан код менее 1500, то это поле характеризует длину кадра. В противном случае – это код протокола, пакет которого инкапсулирован в кадр Ethernet.

Поле "***Данные***" должно содержать от 46 до 1500 байтов данных. Если данных менее 46 байтов, то поле дополняется символами заполнения.

Поле "***КПК***"– контрольная последовательность кадра, образуется в результате циклического кодирования и содержит инверсию остатка от деления на образующий полином 32-й степени вида:

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

Таким образом, минимальная длина кадра без преамбулы содержит **64** байта (512 бит). Эта величина определяет максимально допустимую двойную задержку распространения сигнала по сети в 512 битовых интервала. Стандартом предполагается, что преамбула может уменьшаться при прохождении кадра через различные сетевые устройства, поэтому она в расчет не принимается. Максимальная длина кадра равна **1518** байтам. Этот параметр учитывается при расчете объема буферной памяти сетевого оборудования. В процессе передачи кадров непрерывно контролируется их размер. Если кадр оказывается короче 64 или длиннее 1518 байтов, то он отбрасывается. Передача единичных элементов кадра осуществляется старшими битами вперед сигналами с манчестерским линейным кодированием.

В настоящее время все сетевые адаптеры, их драйверы, мосты, коммутаторы и маршрутизаторы могут работать со всеми используемыми на практике форматами кадров, а распознавание их типов происходит автоматически.

3.2.3. Стандартная Ethernet 10BASE-5

Ethernet, построенная по шинной топологии на основе коаксиального кабеля диаметром наружной изоляции около 12 мм (0,5 дюйма), называется "**толстая сеть**" и обозначается 10BASE-5. Исторически она появилась первой и получила название "стандартная Ethernet". Длина сегмента в сети не более 500 м. К нему могут быть подключено до 100 станций. Несколько сегментов сети могут быть соединены в единую сеть с помощью повторителей. Повторители осуществляют регенерацию сигналов без изменения их логической структуры. Стандартом установлено так называемое **правило 5-4-3**, в соответствии с которым разрешается использование в сети не более 5 сегментов, соединенных посредством 4-х повторителей. Компьютеры можно подключать только к трем сегментам. Остальные служат для увеличения **диаметра сети** – максимального расстояния между станциями данной сети. На концах сегментов включается терминаторы сопротивлением 50 Ом.

Доступ к среде осуществляется с помощью трансивера с отводом типа "*вампир"*, который размещается на кабеле. В таком ответвителе с помощью винтового механизма подвижный игольчатый контакт "вгрызается" в кабель и посредством иглы обеспечивается соединение с центральной жилой кабеля. Трансиверы в этом случае только внешние. Трансивер соединяется с сетевым адаптером интерфейсным кабелем AUI (*Attachment Unit Interface*), состоящим из 4-х витых пар. Адаптер должен иметь разъем типа DB-15 с цепями, соответствующими интерфейсу **AUI**. Максимальная длина соединительного кабеля равна 50 м.

Сеть 10BASE-5 практически полностью вытеснена модификациями 10BASE-2 и 10BASE-T по причине высокой стоимости толстого кабеля, сложности его прокладки из-за большой жесткости, потребности в специальном монтажном инструменте.

3.2.4. Тонкая Ethernet 10BASE-2

В качестве среды в сети 10BASE-2 используется "тонкий" коаксиальный кабель наружным диаметром около 6 мм. Длина кабеля связи (сегмента) не должна превышать 185 метров. Каждый сегмент в среде Ethernet считает-

ся отдельной сетью. К сегменту 10BASE-2 разрешается подключать не более 30 станций. Структура "тонкой" Ethernet 10BASE-2 показана на рисунке 3.9.

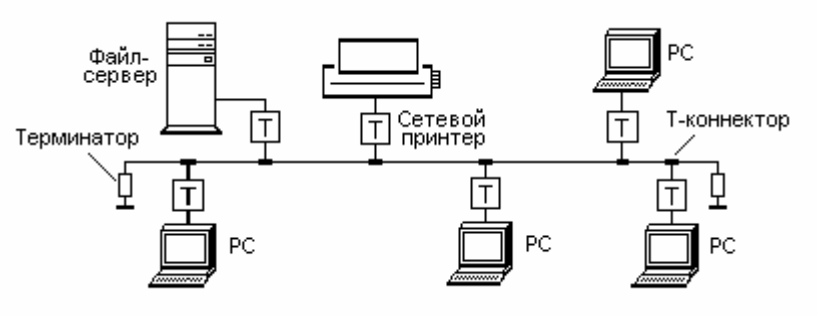


Рисунок 3.9 – Тонкая Ethernet 10BASE-2

Сеть имеет топологию шины. Каждая рабочая станция PC подключается к линии связи через трансивер Т. Задачей трансивера (*приемопередатчика*) является подключение к среде и обнаружение коллизий. Практически во всех 10BASE-2 трансиверы встроены в сетевую интерфейсную плату (сетевой адаптер). Расстояние между трансиверами в кабельном сегменте "тонкой" Ethernet должно составлять не менее 50 см. Подключение трансивера к кабелю осуществляется через Т-образный коаксиальный разъем (Т-коннектор). Соединение сегментов осуществляется разъемом со штыковым креплением типа **BNC** (*Bayonet Nut Connector*).

Для исключения появления отраженных волн, на концах сегментов включается нагрузочный резистор (*терминатор*), имеющий сопротивление, равное волновому сопротивлению тонкого коаксиального кабеля (50 Ом). Конструктивно он вмонтирован в разъем кабеля типа РК50-9-11.

Как и в толстой сети, несколько сегментов тонкой сети могут быть соединены в единую сеть с помощью повторителей. При этом также должно соблюдаться правило **5-4-3**, т.е. разрешается использование в сети не более 5 сегментов, из них два должны быть ненагруженными. Очевидно, что в этом случае устанавливается четыре повторителя. Компьютеры можно подключать только к трем сегментам. Два промежуточных сегмента увеличивают диаметр сети.

3.2.5. Ethernet на основе витой пары 10BASE-T

Сеть на основе витой пары – 10BASE-T является самой распространенной разновидностью локальной компьютерной сети Ethernet. В этой сети используется топология “звезда”. В роли центрального узла сети выступает *концентратор (хаб* – от англ. *Hub*). Он представляет собой многопортовый

повторитель, который передает полученные пакеты во все свои выходные порты, за исключением порта источника пакета, независимо от адресата получателя. В качестве физической среды применяется телефонный неэкранированный провод (витая пара). Длина отвода между компьютером и концентратором не должна превышать 100 м. Рабочие станции подключаются с помощью стандартного сетевого разъема типа RJ-45 к кабельному концентратору (рисунок 3.10). Трансивер в большинстве случаев является внутренним и монтируется на сетевой плате.

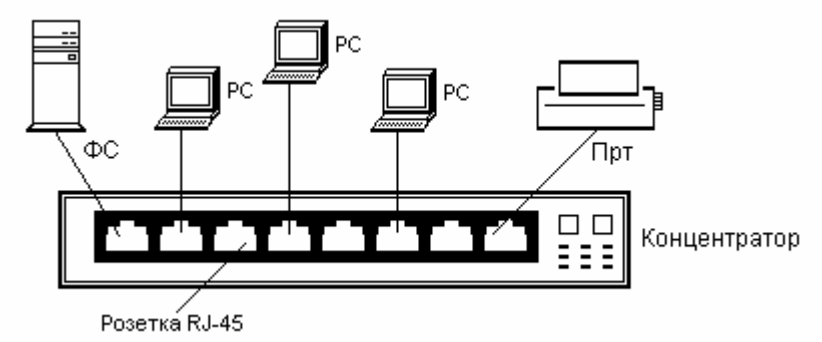


Рисунок 3.10 – Схема сети Ethernet 10 BASE -T

С точки зрения производительности концентраторы просто передают пакеты с использованием всей пропускной способности (полосы) линии связи. Задержка, вносимая повторителем, весьма мала (в соответствии с IEEE 802.3 – менее 3 мкс). Сети, содержащие концентраторы, имеют пропускную способность 10 Мбит/с подобно сегменту на основе коаксиального кабеля и прозрачны для большинства сетевых протоколов, таких как TCP/IP и IPX. При попытке передачи пакетов несколькими станциями одновременно возникает коллизия. В этом случае концентратор выдает соответствующую Jam-последовательность на все выходные порты.

Следует заметить, что концентратор в процессе функционирования сети не только транслирует кадры, но и осуществляет контроль состояния линий связи с компьютерами, отключая их в случае обрыва или короткого замыкания. При этом он может выдавать администратору диагностическое сообщение о возникших неисправностях.

Путем соединения концентраторов 10BASE-T друг с другом, можно создать иерархическую структуру сети (рисунок 3.11). Общее количество рабочих станций в такой сети не должно превышать 1024. При этом следует придерживаться правила: между любыми двумя станциями сети должно находиться не более четырех повторителей. Это требование получило название **правило 4-х хабов**. При выполнении такого условия обеспечивается синхронизация станций и надежное распознавание коллизий.

Недостатком сети на основе концентратора, как и любой сети с шинной топологией, является возможность осуществлять обмен данными в одно и то же время только между двумя компьютерами, в противном случае возникают коллизии.

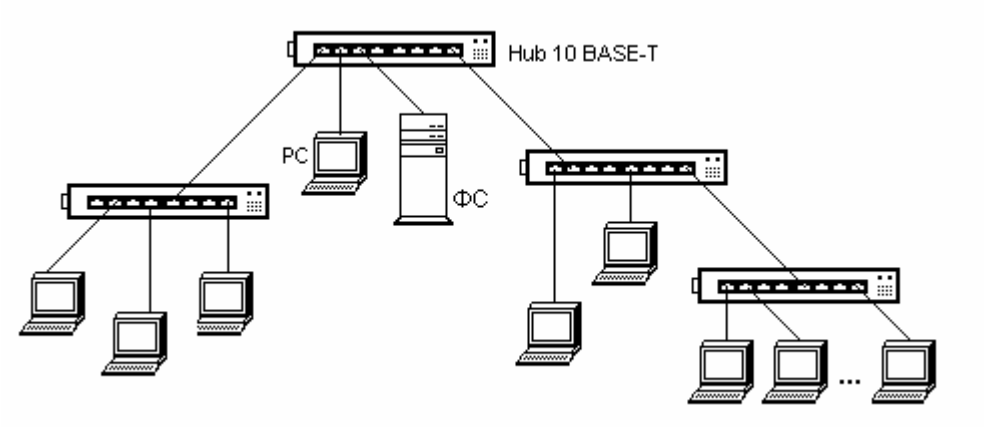


Рисунок 3.11 – Иерархическая структура сети 10BASE-T

Таким образом, в сети со звездной топологией в течение сеанса связи между компьютерами оказываются занятыми только два кабельных сегмента, а остальные не используются. Для устранения этого недостатка были разработаны сетевые коммутаторы, призванные заменить концентраторы (хабы) в сетях 10BASE-T.

3.2.6. Сетевые адаптеры Ethernet

Сетевые адаптеры (*карты*) для персональных компьютеров предназначены для сопряжения параллельной шины компьютера с двухпроводной линией связи. Структурная схема карты сети Ethernet 10BASE-2 изображена на рисунке 3.12. Функционально схема состоит из четырех блоков. До недавнего времени каждый из блоков был реализован в виде одной специализированной БИС. В адаптере, изображенном на рисунке 3.12, используется набор микросхем фирмы *National Semiconductor* типа DP-8390…8392. В последних моделях адаптеров почти все эти микросхемы интегрированы в один чип, который располагается на системной плате компьютера, однако принцип действия адаптера остается прежний. Сетевые адаптеры выпускаются для шины PCI и других типов шин.

Контроллер сетевого интерфейса адаптера обеспечивает функции сетевого протокола по стандарту IEEE 802.3. Он осуществляет преобразование последовательного кода в параллельный и обратно, вычисляет контрольную последовательность кадра (КПК), управляет обменом с внешним буфером

объемом до 64 КБ. В нем содержится буферы для передаваемого и принимаемого пакетов. Адреса буферов располагаются в области верхней памяти (UMA) в диапазоне A0000h…FFFFFh.

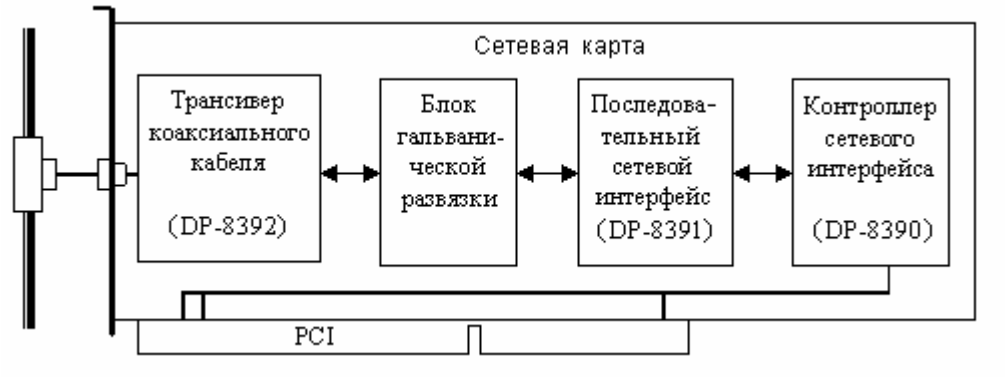


Рисунок 3.12 – Структурная схема сетевой карты Ethernet 10BASE-2

В последовательном сетевом интерфейсе производится кодирование и декодирование сигналов манчестерского кода и преобразование уровней входных и выходных сигналов, а также сигналов коллизий. Приемопередатчик (трансивер) коаксиального кабеля служит для усиления и преобразования линейных сигналов, а также обнаружения коллизии (конфликта). На рисунке 3.13,*а* показана схема трансивера коаксиального кабеля вместе с трансформаторной схемой гальванической развязки.

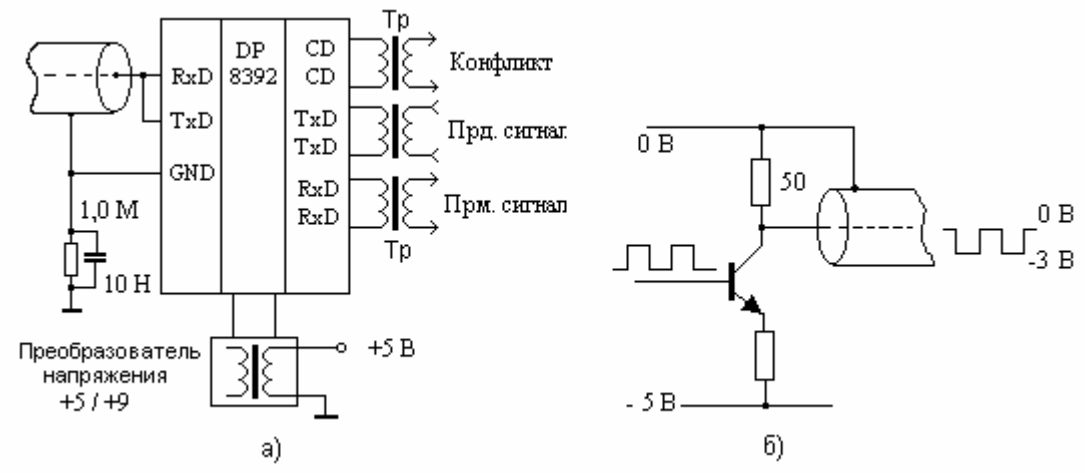


Рисунок 3.13 – Схема трансивера сетевого адаптера а) и выходной каскад б) для коаксиального кабеля

Для разделения цепей питания по постоянному току (гальванической развязки) БИС трансивера применяется преобразователь напряжения

(+5/+9)В с гальванической развязкой цепей питания. Гальваническая развязка цепей сигналов передатчика, приемника и сигнала коллизии осуществляется посредством высокочастотных трансформаторов Тр. Практически все сетевые адаптеры снабжаются трансиверами для коаксиального кабеля (соединитель BNC) и для витой пары (соединитель RJ-45).

Пример реализации схемы выходного каскада трансивера для коаксиального кабеля показан на рисунке 3.13,б.

3.3. Локальная сеть Token Ring

3.3.1. Состав сети и типы кадров

Сеть Token Ring построена в соответствии со стандартом IEEE 802.5. Первоначально скорость передачи в канале была 4 Мбит/с. Затем скорость удалось повысить до 16 Мбит/с. Основным отличием между сетями со скоростью 4 Мбит/с и 16 Мбит/с является введение в 16-Мбитовую архитектуру Token Ring процедуры раннего освобождения маркера (*Early Token Rel*ease – ETR). Это позволило перемещать по кольцу два маркера одновременно. В сети со скоростью 16 Мбит/с и процедурой ETR передающая станция освобождает маркер сразу же после отправки кадра данных. Максимальное число компьютеров в кольце 260, а максимальная длина кольца равна 4 км. При объединении сетей с помощью моста количество компьютеров можно увеличить. К важнейшим преимуществам сетей Token Ring относится **фиксированная задержка** (а не случайная, как у Ethernet). Это позволяет использовать их в системах управления производственными процессами, критичными к задержкам.

Большинство устройств, подключаемых к сети, – это персональные компьютеры (ПК), или рабочие станции (РС). Один ПК служит в качестве файл-сервера (ФС), т.е. устройства, на котором размещается сетевая операционная система (ОС). Периферийные устройства – принтеры, модемы, подключаются либо к РС, либо к файл-серверу. Сервер и рабочие станции взаимодействуют между собой через сетевые платы (карты). Такая плата устанавливается на каждом компьютере, подсоединяемом к сети. Сетевые платы подключаются к устройствам многостанционного доступа MAU (*Multistation Access Unit*), служащими концентраторами сетевых кабелей.

Сетевая плата имеет 9-контактный порт, заканчивающийся разъемом, через который кабель подключается к порту MAU. Она обеспечивает связь рабочей станции с остальной частью сети. В ее состав входят микросхемы Token Ring, реализующие стандартные программы, называемые агентами. ***Агент*** интерпретирует и маршрутизирует все кадры данных, передаваемых

между соответствующим устройством и сетью.

Передача сигналов по кольцу осуществляется биимпульсными сигналами с дифференциальным манчестерским кодированием, при котором "1" передается биимпульсом 01, а "0" $\Rightarrow$ 10. После включения и инициализации сети в кольце циркулирует свободный маркерный блок. Получив свободный маркер, станция, имеющая данные для передачи, меняет состояние маркера на занятый, добавляет к нему блок данных вместе с адресами отправителя и получателя и отправляет дальше по кольцу. Все станции кольца ретранслируют кадр побитно, как повторители. При поступлении кадра на станцию назначения она копирует его во внутренний буфер и устанавливает в кадре признак подтверждения приема. Станция, отправившая кадр в кольцо, после получения его с подтверждением приема изымает этот кадр из кольца и передает в сеть новый свободный маркер.

Существует 3 типа кадров: *кадр маркера, кадр данных и кадр аварийного завершения* (для очистки кольца при возникновении отклонений от нормальной ситуации). Подробнее форматы этих типов кадров рассмотрены ниже.

3.3.2. Упрощенная схема подключения к физическому кольцу

Рабочая станция включается в сеть через сетевую плату. Эта плата подключается к устройствам многостанционного доступа **MAU** (иногда их обозначают **MSAU**). В каждом блоке MAU расположено обычно восемь портов для сетевых устройств, подключаемых к портам MAU абонентскими кабелями. В состав порта MAU входит внутреннее реле Р, получающее питание от компьютера в момент подключения к MAU разъема кабеля сетевой платы (рисунок 3.14).

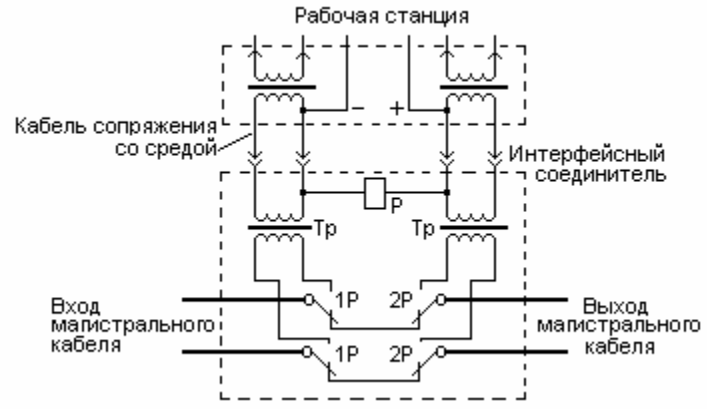


Рисунок 3.14 – Схема подключения рабочей станции к кольцу

При переключении реле магистральный кабель соединяется с входом и выходом рабочей станции, включая тем самым станцию в кольцо.

Расширение сети выполняется с помощью портов RO и RI, размещенных на обоих концах блока MAU. С помощью этих портов и небольшого соединительного кабеля (*патч-кабеля*) можно соединить два MAU. Цепь сигналов между MAU и в самих блоках MAU называются путем основного кольца. К сетевыми картам порты MAU подключаются через абонентский кабель (рисунок 3.15).

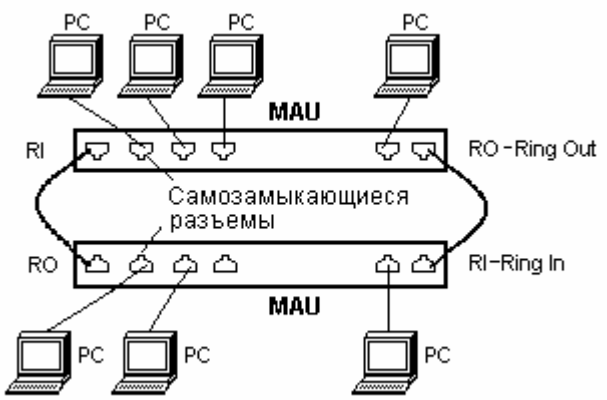


Рисунок 3.15 – Физическая звезда / электрическое кольцо Token Ring

С точки зрения компоновки сеть Token Ring представляет собой *звезду*, а с точки зрения электрических соединений – *кольцо*. Эта схема обеспечивает высокую гибкость при проектировании и компоновке сети.

3.3.3. Управление средой кольца

Время владения линией связи в сети Token Ring ограничивается **временем удержания маркера** (*Token Holding Time*), после окончания которого станция обязана прекратить передачу собственных данных и отправить маркер следующей станции. Компьютер может успеть передать один или несколько кадров, в зависимости от их размера и допустимого времени удержания. Стандартное время удержания кадра установлено равным 10 мс. Если это время истекло, а передача кадра не закончена, то кадр разрешается завершить.

Управление физическим уровнем Token Ring осуществляется посредством выполнения ряда функций, присущих этой архитектуре. Каждая станция в кольце имеет сетевую плату, которая содержит **агента** (стандартные программы, интерпретирующие и обрабатывающие передаваемые кадры).

Этот агент взаимодействует с определенными управляющими станциями Token Ring посредством передачи кадров управления доступом к среде MAC (*Medium Access Control*). Роль управляющей станции заранее определена архитектурой Token Ring.

Управляющие станции обеспечивают функции контроля локального кольца и функции сервера, связанные с управлением кольцом. В процессе управления кольцом контролирующая станция может выполнять функции: пассивного монитора; активного монитора; сервера отчетов о конфигурациях; сервера параметров кольца; монитора ошибок кольца; сервера моста LAN (ЛКС); механизма выдачи отчетов.

Пассивные мониторы представляют собой станции общего назначения. Они также служат для обнаружения сбоев в активном мониторе. Если пассивные мониторы не обнаруживают MAC-кадра активного монитора, то они вступают в состязание за выполнения роли активного монитора.

Активный монитор является главным менеджером связи в кольце. Он отвечает за поддержание передачи данных и управляющую информацию, циркулирующую между всеми станциями кольца. Активный монитор обеспечивает выполнение следующих основных функций:

1) поддержку главного генератора тактовых импульсов (ГТИ) с целью обеспечения синхронности ГТИ всех станций;

2) инициирование уведомления соседа о своем адресе;

3) мониторинг уведомления соседа (поддерживает постоянный процесс уведомления, используя таймер протокола);

4) контроль и поддержание надлежащей задержки в кольце;

5) мониторинг передачи маркера и кадров; проверку бита завершения передачи кадра между станциями, а при ошибках – выполнение очистки кольца;

6) выявление утерянных маркеров и кадров; проверяет максимальное время оборота кадра в кольце (10 мс); если кадр не обнаружен, а интервал истек, монитор выполняет очистку кольца;

7) очистку кольца; при обнаружении сбоев, программой синхронизации передается специальный широковещательный кадр (*Ring Purge*), который очищает кольцо перед передачей нового маркера.

В процессе функционирования сети только одна рабочая станция может быть активным монитором. Эта роль выполняется любой рабочей станцией кольца. Она назначается динамически в соответствии с процессами Token Ring в данном кольце.

3.3.4. Кадры сети Token Ring

Как уже отмечалось выше, в сети Token Ring используются 3 различных формата кадров: ***маркер, данные*** и ***кадр аварийного завершения*** (для очистки кольца при возникновении отклонений от нормальной ситуации).

Маркер содержит лишь 3 байта: стартовый разделитель (*Start Delimiter*) **СтР**; байт управления доступом **УД** (*Access Control*) и конечный разделитель **КР** (*End Delimiter*) (рисунок 3.16).

Рисунок 3.16 – Формат маркерного кадра сети Token Ring

Кодовая комбинация стартового разделителя СтР представляет собой уникальную последовательность сигналов манчестерского кода с нарушением чередования полярностей вида JK0JK000. Здесь *J* и *K* – запрещенные сигналы для манчестерского кода, причем символом ***J*** обозначен сигнал низкого уровня, а ***K*** – высокого. Использование таких сигналов позволяет легко обнаружить на приемной стороне комбинацию стартового и конечного разделителей и произвести синхронизацию по блокам.

Байт управления доступом УД содержит три бита приоритета (P), бит признака маркерного кадра (T=1), бит индикации активного монитора (M=1) и три резервных бита (R). В сети Token Ring предусмотрено восемь уровней приоритета, начиная с 000 (низший) до 111 (высший). Бит монитора M устанавливается в 1 активным монитором и сбрасывается в 0 любой другой станцией, передающей маркерный или информационный блок. При обнаружении активным монитором кадра с установленным битом M, он констатирует, что кадр обошел кольцо и не был обработан ни одной из станций сети. Если это информационный кадр, то он исключается из кольца, а если маркер, то передается дальше по кольцу.

Конечный разделитель КР содержит шестибитовую комбинацию манчестерского кода с нарушением правил образования сигналов вида **JK1JK1**, а также биты признаков ***I*** (*Intermediate*) и ***E*** (*Error*). Значение *I*=1 индицирует промежуточный кадр, а *I*=0 – последний. Признак E устанавливается станцией-отправителем в нулевое состояние. Если транзитная станция обнаруживает в кадре ошибку, то она переводит бит E в единицу.

Кадр данных предназначен для передачи информационного сообще-

ния либо команд управления кольцом. Кроме трех байтов СтР, УД и КР, входящих в состав маркерного блока, кадр данных содержит несколько дополнительных полей. Общий формат кадра Token Ring показан на рисунке 3.17. Цифры в полях кадра показывают длину соответствующего поля в байтах.

СтР	УД	УК	Адрес получателя	Адрес отправителя	Данные	КПК	КР	СК
1	1	1	6	6	< 4502	4	1	1

Рисунок 3.17 – Формат кадра сети Token Ring

В поле управление кадром (УК) задается тип *кадра* (сообщение MAC-уровня или пользовательские данные LLC-уровня). Если кадр определен как MAC, то поле указывает, какой из типов кадров представлен данным кадром. В сети используется 6 типов управляющих кадров.

1. "**Тест дублирования адреса**" – используется при первом подключении к среде, чтобы удостовериться, что адрес подключаемой станции уникальный.
2. "**Существует активный монитор**"– периодически посылается в кольцо активным монитором, чтобы сообщить о том, что он работоспособен.
3. "**Существует резервный монитор**"– отправляется любой станцией, не являющейся активным монитором.
4. "**Маркер заявки**"– посылается резервным монитором, когда подозревается, что активный монитор вышел из строя. Затем резервные мониторы договариваются о том, какой их них станет активным монитором.
5. "**Сигнальный кадр**"– передается станцией в случае обнаружения серьезных сетевых проблем (обрыв кабеля, обнаружение станции, передающей кадр без получения маркера). Диагностирующая программа по станции, отправившей "Сигнал", локализует неисправность.
6. "**Очистка**"– используется активным монитором для перевода всех станций в исходное состояние и очистки кольца от всех ранее посланных кадров.

Адреса получателя и отправителя имеют структуру, аналогичную сети Ethernet. Старший бит адреса получателя определяет групповой (1) или индивидуальный (0) адрес, а "1" в старшем бите адреса отправителя индицирует, что в кадре имеется специальное поле маршрутной информации.

Размер *поля данных*, следующего за адресом отправителя, может иметь произвольную минимальную длину, в том числе и нулевую. Максимальный размер зависит от скорости передачи: около 4500 байт при скорости 4 Мбит/с и 16 кбайт – при 16 Мбит/с. В поле данных может быть вложен пакет другого протокола, например LLC.

Контрольная последовательность кадра КПК содержит избыточные элементы, полученные в результате циклического кодирования аналогично стандарту IEEE 802.3 сети Ethernet.

Последним полем кадра является *байт состояния кадра* СК. Первые и последние четыре бита этого поля повторяют друг друга. Это повышает достоверность записанной там информации, так как она не кодируется помехозащищенным кодом. Первым следует *бит распознавания адреса*, выполняющий функцию флага обнаружения получателем своего адреса. При совпадении адреса в кадре с адресом станции, получатель перед отправкой кадра далее по кольцу, устанавливает бит распознавания в единичное состояние.

Второй бит (*индикатор копирования кадра*) служит для индикации успешного копирования информации из полученного кадра. Если получатель распознал свой адрес, и в буфере имеется свободное место, а также скопировал информацию из полученного пакета, он устанавливает этот бит в единичное состояние. Биты распознавания адреса и копирования кадра активно используются управляющими станциями кольца. Для отправителя они носят второстепенный характер, ибо решение о повторной пересылке при утере кадра принимается обычно на более высоком, транспортном уровне. Два следующих бита имеют нулевое значение.

Кадр "Аварийное завершение" содержит только два байта: стартовый и конечный разделители. Этот кадр может быть выдан в произвольном месте потока битов. Он сигнализирует об отмене передачи кадра данных или маркерного кадра.

При построении больших сетей Token Ring возможно использование нескольких колец. Отдельные кольца связываются друг с другом, как и в иных сетях, с помощью мостов (рисунок 3.18). Мосты бывают "прозрачными" и с маршрутизацией от источника. Последние позволяют связать в единую сеть несколько колец, использующих общий сетевой IPX- или IP-адрес.

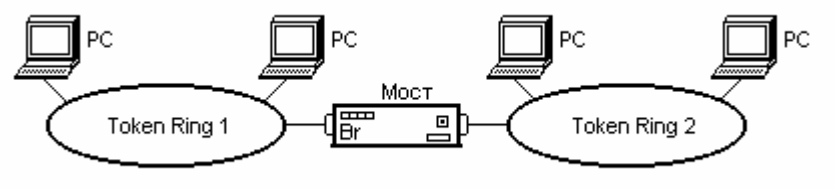


Рисунок 3.18 – Соединение колец Token Ring с помощью моста

Кадры из кольца 1 адресованные компьютеру этого же кольца никогда не попадут в кольцо 2 и наоборот. Через мост пройдут лишь блоки, адресованные станциям соседнего кольца. Фильтрация кадров осуществляется по физическому адресу и номеру порта. На основе этих сведений формируется

собственная база данных, содержащая информацию об объектах колец, подключенных к мосту.

3.4. Высокоскоростные локальные сети

3.4.1. Сети FDDI

Сеть **FDDI** – *Fiber Distributed Data Interface* основывается на технологии Token Ring, развивая и совершенствуя ее основные технические решения. Задача сетевой технологии – повысить скорость и надежность передачи (отказоустойчивость) в случае повреждении кабеля или некорректной работы узла из-за высокого уровня помех.

Особенностью сети является использование в качестве среды волоконно-оптической линии связи (ВОЛС). Передача данных выполняется со скоростью 100 Мбит/с по двойному кольцу длинной до 100 км. Передаваемые данные подвергаются логическому кодированию вида 4B/5B. Из 32 кодовых комбинаций символов для передачи данных используются 16, девять комбинаций служебные, остальные – запрещенные. Значение управляющих и информационных комбинаций приведены в таблице 3.1.

При таком кодировании четырехбитовые данные и управляющие кодовые слова передаются пятью битами. В этом случае любые группы из четырех битов имеют минимум две смены позиций, благодаря чему исключаются длинные паузы изменения уровней сигналов в линии и тем самым обеспечивается устойчивая синхронизация приемного устройства по битам.

Пятибитовые последовательности кодируются линейным кодом NRZI. Эффективная скорость передачи кодом 4B/5B составляет всего лишь 80%. Поэтому для обеспечения эффективной скорости 100 Мбит/с модуляция производится с частотой 125 МГц. В паузах передачи данных между портами непрерывно передается 5-битовая синхронизирующая комбинация 11111 – ***Idle*** (пустой).

Область применения FDDI – ответственные участки сетей: магистральные соединения между крупными сетями (зданиями), а также подключение высокопроизводительных серверов. Максимальное число станций в сети – 500.

Сеть строится на основе двух оптоволоконных колец, которые образуют основной и резервный пути передачи (рисунок 3.19). Наличие двух колец – это основной способ повышения надежности передачи данных между узлами. Различают первичное (*Primary*) и вторичное (*Secondary*) кольцо.

Таблица 3.1 – Значения символов таблицы кодирования 4B/5B

Символы		Значение кодового слова
Имя	Кодовые слова	
I	11111	Idle (Дополнительный символ для синхронизации)
H	00100	Halt (Разрешение активизации для останова)
Q	00000	Quiet (Отсутствие переходов)
J	11000	Frame delimiter (Разделитель кадра)
K	10001	Frame delimiter
L	00101	Frame delimiter (только для FDDI–II)
T	01101	Frame delimiter
R	00111	Логический "0"
S	11001	Логическая "1"
0	11110	Данные 0 – Исходная комбинация "0000"
1	01001	Данные 1 – "0001"
2	10100	Данные 2 – "0010"
3	10101	Данные 3 – "0011"
. . .	. . .	. . .
E	11100	Данные E – "1110"
F	11101	Данные F – "1111"
V	00001	Запрещенный символ V
. . .	. . .	. . .

В нормальном режиме работы сети данные проходят по участкам только первичного кольца. Этот режим называют "сквозным" или "транзитным". Второе кольцо в таком режиме не используется.

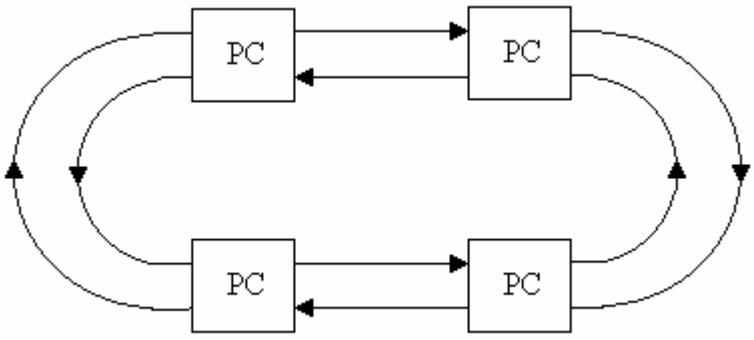


Рисунок 3.19 – Топология двухкольцевой сети FDDI

В случае какого-либо отказа когда часть первичного кольца не может передать данные, (обрыв кабеля или отказ узла) первичное кольцо объединяется со вторичным, вновь образуя единое кольцо (рисунок 3.20 и 3.21).

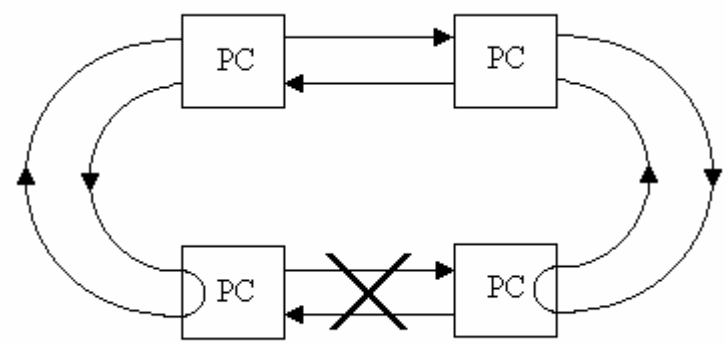


Рисунок 3.20 – Восстановление сети FDDI при повреждении кабеля

Режим объединения называют *свертыванием* кольца (или сворачиванием). Операция свертывания кольца осуществляется средствами концентраторов или сетевых адаптеров FDDI. Для упрощения этой процедуры данные по первичному кольцу передаются в одном направлении, а по вторичному – в обратном. В стандартах FDDI много внимании уделяется различным процедурам, которые позволяют определить наличие отказа в сети, а затем произвести необходимую реконфигурацию.

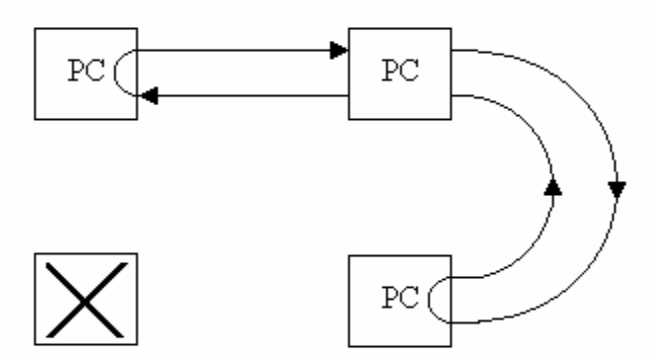


Рисунок 3.21 – Восстановление сети FDDI при повреждении рабочей станции

Сеть FDDI может полностью восстанавливать свою работоспособность в случае единичных отказов ее элементов. При множественных отказах сеть распадается на несколько не связанных сетей. Реконфигурация кольца осуществляется за счет специальных оптических переключателей.

Доступ к среде в FDDI очень похож на доступ в Token Ring. Здесь применяется алгоритм раннего освобождения маркера (*Token*), т.е. станция передает маркер доступа (свободный маркер) следующему компьютеру сразу же после окончания передачи последнего бита кадра, не дожидаясь возвращения по кольцу этого кадра с битом подтверждения приема. Таким образом, по кольцу могут одновременно продвигаться кадры нескольких станций.

Отличие метода доступа состоит в том, что время удерживания маркера в сети FDDI не является постоянным, как в Token Ring. **Время удерживания** $t_{уд}$ – это время владения средой. По истечении установленного интервала $t_{уд}$ станция обязана прекратить передачу собственных данных и передать маркер далее по кольцу. Для Token Ring время удерживания $t_{уд}$ =10 мс. Такой интервал установлен по той причине, чтобы за время удерживания станция успела передать хотя бы один кадр. Время удерживания в FDDI зависит от загрузки кольца. При небольшой загрузке оно увеличивается, а при перегрузках может уменьшаться до нуля. Рабочие станции на этапе инициализации сети договариваются о величине времени удерживания маркера. Каждая станция передает свое значение, а в итоге для кольца устанавливается минимальное время.

Формат кадра FDDI близок к формату Token Ring, основное отличие заключается в отсутствии полей приоритетов (в Token Ring имеется 8 уровней). В FDDI установлено два класса трафика: *асинхронный* (допускающий задержку) и *синхронный*. Трафик второго класса обслуживается всегда, даже при перегрузках кольца.

В FDDI сети нет выделенного активного монитора, а его функцию может выполнить любая рабочая станция, причем все станции и концентраторы равноправны. В случае обнаружения отклонений от нормальной работы сети они начинают процесс повторный инициализации сети, а затем ее реконфигурации. Основным параметром правильного функционирования кольца является максимально допустимое время *оборота маркера* (**TRT**–*token rotation time*). Если время, прошедшее с момента последнего прохождения маркера, больше допустимого, то предполагается наличие ошибки в кольце (нарушение функционирования). Для обнаружения и устранения нарушения работы кольца в сети используются специальные процессы:

- посылка маркера заявки (*Claim Token*);
- инициализация – посылка сигнального (*маячного*) кадра (*Beacon*).

В случае обнаружения одной из станций неполадок в кольце, она посылает маркер заявки *Claim Token*, в котором содержится известное этой станции допустимое время оборота маркера. После обнаружения такого кадра в кольце все остальные станции посылают в сеть маркер заявки со своим допустимым временем оборота маркера. Процесс инициализации обнаруживается по наличию в кольце многих пакетов с маркером заявки. Все кадры, за исключением имеющего минимум времени оборота, подавляются. Процесс заканчивается, как только рабочая станция зафиксирует свой *Claim Token*, прошедший по всему кольцу. В результате получается, что все станции запоминают минимальное значение TRT. Станция, имеющая минимальное время оборота маркера, является ответственной за посылку в кольцо свободного маркера.

Сигнальный процесс *Beacon* служит для локализации места нарушения в кольце. Во время этого процесса станции посылают продолжительное время в кольцо сигнальные кадры. Каждая станция, получившая кадр от верхнего соседа (откуда приходит сообщение), прекращает передачу кадров *Beacon*. В конце концов, остаются передавать только те станции, которые не обнаружили кадр от своих соседей. На основании этого определяются поврежденные станции.

3.4.2. Сети Fast Ethernet

С начала 90-х годов прошлого века стал ощущаться недостаток пропускной способности сети Ethernet. При разработке новой Ethernet специалисты разделились на 2 лагеря, что привело к появлению двух новых стандартов: Fast Ethernet и 100 VG-Any LAN**.** Сеть **Fast Ethernet** оказалась максимально приближена к классической Ethernet, вторая – сеть **100 VG-Any LAN,** совместимая с Token Ring.

Все отличия Fast Ethernet (спецификация IEEE 802.3u) от классической Ethernet сосредоточены на физическом уровне. Верхние уровни остались прежними. Для технологии Fast Ethernet разработаны различные варианты физического уровня, отличающиеся не только типом кабеля и электрическими параметрами импульсов, но и способом кодирования сигналов, а также количеством используемых в кабеле проводников. В предыдущих подразделах отмечалось, что в классической Ethernet для передачи линейных сигналов применяется манчестерское кодирование. Однако использование его для более высокоскоростных сетей (100 или 1000 Мбит/с) является неприемлемым, так как электрические кабели не рассчитаны на работу при столь высоких частотах. Поэтому в сетях Fast Ethernet применяются другие линейные коды (**NRZI** и **MLT-3**), а для улучшения их синхронизирующих свойств входные данные подвергаются дополнительному кодированию. Такая дополнительная обработка состоит в логическом блочном кодировании, при котором одна группа бит по определенному алгоритму заменяется другой группой. Наиболее распространенными типами подобного кодирования являются избыточные коды 4B/5B, 8B/6T и 8B/10T. По этой причине физический уровень Fast Ethernet имеет более сложную структуру, чем классический Ethernet.

Стандартом предусмотрено несколько вариантов сети Fast Ethernet: 100BASE-T4, 100BASE-TX и 100BASE-FX.

100BASE-TX предназначена для передачи данных по двум витым парам кабеля 5-й категории, причем одна пара используется для передачи данных, а вторая – для их приема. Максимально допустимая длина кабеля 100

метров. Кабель связи может быть экранированным, либо неэкранированным. Используется алгоритм преобразования кодов данных 4B/5B и способ линейного кодирования MLT-3.

100BASE-FX – в качестве сегментов применяется два световода оптоволоконного кабеля (один для передачи другой для приема), в частности мультимодовое волокно диаметром 62,5/125 мкм, работающее в инфракрасном диапазоне 1350 нм. Максимальная длина сегмента составляет 412 метров при полудуплексном режиме и до 2-х км при полном дуплексе. Используется алгоритм преобразования кодов данных **4B/5B** и способ линейного кодирования **NRZI**. Спецификации 100BASE-TX и 100BASE-FX разработаны американским национальным институтом стандартов ANSI, их иногда в общем виде обозначают как 100BASE-X.

100BASE-T4 – это особая спецификация, предложенная комитетом IEEE 802.3u, согласно которой передача данных осуществляется по четырем витым парам неэкранированного телефонного кабеля UTP категории 3 длиной до 100 метров. Рекомендован алгоритм преобразования кодов данных 8B/6T и способ линейного кодирования NRZI.

Коаксиальный кабель в перечне используемых линий связи не значится. Это поясняется тем, что на небольших расстояниях витая пара может передавать данные с той же скоростью, что и коаксиальный кабель, однако сеть получается более дешевой и удобной в эксплуатации. На больших расстояниях оптоволоконные линии обладает более широкой полосой пропускания, чем коаксиальные кабели, а стоимость сети на оптической линии не намного выше. Отказ от коаксиального кабеля привел к тому, что сеть Fast Ethernet всегда имеет иерархическую древовидную структуру, построенную на концентраторах-повторителях, аналогичную сети Ethernet 10BASE-T (рисунок 3.22).

Для идентификации варианта сети разработан специальный протокол распознавания, позволяющий строить сети, содержащие оборудование и кабельные сегменты, отвечающие разным требованиям. Для всех трех вариантов Fast Ethernet характерны следующие особенности:

- форматы кадров не совпадают с форматом классической Ethernet;
- межкадровый интервал равен 0,96 мкс (в классической сети Ethernet – 9,6 мкс), а длительность единичного элемента (битовый интервал) составляет 10 нс.
- метод линейного кодирования отличается от кодирования в сети Ethernet, в частности, применяется логическое кодирование типа 4B/5B или 8B/6T и линейное кодирование MLT-3 или NRZI (в 10BASE-T используется манчестерское кодирование);
- для индикации не занятого состояния среды передается специальный символ "**Idle**" (пустой, незанятый), в качестве которого применяется одна из

неиспользуемых комбинаций линейного кода (в классической Ethernet при свободной линии сигнал в ней отсутствует полностью).

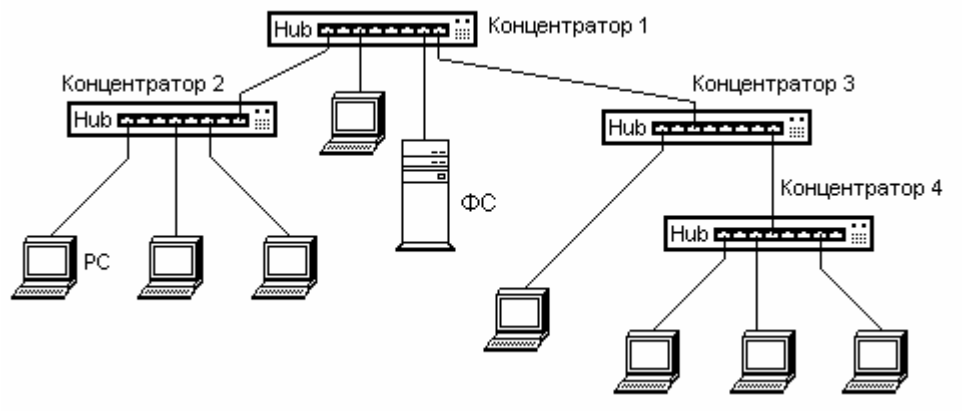


Рисунок 3.22 – Структура сети Fast Ethernet

Служебный символ Idle (11111), постоянно передается трансмиттером концентратора для контроля физического состояния соединений и поддержки приемных устройств сетевых адаптеров в синхронном и синфазном состоянии. Для отделения кадра Ethernet от символов Idle в потоке сигналов манчестерского кода используется пара служебных кодовых слов 11000 и 10001 (символы J и K соответственно, см. таблицу 3.1), а после завершения кадра перед первым символом Idle вставляется управляющее слово 01101 (символ T).

Физический уровень содержит три компонента: независимый от среды интерфейс **MII** (*Media Independent Interface*); устройство физического уровня **PHY** (*Physical layer device*) и подуровень согласования (*reconciliation sublayer*). Подуровень согласования ввели для того, чтобы MAC-уровень, рассчитанный на интерфейс AUI, мог работать с физическим уровнем через интерфейс *MII*.

Универсальная схема подключения компьютера или любого другого оборудования (например, сетевого принтера) к 100-мегагерцовой Ethernet показана на рисунке 3.23.

Физическая среда служит для передачи сигналов Ethernet от одной станции к другой. Для перечисленных видов физической среды, используемых Fast Ethernet (T4, TX и FX) применяется зависимый от среды интерфейс **MDI** (*Medium Dependent Interface*), в частности 8-контактный разъем RJ-45 для неэкранированных витых пар. Для экранированных витых пар в качестве разъема MDI необходимо использовать разъем STP IBM типа 1, который является разновидностью экранированного разъема DB-9. Подключение оптических сегментов рекомендуется выполнять посредством соединителей типа

MIC (*Media Interface Connector*), используемых также в сетях FDDI, или дуплексным соединителем типа SC, являющимся единственным рекомендованным комитетом IEEE для употребления в сети 100BASE-FX Fast Ethernet.

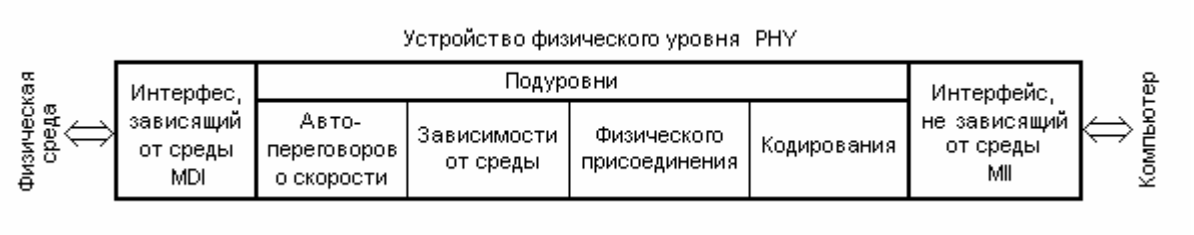


Рисунок 3.23 – Структурная схема подключения оборудования в Fast Ethernet

Устройство физического уровня *PHY* выполняет ту же функцию, что и трансивер в классической Ethernet. Поскольку Fast Ethernet может использовать различный тип кабеля, то для каждой среды требуется уникальное предварительное преобразование сигнала, обеспечивающее снижение искажений сигналов данных и повышение помехоустойчивости их приема, а также улучшение условий синхронизации тактовых генераторов на передающей и приемной сторонах. Устройство физического уровня в свою очередь делится на четыре подуровня (рисунок 3.23). В **подуровне кодирования** выполняется кодирование и декодирование данных с использованием алгоритмов логического кодирования 4B/5B или 8B/6T.

В **подуровнях физического присоединения** и зависимости от физической среды осуществляется связь между подуровнем кодирования и зависящим от среды интерфейсом MDI. При этом выполняется преобразование сигналов по алгоритму NRZI или MLT-3 (см. подразделы 2.2.2 и 2.2.3).

В **подуровне автопереговоров** реализуется взаимодействие двух взаимосвязанных портов, в результате которого автоматически выбирается наиболее эффективный режим работы: дуплексный или полудуплексный со скоростью обмена 10 или 100 Мбит/с.

Конструктивно физический уровень может представлять собой набор интегральных схем в сетевом порту или выполнен в виде небольшого автономного блока, закрепленного на соединительном кабеле, длина которого не должна превышать 0,5 м.

Сопряжение с компьютером производится посредством независимого от среды интерфейса *MII (Media Independent Interface*). Разъем *MII* отличается от соединителя типа AUI и имеет 40 контактов. Интерфейс *MII* является опционным, он может поддерживать работу с 10- и 100-мегабитной Ethernet. Задачей *MII* является преобразование сигналов, поступающих от *PHY*, в форму, приемлемую для стандартного набора микросхем Ethernet. *PHY* и *MII*

интерфейсы могут быть объединены на одной сетевой карте, вставляемой в компьютер.

Существует два варианта реализации интерфейса *MII*: внутренний и внешний. При внутреннем варианте микросхема, реализующая подуровни MAC и согласования, с помощью интерфейса MII соединяется с микросхемой трансивера внутри платы сетевого адаптера системного блока компьютера или модуля маршрутизатора. Микросхема трансивера реализует все функции устройства PHY. Внешний вариант соответствует случаю, когда трансивер вынесен в отдельное устройство и соединен кабелем *MII* через разъем *MII* с микросхемой MAC-подуровня.

В спецификации 100BASE-T4 выполняется предварительное преобразование передаваемых данных по способу 8B/6T, в соответствии с которым восьмибитовые последовательности преобразуются в шестиэлементные трехуровневые (тернарные) посылки: "–", "0" и "+". Способ кодирования аналогичен рассмотренному в подразделе 2.2.3 способу 4B/3T. Кодирование осуществляется в соответствии с таблицей преобразования, содержащей все 256 возможных 8-битовых комбинаций. Из 729 возможных значений результирующего кода выбираются те значения, которые отвечают следующим критериям:

- результирующий уровень постоянного напряжения кодового символа равен 0;
- в пределах символа происходит как минимум два изменения уровня выходного напряжения.

В таблице 3.2, в качестве примера, приведена часть комбинаций кода 8B/6T.

Особенностью реализации 100BASE-T4 при использовании витой пары 3-й категории является передача данных по трем парам кабеля одновременно со скоростью передачи информации 33,3 Мбит/с по каждой из пар, что дает в итоге скорость 100 Мбит/с. Четвертая пара кабеля применяется для контроля коллизий. Благодаря трехпозиционной модуляции скорость передачи тернарных сигналов по каждой витой паре составляет 6/8 от 33,3 Мбит/с, что соответствует тактовой частоте 25 МГц. Именно с такой частотой работает задающий генератор интерфейса *МП*.

В сети Fast Ethernet поддерживается функция автовыбора взаимодействия – *Auto-negotiation*, с помощью которой два взаимодействующих устройства физического уровня PHY могут автоматически выбрать наиболее эффективный режим работы.

Всего в настоящее время определено 5 различных режимов, поддерживающих устройства на витых парах:

- 10BASE-T (2 пары категории 3);
- 10BASE-T full–duplex (2 пары категории 3);

- 100BASE-TX (2 пары категории 5);
- 100BASE-TX full-duplex (2 пары категории 5);
- 100BASE-T4 (4 пары категории 3).

Таблица 3.2 – Пример фрагмента таблицы кодирования 8B/6T

8-битовые комбинации	6-элементные тернарные последовательности
00000000	+ − 0 0 + −
00000001	0 + − + − 0
00000010	0 − + 0 − +
00000011	0 − + + 0 −
00000100	− + 0 + 0 −
00000101	+ 0 − − + 0
.....	.....
11111110	− + 0 + 0 0
11111111	+ 0 − + 0 0

Режим 10BASE-T имеет самый низкий приоритет при переговорном процессе, а режим 100BASE-T4 – самый высокий. Переговорный процесс происходит после включения питания устройства, а также может быть инициирован в любой момент времени модулем управления. Для организации переговорного процесса используются служебные сигналы проверки целостности линии технологии 10BASE-T – *link test pulses*, если узел-партнер поддерживает только стандарт 10BASE-T. Узлы, поддерживающие функцию автовыбора, также используют существующую технологию сигналов проверки целостности линии, при этом они посылают пакеты служебных сигналов, в которые инкапсулирована информация переговорного процесса *Auto-negotiation*. Такие пакеты носят название ***FLP*** *(Fast Link Pulse burst).*

Устройство, начавшее процесс автовыбора, посылает своему партнеру пачку сигналов FLP, в которой содержится 8-битовое слово, отображающее предлагаемый режим взаимодействия, начиная с самого приоритетного, поддерживаемый данным узлом. Если в удаленном компьютере поддерживается функция *Auto-negotiation* и его устройство связи может реализовать предложенный режим, то он отвечает пачкой импульсов FLP, подтверждающей данный режим и на этом переговоры заканчиваются. Если же вызываемый компьютер поддерживает менее приоритетный режим, то он указывает его в ответе и этот режим выбирается в качестве рабочего. Таким образом, всегда выбирается наиболее приоритетный общий режим функционирования рабочих станций.

Компьютер, способный поддерживать только технологию 10BASE-T, каждые 16 миллисекунд посылает импульсы для проверки целостности ли-

нии, связывающей его с соседней станцией. Такой компьютер не реагирует на запрос FLP, посылаемый ему компьютером с функцией *Auto-negotiation*, и продолжает посылать свои импульсы. Станция, получившая в ответ на запрос FLP только импульсы проверки целостности линии, фиксирует, что его партнер может работать только по стандарту 10BASE-T и устанавливает этот режим работы и для себя.

Станции, выполненные по спецификации FX и TX, могут работать в дуплексном режиме. При таком способе обмена не используется метод доступа к среде CSMA/CD, и отсутствуют коллизии, потому что каждый узел одновременно передает и принимает кадры данных по отдельным линиям передачи Tx и приема Rx. Полнодуплексная работа возможна только при соединения сетевого адаптера с коммутатором или же при непосредственном соединении коммутаторов между собой.

3.4.3. Технология 100VG - Any LAN

Технология 100VG-Any LAN существенно отличается от классической Ethernet. Основными отличительными признаками является следующие:

а) другой метод доступа (*Demand Priority* – запрос приоритета), который обеспечивает более справедливое распределение пропускной способности сети; кроме того, он поддерживает приоритетный доступ для синхронных приложений;

б) кадры передаются не всем станциям, а только станции назначения;

в) в сети имеется выделенный арбитр доступа – концентратор;

г) поддерживаются кадры двух технологий Ethernet и Token Ring (поэтому появилась добавка Any LAN);

д) данные передаются одновременно по четырем парам UTP кабеля 3-й категории, по каждой паре со скоростью 25 Мбит/с;

е) для логического кодирования применяется линейный код 5B/6B, обеспечивающий спектр линейного сигнала шириной до 16 МГц.

Сеть состоит из центрального концентратора, называемого корневым, и соединенных с ним узлов и концентраторов (рисунок 3.24). Метод доступа *Demand Priority* основан на передаче функций арбитра концентратору.

Допускаются 3 уровня каскадирования. Каждый концентратор должен быть заранее настроен либо на работу по протоколу Ethernet, либо Token Ring. Концентратор в паузах между передачей опрашивает порты. Рабочая станция, желающая передать сообщение, посылает специальный сигнал, запрашивая передачу и указывая свой приоритет (высокий/низкий). Если сеть свободна, концентратор разрешает передачу пакета. После анализа адреса

концентратор отправляет пакет получателю.

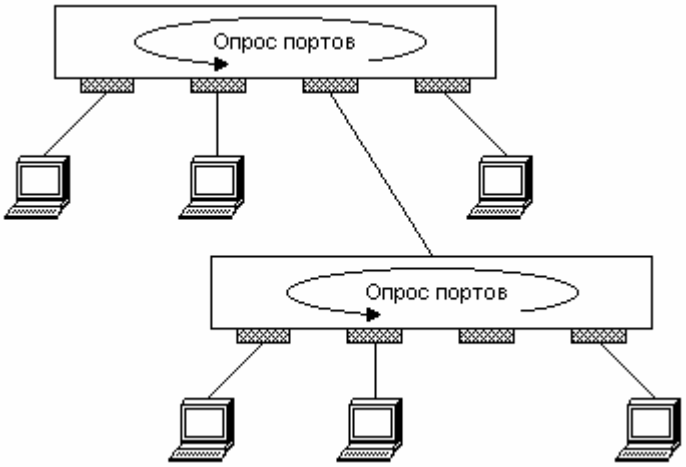


Рисунок 3.24 – Компьютерная сеть 100 VG - Any LAN

Технология не приобрела особой популярности, поскольку более распространенной оказалась Fast Ethernet. Кроме того, появилась новая более эффективная технология Gigabit Ethernet.

3.4.4. Гигабитовые технологии в сетях Ethernet

Из гигабитовых технологий первой была утверждена в 1998 году сеть Ethernet IEEE 802.3z, получившая обозначение **1000BASE-FX**. Эти сети ориентированы на применение 4-х витых пар категории 5 или выше (длиной до 100 м, с разъемом RJ-45) и оптоволоконных кабелей. В сети применяется логическое кодирование типа 8B/10B, т.е. каждый байт кодируется в процессе передачи десятью битами. При этом из 2^{10} возможных 10-разрядных кодовых комбинаций выбираются такие, в которых не содержится более 4 одинаковых битов подряд, и ни в одной кодовой комбинации нет более 6 нулей или 6 единиц. Этим приемом обеспечиваются удовлетворительные условия тактовой синхронизации и высокая стабильность постоянной составляющей передаваемых сигналов. Кроме этого, применение такого кода исключает перегрев лазерных диодов передатчика при поступлении от источника информации длинных последовательностей единиц.

В гигабитовой Ethernet-технологии имеется много общих признаков со своими предшественниками 10BASE и100BASE. Прежде всего – это метод доступа к среде передачи данных CSMA/CD, полудуплексный и полнодуплексный режимы работы, а также форматы кадров Ethernet. Топология сети имеет вид иерархической звезды. В то же время использование витой пары

кабеля 5-й категории потребовало внести серьезные изменения в реализацию физического уровня адаптера. Если отличия между Ethernet и Fast Ethernet минимальны и не затрагивают MAC-уровня, то при разработке стандарта *Gigabit Ethernet* 1000BASE-T разработчикам пришлось внести коррективы в физический уровень, а также изменить и MAC-уровень.

Чтобы обеспечить максимальный диаметр сети размером 200 м (два кабеля по 100 м и коммутатор), минимальная длина кадра в стандарте Gigabit Ethernet была увеличена до **512 байт**. По этой причине при недостаточном объеме информации сетевой адаптер дополняет поле данных до длины 448 байт запрещенными комбинациями (так называемым расширением). В то же время увеличение минимальной длины кадра негативно сказывается при передаче коротких служебных сообщений, например квитанций, так как полезная информация в кадре становится существенно меньше общего количества передаваемых битов. С целью сокращения избыточности при использовании длинных кадров для передачи коротких квитанций стандартом Gigabit Ethernet разрешена выдача в линию нескольких кадров подряд в режиме монопольного захвата среды, т.е. без передачи линии другим станциям. Такой монопольный режим захвата называется **Burst Mode**. В этом режиме станция может передавать подряд несколько кадров с общей длиной не более 8192 байт (*BurstLength*).

В технологии **1000BASE-T** применяется помехозащищенное восьмипозиционное сверточное кодирование (на восемь различных состояний). Символы передаются по всем четырем витым парам кабеля одновременно с использованием пятиуровневого кодирования PAM-5 (–2; –1; 0; 1; 2) , т.е. один единичный элемент τ_0 отображает 2,32 бита информации. Передача ведется одновременно по 4-м парам кабеля, тактовая частота при этом снижается с 250 до 125 МГц. Такое кодирование получило название четырехмерного 4D/PAM-5.

Работы над стандартом **10Gigabit Ethernet** начались в 2002 году. В качестве среды были определены волоконно-оптические линии связи. Тогда же комитетом IEEE 802.3 была сформирована исследовательская группа, задача которой состояла в определении возможностей для передачи 10-гигабитного трафика с использованием технологии Ethernet по витой паре с длиной линии до ста метров. Это приложение получило обозначение 10GBASE-T – широкополосная передача данных со скоростью 10 Гбит/с по витой паре. Следует отметить, что скрученные пары предлагаются в качестве дешевого решения, при больших расстояниях между станциями оптическое волокно остается вне конкуренции.

Основу функционирования оборудования в **10GBASE-T** составляет полнодуплексная передача по всем четырем парам кабеля 7-й категории. 10-гигабитный поток расщепляется на четыре потока со скоростями 2,5 Гбит/с.

В процессе передачи применяется 10-уровневая амплитудно-импульсная модуляция, при этом один передаваемый единичный элемент отображает три бита. В итоге получается скорость передачи 833, 33 Мбод/с.

При использовании оптических линий предельные длины ВОЛС составляют 220 м или 500 м, в зависимости от вида кабеля (применяется многомодовый кабель). На одномодовом кабеле предельная длина увеличена до 5000 м.

Стандартизированы разновидности 10-гигабитовых сетей на основе волоконно-оптических линий, в частности следующие:

- 10GBASE-LR– передача на расстояние до 10 км по одномодовому волокну; область использования – высокопроизводительные магистральные и корпоративные каналы;
- 10GBASE-ER – на дальности до 40 км по одномодовому волокну;
- 10GBASE-SR – передача на расстояние до 28 м по мультимодовому волокну, предполагается использовать для соединений коммутаторов друг с другом;
- 10GBASE-LX4, дальность связи до 300 м по мультимодовому волокну стандарта FDDI – для сетей в пределах одного здания.

Первый символ, стоящий после 10GBASE, обозначает длину волны, на которой осуществляется передача по оптическому кабелю: **S** (*Short Wawelength*) -850 нм; **L** (*Long Wawelength*) -1310 нм; **E**-1550 нм. Второй символ указывает способ кодирования выходного потока: **W**-поток, сформированный в кадры, совместимые с глобальными сетями SDH (SONET); **R**-без ориентации на глобальные сети; **X**-для локальных сетей.

В версии 10GBASE-X4 реализовано кодирование 8B/10B. В процессе передачи формируется 4 потока по 3,125 Гбит/с, которые передаются по одному волокну (1310 нм) с привлечением техники мультиплексирования длин волн WWDM. В случае 10GBASE-W на уровне MAC увеличена минимальная длина межкадровой паузы IPG.

В 10GBASE-LX для локальных сетей применяется логическое кодирование 64B/66B вместо 8B/10B, используемого в обычной гигабитной сети Ethernet. Передача осуществляется параллельно четырмя потоками по волоконно-оптическому кабелю на различных длинах волн. Суммарная техническая скорость составляет 12,5 Гбит/с.

Сетевой адаптер 10-гигабитовой Ethernet основывается на цифровой обработке сигналов и использовании цифровых сигнальных процессоров (рисунок 3.25). Обработка сигналов (кодирование и декодирование, модуляция и демодуляция, формирование кадров и анализ полей заголовков и др.) осуществляется цифровым сигнальным процессором. Преобразование входных аналоговых сигналов в цифровые эквиваленты выполняет аналого–цифровой преобразователь (АЦП). Обратное преобразование цифровых сиг-

налов в аналоговые осуществляет цифро-аналоговый преобразователь (ЦАП).

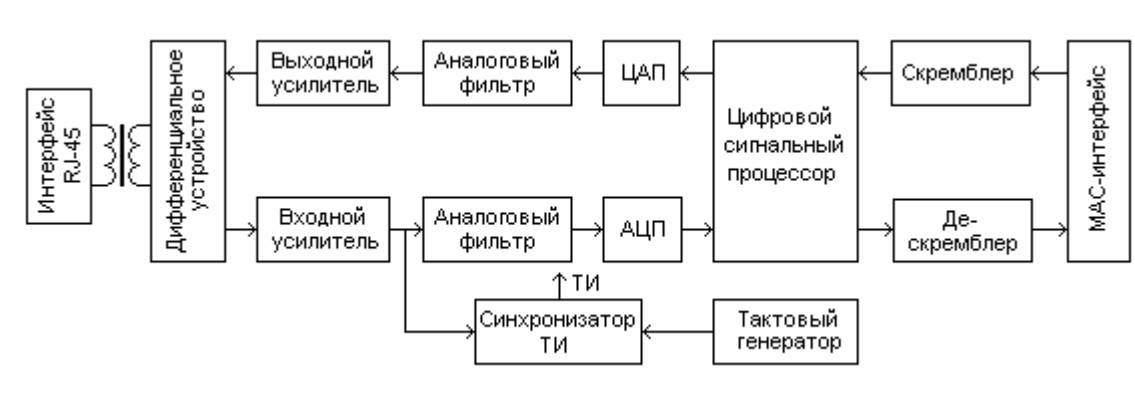


Рисунок 3.25 – Структурная схема сетевого адаптера 10G BASE-T

Усиление входных и выходных сигналов и их фильтрация выполняется соответствующими усилителями и аналоговыми фильтрами. Для улучшения условий синхронизации по тактам применяется процедура скремблирования (и соответственно, на приемной стороне – дескремблирования), увеличивающая количество изменений позиций сигналов при длительных последовательностях нулей или единиц, поступающих от источника информации. Разделение сигналов на два разнонаправленных потока осуществляется в дифференциальном устройстве, которое содержит как собственно дифференциальную систему, так и схему эхокомпенсации. Соединение сетевого адаптера со средой реализовано посредством интерфейса на основе соединителя RJ-45.

Основной областью применения 10-гигабитовой Ethernet на данном этапе развития является построение магистральных сетей. В этой области они становятся серьезным конкурентом ATM-сетей и цифровых систем передачи данных SDH.

3.4.5. Оценка производительности локальных сетей

Производительность сетей оценивается количеством кадров (пакетов), передаваемых от компьютера источника к получателю в единицу времени. Она зависит как от скорости передачи сигналов по линии связи, так и от скорости обработки кадров в коммуникационных устройствах, передающих кадры между своими портами. Скорость передачи по линиям в свою очередь зависит от используемой среды передачи и типа сети: Ethernet, Token Ring или FDDI.

Для коммуникационного оборудования наиболее напряженным режимом является обработка кадров минимальной длины. Это вызвано тем, что на обработку каждого кадра мост, коммутатор или маршрутизатор выделяет приблизительно одно и то же время, которое в основном затрачивается на просмотр таблиц продвижения кадров, формирование новых кадров и т.п. А при постоянной битовой скорости количество кадров, поступающих в коммуникационное устройство в единицу времени максимально при их минимальной длине.

Если на передачу кадра по одному или нескольким сегментам сети затрачивается $T_{\text{пк}}$ секунд, то производительность компьютерной сети $C_{\text{кс}}$ определяется известным выражением

$$C_{\text{кс}} = 1 / T_{\text{пк}} \text{ кадров/с.} \quad (3.4)$$

Рассчитаем максимальную производительность компьютерной сети Ethernet при отсутствии коллизий. Время, требуемое на передачу кадра минимальной длины с учетом преамбулы $t_{\text{пра}}$, стартового разделительного байта $t_{\text{стрб}}$ и межкадровой паузы $t_{\text{мкп}}$ (рис.3.7) равно:

$$T_{\text{пкЕ мин}} = t_{\text{пра}} + t_{\text{стрб}} + t_{\text{к мин}} + t_{\text{мкп}} = 8(7+1+64) / \text{В} + 9{,}6 \cdot 10^{-6}. \quad (3.5)$$

При битовой скоростив моноканале B=10 Мбит/с

$$T_{\text{пкЕ мин}} = 576 / 10^7 + 9{,}6 \cdot 10^{-6} = (57{,}6 + 9{,}6) \cdot 10^{-6} = 67{,}2 \text{ мкс.}$$

Отсюда максимальная пропускная способность компьютерной сети Ethernet $C_{\text{ксЕ}}$ при технической битовой скорости передачи 10 Мбит/с достигает величины $C_{\text{ксЕ}} = 1 / T_{\text{пк мин}} = 1 / (67{,}2 \cdot 10^{-6}) \approx 14\,880$ кадр/с.

При максимальном размере Ethernet-кадра длиной 1518 байтов время его передачи равно $T_{\text{пк макс}} = 8\,(8+1518) /10^7 + 9{,}6 \cdot 10^{-6} = 1230{,}4$ мкс, а пропускная способность компьютерной сети в Ethernet этом случае составляет 812 кадр/с.

В связи с передачей по сети наряду с полезной и служебной информации эффективная битовая скорость передачи данных $V_{\text{эф}}$, т.е. скорость выдачи данных получателю, всегда меньше технической скорости. Для кадров минимальной длины объем полезной информации в кадре равен 46 байтам. Эффективная битовая скорость в этом случае равна $V_{\text{эфЕ}} = 14880 \cdot 46 \cdot 8 =$ 5,476 Мбит/с, что почти в два раза ниже технической скорости, которая равна 10 Мбит/с. При передаче кадров максимальной длины эффективная ско-

рость приближается к технической, т.е. $V_{\text{эфЕ макс}} = 812 \cdot 1500 \cdot 8 = 9{,}74$ Мбит/с.

В случае обмена кадрами среднего размера с полем пользовательских данных 512 байтов пропускная способность сегмента составляет 2273 кадров в секунду, а эффективная скорость - 9,3 Мбит/с. Это не намного меньше технической скорости передачи данных в среде.

Наличие коллизий в сети приводит к заметному снижению пропускной способности сегмента. Для расчета времени передачи кадра в формулу (3.4) добавляется время обработки коллизии, которое является случайной величиной, зависящей от количества компьютеров, входящих в домен коллизий и от активности пользователей.

Время передачи кадра минимальной длительности при возникновении коллизии увеличивается на величину интервала ожидания, т.е.

$$T_{\text{пкк}} = T_{\text{пкЕмин}} + J\Delta t_{\text{ож}} , \qquad (3.6)$$

где $\Delta t_{\text{ож}}$ – минимальный интервал ожидания (см. 3.3); J –количество повторений коллизий. Предположим для упрощения, что интервал $\Delta t_{\text{ож}}$ постоянный и равен 51,2 мкс.

Пусть после обнаружения коллизии имеется k станций, готовых к передаче, а значение вероятности передачи каждой из станций постоянно и равно p. Тогда вероятность того, что какой то станции удастся захватить канал, определяется выражением

$$P_{\text{зк}} = k\, p(1- p)^{k-1} . \qquad (3.7)$$

Можно показать, что максимальная вероятность захвата канала $P_{\text{зк}}$ будет при равных вероятностях $p=(1/ k),$ а при больших k вероятность $P_{\text{зк}}$ стремится к величине $1/e$.

Вероятность того, что интервал борьбы за канал будет состоять ровно из J минимальных интервалов, определяется как $p(1- p)^{J-1}$, а среднее число повторений $J = 1/ P_{\text{зк}} = e$. Тогда общая продолжительность передачи пакета минимальной длины при возникновении коллизий $T_{\text{пкк}} = T_{\text{пкЕмин}} + 51{,}2\ e = 67{,}2 + 2{,}7 \cdot 51{,}2 = 205{,}4$ мкс. Производительность сети Ethernet снижается при этом до величины $C_{\text{ксЕк}} = 1/ T_{\text{пкк}} = 4868$ кадр/с, а эффективная скорость – до 1,8 Мбит/с.

В худшем случае, когда количество коллизий, возникающих подряд, возрастает до 10, а общий интервал ожидания увеличивается до 52,4 мс (см. 3.3), пропускная способность сети падает до 19 кадр/с, т.е. сеть практически блокируется.

В отличие от сетей Ethernet со случайным доступом в сетях Token Ring гарантируется стабильность пропускной способности благодаря отсутствию коллизий. Время передачи кадра в такой сети $T_{пкTR}$ состоит из времени, затрачиваемое на передачу служебных символов $t_{слс}$ и времени передачи пользовательской информации кадра $t_{пик}$.

Максимальная пропускная способность сети достигает в случае, если среда передачи не простаивает, т.е. если время передачи кадра равно времени удержания кольца, величина которого регламентирована стандартом и составляет 10 мс.

Время передачи кадра при битовой скорости 4 Мбит/с и максимальной длине пользовательской информации в кадре равной 4500 байтов не превышает интервал удержания и составляет (см. рисунок 3.17):

$$T_{пкTR\ макс} = 8\ [(1+1+1+6+6+4+1+1) + 4500]\ /(4 \cdot 10^6) = 9\ мс.$$

Отсюда минимальная пропускная способность сети Token Ring составит $C_{ксTR\ мин} = 1\ /\ T_{пкTR\ макс} = 1\ /\ (9 \cdot 10^{-3}) \approx 111$ кадр/с. Эффективная минимальная битовая скорость в сети при этом практически равна номинальной:

$$V_{эфTR} = 111 \cdot 4500 \cdot 8 = 3{,}99\ Мбит/с.$$

Для сети Token Ring со скоростью передачи 16 Мбит/с и с ранним освобождением маркера минимальная пропускная способность равна

$$C_{ксTR16\ мин} = 1\ /\ \{[8\ (21+16000)]\ /\ (16 \cdot 10^6)\} = 124{,}8\ кадр/с.$$

Эффективная битовая скорость в сети в этом случае составит $V_{эфTR16} = 124{,}8 \cdot 16000 \cdot 8 = 15{,}97$ Мбит/с, что практически равно номинальной.

Из приведенных расчетов видно, то эффективность сети Token Ring-16 выше, чем Ethernet за счет более длинных кадров. Другим преимуществом Token Ring является предсказуемость задержки в сети и независимость пропускной способности от активности пользователей сети, так как в ней отсутствуют коллизии и обеспечивается регулярность доступа.

Однако относительно высокая стоимость сетевого оборудования Token Ring, более поздний выход ее на рынок и появление высокоскоростной Ethernet привело к тому, что сеть Token Ring находит ограниченное применение, преимущественно в системах управления, где предъявляются жесткие требования к задержкам доступа к среде.

3.5. Оборудование локальных сетей

Развитие корпоративных и объединенных сетей тесно связано с разработкой специальных средств объединения (*комплексирования*) и межсетевых протоколов. Комплексирование может осуществляться на различных уровнях эталонной модели ВОС. Оборудование локальных сетей различается по занимаемому уровню эталонной модели и выполняемым функциям. В перечень такого оборудования входят повторители и концентраторы, сетевые мосты, коммутаторы, маршрутизаторы и шлюзы.

3.5.1. Повторители и концентраторы

Повторители (*Repiters*) объединяют сети на **физическом уровне**, осуществляя согласование электрических параметров сопрягаемых сетей, усиление и регенерацию сигналов. Эти устройства вносят задержку на один тактовый интервал, поддерживая побитовый синхронизм входного и выходного потоков. Повторители используются для сопряжения кабельных сегментов как с одинаковыми, так и с различными характеристиками физической среды. В рамках однородной физической среды повторители применяются с целью увеличения диаметра сети и количества подключаемых абонентов.

Концентратор или **хаб** (англ. *Hub*) представляет собой многопортовый Ethernet-повторитель, служащий в качестве центральной точки сети со звездообразной технологией, в которой концентрируются (соединяются) кабели рабочих станций. Как и повторитель, концентратор работает на первом, физическом уровне эталонной модели.

3.5.2. Сетевые мосты

Мост (Bridge) - это устройство, которое обеспечивает взаимосвязь двух (реже нескольких) локальных сетей посредством передачи кадров из одной сети в другую, предварительно осуществляя их промежуточную буферизацию. Мост, в отличие от повторителя, не поддерживает побитовый синхронизм в обеих объединяемых сетях. Он выступает по отношению к каждой из сетей как конечный узел. Мост принимает кадр, буферизует его, анализирует адрес назначения кадра, и только в том случае, когда адресуемый узел действительно принадлежит другой сети, он передает кадр в эту сеть.

Мосты функционируют на канальном уровне и применяются для решения следующих задач в локальных сетях:

- увеличения максимального количества компьютеров в сети;
- разделения перегруженной сети на отдельные сегменты с уменьшенным трафиком, в результате чего каждая подсеть работает более эффективно;
- соединения разнородных физических сред (витая пара – коаксиальный кабель).

Принципы работы моста. Мосты допускают использование в сети всех протоколов сетевого уровня, не отличая при этом один протокол от другого по той причине, что они работают на канальном уровне и им не доступна информация, содержащаяся на более высоких уровнях этой модели. Мосты работают на подуровне управления доступом к среде. Поэтому в их функции входит «прослушивание» всего трафика сегментов, подключенных к их портам; проверка адресов источника и получателя каждого пакета; построение таблицы адресации; передача пакетов.

Передача фреймов осуществляется следующим образом. Если получатель не указан в таблице адресации, мост передает кадр во все сегменты (волновая маршрутизация), а если адресат содержится в таблице, то мост передает кадр в указанный сегмент.

Работа моста основана на принципе, согласно которому каждый узел сети имеет собственный MAC-адрес. Мост передает блоки, исходя из физического адреса узла назначения. В начале работы таблица адресации моста пуста и в ней находится только база данных пользователей сети. В процессе прохождения кадров через узел адреса источников копируется в таблицу. Таким образом, мост изучает расположение компьютеров в сегментах сети.

Принимая блок, мост ищет MAC-адрес источника в своей таблице адресации. В случае если адрес источника не найден, он добавляется в базу данных таблицы. При наличии адреса получателя в таблице адресации, если адресат располагается в одном сегменте с источником, то блок отбрасывается. Такая фильтрация уменьшает сетевой трафик и изолирует сегменты сети. В противном случае мост передает кадр адресату через соответствующий порт. В случае отсутствия адреса получателя в таблице, мост передает кадр во все свои выходные порты, за исключением того номера порта, с которого блок был принят.

Большая компьютерная сеть не ограничивается одним мостом. Если между узлами имеется несколько путей, то определяется самый короткий из них. Мост может работать как автономное устройство (внешний мост), так и на сервере (внутренний), если сетевая ОС допускает установку на сервере нескольких сетевых плат.

С электрической точки зрения мост представляет собой многопроцес-

сорное коммуникационное устройство, блоки которого объединены внутренней системной шиной (рисунок 3.26). Блоки MAC осуществляют доступ к среде и выполняют функции портов моста. Они являются высокоскоростными процессорами ввода/вывода. На схеме показано два порта. В общем случае их может быть и больше.

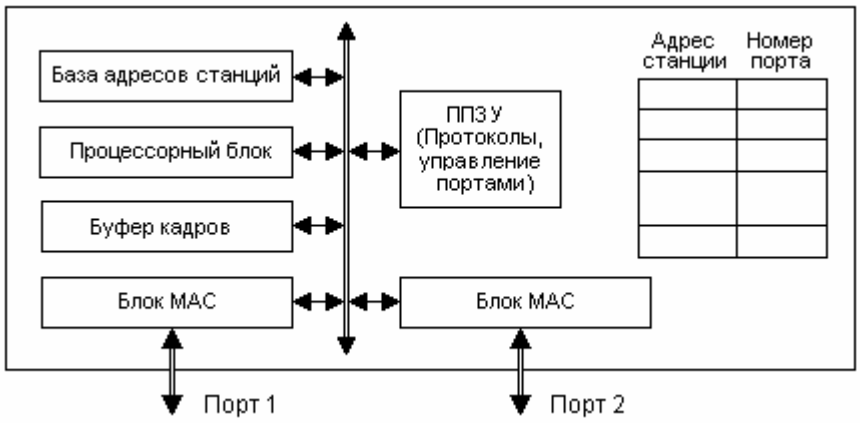


Рисунок 3.26 – Структура сетевого моста с таблицей адресации

Как и повторитель, мост не проводит никакой обработки кадра по изменению его содержания. Он осуществляет только регенерацию сигналов данных и контроль кадров на наличие ошибок (собственно ошибок кадра и выхода размера кадров за допустимые пределы). При обнаружении ошибок кадр не транслируется в сегмент кабеля, к которому подключен адресат. Такая функция моста защищает сегмент получателя от поврежденных кадров, так как они требуют затрат ресурсов сегмента, а устройство-получатель в любом случае отбросит поврежденный кадр.

3.5.3. Сетевые коммутаторы

Коммутатор (*Switch*) представляет собой мультипроцессорный мост, способный независимо транслировать кадры между всеми парами своих портов. Оба устройства, мост и коммутатор, передают кадры на основании одного и того же алгоритма, называемого **алгоритмом прозрачного моста**, изложенного в стандарте IEEE 802.1D. Понятие "прозрачный мост" означает, что мосты и коммутаторы в процессе функционирования не учитывают существование в сети адаптеров, повторителей и концентраторов. С другой стороны и перечисленные сетевые устройства "не замечают" присутствия в сети мостов и коммутаторов. Отличие многопроцессорного коммутатора от моста состоит в установке в коммутаторе коммутационной матрицы, связы-

вающей между собой порты устройства. Существуют два способа передачи кадров через коммутатор.

1. **Запоминание кадра и последующая передача** (*Store-and-Forward*). При этом анализируется адрес источника и получателя, возможно использование фильтрации кадров по определенным признакам, а также контроль наличия ошибок. Недостаток такого способа – задержка кадров, которая растет с увеличением их размеров, преимущество – высокая вероятность обнаружения ошибок.

2. **Коммутация "на лету"** (*On The Fly*). Данные передаются в выходной порт после считывания адреса назначения, не дожидаясь окончания приема всего кадра. При таком способе задержка кадров существенно меньше, но практически исключена возможность контролировать ошибки. Следует заметить, что при передаче пакетов из низкоскоростного порта в высокоскоростной, например, из порта 10 Мбит/с в порт 100 Мбит/с, коммутацию "на лету" использовать вообще невозможно, так как при организации соединения между портами с разной скоростью требуется буферизация пакетов.

Структурная схема коммутатора ЛКС изображена на рисунке 3.27.

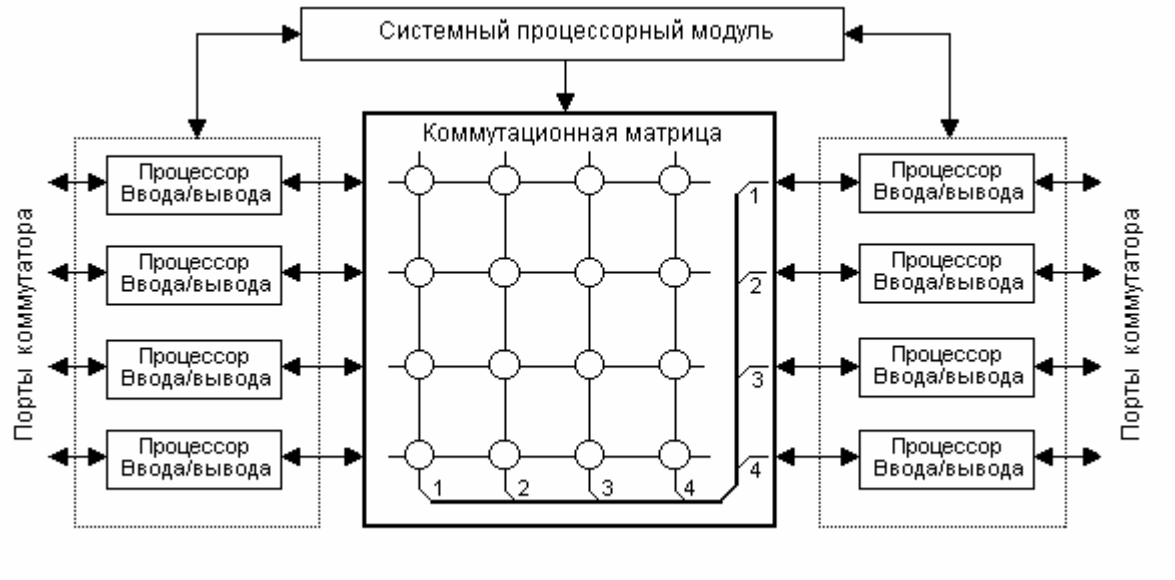


Рисунок 3.27 – Структура сетевого коммутатора

Каждый из 8 портов обслуживается одним процессором ввода/вывода (линейный процессор). Передатчик и приемник процессора работают независимо друг от друга и могут реализовать дуплексный режим обмена. Системный процессорный модуль поддерживает общую адресную таблицу коммутатора и управляет коммутационной матрицей. Этот модуль работает в многозадачном режиме, параллельно обслуживая запросы всех процессоров

ввода/вывода. Коммутационная матрица функционирует по принципу коммутации каналов, рассмотренном в первом разделе.

При поступлении кадра в какой-либо порт линейный процессор буферизирует несколько первых байтов кадра, чтобы прочитать адрес назначения. После дешифрации адреса процессор принимает решение о передаче пакета, не дожидаясь прихода последующих байтов кадра. Для этого он просматривает свой собственный *кэш адресной таблицы*, а если не находит там нужного адреса, обращается к системному процессорному модулю. Системный модуль производит просмотр общей адресной таблицы и возвращает линейному процессору найденную строку, которую тот буферизирует в своем кэше до следующего использования. Во время этих действий продолжается процесс буферизации поступающего кадра. Если кадр нужно **отфильтровать** (не пропустить), процессор просто прекращает запись, очищает буфер и ждет поступления нового кадра.

В случае необходимости передачи кадра в другой порт линейный процессор пытается с помощью коммутационной матрицы соединиться с выходным портом назначения. Эта процедура возможна, если порт назначения свободен. В случае занятости порта кадр буферизируется полностью, а процессор ожидает освобождения требуемого выходного порта. Способ коммутации кадра "на лету" реализует конвейерную обработку кадров, что существенно повышает быстродействие коммутатора сети.

Несмотря на малую стоимость и высокую производительность, коммутаторы на основе коммутационной матрицы слишком примитивны для эффективной трансляции между низкоскоростными интерфейсами Ethernet или Token Ring и высокоскоростными портами сетей ATM и FDDI.

Другим вариантом реализации сетевого коммутатора является использование вместо коммутационной матрицы разделяемой системной памяти (рисунок 3.28). Принцип соединения портов в таком коммутаторе аналогичен принципу коммутации в системах с временным разделением каналов. Прием запросов на установление соединения и управление буферами разделяемой памяти осуществляет системный процессорный модуль.

В зависимости от соотношения скоростей ввода и вывода различают коммутаторы с *симметричной* и *асимметричной* коммутацией. Все порты симметричного коммутатора имеют одинаковую пропускную способность, например 10 или 100 Мбит/с. В асимметричных коммутаторах данные могут передаваться как между портами с равными скоростями передачи, так и с разными (10/100 Мбит/с). В таких коммутаторах производится мультиплексирование/демультиплексирование нескольких низкоскоростных потоков. В случае необходимости передачи пакета, поступившего через порт с более высокой скоростью на порт с меньшей пропускной способностью, например 100/10, производится буферизация поступившего пакета и последующая вы-

дача его с меньшей скоростью.

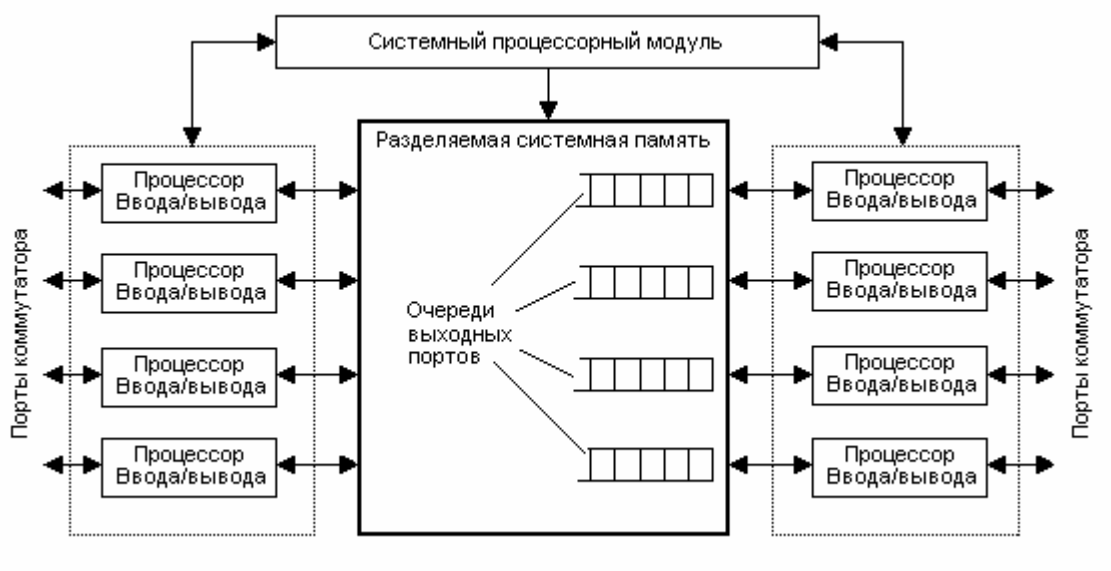


Рисунок 3.28 – Сетевой коммутатор с разделяемой памятью

Высокая производительность коммутаторов и возможность параллельной передачи кадров привели к тому, что они практически полностью вытеснили однопроцессорные мосты. Основным параметром коммутатора является его **производительность**. Для того чтобы охарактеризовать ее используются несколько параметров:

- скорость передачи между портами, измеряемая в пакетах/с;
- общая пропускная способность, характеризующая максимальную суммарную скорость, с которой пакеты могут передаваться адресатам через коммутатор;
- задержка – промежуток времени между получением пакета от отправителя и передачей его получателю. Обычно задержку измеряют относительно первого бита пакета.

В зависимости от вида решаемых задач коммутаторы подразделяют на три класса: коммутаторы уровня доступа, коммутаторы для рабочих групп и магистральные коммутаторы.

Уровень доступа является ближайшим к пользователю уровнем и предоставляет ему доступ к ресурсам сети. Работающие на этом уровне коммутаторы должны поддерживать подключение отдельных компьютеров к объединенной сети.

Коммутаторы для рабочих групп обеспечивают связь между группой компьютеров одной сети. Они являются точкой концентрации для нескольких коммутаторов уровня доступа и должны справляться с большими

объемами передаваемых данных. В зависимости от способа реализации, на уровне доступа могут выполняться следующие функции:

- обеспечение маршрутизации, качества обслуживания и безопасности сети;
- переход от одной технологии к другой (например, от 10BASE-T к 100BASE-TX или от 100BASE-TX к 1000BASE-T) и т.п.;
- объединение пропускных способностей низкоскоростных каналов доступа в высокоскоростные магистральные каналы.

Выпускаемые промышленностью современные коммутаторы для рабочих групп представляют собой настраиваемые коммутаторы, которые обеспечивают коммутируемые каналы 10/100 Мбит/с и 10/100/1000Мбит/с и поддерживают до 24 пользователей и 2 порта Gigabit Ethernet для серверов. Коммутаторы для рабочих групп позволяют полностью сохранить сетевую инфраструктуру со стороны клиентов, включая программы, сетевые адаптеры, кабели.

Магистральные коммутаторы служат для передачи данных между отдельными сегментами компьютерных сетей. Они выполняются преимущественно в виде модульных устройств и предназначены для работы в сетях операторов связи. Основное их отличие – способность осуществлять коммутацию на сверхвысоких скоростях (до 10 Гбит/с). Магистральные коммутаторы обычно имеют в своем составе от 8 до 16 портов со скоростью 10, 100 или 1000 Мбит/с, используемых для подключения сегментов рабочих групп, и несколько магистральных портов со скоростью передачи до 10 Гбит/с. Промышленностью выпускаются коммутаторы в исполнении для одной определенной среды передачи с оптоволоконными портами 1000BASE-SX, так и в универсальном исполнении с портами 100/1000BASE-T для медного кабеля и несколькими портами для оптоволоконных линий связи.

В последнее время появились комбинированные коммутаторы, способные кроме коммутации на канальном уровне выполнять функции маршрутизаторов, а в ряде вариантов и функции транспортного уровня. Такие устройства получили название соответственно "коммутаторы 3-го" или "4-го уровня". Работа комбинированного коммутатора на втором, третьем или четвертом уровне зависит от его настройки.

Коммутаторы 3-го уровня принимают решения на основе информации сетевого уровня, а не на основе MAC-адресов. Основная цель коммутации такого типа – получить скорость коммутации 2-го уровня и масштабируемость маршрутизации. Обработку пакетов коммутатор 3-го уровня выполняет таким же образом, как и маршрутизатор:

- на основе информации 3-го уровня (сетевых адресов) определяет путь к месту назначения пакета;

- проверяет целостность заголовка 3-уровня путем вычисления контрольной суммы;
- контролирует время жизни пакета;
- обновляет статистику в Информационной базе управления MIB (*Management Information Base*);
- обеспечивает необходимое качество сервиса (QoS) для мультимедийных приложений, чувствительных к задержкам передачи.

Основное отличие между маршрутизаторами и коммутаторами третьего уровня заключается в том, что в основе коммутации 3-го уровня лежит аппаратная реализация. В маршрутизаторах общего назначения коммутация пакетов обычно выполняется программным образом. В связи с тем, что коммутаторы 3-го уровня обычно быстрее и дешевле маршрутизаторов, то их использование в локальных сетях предпочтительнее.

Под **коммутацией 4-го уровня** понимают возможность коммутатора принимать решение о передаче пакета, основываясь не только на MAC- или IP-адресах, но и на параметрах четвертого уровня, таких как номер порта TCP/UDP. При выполнении функции 4-го уровня, коммутатор читает поля TCP- и UDP-заголовка и определяют, какой тип информации передается в этом пакете. Администратор сети также может запрограммировать коммутатор обрабатывать трафик в соответствии с приоритетом приложений. Эта функция позволяет задавать качество сервиса для конечных пользователей. При указании качества обслуживания коммутатор 4-го уровня будет выделять, например, трафику видеоконференции, большую полосу пропускания по сравнению с почтовым сообщением или пакетом FTP.

Для настройки коммутатора необходимо установить физическое соединение между коммутатором и рабочей станцией. Существуют два типа кабельного соединения, используемых для управления коммутатором. Первый тип – через консольный порт (интерфейс RS-232C, если он имеется у устройства), второй – через порт Ethernet (по протоколу Telnet или через Web-интерфейс). Консольный порт используется для первоначальной конфигурации коммутатора и обычно не требует настройки. Для того чтобы получить доступ к коммутатору через порт Ethernet, устройству необходимо назначить IP-адрес.

3.5.4. Маршрутизаторы и шлюзы

Маршрутизаторы (*Routers*) работают на сетевом уровне. Это означает, что они могут переадресовывать и маршрутизировать *пакеты* через множество сетей, обмениваясь служебной информацией, зависящей от протокола между различными сетями. Электрическая схема маршрутизатора отлича-

ется от схемы моста наличием не менее трех портов и алгоритмом перенаправления кадров между портами. Маршрутизаторы могут также выполнять функции мостов, в частности *фильтровать и изолировать трафик*, *соединять сегменты сети*. Однако маршрутизаторам доступен больший объем управляющей информации, чем мостам, и они используют ее для оптимизации доставки пакетов.

Таблица маршрутизации, размещаемая в маршрутизаторе, содержит **сетевые адреса**. Для каждого протокола, используемого в сети, строится своя таблица, которая помогает маршрутизатору определить адреса назначения для поступающих данных. Она включает следующую информацию:

- все известные сетевые адреса;
- способы связи с другими сетями;
- возможные пути между маршрутизаторами;
- стоимость передачи данных по этим путям.

Маршрутизатор выбирает наилучший маршрут для пакетов, сравнивая стоимость и доступность различных путей. Следует заметить, что таблица маршрутизации существует и для мостов, однако она содержит адреса подуровня управления доступом к среде, тогда как таблица маршрутизатора содержит номера сетей. Поэтому термин «таблица маршрутизации» для мостов и маршрутизаторов имеет разный смысл. Маршрутизаторы взаимодействуют только с другими маршрутизаторами, а не с рабочими станциями.

В связи с тем, что маршрутизаторы выполняют более сложную обработку пакета, они обладают меньшим быстродействием по сравнению с мостами. Когда пакеты передаются от одного маршрутизатора к другому, адреса источника и получателя канального уровня маршрутизатором отсекаются, а затем создаются заново. Это позволяет передавать сообщения между разнотипными сетями, например TCP/IP – Ethernet или TCP/IP –Token Ring.

Воспринимая только адресованные сетевые пакеты, маршрутизаторы препятствуют проникновению в сеть некорректных пакетов. Они *не пропускают* также *широковещательные сообщения*. Все это уменьшает межсетевой трафик и повышает эффективность сетей.

Маршрутизаторы способны соединять сегменты с абсолютно разными схемами упаковки данных в пакеты и доступа к среде, им часто доступны несколько путей. При отказе одного из маршрутизаторов, данные будут передаваться по другим. Используя сведения о загруженности участков сети и о стоимости пути, маршрутизатор выбирает оптимальный путь. Маршрутизаторы подразделяются на два основных вида:

- *статические*, когда администратор вручную создает и конфигурирует таблицу маршрутизации, а также указывает каждый маршрут;
- *динамические,* в которых маршруты определяются автоматически на основе анализа информации от соседних маршрутизаторов.

Маршрутизаторы наилучшим образом подходят для соединения удаленных сетей, так как передают по каналу связи только те данные, которые предназначены для этих сетей.

Маршрутизаторы обладают собственной операционной системой. Управлять ими можно путем задания инструкций в командной строке данной операционной системы с помощью любой терминальной программы через последовательный порт компьютера, связанного с консольным портом маршрутизатора. Конфигурация маршритизатора возможна и дистанционным способом с помощью программы *Telnet* на IP-адрес любого из его интерфейсов.

Мосты-маршрутизаторы (*Brouter*s) соединили лучше свойства моста и маршрутизатора. Для одних протоколов они могут действовать как мост, а для других – как маршрутизатор.

С маршрутизаторами работают не все протоколы. Поэтому если есть необходимость применить в сети маршрутизаторы, следует убедиться, что в сети не используются немаршрутизируемые протоколы. К маршрутизируемым протоколам относятся: **IP, IPX XNS, DECnet**; к немаршрутизируемым – протокол **NetBEUI**.

Шлюзы (*Gateways*). Этим термином раньше называли маршрутизирующее устройство. В настоящее время узлы, выполняющие эти функции, называются маршрутизаторами. Под шлюзами (в узком смысле слова) понимают устройства, обеспечивающие взаимодействие между сетями различного типа, в частности, шлюзы связывают две системы, которые используют разные коммуникационные протоколы, структуры и форматы данных, языки, архитектуры.

Шлюзы создаются для выполнения определенного типа задач, т.е. для конкретного типа преобразования данных, например, шлюз Windows NT Server to SNA Gateway. Шлюз принимает данные из одной среды, удаляет старый протокольный стек и переупаковывает их в протокольный стек системы назначения. Некоторые шлюзы используют все семь уровней модели ВОС, однако, чаще шлюзы выполняют преобразования на *прикладном* уровне. Обычно функции шлюзов в сети реализуют выделенные серверы.

Шлюзы осуществляют преобразования протоколов данных. Однако они имеют некоторые ограничения, а именно:

- предназначены для выполнения одной конкретной задачи;
- часто работают с низкой производительностью;
- имеют достаточно высокую стоимость.

Таким образом, шлюзы следует использовать, если необходимо установить связь между различными сетями.

3.6. Сегментация локальных компьютерных сетей

3.6.1. Домен коллизий и необходимость сегментации сетей

В сетях со случайным доступом типа CSMA/CD при попытке одновременного начала передачи данных двумя или несколькими рабочими станциями возникают коллизии (см. пп. 3.1.2). В расширенных сетях Ethernet, построенных на основе повторителей или концентраторов, сигналы коллизий проходят беспрепятственно через такие устройства и распространяются на всю сеть. Участок сети, состоящий из нескольких сегментов, в пределах которого распространяется коллизия, называется **доменом коллизий**. Для исключения влияния скорости на расчет параметров домена коллизий его размер целесообразно измерять не в единицах длины (м), а в бит-интервалах, то есть, в интервалах времени, необходимого для передачи одного бита. Для сети Fast Ethernet эта величина равна 10 нс, а для сети Ethernet - 100 нс. В пп. 3.1.2 показано, что минимальная длина кадра в сети CSMA/CD, измеряемая числом битовых интервалов равняется $2Bd/v$, где B – скорость передачи сигналов, d – диаметр сети и v – скорость распространения сигналов в среде передачи. Время, затрачиваемое на распространения сигнала по сегменту в прямом и обратном направлении, равное $2d/v$, называется временем двойного оборота **PDV** (*Path Delay Value*).

Расчет домена коллизий сводится к определению **времени двойного оборота** PDV и **времени уменьшения межкадрового интервала** IPG (*Inter Packet Gap*). Сеть Ethernet будет устойчиво функционировать только в том случае, если эти параметры не превышают допустимые значения для данной скорости передачи. При расчете необходимо учитывать задержки распространения сигналов, которые вносят повторители (концентраторы) и среда распространения, а также уменьшение интервала времени межпакетной паузы в процессе прохождения кадров через цепочку повторителей.

Время двойного оборота сети определяется по формуле

$$PDV = \sum_{k=1}^{KH} T_{Hi} + \sum_{j=1}^{KS} (l_{CSj} T_{CSj}) \ ,$$

где T_{Hi} – базовая задержка в i-м концентраторе (хабе) в бит-интервалах; l_{CSj} – длина j-го сегмента кабеля в метрах; T_{CSj} – удвоенный удельный интервал задержки в j-м кабельном сегменте в бит-интервалах на метр; KH – максимальное количество концентраторов в сети; KS – максимальное количество кабельных сегментов в сети. Термин «базовая задержка» включает задерж-

ку, вносимую приемником, блоком регенерации и передатчиком концентратора (повторителя), находящегося между кабельными сегментами сети.

Для расширенных сетей, состоящих из нескольких кабельных сегментов, стандартом IEEE 802.3 установлено правило **5-4-3**, в соответствии с которым в такой сети допускается не более 5 сегментов и использование не более 4-х повторителей (концентраторов). Для надежного распознавания коллизий в сетях Ethernet стандартом регламентируется время двойного оборота PDV, которое не должно превышать 575 битовых интервалов. В процессе прохождения кадра через несколько повторителей межкадровый интервал может уменьшиться настолько, что сетевым адаптерам в последнем сегменте не хватит времени на обработку предыдущего кадра, в результате чего он может быть потерян. Номинальное значение межкадрового интервала, установленное IEEE 802.3, равно 9,6 мкс, что составляет 96 битовых интервалов (см. рисунок 3.7). Стандартом допускается уменьшение этого интервала до граничной величины, равной 47 битовых интервалов. Отсюда следует, что допустимое суммарное время уменьшения IPG в сегментах сети может быть не более 96-47 = **49 битовых интервалов**.

В стандарте IEEE 802.3 приведены максимальные значения задержек распространения сигналов и сокращения межкадрового интервала в передающем (прд) и промежуточном (промежуточ.) сегментах (таблица 3.3) для сети Ethernet со скоростью передачи 10 Мбит/с.

Таблица 3.3 – Типовые значения PDV и межпакетной паузы IPG

Тип сегмента	Базовая задержка сегмента в битовых интервалах (б.и.)				Уменьшение IPG сегмента, б.и. прд/промежуточ.	Максимальная длина сегмента, м
	левого	промежуточного	правого	среды на 1 м		
10BASE-5	11,8	46,5	169,5	0,0866	16 / 11	500
10BASE-2	11,8	46,5	169,5	0,1026	16 / 11	185
10BASE-T	15,3	42,0	165,0	0,1130	10,5 / 8	100
10BASE-F	12,3	33,5	156,5	0,1000	10,5 / 8	2000

Левым по терминологии стандарта называется сегмент, в котором начинается путь сигнала от компьютера-источника, а правым – самый удаленный конечный сегмент сети, в котором может возникнуть коллизия. Сегменты, располагаемые на пути сигнала между левым и правым сегментами относят к промежуточным.

Особенностью таблицы является то, что в ней приведены значения удвоенной удельной задержки среды, чем учитывается распространение сигнала данных в прямом направлении и сигнала коллизии в обратном.

Для расчета сегментов сетей Fast Ethernet комитет 802.3 также дает исходные данные для вычисления времени двойного оборота. Задержка сегмента определяется типом сетевого адаптера и типом и категорией соединительного кабеля (таблица 3.4). Все сегменты предполагаются однородными без разделения на правый, левый и промежуточный.

Таблица 3.4 – Максимальные задержки в сегментах и адаптерах сети Fast Ethernet на 100 Мбит/с

Тип кабеля или адаптера	Максимальная длина сегмента, м	Удвоенная задержка кабеля в б.и.		Максимальная задержка адаптеров при двойном обороте, б.и.
		1 метр	Максимальной длины	
UTP кат. 3	100	1,14	114	–
UTP кат. 4	100	1,14	114	–
UTP кат. 5	100	1,112	111,2	–
STP	100	1,112	111,2	–
Оптический	412	1,0	412	–
UTP кат. 3	100	1,14	114	–
Два адаптера TX/FX	–	–	–	100
Два адаптера T4	–	–	–	138
Один TX/FX и один T4	–	–	–	127

В связи с тем, что в задержки, вносимые сетевыми адаптерами, включены преамбулы кадров, время двойного оборота при расчете конфигурации следует сравнивать со временем передачи кадра минимальной длины без преамбулы, т.е. с величиной 512 битовых интервалов. Если расчетное значение PDV оказывается меньше 512, то сеть будет функционировать корректно. Комитет 802.3 рекомендует для устойчивой работы оставлять запас в 4 битовых интервалов, т.е. 516 б.и.

Увеличение количества компьютеров, подключаемых к одному сегменту сети Ethernet, привдит к возникновению коллизий и снижению ее результирующей пропускной способности. Одним из способов повышения производительности локальных сетей является их **сегментация**. Суть ее состоит в том, что локальную сеть Ethernet разделяют на участки меньшего размера, которые называют сегментами. Сегментация позволяет уменьшить число пользователей на один сегмент, снизить объем широковещательного трафика и тем самим повысить производительность сети в целом.

Каждый сегмент большой сети Ethernet также использует метод доступа CSMA/CD, но функционирует как отдельная независимая **подсеть**. По разделяемой среде сегмента циркулирует трафик компьютеров, подключенных только к данной среде. Благодаря этому пропускная способность среды делится между компьютерами, которые непосредственно соединены с ней. Обычно в подсеть включают компьютеры, выполняющие однотипные задачи (бухгалтерия, служба главного механика и т.п.), или входящие в одну административную единицу (отдел, лаборатория). Такую совокупность компьютеров называют **рабочей группой**. Для разделения сегментов применяют мосты, маршрутизаторы или сетевые коммутаторы. На рисунке 3.29 показана структура сегментированной сети, реализованная различными сетевыми устройствами.

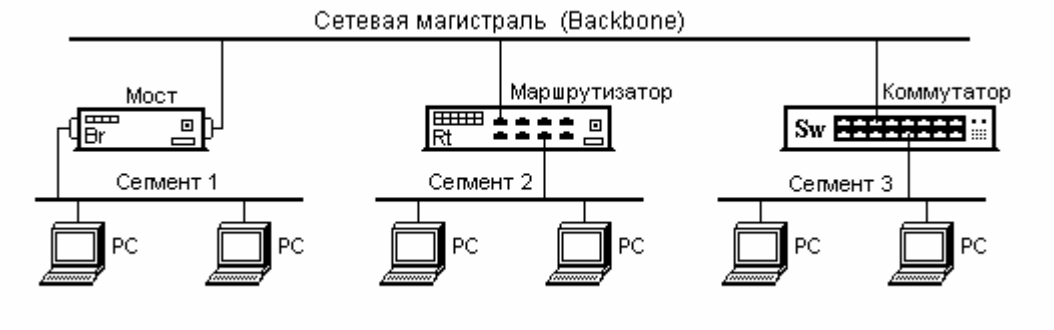


Рисунок 3.29 – Разделение локальной сети на сегменты

Вся сеть состоит из трех сегментов. Они подключены к общей магистральной среде, соединяющей все три подсети в одну сеть, через мост, маршрутизатор или коммутатор, которые передают кадры в магистральную линию только в случае, если адрес получателя находится за пределами соответствующего сегмента. Таким образом, трафик одного сегмента не загружает сегменты других подсетей.

3.6.2. Сегментация с помощью мостов и коммутаторов

Мост (*Bridge*), как и повторитель, может соединять сегменты или компьютерные сети рабочих групп. Однако в отличие от повторителя, мост также служит для разбиения сети на отдельные сегменты – подсети, что позволяет изолировать трафик или отдельные сетевые проблемы. Например, если трафик одного-двух компьютеров или некоторого отдела "затопляет" сеть пакетами, уменьшая ее производительность в целом, мост изолирует эти компьютеры или отдел от другой части сети.

В связи с тем, что мост работает на канальном уровне модели ВОС

(OSI), он анализирует только заголовок кадра, в котором указываются MAC-адреса источника и отправителя. При соединении сегментов сети с помощью моста осуществляется изоляция доменов коллизий, т.е. коллизии не распространяются между смежными сегментами. Под **доменом коллизий**, как уже упоминалось выше, понимают участок сети, в общем случае состоящий из нескольких сегментов, в пределах которого распространяется коллизия. На рисунке 3.30 показана локальная сеть, состоящая из четырех сегментов Ethernet, объединенных с помощью трех мостов. Благодаря этому в сети образуются четыре независимые домена коллизий (они на рисунке выделены пунктирной линией). Мосты *отфильтровывают* (не пропускают в последующие сегменты сети) кадры отправителя, если получатель находится в одном сегменте с отправителем. Исключением являются широковещательные кадры и кадры групповой рассылки. При получении кадра широковещательного сообщения мост передает его на все выходные порты, за исключением того направления (сегмента) откуда поступило широковещательное сообщение.

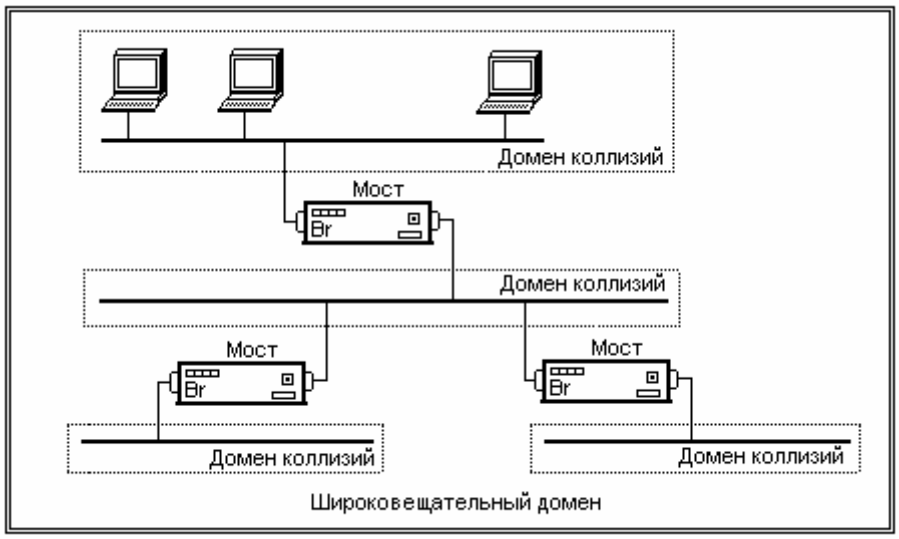


Рисунок 3.30 – Границы распространения коллизий и широковещательных сообщений

Локальная сеть Ethernet, применяющая для сегментации мосты, обеспечивает большую пропускную способность в расчете на одного пользователя, поскольку к одному сегменту подключена только часть компьютеров всей сети. Мосты увеличивают задержку в сети на 10–30%. Это связано с дополнительными затратами времени на буферирование кадров, анализ адресов пунктов назначений и поиск в таблице номеров соответствующих исходящих портов.

Коммутатор (*Switch*) представляет собой мультипроцессорный мост, способный независимо транслировать кадры между всеми парами своих портов. Благодаря этому коммутаторы, разделяя локальную сеть на подсети, делят единый коллизионный домен на отдельные поддомены, свободные от коллизий. Коммутатор создает соединение между своими портами по принципу "точка-точка". Поэтому компьютеры, подключенные к этим портам, имеют в своем распоряжении пропускную способность (10 или 100 Мбит/с), которую способны обеспечить соответствующие порты коммутатора.

Большинство современных коммутаторов, независимо от производителя, поддерживают несколько дополнительных возможностей, отвечающих общепринятым стандартам. Среди них к самым распространенным и наиболее используемым относятся следующие:

- реализация технологии виртуальных сетей – *VLAN;*
- поддержка протокола *Spanning Tree* IEEE 802.1d и *Rapid Spanning Tree* IEEE 802.1w;
- объединение каналов Ethernet в единый магистральный поток;
- поддержка SNMP-управления потоком данных;
- обеспечение функции безопасности *Port Security*, или привязка MAC-адреса к определенному порту и др.

3.6.3. Сегментация на основе маршрутизаторов

Маршрутизаторы способны соединять сегменты с абсолютно разными схемами упаковки данных в пакеты и доступа к среде, им часто доступны несколько путей. При отказе одного из маршрутизаторов или части его портов, данные будут передаваться по другим маршрутизаторам. Используя сведения о загруженности участков сети и о стоимости пути, маршрутизатор выбирает оптимальный путь.

Маршрутизаторы работают на третьем уровне эталонной модели и исполняют больше функций, по сравнению с мостами. Они, подобно мостам, дают возможность расширить сеть и позволяют ограничивать домены коллизий. Кроме этого маршрутизаторы предотвращают распространение широковещательных сообщений в сети, что позволяет создавать отдельные широковещательные домены. Блокировка распространения широковещательных сообщений маршрутизаторами определяет границы **широковещательного домена** – области, за пределы которой не выходят широковещательные сообщения, генерируемые компьютерами сети. На рисунке 3.31 изображена сеть, построенная на основе маршрутизаторов, и показаны границы доменов коллизий (штриховая линия) и широковещательных доменов (двойная ли-

ния).

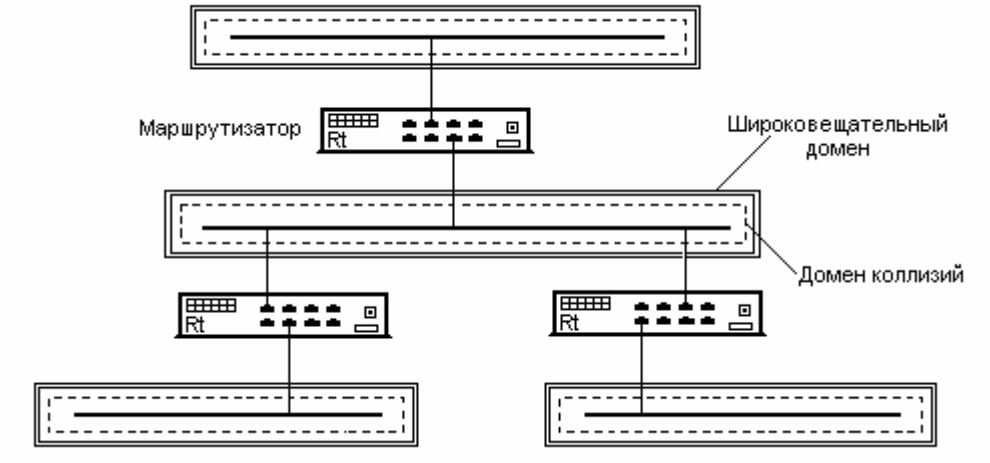


Рисунок 3.31 – Разделение сети на домены посредством маршрутизаторов

Разделение сегментов с помощью маршрутизаторов приводит к тому, что в сети создается несколько широковещательных доменов, а каждый сегмент принадлежит отдельной подсети. В связи с этим принципы функционирования рабочих станций в сети с маршрутизаторами отличаются от принципов работы в сетях, объединенных с помощью мостов. В сетях с мостами и повторителями рабочие станции передают кадры так же, как если бы отправитель и получатель находились в одном домене коллизий.

Рассмотрим пример обмена кадрами в сети, состоящей из двух сегментов А и Б, разделенных маршрутизатором (рисунок 3.32). MAC- и IP-адреса взаимодействующих станций и маршрутизатора указаны на рисунке.

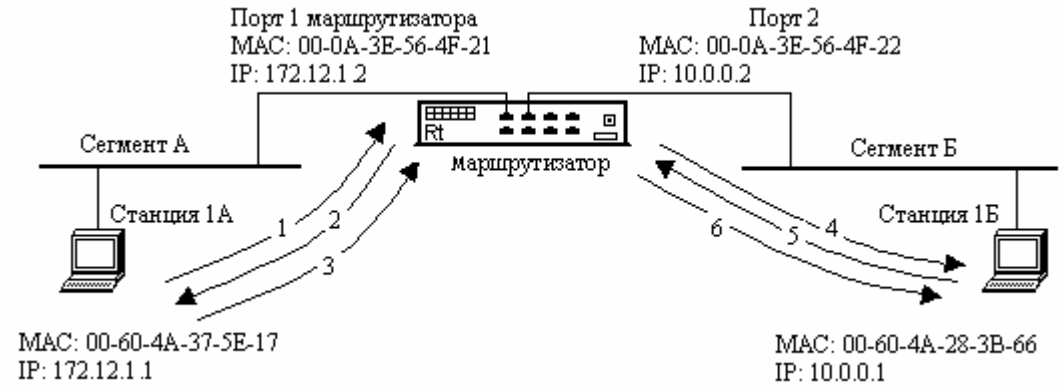


Рисунок 3.32 – Схема обмена кадрами в сети с маршрутизатором

На рисунке стрелками показана последовательность номеров кадров и направление их перемещения.

Пусть рабочей станции 1А сегмента А необходимо отправить пакет станции 1Б сегмента Б. Причем, станции 1А известен лишь IP-адрес компьютера 1Б. По таблице адресов станция 1А определяет, что получатель находится в другой сети. В этом случае отправитель должен осуществлять обмен через маршрутизатор.

Станции 1А известен только IP-адрес маршрутизатора. Для определения MAC-адреса маршрутизатора она отправляет широковещательный кадр (номер1) с указанием IP-адреса маршрутизатора. На это маршрутизатор отвечает кадром (номер 2), в котором сообщает свой MAC-адрес. Станция 1А отправляет на этот MAC-адрес маршрутизатора пакет с данными, указывая в заголовке сетевого уровня IP-адрес компьютера получателя (таблица 3.5).

Таблица 3.5 – Изменения параметров кадров в процессе обмена по сетям, разделенных маршрутизатором

Пакет	Заголовок канального уровня		Заголовок сетевого уровня	
	MAC-адрес получателя	MAC-адрес отправителя	IP-адрес отправителя	IP-адрес получателя
1 – зпр	FF–FF–FF–FF–FF–FF	00–60–4A–37–5E–17	172.12.1.1	172.12.1.2
2 – отв	00–60–4A–37–5E–17	00–0A–3E–56–4F–21	172.12.1.2	172.12.1.1
3 – дан	00–0A–3E–56–4F–21	00–60–4A–37–5E–17	172.12.1.1	10.0.0.1
4 – зпр	FF–FF–FF–FF–FF–FF	00–0A–3E–56–4F–22	10.0.0.2	10.0.0.1
5 – отв	00–0A–3E–56–4F–22	00–60–4A–28–3B–66	10.0.0.1	10.0.0.2
6 – дан	00–60–4A–28–3B–66	00–0A–3E–56–4F–22	172.12.1.1	10.0.0.1

Если маршрутизатору не известен MAC-адрес станции-получателя, то он путем посылки широковещательного кадра с IP-адресом компьютера 1Б узнает его MAC-адрес (кадры 4 и 5) и затем отправляет по нему кадр данных станции 1А, хранящийся у него в буфере (кадр 6). Как видно из таблицы 3.3 в процессе обмена через маршрутизатор изменяются только заголовки канального уровня, в то время как адреса сетевого уровня остаются неизменными. В таблице 3.3 отображены изменения параметров заголовков кадров в процессе обмена пакетами между станциями 1А и 1Б сетевых сегментов А и Б соответственно.

3.6.4. Виртуальные локальные сети

Виртуальной локальной сетью (*Virtual LAN,* ***VLAN***) называется совокупность узлов некоторой компьютерной сети, трафик которой, в том числе широковещательный, на канальном уровне полностью изолирован от трафи-

ка других узлов этой сети. Это означает, что передача кадров между разными виртуальными сетями на основании MAC-адреса невозможна.

Основное назначение технологии *VLAN* – недопущение трафика из одной сети в другую. Это делается либо с целью увеличения реальной пропускной способности сегментов сети, или с целью защиты от несанкционированного доступа. Виртуальные сети возможно создавать на основе коммутаторов из групп пользователей, основываясь на их задачах, а не по физическому расположению в сети. *VLAN* могут быть построены на базе одного или нескольких коммутаторов.

Виртуальные сети на основе одного коммутатора создаются в небольших организациях, в которых рабочие группы состоят из 2…6 компьютеров. В таких сетях применяется механизм *группирования портов* коммутатора. На рисунке 3.33 показано, как компьютерная сеть одной организации, содержащей два файл-сервера ФС и 8 рабочих станций, разделена на три виртуальные сети. При использовании механизма группировании портов каждый порт программным образом назначается одной из виртуальных сетей. Обмен данными в таком случае будет осуществляться только между указанными портами. Порт можно приписать нескольким виртуальным сетям, однако, в случае требований повышенной безопасности это действие исключается.

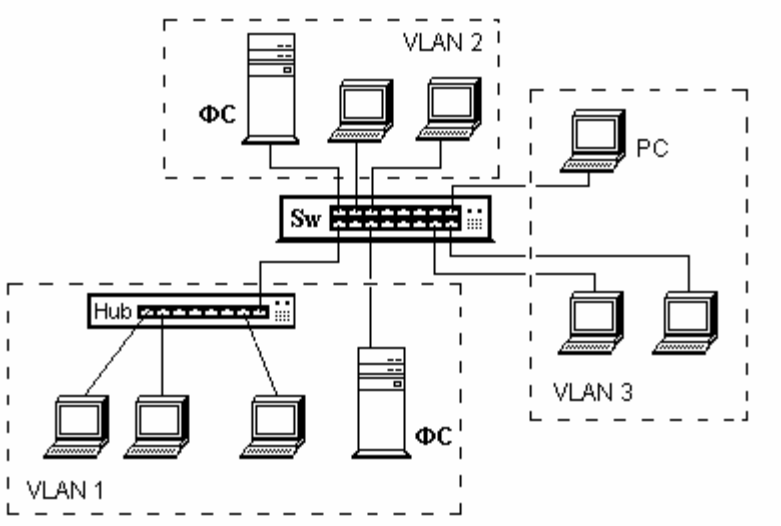


Рисунок 3.33 – Виртуальные сети, построены на основе одного коммутатора

Достоинством VLAN на базе портов является высокий уровень управляемости и безопасности. К недостаткам такого вида сетей следует отнести необходимость физического переключения устройств при изменении конфигурации отдельных сетей. Другим способом создания виртуальных сетей на

базе одного коммутатора является группирование MAC-адресов, при котором каждый физический адрес приписывается той или иной виртуальной сети.

На рисунке 3.34 показана схема реализации двух виртуальных локальных сетей VLAN 1 и VLAN 2, созданных на основе двух коммутаторов. На рисунке узлы, относящиеся к VLAN 1, заштрихованы.

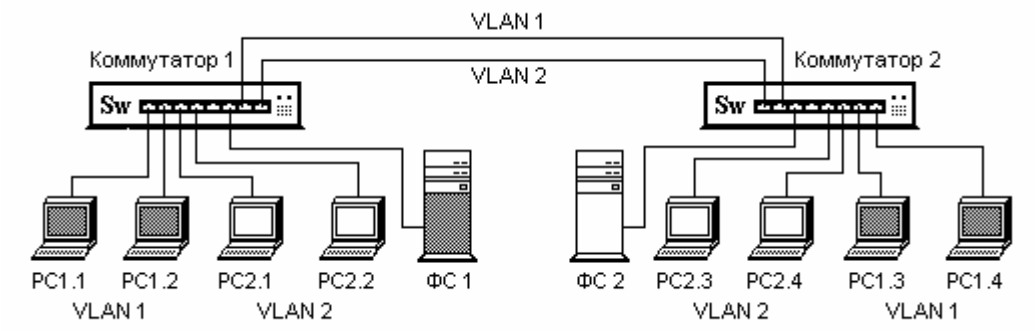


Рисунок 3.34 – Виртуальные сети на основе двух коммутаторов

При использовании механизма группирования портов между коммутаторами должно быть установлено столько связей (выделено портов), сколько виртуальных сетей они поддерживают. Это приводит к повышению расходов кабелей на создание сетей. Группирование MAC-адресов по виртуальным сетям избавляет от необходимости связывать коммутаторы посредством нескольких кабелей. В таком случае распределение кадров по сетям выполняется коммутаторами на основе MAC-адресов, являющихся признаком принадлежности к конкретной виртуальной сети.

Способы создания VLAN во многом определяются возможностями коммутаторов, с помощью которых строятся виртуальные сети. С каждым годом эти возможности расширяются. В настоящее время существуют коммутаторы и программные средства (например, коммутаторы и программа конфигурации фирмы Xylan), которые позволяют создавать VLAN на **сетевом уровне**, **на базе протоколов** и **на базе правил**. Виртуальные ЛВС сетевого уровня дают возможность администратору связать трафик для того или иного протокола в соответствующей виртуальной сети. Администратор может самостоятельно выбрать поля в заголовках кадров, по которым будет определяться принадлежность к виртуальной сети, и загрузить подготовленные правила во все коммутаторы сети.

Виртуальная локальная сеть на базе правил – наиболее мощная реализация VLAN, позволяющая администратору использовать любые комбинации критериев для создания виртуальных сетей. После того, как правила загружены во все коммутаторы, они обеспечивают организацию VLAN на ос-

нове заданных администратором критериев. Поскольку в таких сетях кадры постоянно анализируются коммутаторами на предмет соответствия заданным критериям, принадлежность пользователей к виртуальным сетям может меняться в зависимости от текущей деятельности пользователей.

Для упрощения управления обмена информацией в виртуальных сетях Ethernet был разработан стандарт **IEEE 802.1q**. В соответствии с этим стандартом к кадру Ethernet добавлены четыре байта. Эти 32 бита содержат информацию по принадлежности кадра Ethernet к конкретной VLAN и о его приоритете. Очевидно, что изменение структуры кадра Ethernet влечет за собой возникновение серьезных проблем, так как нарушается совместимость со всеми традиционными устройствами Ethernet, ориентированными на старый формат кадра. Это связано с тем, что данные 802.1q размещаются перед полем с информацией о длине полезной нагрузки (или типе протокола). Традиционное сетевое устройство в процессе анализа заголовка не обнаружит эту информацию на обычном месте. На его месте располагается "метка" виртуальной сети (рисунок 3.35). Новое поле состоит из так называемого тэга протокольного идентификатора **TPID** (*Tag Protocol Identifier*) и тега управляющей информации **TCI** (*Tag Control Information*). Поле TPID имеет длину два байта и содержит код 0x8100. Поскольку это число больше 1500, сетевые карты *Ethernet* будут интерпретировать его как тип, а не как длину. Структура полей TCI изображена в нижней части рисунка.

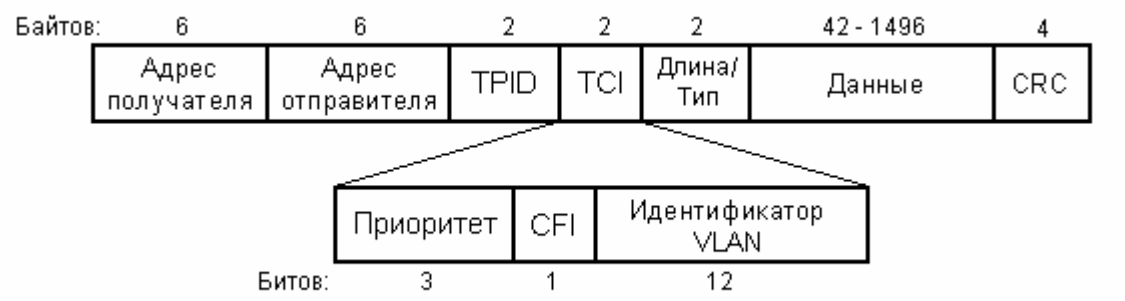


Рисунок 3.35 – Формат кадра Ehternet с меткой виртуальной сети

Трехбитовое поле "**Приоритет**" позволяет выделять *трафик реального времени*, *трафик со средними требованиями* и трафик, для которого *время доставки не критично*. Это открывает возможность использования сети Ethernet для задач управления и обеспечения качества обслуживания при транспортировке мультимедийных данных. Однобитовое поле **CFI** (*Canonical Format Indicator*) зарезервировано для обозначения кадров сетей других типов (Token Ring, FDDI), передаваемых по магистрали Ethernet. В настоящее время его функцией (CFI=1) является указание того, что в поле

данных содержится кадр 802.5. Поле "**Идентификатор VLAN**" длиной 12 бит определяет, какой виртуальной сети принадлежит кадр.

12-битовое поле позволяет коммутаторам разных производителей создавать до 4096 общих виртуальных сетей. Длина модифицированного кадра Ethernet увеличивается на 4 байта, так как, помимо двух байтов собственно тега, добавляются еще два байта. Чтобы не нарушить стандартную максимальную длину кадра при использовании заголовка 802.1p/q поле данных уменьшают на два байта.

3.6.5. Алгоритм покрывающего дерева

В сетях Ethernet коммутаторы поддерживают только древовидные связи т.е. которые не содержат петель. При построении или модернизации сегментированной сети с большим количеством мостов и/или коммутаторов в результате ошибок монтажа или попыток резервирования соединений возможно образование дополнительных связей между сегментами, когда от одного сегмента к другому пакет может попасть более чем одним путем (рисунок 3.36). Это приведет к циркуляции пакетов в замкнутых петлях и перегрузке сети. Кроме того, каждый посланный пакет, поступающий через разные порты, мосты/коммутаторы принимают за два различных пакета и постоянно обновляют свои таблицы. В приведенном примере проблема может быть решена удалением моста 1 и разрывом связи, помеченной знаком “X”.

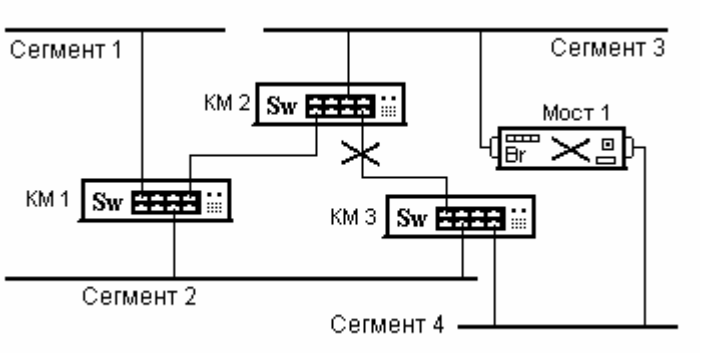


Рисунок 3.36 – Пример иллюстрации наличия петель в сети

Для автоматического решения проблемы зацикливания пакетов был предложен так называемый "алгоритм покрывающего дерева". Алгоритм покрывающего дерева **STA** (*Spanning Tree Algotithm*) обеспечивает построение древовидной топологии связей сети с единственным путем минимальной стоимости от каждого коммутатора и от каждого сегмента до некоторого выделенного корневого коммутатора – корня дерева. Реализация алгоритма

осуществляется на основе протокола **STP** (*Spanning Tree Protocol*), в результате действия которого мост или коммутатор самостоятельно обнаруживает лишние связи и автоматически блокирует ряд соединений, приведших к образованию петель. В случае возникновения аварийных ситуаций и недоступности основного пути, заблокированные соединения могут быть вновь открыты. Этим обеспечивается высокая надежность сети. STP входит в состав протокола мостов и коммутаторов IEEE 802.1d.

При использовании протокола STP при начальной конфигурации каждой линии связи присваивается определенный вес (чем выше приоритет, тем меньше вес). Мосты и коммутаторы периодически рассылают специальные сообщения – **протокольные блоки данных моста BPDU** (*Bridge Protocol Data Unit*), которые содержат коды уникальных идентификаторов, присвоенных им при изготовлении. Мост или коммутатор с наименьшим значением такого кода становится корневым ("корень дерева"). Затем выявляется наикратчайшее расстояние от корневого моста/коммутатора до любого другого моста в сети. В основу алгоритма STA положена теорема из теории графов, которая утверждает, что *структура любого связного графа, содержащего петли, может быть изменена путем удаления ребер таким образом, что он сохранит прежнюю связность, и при этом не будет иметь петель.* Граф, описывающий дерево наикратчайших связей, и является "покрывающим деревом". Такое дерево включает все узлы сети, но необязательно все мосты или коммутаторы. Алгоритм STA функционирует постоянно, отслеживая все топологические изменения. Реализация алгоритма в компьютерной сети возможна только при условии поддержки его всеми коммутаторами/мостами.

Мосты и коммутаторы, поддерживающие алгоритм STA, автоматически создают активную древовидную конфигурацию связей, то есть связную конфигурацию без петель. Древовидная структура сети строится на основании информации, полученной в результате обмена служебными пакетами, и адаптивно перестраивается при возникновении изменений в сети. В последующем при описании протокола мы будем для упрощения связывать его с сетевыми коммутаторами, подразумевая, что это относится и к мостам.

Сообщения протокольного блока данных BPDU размещаются в информационном поле блоков данных канального уровня – кадров Ethernet или Token Ring. **Формат BPDU** включает различные идентификаторы.

1. "Идентификатор протокола" (*Protocol identifier*) – два байта.
2. "Версия протокола" (*Version*) длиной 1 байт.
3. "Тип сообщения" (*Message type*) размером 1 байт. Сообщение может быть конфигурационным (0x00) и извещающим о смене топологии (0x80).
4. "Флаги" (Flags) – 1 байт. Используются только два бита этого поля. Первый из них, TC-бит, сигнализирует о смене топологии (Topology

Change), а восьмой, TCA-бит, подтверждает изменение конфигурации (Topology Change Acknowledgment).

5. "Идентификатор корневого коммутатора " (Root ID) состоит из 8 байтов. Первые два байта отображают **приоритет** данного устройства, который устанавливается системным администратором. Последующие 6 байтов представляют собой MAC- адрес блока управления корневого моста.

6. "Стоимость пути до корня" (Root path cost). Поле длиной 4 байта отображает суммарную стоимость кратчайшего пути от коммутатора–источника сообщения до корневого моста;

7. "Идентификатор моста/коммутатора" (Bridge ID) – 8 байтов. Содержит идентификатор моста, являющегося источником данного сообщения.

8. "Идентификатор порта" (Port ID) указывает порт, через который было передано данное сообщение. Длина его составляет 2 байта, старший из которых может изменяться администратором и является приоритетом порта, а второй представляет собой порядковый номер порта для данного коммутатора (номера портов начинаются с единицы).

9. "Возраст сообщения" (Message age). Поле занимает 2 байта и предназначено для указания времени существования сообщения об изменении топологии в сети. Каждый коммутатор при прохождении через него сообщения модифицирует содержимое этого поля путем добавления значения задержки, соответствующей данному каналу;

10. "Максимальный возраст сообщения" (Maximum age) – время, по истечении которого сообщение BPDU игнорируется коммутатором;

11. "Интервал Hello" (Hello time) – промежуток времени между отправкой конфигурационных сообщений корневым коммутатором (1…4 с);

12. "Задержка перехода" (Forward delay). Значение этого поля определяет величину интервала времени, который должен предшествовать переходу коммутатора в новое состояние при изменении топологии системы. Эта задержка предназначена для исключения возникновения циклических маршрутов во время переходных процессов в сети. Три последних поля BPDU-блока занимают по 2 байта каждый.

В начале работы администратор указывает корневой коммутатор путем задания ему нулевого (самого высокого) уровня приоритета, т.е. значение двух старших байт его идентификатора равно нулю. Приоритет других коммутаторов должен быть отличным от нуля. В этом случае, не зависимо от значения MAC-адресов коммутаторов сети, корневой будет всегда иметь минимальное значение идентификатора. Кроме этого администратор должен задать стоимость портов каждого из коммутаторов. Стоимость порта (*Port Cost*) определяется как условное время передачи бита через данный порт. На практике стоимость порта часто вычисляют по формуле:

Стоимость порта = 1000 / (*скорость передачи порта*, Мбит/с).

Например, стоимость порта 10Base-T=100; 100Base-TX =10; Token Ring –250 или 63, канала T1 = 651 и т.д. Значения другого варианта задания стоимостей, регламентированные стандартом IEEE 802.1d, приведены в таблице 3.6.

Таблица 3.6 – Стоимость порта в зависимости от скорости передачи

Скорость передачи, Мбит/с	4	10	16	45	100	155	622	1000	10000
Стоимость порта	250	100	62	39	19	14	6	4	2

Построение сети без петель по протоколу STP осуществляется следующим образом. Каждый из сетевых коммутаторов путем выдачи на все свои порты BPDU-блоков анонсируют себя в качестве корневого, помещая свой идентификатор в полях "Идентификатор корневого коммутатора" и "Идентификатор коммутатора". Создание древовидной структуры проиллюстрируем на примере сети, изображенной на рисунке 3.37.

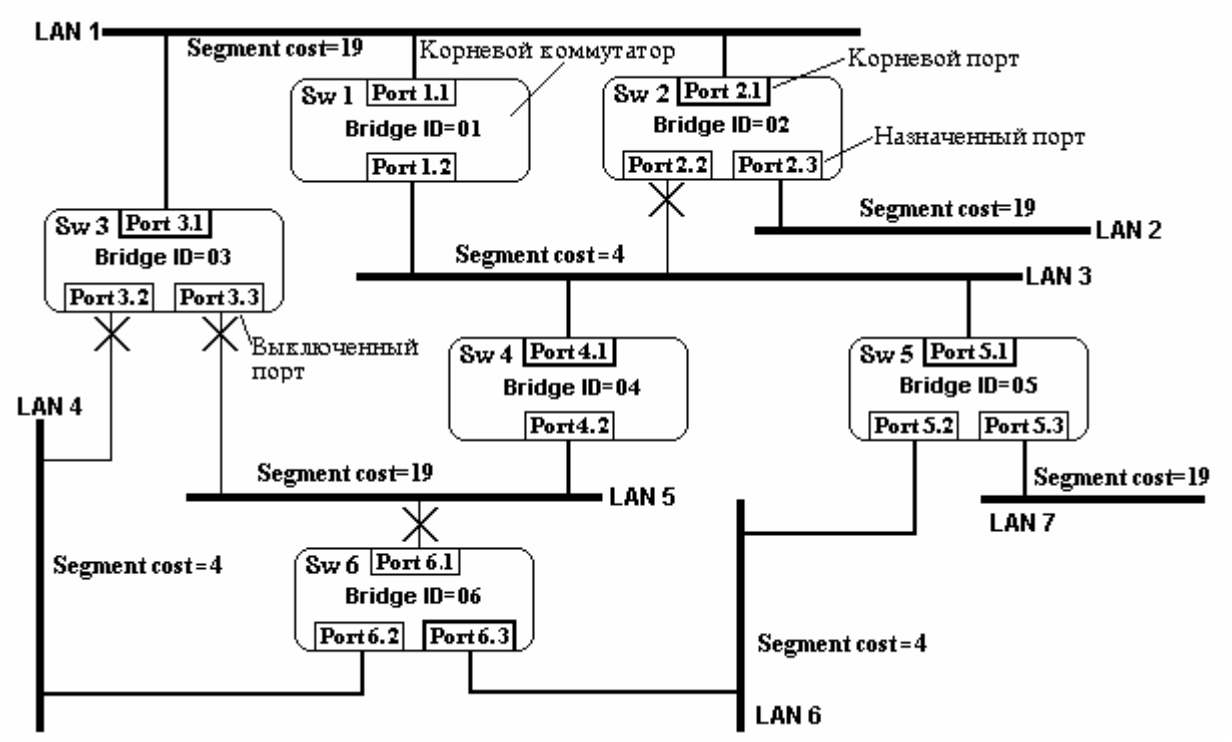


Рисунок 3.37 – Пример построения покрывающего дерева

Она состоит из семи сегментов локальных сетей LAN, объединяемых в единую сеть посредством шести коммутаторов Sw 1…6. Скорости передачи

данных в сегментах LAN1, 2, 5 и 7 составляет 100 Мбит/с, а в остальных –1 Гбит/с. Построение дерева начинается с корневого коммутатора. В соответствии с алгоритмом STA в качестве корневого назначается коммутатор, обладающий минимальным значением идентификатора. В данном примере это коммутатор с идентификатором Bridge ID=01.

Затем для каждого сетевого коммутатора из всех портов данного коммутатора определяется *корневой порт (root port),* т.е. такой порт, путь от которого до любого из портов корневого моста имеет минимальную стоимость. Для этого корневой коммутатор рассылает BPDU-блоки на все свои выходные порты. В поле "Стоимость пути до корня" вначале устанавливается нулевое значение. Следующие коммутаторы прибавляют к этому полю стоимость своих портов и отправляют блоки далее смежным узлам. Это дает возможность каждому коммутатору определить свой корневой порт, через который можно попасть в корневой коммутатор с минимальной стоимостью. Как только коммутатор получает BPDU-блок, содержащий идентификатор корневого коммутатора со значением меньше его собственного, он перестает генерировать свои собственные кадры BPDU, а начинает ретранслировать только кадры нового претендента на статус корневого коммутатора.

В процессе ретрансляции кадров каждый коммутатор увеличивает указанную в пришедшем блоке BPDU стоимость пути до корня дерева на величину стоимости сегмента (*Segment cost*), через который поступил данный блок. Тем самим в кадре BPDU, по мере прохождения через коммутаторы, аккумулируется стоимость пути до корневого узла. В течение этой процедуры каждый коммутатор для каждого из своих портов запоминает параметры путей минимальной стоимости до корня, содержащиеся во всех принятых этим портом кадрах BPDU. Так стоимости маршрутов от коммутатора Sw 06 до корневого коммутатора Sw 01 через Port 6.1 и коммутатор Sw 04 составляет 19+4=23 единицы; через Port 6.1 и Sw 03 – 38; через Port 6.2 и Sw 03 – 23; и, наконец, через Port 6.3 и Sw 05 – 8. Затем эти коммутаторы выделяют из всех своих портов тот, который имеет минимальную стоимость до корневого коммутатора и назначает его своим **корневым портом**. Для коммутатора Sw 06 корневым портом назначается Port 6.3. Аналогичным образом находятся корневые порты остальных коммутаторов сети. На рисунке 3.34 они выделены утолщенной линией.

На следующем этапе функционирования алгоритма для каждого логического сегмента сети из всех портов всех коммутаторов, подсоединенных к данному сегменту, выбирается порт, через который будут передаваться пакеты от этого сегмента в направлении **корня** через **корневой порт** одного из коммутаторов. Для этого сначала из рассмотрения исключаются корневые порты коммутаторов, подключенных к данному сегменту. Затем из всех оставшихся портов выбирается порт с минимальной стоимостью пути до кор-

ня. Этот порт называется "**назначенный порт"** (*designated port*), а коммутатор, которому он принадлежит, получил название ***назначенный коммутатор*** *(designated switch)*. Если в данном коммутаторе имеется несколько портов с одинаковой стоимостью, то назначенным определяется порт, имеющий минимальный идентификатор.

Все остальные порты, кроме корневых и назначенных, отключаются и переводятся в резервное состояние, то есть такое, при котором они не передают обычные кадры данных. На рисунке 3.34 назначенные порты соединены с сегментами LAN утолщенными линиями, а отключенные связи помечены знаком X. При таком выборе активных портов в сети исключаются петли и оставшиеся связи образуют покрывающее дерево. Обратите внимание, что у сегмента может быть только один назначенный порт. У корневого коммутатора все порты являются назначенными, а их стоимости до корня полагаются равными нулю. Корневой порт у корневого коммутатора отсутствует.

В процессе нормальной работы корневой коммутатор продолжает генерировать служебные пакеты BPDU, а остальные коммутаторы принимают их своими корневыми портами и ретранслируют через назначенные порты. Если по истечении максимального времени жизни сообщения (по умолчанию — 20 с) корневой порт любого коммутатора сети не получит служебный пакет BPDU, то он инициализирует новую процедуру построения покрывающего дерева.

Алгоритм STA включает кроме процедуры инициализации активной конфигурации и процедуру изменения конфигурации при отказах элементов сети.

Развитием стандарта 802.1d STP стал стандарт IEEE 802.1w — протокол *Rapid Spanning Tree Protocol* (**RSTP**). Он был разработан для преодоления отдельных ограничений STP, которые мешали внедрению ряда новых функций коммутаторов, в частности, функций 3-го уровня, всё больше и больше применяемых в коммутаторах Ethernet. Процесс вычисления связующего дерева у обоих протоколов одинаков. Однако при работе RSTP, порт может перейти в состояние передачи значительно быстрее, так как он не зависит от настройки таймеров. Порты больше не должны ждать стабилизации топологии, чтобы перейти в режим продвижения.

3.7. Архитектура беспроводных сетей

3.7.1. Целесообразность и особенность применения беспроводных сетей

В практической деятельности встречаются ситуации, в которой принципиально невозможна прокладка кабеля. Это, как правило, старинные здания либо очень дорогие интерьеры, где стоимость прокладки кабельной сети непомерно высока, или важна скорость установки и запуска сети (временные сети на выставках, семинарах и т.п.). В такой ситуации проблема решается путем развертывания беспроводной локальной компьютерной сети **WLAN** (*Wireless Local Area Network*).

Беспроводные решения широко применяются также в корпоративных компьютерных сетях для объединения удаленных локальных сетей в случае отсутствия между ними проводных или волоконно-оптических каналов или необходимости их резервирования. В зависимости от требований к беспроводной сети она строится либо на беспроводных сетевых адаптерах с использованием *точки доступа* в качестве базовой станции, что обеспечивает минимальную стоимость, либо на беспроводных маршрутизаторах, применение которых позволяет достичь максимальной производительности. В зависимости от технологии беспроводные сети можно разделить на три типа: *локальные*; *расширенные локальные* и *мобильные* (на основе переносных компьютеров).

Основные различия между этими типами сетей – параметры передачи. Локальные и расширенные локальные вычислительные сети используют передатчики и приемники, принадлежащие той организации, в которой функционирует сеть. Для переносных компьютеров в качестве среды передачи сигналов применяются радиоканалы операторов связи (UkrStar, UMC) либо местные телефонные компании и их общедоступные службы.

Типичная беспроводная локальная сеть выглядит и функционирует практически так же, как обычная, за исключением среды передачи. В ее состав входит два типа оборудования: компьютеры, укомплектованные беспроводными сетевыми картами, и *базовая станция*, называемая **точкой доступа** (*access point*), которая может выполнять роль моста между беспроводной и проводной сетями.

Точка доступа содержит *радиотрансивер* (приемопередатчик), интерфейс проводной сети, а также встроенный микрокомпьютер и программное обеспечение для обработки данных. Базовая станция определяет не только радиус действия и скорость передачи данных, но и решает задачи управления сетью и обеспечения ее безопасности. Многие точки доступа оснащаются двумя антеннами, причем в каждый момент времени работает антенна с

лучшим качеством принимаемого сигнала. Переключение антенн повышает качество связи, которое ощущается уже на удалении в несколько метров. Соответственно увеличивается скорость передачи по сравнению с одноантенными точками доступа. Базовые станции оснащаются последовательным USB интерфейсом. Это позволяет непосредственно подключать их к консоли управления. Программное обеспечение точек доступа поддерживает протоколы HTTP и Telnet, что обеспечивает удобное администрирование радиосети через Internet и интерфейс браузера. При этом обязательно применяются специальные протоколы защиты удаленного управления. Для упрощения монтажа сети многие точки доступа не требуют дополнительной подводки электропитания. Его они получают через кабель, посредством которого точки доступа связаны с компьютерной сетью.

Сеть на беспроводных адаптерах позволяет подключать по радио к точке доступа базовой станции как отдельные компьютеры, так и проводные сети, при назначении одного из компьютеров такой сети сервером радиодоступа и оснащении его проводным и беспроводным сетевыми адаптерами. Более эффективные сети получаются при использовании оборудования, специально созданного для этих целей – беспроводных маршрутизаторов. Такие устройства могут использоваться как для создания каналов "точка-точка", так и для развертывания масштабных сетей сложной топологии с возможностью многократной ретрансляции сигналов.

Совместимость беспроводных компьютерных сетей обеспечивается международными стандартами IEEE 802.11a/b/g/n, которые регламентируют способы и скорости передачи, способы модуляции и используемые частотные диапазоны.

3.7.2. Способы построения WLAN

В зависимости от количества компьютеров в сети и расстояния между ними, беспроводные сети могут быть созданы двумя различными способами:

1) сеть без базовой станции (режим *Ad Hoc*);
2) сеть с точкой доступа (*Infrastructure Network*).

В режиме **Ad HOC** (рисунок 3.38) без базовой станции, который называют также IBSS (*Independent Basic Service Set*) или режимом «точка-точка», компьютеры непосредственно взаимодействуют друг с другом, пока они находятся в пределах устойчивой радиосвязи.

Для этого режима нужен минимум оборудования: каждый компьютер должен быть оснащен только беспроводным адаптером. При такой конфигурации не требуется создания сетевой инфраструктуры. Основными недос-

татками режима Ad Hoc являются ограниченный диапазон действия возможной сети и невозможность подключения к внешней сети (например, к Интернету).

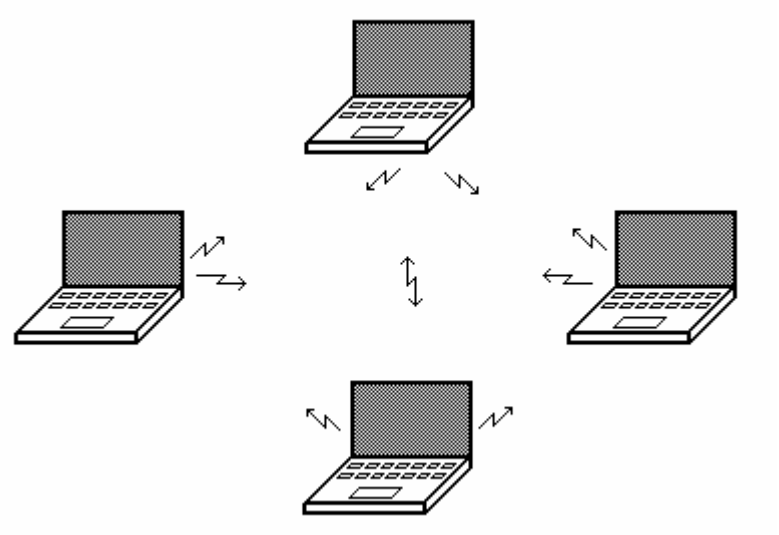


Рисунок 3.38 – Беспроводная сеть без базовой станции, режим Ad HOC

На практике рекомендуется использовать его при наличии в сети 3…4-х компьютеров и расстояния между ними не более 30 м. При втором способе построения – ***Infrastructure Network***, компьютеры взаимодействуют друг с другом не напрямую, а через **точку доступа AP** (*Access Point*), которая выполняет в беспроводной сети роль своеобразного концентратора (аналогично тому, как это происходит в традиционных кабельных сетях). Рассматривают два режима взаимодействия с точками доступа: **BSS** *(Basic Service Set*) и **ESS** (*Extended Service Set*). В режиме BSS все станции связываются между собой только через точку доступа, которая может выполнять также роль моста к внешней сети (рисунок 3.39). Как видно из рисунка, расстояние между компьютерными станциями увеличивается, как минимум, вдвое. Одна точка доступа обеспечивает обслуживание от 15 до 250 абонентов в зависимости от конфигурации сети и технологии доступа. Увеличить емкость сети можно просто, добавив новые точки доступа, при этом не только расширяется зона обслуживания, но и снижается вероятность перегрузки.

В **расширенном режиме** ESS существует инфраструктура нескольких базовых сетей BSS, причем сами точки доступа взаимодействуют друг с другом через некоторую систему, называемую **системой распространения** (DS, *Distribution System*), позволяющую передавать трафик от одной BSS к другой (рисунок 3.40). Связь базовых сетей с DS осуществляется посредством точек доступа. Между собой точки доступа соединяются с помощью сегментов ка-

бельной сети, либо радиомостов, т.е. система распространения представляет собой либо совокупность устройств AP либо сегмент локальной сети.

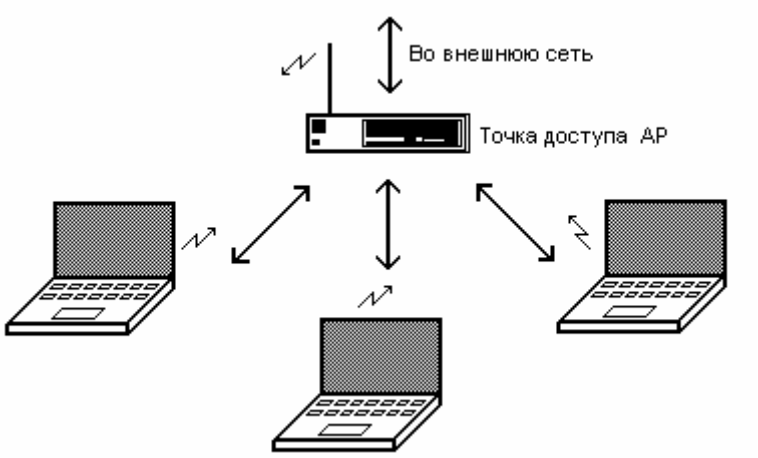


Рисунок 3.39 – Беспроводная сеть с точкой доступа – *Infrastructure Network*

Кроме двух различных режимов функционирования беспроводных сетей на MAC-уровне определяются правила коллективного доступа к среде.

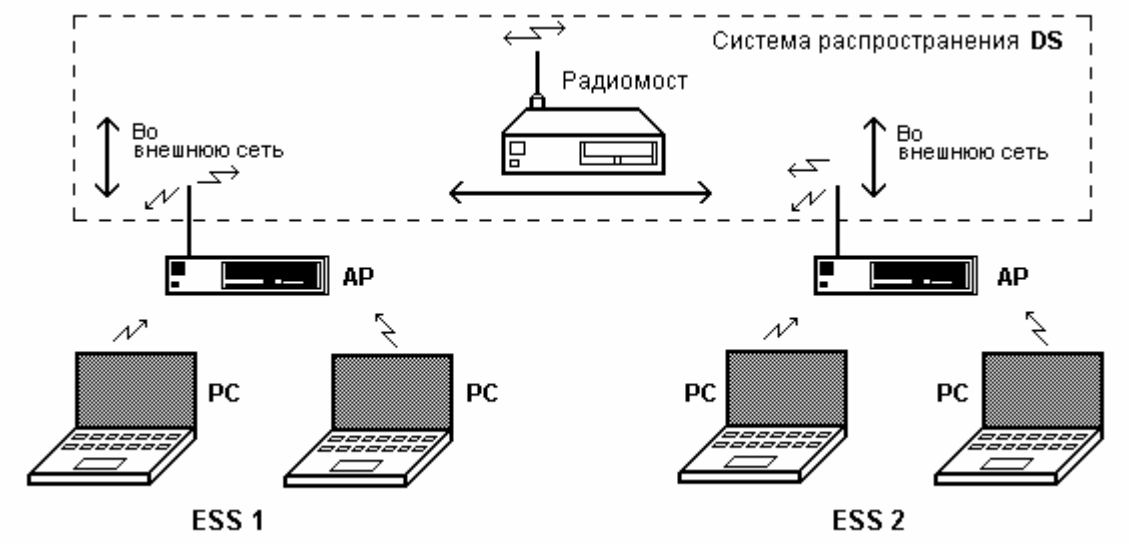


Рисунок 3.40 – Объединение беспроводных сетей

Существуют жесткие правила, регламентирующие коллективный доступ к среде передачи сигналов. Это вызвано тем, что при одновременной передаче сигналов в эфир двух и более станций одновременно выделить отдельные пакеты весьма затруднительно, а в ряде случаев и невозможно.

В беспроводных сетях при организации доступа используется два спо-

соба обслуживания: **асинхронный** (*Asynchronous Data Service*) и **обслуживание с ограниченным временем** (*Time Bounded Service*).

При асинхронном обслуживании каждая из станций может получить доступ к разделяемой среде, но при этом возможны коллизии, которые станции пытаются предотвратить. Этот способ не гарантирует номинальную пропускную способность сети. Служба используется как в сетях без базовой станции, так и в сетях с точкой доступа. Способ ограниченного времени обслуживания гарантирует максимально допустимое время доступа станции к среде. Он применяется только в сетях с промежуточной станцией (точкой доступа). В процессе реализации способа формируются так называемые суперкадры с двумя временными интервалами:

1) *бесконкурентного доступа*, в течение которого точка доступа поочередно опрашивает станции и при их готовности осуществляет обмен кадрами, при этом гарантируется номинальная производительность сети;

2) состязательного доступа.

Подробнее о способах доступа к WLAN сказано ниже.

3.7.3. Стандартизация построения беспроводных сетей

Для беспроводных сетей разработан стандарт IEEE 802.11. Организация IEEE постоянно совершенствует спецификацию, чтобы обеспечить ее применимость в более широком кругу приложений. В настоящее время реализованы стандарты 802.11, 802.11b и 802.11g. Основные параметры стандарта **802.11b** следующие:

- несущая частота излучаемых сигналов – 2,4 ГГц;
- число непересекающихся частотных каналов – 3;
- модуляция с использованием комплементарных кодовых последовательностей **CCK** (*Complementary Code Keying*), с шириной полосы 22 МГц на канал и одной несущей (см.п.2.4.4);
- метод доступа – CSMA/CA, (подробнее о нем в п. 3.7.4);
- максимальная скорость передачи данных – 11 Мбит/с.

Параметры стандарта **802.11a** заметно отличаются от 802.11b**:**

- несущая частота – 5 ГГц;
- число непересекающихся частотных каналов — 8;
- модуляция — многочастотная передача ортогональными сигналами **OFDM** (*Orthogonal Frequency Division Multiplexing*), с шириной полосы 20 МГц на канал с применением несколько несущих (см. п.2.4.5);
- метод доступа – CSMA/CA;
- максимальная скорость передачи данных – 54 Мбит/с.

Данные по стандарту 802.11b передаются на частоте 2,4 ГГц с одной из фиксированных скоростей 1, 2, 5,5 или 11 Мбит/с с использованием двух– или четырехпозиционной дифференциально–фазовой модуляции BPSK и QPSK. В полосе частот 2,400…2,48354 ГГц можно организовать до 14 каналов. В качестве примера на рисунке 3.41 показано распределение частот для трех неперекрывающихся по полосе каналов: 1-го (2,412 ГГц), 6-го (2,437 ГГц) и 11-го (2,462 ГГц). При таком расположении частот удается не только снизить взаимные помехи между каналами, но и упростить управление.

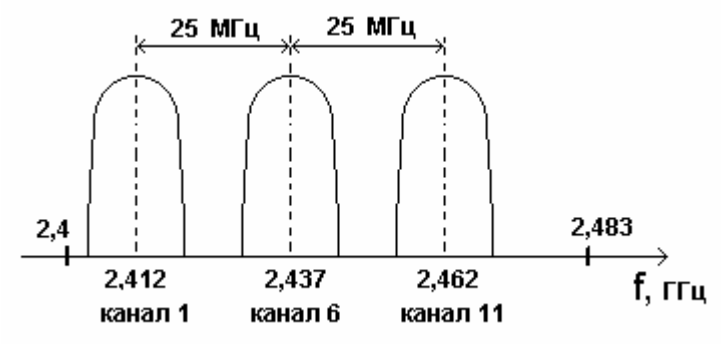


Рисунок 3.41 – Распределение частот по каналам для беспроводных сетей по стандарту 802.11b

Дальность действия абонентского оборудования стандарта 802.11b не превышает 100 м и зависит от скорости передачи, уровня и характера помех, а также от требований к качеству обслуживания. Чтобы обеспечить приемлемое качество соединения при снижении отношения сигнал/помеха, в спецификациях предусмотрена возможность автоматического снижения скорости информационного обмена.

Стандарт 802.11a использует в диапазоне 5 ГГц полосу частот шириной 300 МГц. Этот диапазон менее загружен, чем 2,4 ГГц, а следовательно, свободнее от помех. Суммарная доступная ширина спектра в нем примерно в четыре раза выше, чем в диапазоне 2,4 МГц (83 МГц). Вся полоса частот разделяется на три участка по 100 МГц каждый: 5,15…5,25 ГГц (нижний), 5,25…5,35 ГГц (средний) и 5,725…5,825 ГГц (верхний). В любом из них могут быть организованы четыре неперекрывающихся канала, а во всей выделенной полосе –12 частотных каналов. В каждом из участков спектра допускается своя максимальная мощность излучения: 50 мВт (нижний), 250 мВт (средний) и 1 Вт (верхний).

Следует заметить, что спектральные составляющие сигналов диапазона 5 ГГц более интенсивно поглощаются стенами зданий и другими препятствиями. Следовательно, потери при распространении сигналов в этом диапазоне выше. По этой причине радиус действия станций при прочих равных

условиях вдвое меньше, чем для диапазона 2,4 ГГц. В связи с этим для организации такой же по покрытию сети потребуется большее количество точек доступа, чем для сети на базе стандарта 802.11b.

Следующей разработкой в области беспроводных сетей стал стандарт **802.11g**, цель которого – увеличение скорости передачи данных в диапазоне частот 2,4 ГГц. Повышение скорости до 54 Мбит/с достигнуто за счет новых видов модуляции, в частности, ортогонального частотного мультиплексирования **OFDM**. В качестве базовых использованы две технологии передачи сигналов: OFDM и CCK. Способ OFDM гарантирует передачу данных со скоростью до 54 Мбит/с, а кодовая манипуляция CCK обеспечивает обратную совместимость со стандартом 802.11b.

Кроме обязательных методов передачи, в стандарте 802.11g предложены две дополнительные технологии широкополосного доступа: кодирование с двоичной сверткой пакетов **PBCC** (*Paket Binary Convolution Coding*) и комбинирование способов модуляции **CCK/OFDM**, которые могут быть реализованы в базовом оборудовании по желанию потребителя. Первая из них обеспечивает скорость передачи до 33 Мбит/с (первоначально 22 Мбит/с). Другая, основанная на комбинированном методе CCK/OFDM, позволяет передавать служебную информацию с использованием модуляции CCK, а полезную – при помощи OFDM (54 Мбит/с).

Многие производители беспроводных сетей вводят дополнительные функции в технологию передачи данных. В основе всех технологий расширения протокола 802.11g лежат такие принципы, как пакетная передача (*Packet Bursting*), сжатие данных, быстрые кадры и связывание каналов. В режиме пакетной передачи все кадры, передаваемые в одном блоке, используют сокращенные заголовки, что позволяет уменьшить объем передаваемой служебной информации и тем самым увеличить полезный трафик.

Некоторые производители для увеличения пропускной способности объединяют два канала с полосой пропускания по 22 МГц, чем повышают скорость передачи до 108 Мбит/с. Такой прием возможен по той причине, что стандарт 802.11g разрешает применять одиннадцать каналов в частотной полосе 2,4 ГГц, которые разделены промежутками по 5 МГц. Поскольку общепринятая ширина каждого канала составляет 22 МГц, имеется три канала без частичного наложения (1, 6 и 11), центральные частоты которых отстоят друг от друга на 25 МГц. Очевидно, что передача на двойной скорости возможна только на центральном канале номер 6.

В середине 2006 г. был одобрен новый стандарт **802.11n**. Спецификация 802.11n, предусматривает возможность передачи данных в беспроводных сетях со скоростью до 600 Мбит/с. Стандарт 802.11n использует частотные диапазоны 2,4 ГГц и 5 ГГц и совместим с 802.11a/b/g. Для повышения скорости используется способ с так называемым множественным вво-

дом/выводом **MIMO** (Multiple Input Multiple Output), при котором осуществляется параллельная передача множества сигналов в различных частотных диапазонах. Технология использует мультиплексирование типа **SDM** (*Spatial Division Multiplexing*), при котором сигнал передаётся на нескольких различных частотах одновременно, после приёма превращаясь в единый скоростной поток данных. Однако для реализации MIMO на практике необходимо, чтобы для каждого потока данных использовались свои антенны, цепи приёма/передачи и АЦП.

Дополнительное увеличение скорости передачи сигналов достигается за счет увеличения используемой полосы канала в два раза (до 40 МГц).

3.7.4. Способы доступа пользователей к ресурсам сети

В сетях IEEE 802.11 используется полудуплексный режим передачи, т.е. в каждый момент времени станция может либо принимать, либо передавать информацию, поэтому обнаружить коллизию в процессе передачи невозможно. По этой причине для IEEE 802.11 был разработан модифицированный вариант протокола CSMA/CD с предотвращением коллизий, получивший название **CSMA/CA** (*Carrier Sense Multiple Access with Collision Avoidance*). Функционирование сети в соответствии с этим протоколом происходит следующим образом. Станция, которая собирается передавать информацию, сначала "прослушивает эфир". Если не обнаружено активности на рабочей частоте, станция сначала ожидает в течение некоторого случайного промежутка времени, потом снова "слушает эфир" и, если среда передачи данных все еще свободна, осуществляет передачу. Наличие случайной задержки необходимо для того, чтобы работа сети не нарушалась, если несколько станций одновременно захотят получить доступ к каналу. При получении информационного пакета без искажений принимающая станция посылает источнику сообщение подтверждения. Для обеспечения безошибочности приема применяется контрольная последовательность пакета. Получив подтверждение, передающая станция считает процесс передачи данного информационного пакета завершенным. Если подтверждение не получено, станция-источник полагает, что произошла коллизия, и пакет передается снова через случайный промежуток времени.

Существует еще одна специфичная для беспроводных сетей проблема. Она возникает в ситуации, когда две клиентские станции имеют плохую связь друг с другом, но при этом качество связи каждой из них с точкой доступа хорошее. В таком случае передающая клиентская станция может послать в точку доступа **запрос на очистку эфира**. Тогда по команде с точки

доступа другие клиентские станции прекращают передачу на время "общения" двух точек с плохой связью. Режим принудительной очистки эфира (протокол *Request to Send/Clear to Send* – RTS/CTS) реализован далеко не во всех моделях оборудования IEEE 802.11 и, если он есть, то включается лишь в крайних случаях.

В отличие от проводной Ethernet, в которой при потоковой передаче используется распределенное управление доступом к моноканалу, в беспроводных сетях IEEE 802.11 применяется централизованное управление доступа. Клиентские станции последовательно опрашиваются на предмет передачи потоковых данных. Если какая-нибудь из станций сообщает, что она будет передавать потоковые данные, точка доступа выделяет ей промежуток времени, в который из всех станций сети будет передавать только одна.

На MAC-уровне протокола 802.11 определно два типа коллективного доступа к среде передачи данных: ***функция распределенной координации*** – **DCF** (*Distributed Coordination Function*) и ***функция централизованной координации*** **PCF** (*Point Coordination Function*). Рассмотрим более подробно каждый из этих механизмов.

Функция распределенной координации DCF реализована на основе метода коллективного доступа с обнаружением несущей и механизмом избежания коллизий **CSMA/CA** (*Carrier Sense Multiple Access/Collision Avoidance*). При такой организации каждый узел, прежде чем начать передачу, «прослушивает» среду, пытаясь обнаружить несущий сигнал, и только при условии, что среда свободна, может начать передачу данных.

Однако, в этом случае велика вероятность возникновения коллизий, когда два или более узлов сети одновременно (или почти одновременно) решат, что среда свободна, и начнут предавать данные. Для того чтобы снизить вероятность возникновения подобных ситуаций, используется механизм предотвращения коллизий **CA** (*Collision Avoidance*).

С целью уменьшения вероятности возникновения коллизий в беспроводных сетях после передачи каждого кадра должны выдерживаться межкадровые паузы. Стандартом установлено три вида таких интервалов:

- межкадровый интервал ожидания **DISF** (DCF *Interframe Space*), используемый при асинхронном способе обслуживания; его должны выдерживать все станции с момента освобождения среды передачи;
- межкадровый интервал ожидания **PIFS** (PCF *Interframe Space*), применяется в сетях с гарантированным временем обслуживания; этот интервал должна выдержать ведущая станция перед началом процедуры опроса подчиненных станций;
- укороченный межкадровый интервал **SIFS** (*Short Interframe Space*); это самая короткая пауза, которая используется станциями при передаче пакетов квитирования.

Суть механизма избежания коллизий CA заключается в следующем. Каждый узел сети, убедившись, что среда свободна, прежде чем начать передачу, выжидает в течение определенного промежутка времени. Этот промежуток является случайным и складывается из двух составляющих: обязательного промежутка **DIFS** и выбираемого случайным образом **промежутка обратного отсчета** (*backoff time*). В результате каждый узел сети перед началом передачи выжидает в течение случайного промежутка времени. Это существенным образом снижает вероятность возникновения коллизий, поскольку вероятность того, что два узла сети будут выжидать в течение одного и того же промежутка времени, чрезвычайно мала.

Для того чтобы гарантировать всем узлам сети равноправный доступ к среде передачи данных, необходимо соответствующим образом определить алгоритм выбора длительности промежутка обратного отсчета T_{bot}. Этот промежуток, хотя и является случайным, но в то же время определяется на основании множества некоторых дискретных промежутков времени, т.е. равен целому числу элементарных временных промежутков T_{ts}, именуемых тайм-слотами (*SlotTime*). Для выбора промежутка обратного отсчета каждый узел сети при начальной инициализации формирует так называемое окно конкурентного доступа **CW** (*Contention Window*). Оно используется для определения количества тайм-слотов, в течение которых каждая станция сети выжидает после завершения выдачи в эфир одной из станций текущего кадра, прежде чем начать передачу своего блока. Фактически окно CW – это диапазон для выбора количества тайм-слотов, причем, в реальных сетях минимальной размер окна определяется в 31 тайм-слот, а максимальный – 1023. Промежуток обратного отсчета вычисляется как количество тайм-слотов, рассчитанное на основании размера окна CW:

$$T_{\text{bot}} = Random\,[CW_{\min}, CW_{\max}] \times T_{\text{ts}} .$$

Временная диаграмма, иллюстрирующая равномерный доступ компьютеров к среде на основе способа распределенной координации DCF показана на рисунке 3.42. Когда узел сети пытается получить доступ к среде, то после обязательного промежутка ожидания DIFS запускается процедура обратного отсчета, т.е. включается обратный отсчет счетчика тайм-слотов, начиная от выбранного случайного значения окна CW. Если в течение всего промежутка ожидания среда оставалась свободной, то узел начинает передачу. В данном примере диаграмма иллюстрирует асинхронный доступ к среде трех станций. Черными треугольниками на диаграмме отмечены моменты подачи запросов передачи данных от верхнего уровня.

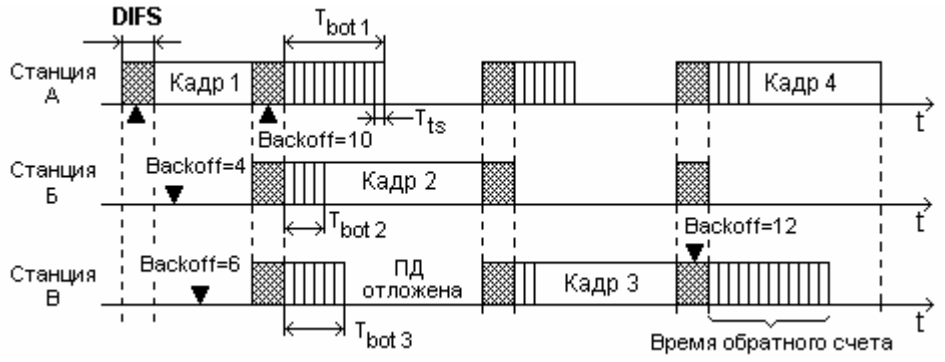


Рисунок 3.42 – Реализация равноправного доступа к среде передачи данных методом DCF

Из рисунка видно, что запрос на передачу от станции А поступил в момент времени, когда среда не занята. Поэтому после истечения обязательного фиксированного межкадрового интервала ожидания **DIFS** станция А начинает передачу кадра. Во время его передачи поступили запросы от станций Б и В. Так как среда передачи занята, обе станции вычисляют случайное значение интервала обратного отсчета (*Backoff*), которое в нашем примере для станции Б равно 4 тайм-слота, а для станции В – 6.

По завершению передачи первого кадра наступает обязательная пауза **DIFS**, во время которой от станции А снова поступил запрос на передачу. Но так как среда занята, а станция А только завершила передачу кадра, то в ней устанавливается максимально возможное окно конкурентного доступа CW, которое в приведенном примере равно 10. Передача второго кадра начнется станцией Б лишь после завершения фиксированного интервала DIFS и уменьшении значения счетчика обратного счета до нуля, которое осуществляется путем его многократного декрементирования с интервалом T_{ts}. Станции А и В при этом задерживают передачу своих данных. Одновременно декрементируются и счетчики остальных станций. По завершению передачи второго кадра и истечению обязательной паузы среду займет станция В, так как ее случайный интервал составляет всего два тайм-слота по сравнению с 6-ю в станции А.

Рассмотренный алгоритм реализации коллективного доступа к среде передачи данных гарантирует равноправный доступ всех узлов сети к среде. Однако при таком подходе вероятность возникновения коллизий хотя и мала, но все-таки существует. Снизить вероятность возникновения коллизий можно путем увеличения максимального размера формируемого окна CW. В то же время это увеличит величины задержек при передаче и тем самым снизит производительность сети. Поэтому в методе распределенного управления DCF для минимизации коллизий используется следующий алгоритм. После каждого успешного приема кадра принимающая сторона через корот-

кий промежуток **SIFS** (*Short Interframe Space*) подтверждает прием, посылая ответную квитанцию – кадр подтверждения ACK (рисунок 3.43).

Если в процессе передачи данных возникла коллизия, то передающая сторона не получает кадр ACK об успешном приеме. В этом случае размер CW-окна для передающего узла увеличивается почти вдвое.

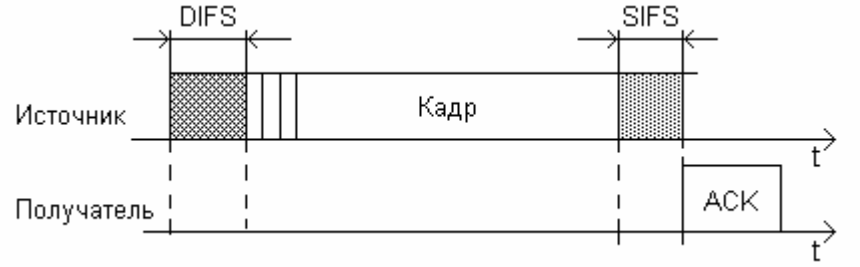


Рисунок 3.43 – Кадры квитанции, отсылаемые в случае успешной передачи данных

Так, если для первой передачи размер окна равен 31 слоту, то для второй попытки передачи он уже составляет 63 слота, для третьей – 127 слотов, для четвертой – 255, для пятой – 511, а для всех последующих – 1023 слота. Тогда для каждой *i*-й передачи (если все предыдущие оказались безуспешными) размер CW-окна увеличивается по следующему правилу:

$$CWi = 2CW_{i-1} + 1 \ .$$

Таким образом, увеличение размера окна происходит динамически по мере роста числа коллизий, что позволяет, с одной стороны, уменьшить временные задержки и, с другой стороны, снизить вероятность возникновения коллизий.

При реализации алгоритма равноправного доступа к среде передачи необходимо также учитывать и размер кадра данных. Так, при длинных кадрах повторная передача вследствие коллизий приведет к резкому снижению производительности сети. Короткие кадры хотя и обеспечивают равномерный доступ, однако из-за большой доли служебной информации эффективная скорость передачи данных снижается.

Как уже упоминалось выше, в беспроводных сетях с точкой доступа возможна ситуация, когда две станции настолько удалены друг от друга, что в связи с затуханием радиосигналов или наличием между ними естественных препятствий они не имеют возможности "слышать" друг друга. Такие станции называют *скрытыми*. Во время сеанса связи одной из станций с точкой доступа, скрытая станция не обнаружив несущей, также пытается связаться с центральным узлом, в результате чего возникает коллизия. Для устранения

таких ситуаций в беспроводных сетях опционально предусмотрено использование алгоритма RTS/CTS (см. также п.1.5.3).

В соответствии с алгоритмом RTS/CTS каждый узел сети, перед тем как послать данные в «эфир», сначала отправляет специальный пакет **RTS** *(Request To Send*), что означает запрос источника на передачу данных. RTS-пакет содержит адрес отправителя и получателя, а также сведения о продолжительности предстоящей передачи. Точка доступа, получив пакет RTS, после истечения обязательной межкадровой фиксированной паузы SIFS<DIFS, отвечает посылкой пакета **CTS** (*Clear To Send*), свидетельствующего о готовности станции к приему информации и дающего команду остальным станциям прекратить передачу – "очистить эфир". В пакете содержится также уточненное время передачи. Это позволяет другим узлам задержать попытку занять среду на время, равное объявленной длительности сеанса связи. После этого передающая станция посылает пакет данных, а приемная станция должна передать кадр ACK, подтверждающий безошибочный прием. Последовательность отправки кадров между двумя узлами сети показана на рисунке 3.44.

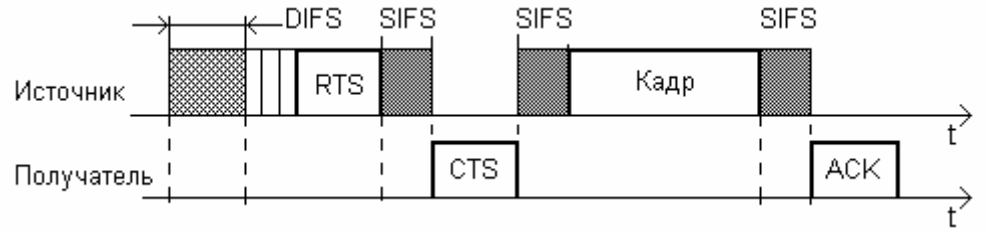


Рисунок 3.44 – Взаимодействие между двумя узлами сети в соответствии с алгоритмом RTS/CTS

Рассмотренный выше механизм распределенной координации DCF является базовым для протоколов 802.11 и может использоваться как в беспроводных сетях без базовой станции *Ad-HOC*, так и в сетях с базовой станцией типа *Infrastructure*. Однако для сетей Infrastructure более естественным является несколько иной механизм регламентирования коллективного доступа, известный как ***функция централизованной координации*** **PCF** (*Point Coordination Function*). Отметим, что механизм PCF является опциональным и применяется только в сетях с точкой доступа.

В случае использования механизма PCF один из узлов сети (точка доступа) является центральным и называется **центром координации** PC (*Point Coordinator*). На центр координации возлагается задача управления коллективным доступом всех остальных узлов сети к среде передачи данных на основе определенного алгоритма опроса (*поллинга*) или исходя из приоритетов узлов сети, т.е. центр координации опрашивает все узлы сети, внесенные в его список, и на основании этого опроса организует передачу данных между

всеми узлами сети. Такой подход полностью исключает конкурирующий доступ к среде, как в случае механизма DCF, и делает невозможным возникновение коллизий, а для времязависимых приложений гарантирует приоритетный доступ к среде. Таким образом, PCF может использоваться для организации приоритетного доступа к среде передачи данных.

Функция централизованной координации не отрицает функцию распределенной координации, а в ряде случаев дополняет. На практике в сетях с механизмом централизованной координации PCF реализуется как механизм PCF, так и традиционный механизм распределенной координации DCF. В течение определенного промежутка времени действует централизованная координация доступа PCF, затем – конкурентная DCF, а потом все повторяется заново.

Для того чтобы иметь возможность чередовать режимы PCF и DCF, необходимо, чтобы точка доступа, выполняющая функции центра координации и реализующая режим PCF, имела бы приоритетный доступ к среде передачи данных. Это можно сделать, если использовать конкурентный доступ к среде передачи данных (как и в методе DCF), но для центра координации разрешить использовать промежуток ожидания, меньший DIFS. В этом случае, если центр координации пытается получить доступ к среде, то он ожидает (как и все остальные узлы сети) окончания текущей передачи и, поскольку для него определяется минимальный режим ожидания после обнаружения «тишины» в эфире, первым получает доступ к среде. Промежуток ожидания, определяемый для центра координации, называется PIFS (PCF *Interframe Space*), причем SIFS<PIFS<DIFS.

В способе с чередованием механизмов режимы распределенной и централизованной координации объединяются в так называемый **суперфрейм**, который образуется из промежутка бесконкурентного доступа к среде, называемого **CFP** (*Contention-Free Period*), и следующего за ним промежутка конкурентного доступа к среде **CP** (*Contention Period*) (рисунок 3.45).

Рисунок 3.45 – Объединение режимов PCF и DCF в одном суперфрейме

Суперфрейм начинается с управляющего сигнального (маячного) кадра Beacon, получив который, все узлы сети приостанавливают попытки передавать данные на время, определяемое периодом бесконкурентного досту-

па CFP. Кадры-маячки несут служебную информацию о продолжительности CFP-промежутка и позволяют синхронизировать работу всех узлов сети. Во время режима бесконкурентного доступа точка доступа опрашивает все узлы сети о кадрах, которые стоят в очереди на передачу, посылая им служебные опросные (*поллинговые*) кадры CF_POLL. Опрашиваемые узлы в ответ на получение кадров CF_POLL посылают подтверждение CF_ACK. Если подтверждения не получено, то точка доступа переходит к опросу следующего узла.

Для осуществления передачи данных между всеми узлами сети, точка доступа может передавать кадр данных (DATA) и совмещать кадр опроса с передачей данных (кадр DATA+CF_POLL). Аналогично узлы сети могут совмещать кадры подтверждения с передачей данных DATA+CF_ACK (рисунок 3.46).

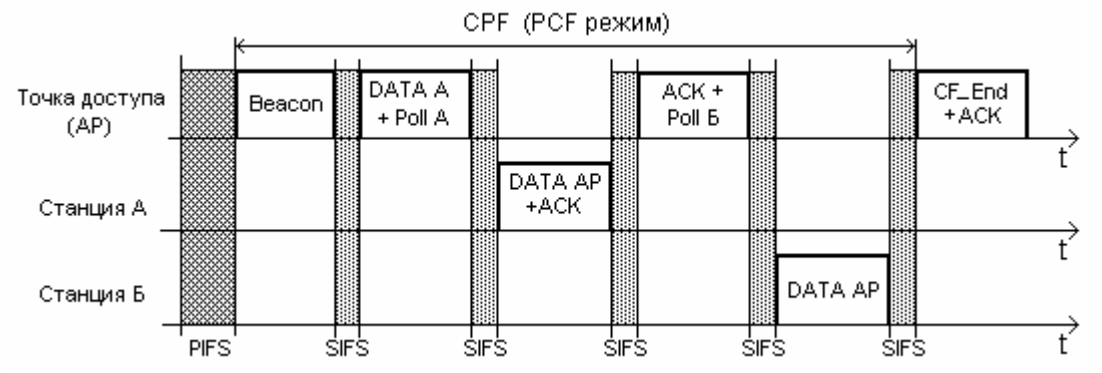


Рисунок 3.46 – Процедура совмещения кадров подтверждения с передачей данных

Во время режима PCF допускаются следующие типы кадров:

- DATA – кадр данных;
- CF_ACK – кадр подтверждения;
- CF_POLL – кадр опроса узла;
- DATA+CF_ACK – комбинированный кадр данных и подтверждения;
- DATA+CF_POLL – комбинированный кадр данных и опроса;
- DATA+CF_ACK+CF_POLL — комбинированный кадр данных, подтверждения и опроса;
- CF_ACK+CF_POLL – комбинированный кадр подтверждения и опроса узла.

3.8. Выводы по разделу

1. Расположение ЛКС на ограниченном пространстве позволяет использовать в этих сетях собственные линии связи на базе проводных и волоконно-оптических кабелей. За счет этого скорости передачи данных в ЛКС составляют от 4 до 10000 Мбит/с. Вероятность ошибки при передаче по таким линям на несколько порядков ниже, чем в глобальных сетях.

2. В локальных сетях преимущественно используются звездная, шинная кольцевая и звездно-иерархическая топология.

3. Особенностью ЛКС является коллективное использование линии связи, поэтому возникает проблема регулировать доступ отдельных компьютеров к моноканалу. Доступ может осуществляться состязательным или организационным способом. При состязательном способе возможно возникновение коллизий и нарушение функционирования всей сети.

4. При реализации состязательного доступа к среде передачи применяются метод множественного доступа с контролем передачи и обнаружением коллизий CSMA/CD. Для эффективного использования среды передачи ограничивается минимальная и максимальная длина кадра. Минимальная длина выбирается таким образом, чтобы при возникновении коллизии на максимальном удалении от передатчика, и достижении сигнала коллизии передающей станции, передача кадра не была бы завершена. Максимальная длина связана с допустимой задержкой ожидания станциями освобождения линии и размером буфера промежуточного хранения кадра. Состязательный доступ характерен для топологии "шина" и "звезда".

5. В сетях с организационным способом доступа право работы с каналом реализуется посредством посылки специального кадра разрешения – маркера. Станция, получившая маркер, может начинать передачу данных. После завершения передачи право на использование общей линии передается другому компьютеру. Этот способ применяется в шинной топологии (логическое кольцо) и в топологии физическое кольцо.

6. Из нескольких десятков типов систем проводных соединений в ЛКС лидируют два стандарта: 802.3 (*Ethernet*) – шинная и звездная топологии, и 802.5 (*Token Ring*) – физическое кольцо. В первом доступ к общей среде осуществляется состязательным способом CSMA/CD, а во втором регулирование доступа выполняется путем посылки в кольцо свободного маркера.

7. К классическим сетям Ethernet относятся сети 10BASE-5,-2,-T. Первые два типа используют для передачи сигналов «толстый» и «тонкий» коаксиальные кабели, при топологии «шина». В сетях 10BASE-T применяются кабели типа «витая пара» и звездная топология. Передача данных осуществляется сигналами с манчестерским линейным кодированием, скорость пере-

дачи данных равна 10 Мбит/с, максимальное расстояние между станциями составляет несколько сот метров.

8. Каждая станция при получении доступа передает один кадр, состоящий из преамбулы, стартового разделителя, адресов отправителя и получателя, поля с типом (или длиной) кадра, информационной части и контрольной последовательности кадра. По окончании передачи кадра все узлы сети должны выдержать технологическую межпакетную паузу длиной 9,6 микросекунд.

9. Передача данных в сети Token Ring осуществляется сигналами с манчестерским кодированием со скоростью 4 или 16 Мбит/с. Второе значение скорости обеспечивается при использовании процедуры раннего освобождения маркера и передаче по кольцу двух кадров. К важнейшим преимуществам сети Token Ring относится фиксированная задержка (а не случайная, как у Ethernet). Станция, передавшая кадр с данными, устанавливает маркер в занятое состояние и изменяет его на свободное только после возвращения кадра с отметкой о копировании его получателем.

10. Маркерный кадр сети Token Ring состоит из трех байтов: стартового разделителя, байта управления доступом и конечного разделителя. Обнаружение разделителей осуществляется по нарушению чередования полярностей передаваемых импульсов. Кадр данных содержит те же байты, что и маркер, к которым добавляются адресные поля получателя и отправителя, поле данных, контрольной последовательности кадра и статусного байта. Для управления работой кольца используется шесть типов управляющих кадров.

11. Сеть FDDI относится к высокоскоростным сетям (100 Мбит/с) и основывается на кольцевой технологии Token Ring. В качестве среды используется волоконно-оптическая линия длиной до 100 км. В процессе передачи производится логическое кодирование 4B/5B. Для обеспечения надежности сети она строится на основе двух оптоволоконных колец, образующих основной и резервный пути передачи. Такой способ построения позволяет при отказах отдельных участков или узлов сети замыкать кольцо по резервному пути и обеспечить функционирование сети.

12. Отличие метода доступа в FDDI от Token Ring состоит в том, что время удержания маркера на является постоянным, а зависит от загрузки кольца. Кроме этого, в формате кадра отсутствует поле приоритетов. Вместо него введено два класса трафика: асинхронный и синхронный.

13. Сеть Fast Ethernet обеспечивает обмен данными по электрическим и оптоволоконным кабелям со скоростью 100 Мбит/с. Все отличия сети от классической Ethernet сосредоточены на физическом уровне. В процессе передачи производится логическое кодирование 4B/5B или 8B/6T и линейное кодирование MLT-3 или NRZI. Существует три разновидности Fast Ethernet:

100BASE-TX, которая предназначена для передачи данных по двум витым парам кабеля 5-й категории длиной до 100 м; 100BASE-FX, где в качестве сегментов применяются два световода оптоволоконного кабеля длиной до 412 м; 100BASE-T4. в которой передача данных осуществляется по четырем витым парам неэкранированного телефонного кабеля UTP категории 3 длиной до 100 метров.

14. Сеть Fast Ethernet имеет иерархическую топологию, форматы кадров несколько отличаются от формата классической Ethernet. Сокращен межкадровый интервал. Не занятое состояние среды индицируется непрерывной передачей служебной комбинации "Idle" (11111). Физический интерфейс делится на подуровни согласования, независимый от среды интерфейс и устройство физического уровня. Для соединения со средой используется зависимый от среды интерфейс.

15. В сети Fast Ethernet поддерживается функция автовыбора взаимодействия, с помощью которой два взаимодействующих устройства физического уровня могут автоматически выбрать скорость передачи (10 или 100 Мбит/с) и режим работы.

16. Сети 100VG-Any LAN позволяют станциям обмениваться данными со скоростью 100 Мбит/с. Передача осуществляется по четырем парам UTP кабеля 3-й категории, по каждой паре со скоростью 25 Мбит/с. Поддерживаются кадры двух технологий Ethernet и Token Ring. Кадры в сети передаются не всем станциям, а только станции назначения. В сети имеется выделенный арбитр доступа – концентратор, который опрашивает порты. Если имеется запрос на передачу и сеть свободна, концентратор разрешает работу.

17. Сети класса Gigabit Ethernet включают сети со скоростями передачи 1 и 10 Гбит/с. Технологии имеется много общих признаков со своими предшественниками 10BASE и 100BASE: это – метод доступа к среде передачи данных CSMA/CD, полудуплексный и полнодуплексный режимы работы, а также форматы кадров Ethernet. Сети 1000BASE-FX ориентированы на применение 4-х витых пар категории 5 или выше (длиной до 100 м, с разъемом RJ-45) и оптоволоконных кабелей.

18. В сети 1000BASE-FX применяется логическое кодирование типа 8B/10B. Минимальная длина кадра составляет 512 байт, при коротких кадрах адаптер автоматически дополняет его до стандартной величины. Для повышения помехоустойчивости используется восьмипозиционное сверточное кодирование (на восемь различных состояний). Символы передаются по всем четырем витым парам кабеля одновременно с использованием пятиуровневого кодирования PAM-5 (–2; –1; 0; 1; 2). Передача ведется одновременно по 4-м парам кабеля, тактовая частота при этом снижается с 250 до 125 МГц.

19. Одним из важнейших параметров локальных сетей является произ-

водительность, измеряемая количеством кадров, передаваемых от источника к получателю за единицу времени. Другой не менее важный параметр - эффективная битовая скорость. Эффективная скорость возрастает с увеличением длины кадра и уменьшением загрузки сети.

20. Для объединения локальных сетей применяется следующее оборудование: повторители и концентраторы; сетевые мосты; коммутаторы; маршрутизаторы. Повторители объединяют сети на физическом уровне, осуществляя согласование электрических параметров сопрягаемых сетей и усиление сигналов. Концентратор представляет собой многопортовый Ethernet-повторитель, служащий в качестве центральной точки сети со звездообразной технологией, в которой соединяются кабели рабочих станций. Как и повторители, концентраторы функционирует на физическом уровне эталонной модели.

21. Мосты функционируют на канальном уровне и разделяют сеть на отдельные подсети, что позволяет изолировать трафик. Коммутатор представляет собой мультипроцессорный мост, способный независимо транслировать кадры между всеми парами своих портов. Маршрутизаторы работают на сетевом уровне, они могут переадресовывать и маршрутизировать *пакеты* через множество сетей, обмениваясь служебной информацией, зависящей от протокола между различными сетями.

22. В сетях Ethernet, построенных на основе повторителей или концентраторов, при возникновении столкновений сигналы коллизий проходят беспрепятственно через такие коммуникационные устройства и распространяются на всю сеть. Участок сети, состоящий из нескольких сегментов, в пределах которого распространяется коллизия, называется доменом коллизий. Расчет домена коллизий сводится к определению времени двойного оборота PDV и времени уменьшения межкадрового интервала IPG.

23. Сеть Ethernet устойчиво функционирует только в случае, если параметры домена коллизий не превышают допустимые значения для данной скорости передачи. С целью уменьшения размера домена коллизии применяют сегментацию сетей на основе мостов, коммутаторов и маршрутизаторов. Такие устройства изолируют сегменты сети, не допуская распространения коллизий в смежные сегменты.

24. В сетях Ethernet, коммутаторы поддерживают только древовидные связи, т.е. которые не содержат петель. Для автоматического решения проблемы зацикливания пакетов разработан "алгоритм покрывающего дерева" STA. Этот алгоритм обеспечивает построение древовидной топологии связей сети с единственным путем минимальной стоимости от каждого коммутатора и от каждого сегмента до некоторого выделенного корневого коммутатора – корня дерева.

25. В беспроводных сетях связь между компьютерами осуществляется

с помощью радиоволн. Беспроводные сети могут быть созданы двумя различными способами: сеть без базовой станции и сеть с точкой доступа. Точка доступа AP выполняет в беспроводной сети роль своеобразного концентратора. Связь отдельных сетей между собой выполняется путем соединения точек доступа сегментом кабельной сети либо радиомостом.

26. В беспроводных сетях при организации доступа используются два способа обслуживания: асинхронный и обслуживание с ограниченным временем. При асинхронном обслуживании каждая из станций может получить доступ к разделяемой среде, но при этом возможны коллизии. Способ ограниченного времени обслуживания гарантирует максимально допустимое время доступа станции к среде. В процессе реализации этого способа формируются суперкадры с двумя временными интервалами: один – бесконкурентный доступ, в течение которого точка доступа поочередно опрашивает станции и при их готовности осуществляет обмен кадрами, второй – состязательный доступ.

27. Для обеспечения совместимости работы беспроводных сетей их основные параметры оговариваются стандартами IEEE 802.11. Стандарты задают частотный диапазон излучаемых сигналов, способ модуляции, метод доступа и избежания коллизий, а также скорости передачи данных.

28. Принципы построения локальных компьютерных сетей, особенности настройки и управления более детально изложены в [2,10,11,22,23,27], моделирование сетей – в [21], а вопросы программирования сетевых коммутаторов и конфигурирования виртуальных локальных сетей – в [17].

3.9. Контрольные вопросы

1. В каких топологиях локальных компьютерных сетей и по какой причине возникают коллизии?
2. Как осуществляется доступ к среде в сетях с шинной топологией?
3. Из каких соображений рассчитывается минимальная длина кадра в сетях с CSMA/CD?
4. По каким признакам рабочие станции распознают коллизию?
5. Каким образом в сети с физической шиной можно регулировать доступ, чтобы избежать коллизий?
6. Каковы преимущества топологии "физическое кольцо"?
7. В чем состоит различие канальных подуровней управления логической передачей данных LLC и управления доступом к среде MAC?
8. Каковы отличия способов подключения к среде в "тонкой" и "толстой" сети Ethernet?
9. С какой целью введена межпакетная пауза в сетях *Ethernet,* и каковы ее

параметры?

10. Почему появились различия в форматах кадров *Ethernet* и как обеспечивается совместимость различных типов кадров?
11. Что определяет диаметр сети? Диаметр используемого кабеля или диаметр сегмента кабельного кольца?
12. Почему в результате внедрения сетей 10Base-T их топология из шинной превратилась в древовидную?
13. Из каких узлов состоит сетевой адаптер *Ethernet* и каковы их функции?
14. Как в сетях Token Ring сохраняется целостность кольца при выключении питания одной из рабочих станций?
15. Как отправитель сообщения сети Token Ring узнает, что кадр был успешно принят получателем?
16. Каким образом в сети FDDI обеспечивается высокая надежность передачи и живучесть сети?
17. Чем вызвано наличие трех вариантов реализации сети Fast Ethernet, и с какой целью в этих сетях применяется логическое кодирование 5B/6B?
18. Почему в одних вариантах сетей Fast Ethernet применяется логическое кодирование 5B/6B, а в других – 8B/6T?
19. Каким образом рабочие станции сети Fast Ethernet определяют возможность доступа к среде?
20. За счет чего в гигабитовых сетях Ethernet удалось обеспечить передачу данных со скоростью 10 Гбит/с по витой паре, полоса пропускания которой ограничивается несколькими сотнями МГц?
21. Какими мерами обеспечивается необходимая помехоустойчивость в гигабитовых компьютерных сетях?
22. Что представляет собой домен коллизий, как осуществляется расчет его параметров?
23. С какой целью и какими средствами выполняется сегментация локальных сетей?
24. За счет чего суммарная пропускная способность коммутаторов выше, чем у концентраторов?
25. Что представляет собой точка доступа в беспроводных сетях и каковы ее функции?
26. С какой целью в беспроводных цепях применяются при модуляции ССК-последовательности или многочастотная модуляция ортогональных сигналов OFDM?
27. Назовите параметры беспроводных компьютерных сетей, которые регламентируются международными стандартами.
28. Как в беспроводных сетях регулируется доступ станций к среде?

Раздел 4
КОМПЬЮТЕРНАЯ СЕТЬ ИНТЕРНЕТ

В чем состоит сложность объединения сетей? Почему стек протоколов TCP/IP отличается от эталонной модели OSI? Чем отличается байт от октета? На основании чего сетевой компьютер определяет, какую маску использовать для выделения номера сети? Зачем нужно преобразовывать IP-адреса в физические? Зачем нужно делить сети на подсети? Как отправитель узнает, что пакет не поступил получателю? С какой целью в сети Интернет ввели два транспортных протокола? Зачем протоколы маршрутизации делят на внутренние и внешние? Какое отношение к маршрутизации имеет политика? Если протоколы маршрутизации не выполняют маршрутизацию, то как же она осуществляется? Как реализованы сервисные службы в сети Интернет? На эти и другие вопросы Вы найдете ответ, изучив материал данного раздела.

4.1. Особенности функционирования объединенных сетей

4.1.1. Цель и проблемы объединения разнородных сетей

Совокупность независимых разнородных сетей, объединенных в единую сеть на основе протоколов TCP/IP, получила название Интернет (*Internetwork* или сокращенно *Internet*). Цель объединения разнородных сетей – создание унифицированной, согласованной и эффективной сетевой системы, которая позволяла бы более эффективно решать задачи на распределенных в пространстве компьютерах, обеспечивала обмен данными между любыми абонентами сети, а также поддерживала ряд универсальных сетевых служб. Осуществить объединение сетей простым их соединением не представляется возможным. Это связано с тем, что объединяемые компьютерные сети зачастую различаются используемыми операционными системами, типами компьютеров и коммуникационным оборудованием, форматами данных, способами кодирования и передачи, электрическими характеристиками.

При достижении этой цели решались также задачи сокрытия от пользователя особенностей низкоуровневой структуры объединенной сети, исключения зависимости взаимодействия пользователей от топологии объединенной сети, использования универсального набора идентификационных па-

раметров – имен и адресов.

Для соединения двух разнородных сетей необходимо использовать сетевой узел на основе специализированного компьютера, осуществляющего преобразование и согласование форматов и протоколов этих сетей, оснащенный двумя сетевыми адаптерами, каждый из которых включен в свою сеть. Очевидно, что параметры кодирования и модуляции, информационные и электрические характеристики каждого из адаптеров должны полностью отвечать параметрам, используемых в соответствующих сетях.

Коммуникационные компьютеры, связывающие две сети и выполняющие пересылку пакетов между ними, называются **межсетевыми шлюзами** (*gateways*) или **маршрутизаторами** (*internet routers*). Термин "шлюз" до сих пор достаточно часто используется в литературе о TCP/IP как синоним маршрутизатора. Однако в узком смысле слова он применяется для обозначения устройства, осуществляющего объединение сетей на **прикладном уровне**, т.е. объединяющего два различных семейства протоколов (например, TCP/IP и IBM SNA) в одном конкретном приложении (чаще всего это электронная почта или передача файлов). Маршрутизатор же осуществляет объединение на **сетевом уровне**. Схема объединения двух сетей с помощью маршрутизатора М изображена на рисунке 4.1.

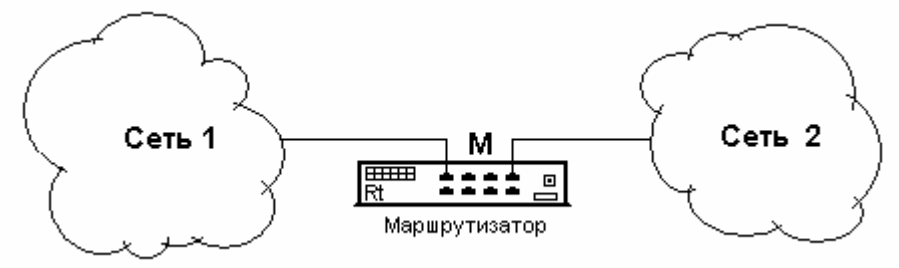


Рисунок 4.1 – Схема объединения разнородных сетей

Маршрутизаторы позволяют объединить сети, построенные на различных физических принципах: *Ethernet, Token Ring, Point-to-Point, FDDI (Fiber Distributed Data Interface),* и так далее. Сетевой механизм взаимодействия обеспечивает доставку небольших пакетов данных от отправителя прямо к получателю без использования промежуточных прикладных программ. Пересылка небольших фрагментов данных вместо огромных и громоздких файлов, как это делалось на прикладном уровне, обладает рядом преимуществ:

1) в обмене непосредственно участвует низкоуровневое сетевое аппаратное обеспечение, что делает его чрезвычайно эффективным;

2) взаимодействие на сетевом уровне позволяет отделить логику при-

ложения от механизма пересылки данных; поэтому компьютеры на промежуточных узлах могут оперировать трафиком, не вникая, для каких сетевых приложений он предназначен;

3) использование взаимодействия на сетевом уровне позволяет создать универсальную систему, предназначенную для выполнения практически любых видов компьютерной связи;

4) структура системы позволяет легко управлять сетью; например, при появлении новых сетевых технологий требуется только внести соответствующие изменения в программное обеспечение сетевого уровня, не затрагивая при этом самих прикладных программ.

Следует особо подчеркнуть, что семейство протоколов TCP/IP было разработано для обеспечения универсального взаимодействия между компьютерами, не зависящего от типов конкретных сетей, к которым эти компьютеры подключены. В связи с этим такая объединенная сеть с точки зрения пользователя выглядит как единая виртуальная сеть, к которой подключаются все компьютеры. При этом для пользователя детали конкретного физического подключения не имеют значения. Поэтому, чтобы облегчить конечному пользователю понимание механизма взаимодействия, объединенная сеть должна представляться в виде единой сети, а не совокупности отдельных сетей. Для реализации такого подхода недостаточно наличия маршрутизаторов, связывающих физические сети. Кроме этого, на каждом компьютере объединенной сети должно быть установлено также специальное программное обеспечение, с помощью которого прикладные программы могут использовать объединенную сеть так, как если бы это была одна физическая сеть.

На рисунке 4.2 показано объединение двух сетей: *Ethernet* и *Token Ring* с помощью маршрутизатора. Несмотря на то, что на нем показана связь только между двумя компьютерами, подсоединенными к маршрутизатору из разных сетей, каждый компьютер в *Ethernet* может общаться с любым компьютером в *Token Ring*.

На этом рисунке можно проследить разницу между конечной системой (*end system*) – два компьютера на каждой стороне, и промежуточной системой (*intermediate system*) – маршрутизатор в середине. Прикладной и транспортный уровни используют протоколы соответственно FTP и TCP, ориентированные на соединение (*end-to-end*). На рисунке эти два уровня применяются только конечными системами. Сетевой уровень, однако, работает с протоколом IP, не требующим соединения, так как применяется способ передачи типа "пересылка-за-пересылкой" (*hop-by-hop*). Протокол сетевого уровня IP используется в данном случае двумя конечными и каждой промежуточной системами.

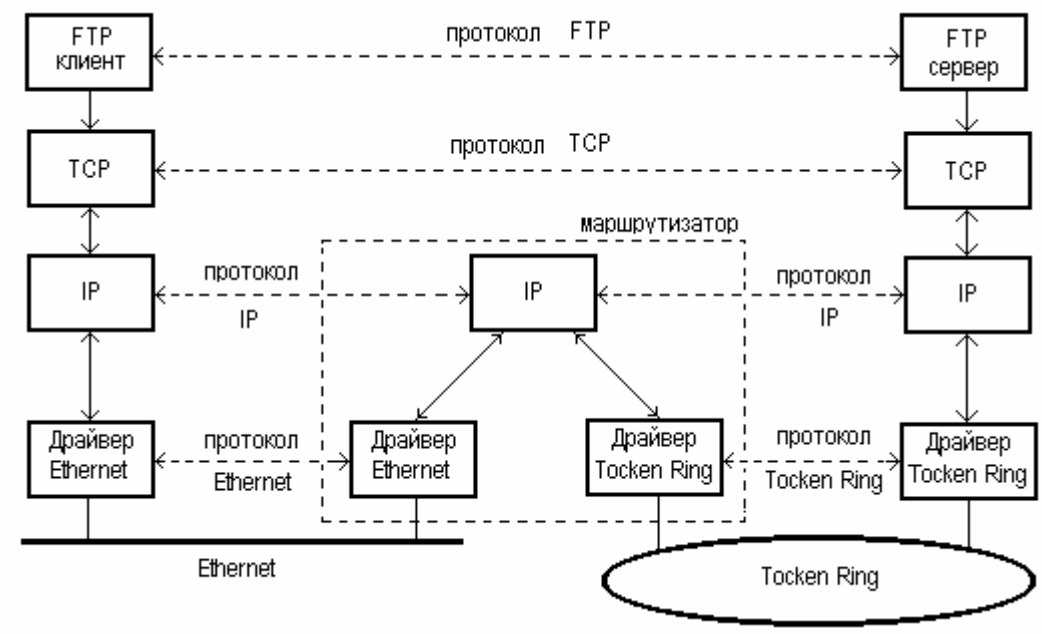


Рисунок 4.2 – Пример соединения двух сетей через маршрутизатор

4.1.2. Стек протоколов TCP/IP

Стек протоколов TCP/IP отличается от стека модели OSI (*Open System Integration*). Базируясь на классификации модели OSI, сопоставим всю архитектуру протоколов семейства TCP/IP с эталонной моделью. На рисунке 4.3 изображены основные программные модули, реализующие протокольные функции стека TCP/IP и их соответствие уровням модели OSI. Прямоугольниками на схеме обозначены модули, обрабатывающие пакеты, а линиями – пути передачи данных. **Модуль** представляет собой программу, взаимодействующую с драйвером сетевого адаптера, с сетевыми прикладными программами или с другими модулями. Схема приведена для случая подключения узла сети через локальную сеть *Ethernet*, поэтому названия блоков данных отражают эту специфику.

Сетевой интерфейс сети *Ethernet* (сокращенно Enet) – это физическое устройство, с помощью которого компьютер подключается к сети. В данном примере это сетевая карта (сетевой адаптер) *Ethernet*. Сетевой интерфейс передает (или принимает) данные в виде группы битов, которая носит название "*Кадр*" или "*Блок*".

Модуль IP обменивается с сетевым интерфейсом отрезками сообщения, получивших название "*IP-пакет*". На транспортном уровне единицей обмена является блок символов, который в зависимости от модуля обрабатывающего этот блок, называется *UDP-дейтаграмма* или *TCP-сегмент*.

Блок данных, которым обмениваются программы сетевых приложений с протоколами транспортного уровня, получил название "*Прикладное сообщение*".

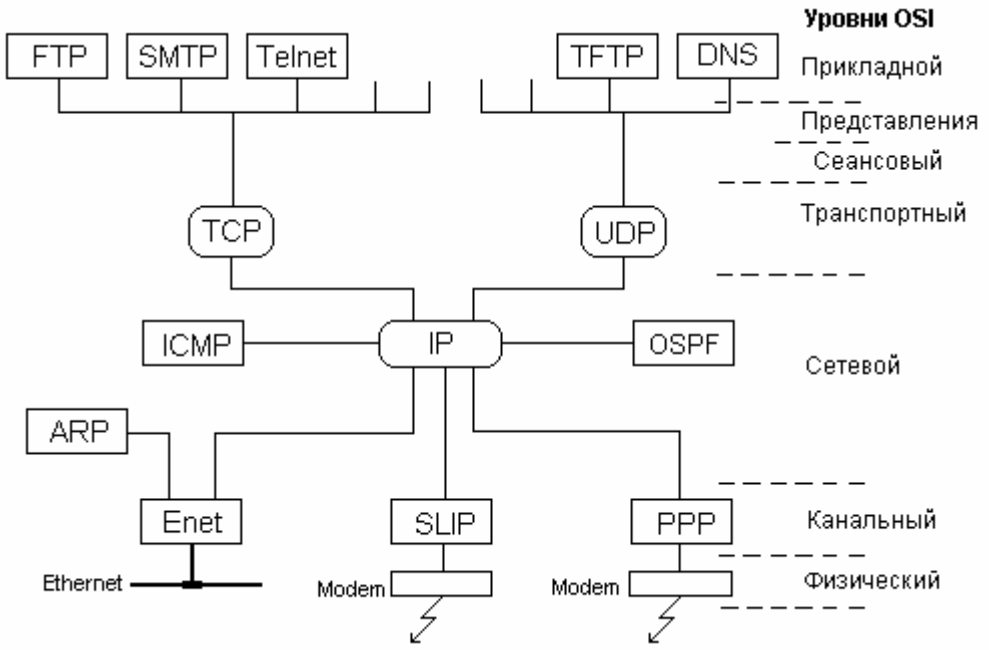


Рисунок 4.3 – Схема взаимодействия модулей, реализующих протокольные функции стека TCP/IP

На рисунке 4.3 сокращенно обозначены следующие протоколы. **ARP** (*Address Resolution Protocol*) – используется для определения соответствия IP-адресов и *Ethernet*-адресов. **SLIP** (*Serial Line Internet Protocol*) – протокол передачи данных по телефонным линиям общего пользования. **PPP** (*Point to Point Protocol*) – протокол обмена данными по схеме "точка-точка". **FTP** (*File Transfer Protocol*) – протокол обмена файлами. **Telnet** – протокол эмуляции виртуального терминала. **TFTP** (*Trivial File Transfer Protocol*) – упрощенный протокол передачи файлов. **DNS** (*Domain Name System*) – система доменных имен, обеспечивающая возможность определить по символическому Интернет-адресу его числовой эквивалент. **SMTP** (*Simple Mail Transfer Protocol*) – упрощенный протокол электронной почты. **ICMP** (*Internet Control Message Protocol*) – протокол межсетевых диагностических и управляющих сообщений. **OSPF** (*Open Shortest Path First*) – протокол маршрутизации.

Когда приложение посылает данные с использованием TCP, они передаются вниз по стеку протоколов, проходя через каждый уровень до тех пор, пока не будут отправлены в виде потока битов по сети. Каждый уровень добавляет свою управляющую информацию к данным путем пристыковки заголовков. Данные верхних уровней как бы обволакиваются заголовком ниж-

них (а иногда и окончанием). Такой процесс называют *инкапсуляцией* (рисунок 4.4). Числа, стоящие под заголовками и окончанием (завершителем) Ethernet, показывают стандартные размеры заголовков в байтах.

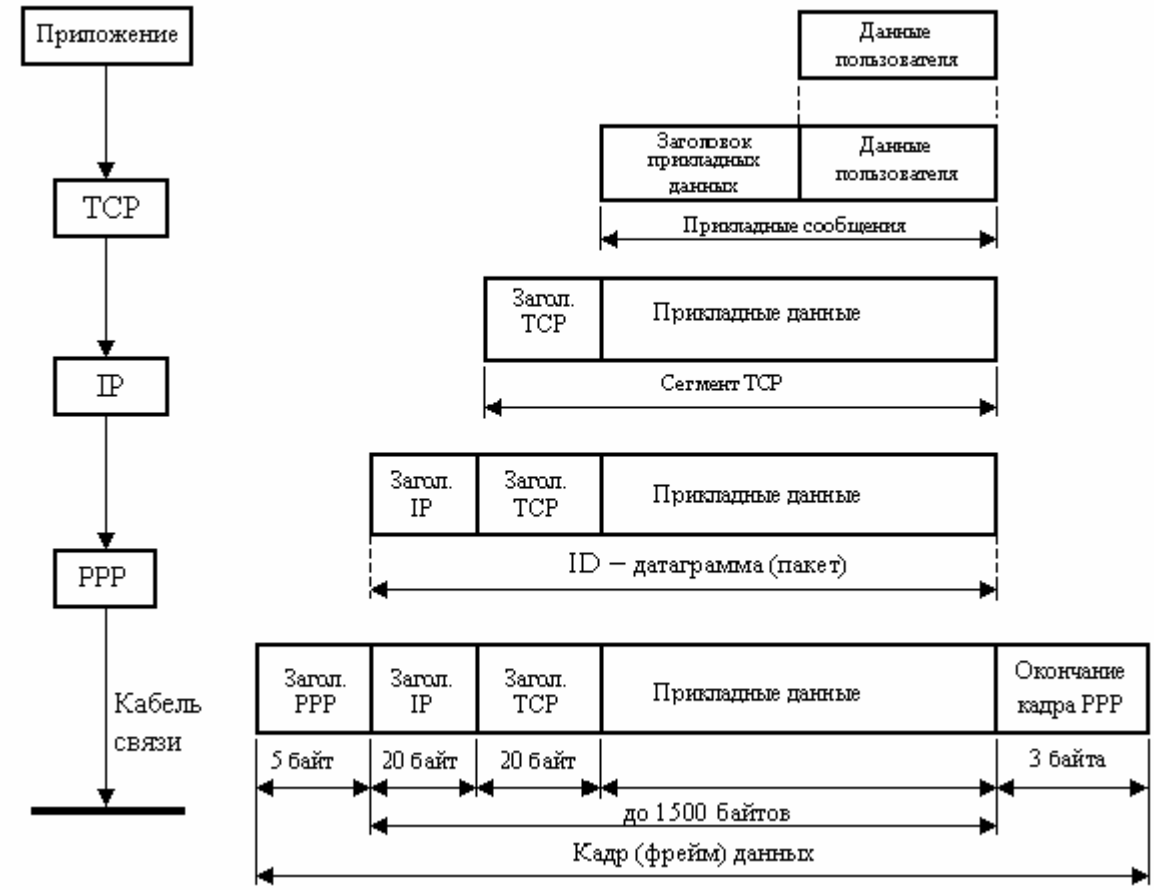


Рисунок 4.4 – Схема инкапсуляции данных

В случае передачи UDP-данных, процесс инкапсуляции выглядит почти аналогично. Различие заключается в том, что блок информации, который UDP передает в IP, называется UDP-дейтаграммой, а размер UDP-заголовка составляет всего 8 байт. В технической литературе и стандартах по Интернет для обозначения минимальной единицы данных используется термин *октет*. Этот термин появился в литературе в связи с тем, что реализация разработок по Интернету производились на вычислительной технике, подобной DEC-10, где байт отнюдь не равнялся восьми битам. Так как в большинстве современных компьютеров байт равен восьми битам, то термины байт и октет в последующем используются как синонимы.

Протоколы TCP, UDP, ICMP и OSPF посылают данные на сетевой уровень IP. Протокол IP должен добавить определенный идентификатор к IP-заголовку, который он формирует, чтобы указать какому уровню принад-

лежат данные. IP делает это путем записи номера протокола в восьмибитном поле своего заголовка, которое называется полем протокола. Значение номера протокола равно 1 для ICMP, 89 для OSPF, 6 для TCP и 17 для UDP. В связи с тем, что различные приложения могут использовать TCP или UDP в одно и то же время, протоколы транспортного уровня располагают в заголовке идентификатор приложения, которое их использует. В качестве такого идентификатора используется шестнадцатибитный *номер порта*. Протоколы TCP и UDP заносят номера портов источника и назначения в своих заголовках.

Сетевой интерфейс посылает и принимает кадры (*фреймы*), относящиеся к протоколам IP, ARP либо RARP. Для идентификации вышестоящего протокола в заголовке Ethernet выделено шестнадцатибитное поле типа фрейма. Когда фрейм *Ethernet* принимается приемником сетевой карты, он начинает свой путь вверх по стеку протоколов. При этом заголовки последовательно удаляются в процессе прохождения стека на соответствующих им уровнях. Каждый протокол просматривает определенные идентификаторы в заголовке, чтобы определить, какой следующий верхний уровень должен получить данные. Этот процесс называется демультиплексированием. Схема демультиплексирования изображена на рисунке 4.5.

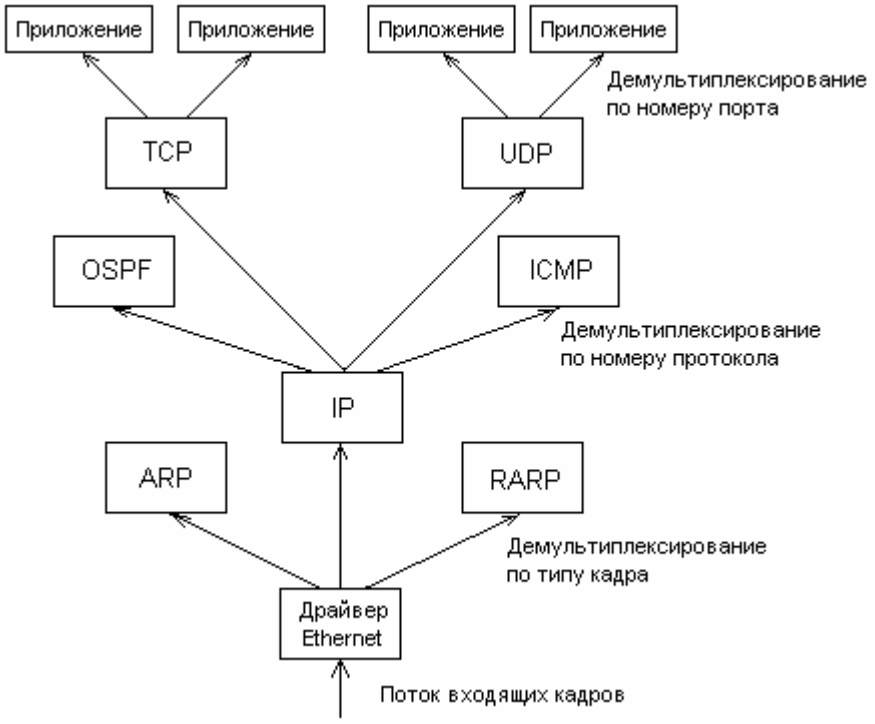


Рисунок 4.5 – Схема демультиплексирования пакетов

Для того чтобы обеспечить развитие Интернета и поддержку его в ра-

ботоспособном состоянии, организовано сообщество специалистов *ISOC* (*Internet Society*). В составе сообщества имеются группы, занимающиеся технической координацией и разработкой стандартов. Все официальные стандарты сообщества *Internet* публикуются в документах **RFC** (*Request for Comment*). Кроме того, существует множество RFC, которые не являются официальными стандартами, однако они публикуются с информационными целями. Все RFC имеют собственный номер. Они доступны бесплатно по электронной почте или через *Internet*.

4.1.3. Адресация в сети Интернет

Каждый компьютер в сети TCP/IP имеет адреса трех уровней.

Локальный адрес узла, определяется технологией, используемой для построения отдельной сети, в которую входит данный узел. Для узлов, входящих в локальные сети – это MAC-адрес сетевого адаптера или порта маршрутизатора, например, 11-A0-17-3D-BC-01. Эти адреса назначаются производителями оборудования и являются уникальными адресами, так как они присваиваются централизовано. Для всех существующих технологий локальных сетей MAC-адрес имеет формат, состоящий из 6 байтов: старшие 3 байта – идентификатор фирмы производителя, а младшие 3 байта назначаются уникальным образом самим производителем. Для узлов, входящих в глобальные сети, таких как X.25 или *Frame Relay*, локальный адрес назначается администратором глобальной сети.

IP-адрес – используется на сетевом уровне. Он назначается администратором сети во время конфигурирования компьютеров и маршрутизаторов. IP-адрес содержит две части: *номер сети* и *номер узла (хоста)*. **Номер сети** часто называют ***префиксом*** сетевого адреса, а номер хоста – ***суффиксом***. Номер сети может быть выбран администратором произвольно (если сеть функционирует автономно), либо назначен по рекомендации специального подразделения *Internet* (*Network Information Center*, NIC), если сеть должна работать как составная часть Интернета. Обычно провайдеры услуг сети Интернет получают диапазоны адресов у подразделений NIC, а затем распределяют их между своими абонентами. **Номер узла** (*сетевого компьютера*) в протоколе IP назначается независимо от его локального адреса. Деление IP-адреса на поле номера сети и номера узла – гибкое, и граница между этими полями может устанавливаться произвольно. Узел может входить в несколько IP-сетей. В этом случае узел должен иметь несколько IP-адресов, по числу сетевых связей. Таким образом, IP-адрес характеризует не отдельный компьютер или маршрутизатор, а одно сетевое соединение.

Символьный адрес, например, SERV1.IBM.COM. Этот адрес назна-

чается администратором и состоит из нескольких частей, например, имени компьютера, имени организации, имени домена. Такой адрес, называемый также DNS-именем, используется на прикладном уровне, например, в протоколах FTP или *Telnet*.

IP-адреса представляют собой 32-битовые идентификаторы, структура которых оптимизирована для решения основной задачи протокола – IP-маршрутиризации. Каждому сетевому интерфейсу присваивается уникальный IP-адрес. Основное преимущество от разделения IP-адреса на две части проявляется при рассмотрении процесса маршрутизации, и, в частности, размеров таблиц маршрутизации. При таком подходе у маршрутизатора появляется возможность хранить в таблице маршрутизации только один элемент записи для всех компьютеров одной сети, а не один элемент для каждого сетевого компьютера. Кроме того, при выполнении маршрутизации нужно проанализировать только сетевую часть IP-адреса.

Для удобства представления IP-адресов для пользователя применяется их цифровое написание, при котором адрес записывается, как десятичное представление 4-х байтов (октетов), разделенных точками, например:

192.171.153.60

В двоичной форме представления этот адрес будет выглядеть следующим образом:

11000000 10101011 10011001 00111100.

Каждый адрес можно представить в виде пары идентификаторов: идентификатор **сети** и идентификатор **хоста** (компьютера): **NetID, HostID.** Все IP-адреса разделены на 5 классов, однако, на практике применяется, в основном, три из них.

Для идентификации класса адреса используется неравномерный неприводимый двоичный код с кодовыми комбинациями 0, 10, 110 и 1110. Такое решение позволяет легко отделить идентификатор класса от остальной адресной части без дополнительных мер по синхронизации.

Класс А определен для сетей с числом хостов от 65535 до 16777215. В адресах этого класса 7 бит отведены под поле *идентификатора сети* **NetID**, а 24 – поле *идентификатора хоста* **HostID**. Первый бит является *идентификатором класса адреса,* он имеет значение **0**.

Класс В используется для среднемасштабных сетей, в которых содержится от 256 до 65535 хостов. Под поля NetID и HostID отводится соответственно 14 и 16 битов. Для идентификации класса адреса выделено два первых бита вида **10**.

Класс С применяется для сетей с числом компьютеров менее 256. Под HostID отведено 8 бит. Идентификатором класса адреса С яв-

ляются первые три бита со значениями **110**.

Класс D предназначен для отправки сообщений определенному множеству (группе) адресатов. Идентификатором класса адреса D служат первые четыре бита вида **1110**.

Класс E зарезервирован для будущих использований.

Все функции протокола IP исполняют хосты и маршрутизаторы. Следует иметь в виду и четко представлять, что IP-адрес идентифицирует сетевое соединение, а не хост. Поэтому, если хост переносится из одной подсети в другую, то ему следует обязательно изменить адрес. Компьютеры, подсоединенные к нескольким сетям, имеют столько же IP-адресов – по одному для каждого сетевого интерфейса.

Помимо адресов, предназначенных для *индивидуального хоста* (**unicast**), существуют *широковещательные* (**broadcast**) и *групповые* (**multicast**) адреса и др. **Широковещательные адреса** позволяют обращаться ко всем хостам сети. В них поля идентификации состоят только из единиц **FFFFFFFFh.** Механизм IP предоставляет возможность широковещательной передачи, но не гарантирует ее, если каждая физическая сеть не обеспечивает такой режим.

Групповые адреса применяются для передачи пакетов нескольким компьютерам. Они используются при проведении телеконференций, передачи почты и т. д. Некоторые из этих адресов зарезервированы для специальных групп, например: **224.0.0.1** – все узлы данной сети; **224.0.0.2** – все маршрутизаторы в данной сети; **224.0.0.5** – все OSPF-маршрутизаторы; **224.0.0.9** – маршрутизаторы RIP-2 и так далее.

Тестовый адрес имеет значение первого байта равное 127 (**01111111 | xxxx…x**), а оставшееся поле не специфицировано (обычно заполняется единицами). Он используется для отладки и тестирования, не является адресом никакой сети и не обрабатывается маршрутизатором.

В адресах класса A выделены две особые сети, их номера 0 и 127. Сеть 0 используется при маршрутизации как указание на маршрут по умолчанию. IP-интерфейс с адресом сети 127 применяется для адресации узлом самого себя (адрес типа loop back). Этот адрес имеет только локальное значение в пределах данного компьютера и может быть использован для проверки функционирования программных компонентов, которые применяются для реализации в данном устройстве стека протоколов TCP/IP. В качестве номера хоста обычно устанавливается значение "1", например 127.0.0.1. Обращение по этому адресу означает связь с самим собой (без выхода пакетов данных на уровень доступа к сети).

В объединенной IP-сети введено понятие "частная сеть". К этому виду относят сети, в которых для обеспечения информационного взаимодействия

не требуется обращения к глобальным ресурсам Интернет. К категории "частная сеть" относят также сети, узлы которых для обращения к информационным ресурсам глобальной сети применяют специальные шлюзы.

В соответствии с рекомендациями RFC 1918 для построения сетей, попадающих под определение "частная сеть", могут быть использованы следующие диапазоны адресов Интернет:

Класс	Начальный адрес	Конечный адрес	Число сетей
A	10.0.0.1	10.255.255.255	1
B	172.16.0.0.	172.31.255.255	16
C	192.168.0.0.	192.168.255.255	255

Таким образом, с учетом всех указанных выше ограничений и особенностей, реальное адресное пространство IP может быть представлено в виде

Класс	Начальный адрес	Конечный адрес
A	1.0.0.0	126.0.0.0
B	128.0.0.0	191.255.0.0
C	192.0.1.0	223.255.255.0
D	224.0.0.0	239.255.255.255
E	240.0.0.0	247.255.255.255

Для того чтобы обеспечить информационное взаимодействие между компонентами сети Интернет, необходимо установить соответствие между сетевыми адресами IP и аппаратными MAC-адресами. Эта функция в стеке протоколов TCP/IP возложена на протокол, который называется ARP (*Address Resolution Protocol*).

Уникальный IP-адрес назначается каждому сетевому интерфейсу специальной организацией *Internet Network Information Center* **(InterNIC)**, которая отвечает за выделение адресов сетям, объединенным в Internet. Назначение идентификаторов хостов не входит в компетенцию InterNIC и находится в ведении системного администратора компьютерной сети.

В связи с бурным ростом Интернет, 32-битовая адресация нынешней версии **IPv4** уже не удовлетворяет потребностям Мировой сети. Новая версия **IPv6** имеет 128-битовый формат IP-адреса и поддерживает автоматическое назначение адресов. Спецификация протокола IPv6 подробно рассмотрена ниже в данном разделе.

4.1.4. Преобразование адресов в IP-сетях

IP-адреса имеют какое-либо значение только в семействе протоколов TCP/IP. Любое устройство, подключенное к локальной сети (*Ethernet, FDDI* и т.д.), обладает только уникальным физическим адресом, заданным аппаратным образом. Так *Ethernet*-адрес сетевого устройства состоит из 6 байтов и назначается изготовителем сетевого интерфейсного оборудования из выделенного для него по лицензии адресного пространства. Если у компьютера меняется сетевой адаптер, то меняется и его аппаратный *Ethernet*-адрес. 4-байтовый IP-адрес задает администратор сети с учетом положения компьютера в сети Интернет. Если компьютер перемещается в другую часть сети Интернет, то его IP-адрес должен быть изменен.

Канальные уровни, такие как *Ethernet* или *Token Ring*, имеют собственную схему адресации, в которой применяются в основном 48-битные адреса. Когда фрейм Ethernet отправляется от одного хоста по локальной сети к другому, по его 48-битному Ethernet-адресу определяется, к какому интерфейсу он должен быть доставлен. Драйвер сетевой платы никогда не реагирует на IP-адрес назначения в IP-дейтаграмме.

Для обеспечения информационного взаимодействия на сетевом уровне необходимо задание однозначного соответствия **логического** (*сетевого*) и **физического** (*аппаратного, канального*) адресов взаимодействующих узлов. Поэтому возникает необходимость установить соответствие между двумя различными формами адресов: 32-битными IP адресами и каким-либо типом адресов канального уровня. Эта процедура реализуется **протоколом разрешения адресов ARP**.

Протокол ARP используется для установления значения физического адреса хоста по известному логическому адресу. Для решения обратной задачи (определения сетевого адреса для конкретной станции) предназначен протокол, который имеет название **RARP** (*Reverse ARP*). Оба эти протокола предполагают выполнение информационного обмена между узлами с использованием кадров одинакового типа.

Протокол ARP применяется в локальных сетях *Ethernet*. Преобразование адресов осуществляется только при отправления IP-пакетов, так как только в этот момент создаются заголовки IP и Ethernet. Для установления соответствия между IP- и Ethernet-адресами используется табличный способ потому, что нет никакой закономерной связи между этими адресами и нет какого-либо алгоритма для их вычисления. Если компьютер перемещается в другой сегмент сети, то его ARP-таблица должна быть изменена. Преобразование адресов осуществляется путем поиска в ARP-таблице, состоящей из двух столбцов (таблица 4.1). В первом столбце содержится IP-адрес, а во

втором – *Ethernet*-адрес.

Таблица 4.1 – Пример ARP-таблицы

IP-адрес	Ethernet-адрес
223.1.2.1	08:00:39:00:2F:C3
223.1.2.3	08:00:5A:21:A7:22
223.1.2.4	08:00:10:99:AC:54

Для определения физического адреса $A_ф$ абонента сети по известному его сетевому адресу A_{ca} станция B, желающая организовать информационное взаимодействие со станцией A, формирует специальный блок канального уровня – *кадр ARP*. В этот кадр наряду со служебной информацией помещается сетевой адрес искомой станции. Для того чтобы этот кадр мог достичь всех абонентов адресуемой сети, в качестве MAC-адреса назначения ARP-кадра используется широковещательный адрес. Сформированный таким образом кадр называется **ARP-запрос** (*ARP-request*). Этот кадр передается в сеть и принимается всеми станциями, подключенными к ней. Станции анализируют содержимое принятого запроса и одна из них, которая обнаружила в кадре принятого запроса свой сетевой адрес, формирует ответ на этот запрос (*ARP-reply*). В кадр ARP-*reply* станция помещает свой MAC-адрес и отправляет его в направлении источника запроса, используя при этом физический адрес станции отправителя.

Для того чтобы не запускать процедуру преобразования адресов всякий раз, когда потребуется организовать обмен с какой либо станцией, применяется аппарат кэширования результатов запросов – **ARP-cache** (буфер ARP). Интернет-адреса и соответствующие им аппаратные адреса содержатся в кэше модуля преобразования адресов, который создается в каждом сетевом компьютере.

Обычно поиск начинается в кэше ARP, и только в случае отсутствия там необходимой информации осуществляется обращение к таблице ARP. Эффективность функционирования ARP во многом определяется размером ARP-кэша. Стандартное время жизни записи в кэше, в зависимости от используемой операционной системы, составляет 10…20 минут с момента создания записи. После истечения этого времени она удаляется. Запись в таблице также удаляется, если она не востребована в первые две минуты после создания. Для просмотра содержимого таблицы на маршрутизаторе предусмотрена команда **show arp**, выполняемая в пользовательском режиме. Если требуется изменить время жизни записей в таблице, применяется команда **arp timeout**. Формат кадра протокола ARP показан на рисунке 4.6. Он со-

держит следующие поля.

"**Тип оборудования**" (*Hardware Type*). В этом поле располагается признак типа применяемого протокола канального уровня. Например, протоколу *Ethernet* значение данного поля соответствует 1, сети X.25 – 2, ATM – 16.

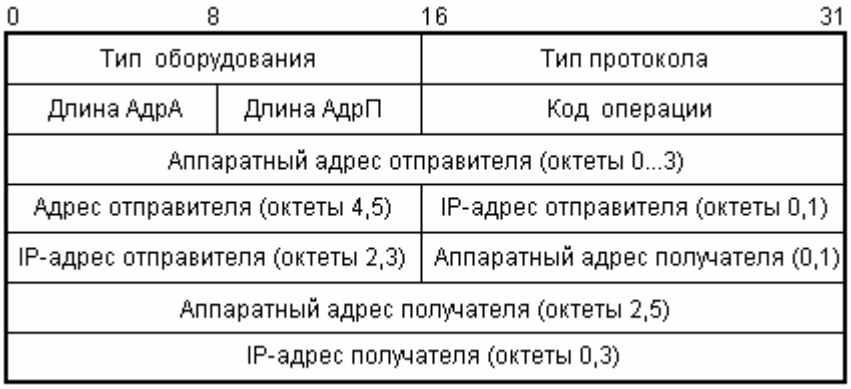


Рисунок 4.6 – Формат кадра протокола

"**Тип протокола**" (*Protocol Type*). В него помещается признак типа используемого протокола сетевого уровня. Например, для протокола IP в это поле помещается число 2048, для X.25 – 2053.

"**Длина АдрА и АдрП**" (HLEN и PLEN). Содержимое этих полей определяет размер адреса канального (аппаратного) и сетевого (протокольного) уровней соответственно. Наличие данных полей обеспечивает возможность использования протокола ARP для определения физического адреса в различных сетях второго и третьего уровней.

"**Код операции**" (*Operation*). В этом поле размещается признак типа информационного кадра: ARP *Request*; ARP *Response*; RARP *Request* или RARP *Response*.

"**Аппаратный адрес отправителя/получателя**" (*Sender/Target Hardware Address)* служат для размещения физических адресов передающей станции и станции назначения соответственно.

"**IP-Адрес сети отправителя/получателя**" (*IP Sender/Target Network Address*). В них располагаются сетевые адреса передающей станции и станции назначения соответственно.

Для выполнения функции, обратной действиям ARP разработан п**ротокол RARP** (*Reverse ARP*). Он предназначен для нахождения логического сетевого адреса узла сети по известному его MAC-адресу. Необходимость применения такого протокола возникает в случаях использования в локальной сети бездисковых рабочих станций. Поскольку специальных запоминающих устройств для хранения сетевого адреса на бездисковой рабочей станции нет и быть не может, то этот адрес должен быть присвоен ей дина-

мически. Динамическое присвоение сетевого адреса бездисковым рабочим станциям выполняет протокол RARP.

Для того чтобы кадр RARP мог достичь всех абонентов своей сети, в качестве MAC-адреса назначения этого блока используется широковещательный адрес (*broadcast*). Сформированный таким образом кадр называется RARP-запрос (*RARP-request*). На кадр данного типа может ответить только устройство, выполняющее функцию RARP-сервера.

Функцию RARP-сервера в сети выполняет специальный компьютер, устанавливающий соответствие между физическим и сетевым адресами станций. Обычно это соответствие устанавливается при помощи специальных динамических таблиц, в которых каждой станции по её физическому адресу ставится в соответствие логический – сетевой адрес. Таким образом, информационное взаимодействие при выполнении протокола RARP состоит из следующих этапов:

- получение RARP-запроса от рабочей станции;
- определение значения MAC-адреса;
- нахождение по таблице значение сетевого адреса;
- формирование ответного кадра RARP-reply.

Для обеспечения достаточной надежности функционирования службы определения соответствия адресов целесообразно применять резервирование RARP-сервера. Простое резервирование, например, дублирование этих устройств, может привести к возникновению дополнительных трудностей. К таким трудностям, в частности, относится возможность возникновения коллизий при одновременном ответе на RARP-запрос двумя RARP-серверами. Для разрешения этой проблемы проводится ранжирование серверов на первичный и вторичные.

4.1.5. Структуризация IP-сетей с помощью масок

Часто администраторы сетей испытывают неудобства из-за малого количества централизованно выделенных им номеров сетей, которых недостаточно для разделения на части (***структурирования***) сети надлежащим образом, например, разместить все редко взаимодействующие компьютеры по разным сетям. В такой ситуации возможны два пути. Первый из них связан с получением дополнительных номеров сетей. Второй способ, употребляющийся более часто, связан с использованием так называемых сетевых *масок*, позволяющих разделять одну сеть на несколько подсетей. *Подсеть* представляет собой подмножество сети, не пересекающееся с другими подсетями. Это означает, что сеть организации (например, сеть класса C) может

быть разбита на части, каждая из которых представляет собой подсеть. На практике каждая подсеть обычно соответствует физической локальной сети (например, сегменту *Ethernet*).

Для указания маршрутизатору, какие биты используются для идентификации сети и подсети, применяется **процедура маскирования**. Маска представляет собой число, двоичная запись которого содержит единицы в тех разрядах, которые должны интерпретироваться как номер сети, в остальных разрядах содержатся нули. Например, для стандартных классов сетей маски имеют следующие значения:

255.0.0.0 – для сети класса A,
255.255.0.0 – для сети класса B,
255.255.255.0 – для сети класса C.

Разбиение сети на подсети использует часть IP-адреса, закрепленную за номерами хостов. В масках, применяемых администратором для увеличения числа сетей, количество единиц в последовательности, определяющей границу номера сети, не обязательно должно быть кратным 8, чтобы повторять деление адреса на байты.

Рассмотрим пример, в котором маска сети имеет значение 255.255.192.0 (11111111 11111111 11000000 00000000). Пусть сеть имеет номер 129.44.0.0 (10000001 00101100 00000000 00000000), из которого видно, что она относится к классу B. После наложения данной маски на этот адрес число разрядов, интерпретируемых как номер сети, увеличилось с 16 до 18, т.е. администратор получил возможность использовать вместо одного, централизованно заданного ему номера сети, четыре:

129.44.0.0 (10000001 00101100 00000000 00000000)
129.44.64.0 (10000001 00101100 01000000 00000000)
129.44.128.0 (10000001 00101100 10000000 00000000)
129.44.192.0 (10000001 00101100 11000000 00000000) .

Например, IP-адрес 129.44.141.15 (10000001 00101100 10001101 00001111), который задает номер сети 129.44.0.0 и номер узла 0.0.141.15, при использовании маски будет интерпретироваться как пара:
129.44.128.0 – номер сети, 0.0.13.15 – номер узла.

Таким образом, установив новое значение маски, можно заставить маршрутизатор по-другому интерпретировать IP-адрес. При этом два дополнительных последних бита номера сети часто интерпретируются как номера подсетей.

Рассмотрим еще один пример. Пусть некоторая сеть относится к клас-

су В и имеет адрес 147.30.0.0 (рисунок 4.7).

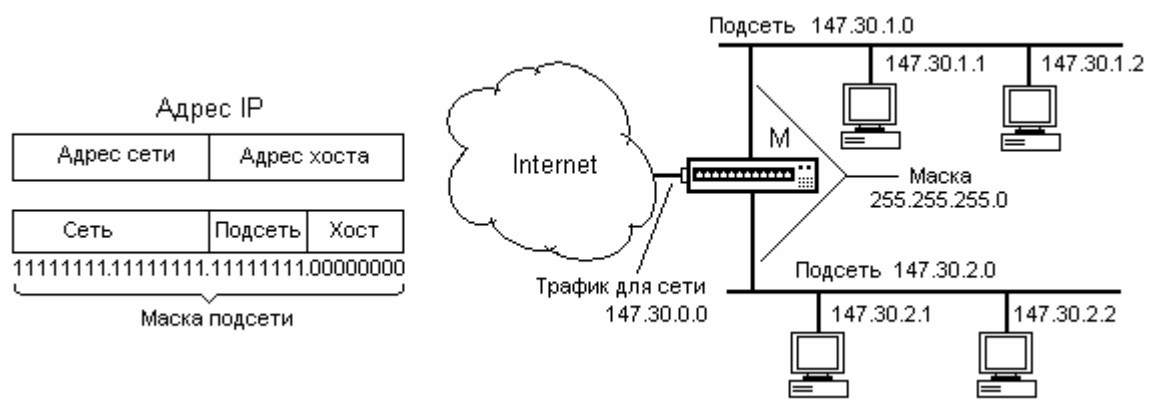


Рисунок 4.7 – Пример использования масок для структурирования сети

Этот адрес используется маршрутизатором, соединяющим сеть с остальной частью объединенной сети. И пусть среди всех станций данной сети есть станции, редко взаимодействующие между собой. Их целесообразно разместить в разных сетях. Для этого сеть следует разделить на две подсети, подключив их к соответствующим портам маршрутизатора, и задать для этих портов в качестве маски, например, число 255.255.255.0. Таким образом, внутри исходной сети создано две подсети с централизовано заданным номером класса С (можно было бы выбрать и другой размер для поля адреса подсети). Извне сеть по-прежнему выглядит, как единая сеть класса В, а на местном уровне это – две отдельные сети класса С. Приходящий общий трафик разделяется местным маршрутизатором М между подсетями. Необходимо заметить, что, если принимается решение об использовании механизма масок, то соответствующим образом должны быть сконфигурированы и маршрутизаторы, и компьютеры сети.

К сожалению, подсети не только решают, но также и создают ряд проблем. Например, невозможно использовать часть адресов из-за самого принципа построения адресов подсети. Так, например, выделение трех битов на адрес подсети не приводит к образованию 8-ми подсетей. Образуется только 6 подсетей в связи с тем, что номера сетей 0 и 7 использовать в силу специального значения IP-адресов, состоящих только из всех нулей и единиц, нельзя. Таким образом, все комбинации адресов хоста внутри подсети, которые можно было бы связать с этими номерами, использовать невозможно. Чем шире маска подсети (чем больше места отводится на адрес хоста), тем больше потерь. В ряде случаев приходится выбирать между приобретением адреса еще одной сети или изменением маски.

4.2. Межсетевые протоколы IP и IPv6

4.2.1. Межсетевой протокол IP

Межсетевой протокол предназначен для использования в системе связанных между собой маршрутизаторами (*межсетевыми шлюзами*) сетей коммутации пакетов. Основным назначением IP-протокола является передача от источников к адресатам блоков данных, называемых **межсетевыми дейтаграммами** (МД). IP обеспечивает определение пути следования дейтаграмм, фрагментацию их на передающей стороне и сборку фрагментов на узле получателя при необходимости передачи длинных дейтаграмм через подсети с малой допустимой длиной пакетов.

IP является протоколом *без установления соединения, не гарантирующим* доставку пакетов. Это означает, что протокол IP не предусматривает обмен служебными сообщениями, подтверждающими готовность узла к приему, установление соединения, подтверждения правильности приема данных и их целостность. Протокол IP обрабатывает каждую дейтаграмму как независимую единицу, не имеющую связи ни с какими другими дейтаграммами в сети. После отправки дейтаграммы в сеть ее дальнейшее продвижение никак не контролируется отправителем (на уровне протокола IP). Если дейтаграмма не может быть доставлена получателю, она удаляется из сети. Узел, уничтоживший дейтаграмму, может оправить по обратному адресу *ICMP-сообщение* о причине сбоя.

Программный модуль, реализующий IP-протокол, функционирует в каждом из компьютеров, включенных в сеть, а также в каждом маршрутизаторе, соединяющим отдельные подсети. Эти модули работают по общим правилам интерпретации поля межсетевого адреса, фрагментации и сборки межсетевых дейтаграмм.

Обобщенный алгоритм работы модуля IP на каком-либо узле сети, принимающего дейтаграмму из сети, сводится к следующим действиям.

1. Прием с одного из интерфейсов уровня доступа к среде (например, с Ethernet-интерфейса) IP-дейтаграммы.
2. Анализ заголовка дейтаграммы.
3. Если пунктом назначения дейтаграммы является данный компьютер, то:

3.1. Если дейтаграмма является фрагментом большей дейтаграммы, принимаются остальные фрагменты, после чего из них собирается исходная дейтаграмма.

3.2. Из дейтаграммы извлекаются данные, определяется номер протокола вышележащего уровня и данные направляются на обработку этому протоколу.

4. Если дейтаграмма не направлена ни на один из IP-адресов данного узла, то дальнейшие действия зависят от дополнительных условий.

4.1. Если ретрансляция разрешена, то определяется следующий узел сети и интерфейс нижнего уровня для доставки дейтаграммы по назначению.

4.2. Передача дейтаграммы на нижний уровень этому интерфейсу для отправки. При необходимости может быть произведена фрагментация дейтаграммы.

5. Если в дейтаграмме обнаружена ошибка или по каким-либо причинам она не может быть доставлена, дейтаграмма уничтожается.

6. Посылка отправителю дейтаграммы ICMP-сообщения об ошибке.

При получении данных от вышестоящего уровня для отправки их по сети IP-модуль формирует дейтаграмму с этими данными, в заголовок которой заносятся адреса отправителя и получателя, полученные от транспортного уровня, и другая информация; после чего выполняются следующие шаги.

1. Если дейтаграмма предназначена этому же узлу, из нее извлекаются данные и направляются на обработку одному из протоколов транспортного уровня (тип протокола указывается в заголовке дейтаграммы).

2. Если дейтаграмма не направлена ни на один из IP-адресов данного узла, то определяются следующий узел сети, на который должна быть переправлена дейтаграмма для доставки ее по назначению, и интерфейс нижнего уровня, после чего дейтаграмма передается на нижний уровень этому интерфейсу для отправки; при необходимости может быть произведена фрагментация дейтаграммы.

3. Если же дейтаграмма ошибочна или по каким-либо причинам не может быть доставлена, она уничтожается.

Формат заголовка IP-дейтаграммы (межсетевой дейтаграммы) показан на рисунке 4.8. Заголовок состоит из 32-разрядных слов и имеет переменную длину, зависящую от размера поля "Опции", но всегда кратную 32 битам. Поля заголовка имеют следующие значения. "**Версия**" указывает версию протокола (для IP равно 0100). "**ДлЗГЛ**" – Длина заголовка в 32-битовых словах. Минимальное значение – 5 (20 байтов), а максимальное – 15 (60 байтов).

"**Тип сервиса**" индицирует степень приоритета пакета (3 бита), а также включает указатели: **D** – задержка, **T** – производительность, надежность **R** (по одному биту на каждый указатель: 0 – нормальная, 1 – низкая задержка, высокая производительность, высокая надежность). "**Полная длина**" указывает длину дейтаграммы в байтах (октетах), включая межсетевой заголовок и данные. Номинальная длина межсетевой дейтаграммы составляет

576 байтов (512 данные + 64 заголовок), она передается целиком или фрагментами. Минимальная длина заголовка 20, а максимальная – 60 байт; 4 байта зарезервировано для заголовков протоколов высших уровней.

"**Идентификатор**" формируется отправителем и служит для сборки фрагментов дейтаграммы.

"**Флаги**" (3 бита), 0 – резервный; **DF** =:0 – данную дейтаграмму можно фрагментировать, 1– нельзя; **MF**=0 – последний фрагмент, 1 – промежуточный.

Рисунок 4.8 – Формат заголовка межсетевой дейтаграммы

"**Указатель фрагмента**" указывает место данного фрагмента в межсетевой дейтаграмме. Измеряется в байтах. Первый фрагмент имеет нулевое смещение.

"**Время жизни**" (**TTL**, *Time To Live*) определяет максимальное время (в секундах), в течение которого дейтаграмма может оставаться в объединенной сети. Номинальное время 30 с. Поле изменяется при обработке межсетевого заголовка. Если время обработки меньше 1с, то все равно оно уменьшится на 1. Если это поле достигает нулевого значения, то дейтаграмма уничтожается. При этом отправителю может быть послано соответствующее ICMP-сообщение (см. п. 4.5).

"**Протокол**" – указывает протокол ближайшего верхнего уровня (TCP или UDP). Для TCP значение идентификатора равно 6, а для UDP –17.

"**Контрольная сумма заголовка**" занимает два байта и рассчитывается только по заголовку. Поскольку некоторые поля заголовка меняют свое значение в процессе передачи по сети (например, TTL) контрольная сумма проверяется и повторно рассчитывается каждым узлом. При обнаружении ошибки дейтаграмма уничтожается.

"**IP-адрес отправителя и получателя**" содержит стандартные 32-битовые IP-адреса.

"**Поле дополнительных услуг – IP опции**" имеет переменную длину и может отсутствовать в дейтаграмме. Опции служат для повышения безопасности передачи, управления маршрутизацией, фрагментацией, для тестирования и отладки сетей. Опция состоит как минимум из октета "Тип опции", за которым могут следовать октеты "Длина опции" и байты сообщения опции. Это сообщение может принимать значения: "*Степень безопасности*", "*Свободная источниковая маршрутизация*", "*Запись маршрута*", "*Конец списка дополнительных услуг*" и т.д. Маршрутная информация состоит из 32 битов. Если указатель маршрутизации больше, чем дополнительная длина поля, то маршрутизация производится на основании только адреса получателя. "**Заполнитель**" нужен для того, чтобы заголовок заканчивался, а данные начинались на границе 32-разрядного слова. В случае использования, заполняется нулями, имеет переменную длину.

В связи с бурным ростом Интернет 32-битовая адресация нынешней версии IPv4 уже не удовлетворяет потребностям Мировой сети. Для устранения этого недостатка разработана новая версия межсетевого протокола IPv6. Подробнее об этой версии протокола будет сказано ниже.

4.2.2. Фрагментация IP-пакетов

В IP-дейтаграмме под поле, содержащее размер пакета, разработчиками было выделено 16 битов. Таким образом, максимальная длина дейтаграммы не может превышать 65535 октетов. Поэтому протоколы транспортного уровня (TCP или UDP), пользующиеся сетевым уровнем для отправки пакетов, считают, что максимальный размер поля данных IP-пакета равен 65535. В связи с этим они могут передать IP-уровню сообщение такой же длины для транспортировки его через объединенную сеть. Однако на практике на размер дейтаграмм налагаются существенные ограничения. Как известно, передача дейтаграмм от одного компьютера к другому всегда выполняется средствами физической сети. Формат и размер кадров канального уровня зависит от вида физической сети и типа используемого оборудования.

В любой сети с коммутацией пакетов существует ограничение на максимальный размер данных, которые могут быть переданы в одном физическом кадре (*фрейме*). Например, в технологии *Ethernet* это значение равняется 1500 октетам, а в технологии FDDI в одном фрейме может находиться 4096 октетов данных, а сети X.25 чаще всего работают с MTU в 128 байтов. Для обозначения подобных ограничений используют специальный термин –

максимальная единица передачи данных в сети, сокращенно **MTU** *(Maximum Transfer Unit)*. Величина MTU может быть достаточно малой. В некоторых сетях значение MTU составляет 128 октетов и даже меньше.

Таким образом, ограничение длины дейтаграммы размерами минимального MTU, существующего в объединенной сети, приводит к падению эффективной скорости передачи данных по причине увеличения избыточности кадра. В то же время, если размер дейтаграммы будет больше, чем размер минимального MTU в объединенной сети, это приведет к тому, что на некоторых участках сети она может не поместиться в один канальный кадр.

Поэтому в функции уровня IP входит разбиение слишком длинного для конкретного типа составляющей сети сообщения на более короткие пакеты. Части, на которые разделяется дейтаграмма, называются *фрагментами,* а сам процесс разделения дейтаграммы – ***фрагментацией***.

Размер фрагмента выбирается так, чтобы он целиком помещался в одном канальном кадре. Кроме того, так как в протоколе IP величина смещения фрагмента от начала заголовка кратна 8 байтам, размер фрагмента также выбирается кратным 8 байтам. Очевидно, что выбор фрагмента, размер которого приблизительно равен MTU участка сети и при этом кратен 8 октетам, приводит к делению дейтаграмм на неравные части. Размер последней части может оказаться намного меньше остальных.

В протоколе IP нет каких-либо рекомендаций по поводу минимального размера дейтаграммы. Поэтому пользователь может выбирать размер дейтаграммы по своему усмотрению. Ее фрагментация и последующая сборка выполняются автоматически на уровне протокола IP, без какого-либо вмешательства отправляющей стороны.

Перед тем, как узел-получатель сможет обработать фрагментированную дейтаграмму, он должен принять все ее части и произвести *сборку* (т.е. восстановить ее первоначальный вид). В связи с тем, что отдельные фрагменты перемещается по сети независимо от других, в заголовке каждого пакета должна быть информация, дублирующая заголовок исходной дейтаграммы. Кроме того, в нем предусматриваются служебные поля, необходимые для сборки фрагментов в исходное сообщение. Размер фрагмента вместе с заголовком не должен превышать размера MTU той сети, по которой пересылается фрагмент.

Обратите внимание, что в заголовке дейтаграммы предусмотрены три поля, управляющие процессом ее фрагментации и последующей сборки: *идентификационные данные, флажки* и *смещение фрагмента*. В поле идентификационных данных указывается уникальное целое число, предназначенное для отождествления текущей дейтаграммы.

После доставки получателю первого фрагмента дейтаграммы запускается специальный *таймер сборки (reassembly timer)*. Если значение таймера

истекает до того, как получены все фрагменты дейтаграммы, получатель не обрабатывает ее и удаляет полученные фрагменты, т.е. фрагментация существенно повышает вероятность потери целой дейтаграммы, поскольку потеря любого фрагмента автоматически означает потерю всей дейтаграммы. Работа протокола IP по фрагментации пакетов в хостах и маршрутизаторах иллюстрируется рисунком 4.9. Здесь К1 и Ф1 – канальный и физический уровни сети 1; К2 и Ф2 – канальный и физический уровни сети 2.

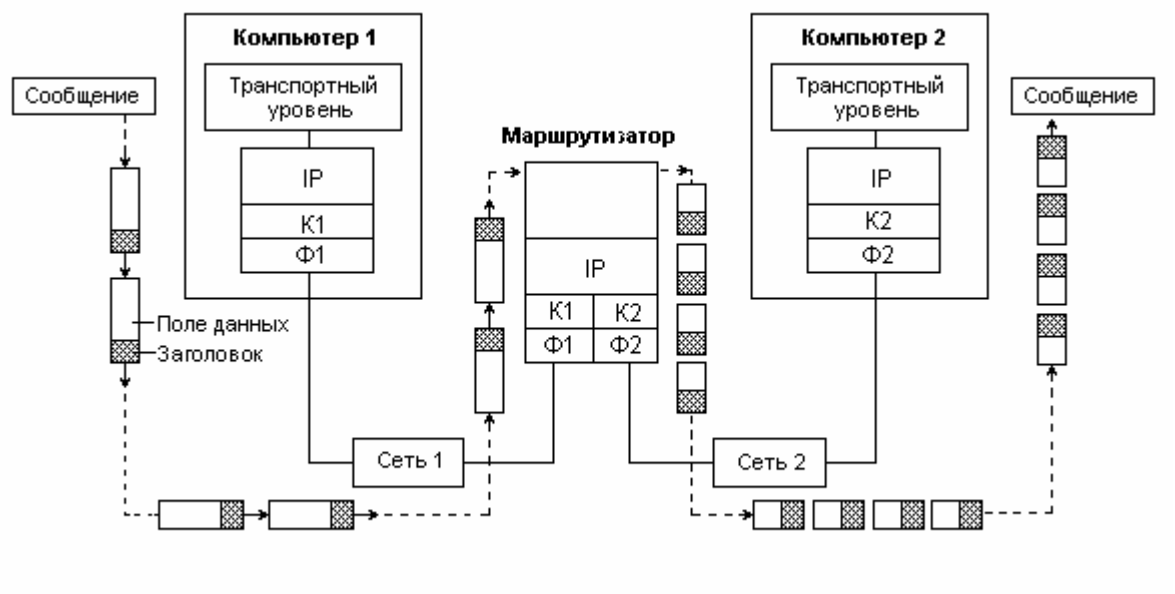


Рисунок 4.9 – Фрагментация IP-пакетов при передаче между сетями с разными максимальными размерами пакетов

Пусть компьютер 1 связан с сетью, имеющей значение MTU в 4096 байтов, например, с сетью FDDI. При поступлении на IP-уровень компьютера 1 сообщения от транспортного уровня размером в 5600 байтов, протокол IP делит его на два IP-пакета, устанавливая в первом пакете признак фрагментации и присваивая пакету уникальный идентификатор, например, 486. В первом пакете величина поля смещения равна 0, а во втором – 2800. Признак фрагментации во втором пакете равен нулю, последнее показывает, что это – завершающий фрагмент пакета. Общая величина IP-пакета составляет 2800+20 (размер заголовка IP), т.е. 2820 байтов, что умещается в поле данных кадра FDDI.

Далее компьютер 1 передает эти пакеты на канальный уровень К1, а затем и на физический уровень Ф1, который отправляет их маршрутизатору, связанному с данной сетью.

Маршрутизатор определяет по сетевому адресу, что прибывшие два пакета нужно передать в сеть 2, которая имеет меньшее значение MTU, равное 1500. Вероятно, это – сеть Ethernet. Маршрутизатор извлекает фрагмент

транспортного сообщения из каждого пакета FDDI и делит его еще пополам, чтобы каждая часть уместилась в поле данных кадра Ethernet. Затем он формирует новые IP-пакеты, каждый из которых имеет длину 1400 + 20 = 1420 байтов, что меньше 1500 байтов, поэтому они нормально помещаются в поле данных кадров Ethernet.

В результате в компьютер 2 по сети Ethernet приходит четыре IP-пакета с общим идентификатором 486, что позволяет протоколу IP, работающему в компьютере 2, правильно собрать исходное сообщение. Если пакеты пришли не в том порядке, в котором были посланы, то смещение укажет правильный порядок их объединения.

Отметим, что IP-маршрутизаторы не собирают фрагменты пакетов в более крупные единицы, даже если на пути встречается сеть, допускающая такое укрупнение. Это связано с тем, что отдельные фрагменты сообщения могут перемещаться в объединенной IP-сети по различным маршрутам, поэтому нет гарантии, что все фрагменты проходят через какой-либо промежуточный маршрутизатор на их пути.

При приходе первого фрагмента пакета узел назначения запускает таймер, определяющий максимально допустимое время ожидания прихода остальных фрагментов этого пакета. Если таймер истекает раньше прибытия последнего фрагмента, то все полученные к этому моменту фрагменты пакета отбрасываются, а в узел, пославший исходный пакет, направляется сообщение об ошибке с помощью протокола ICMP.

4.2.3. Межсетевой протокол IPv6

Изменения, внесенные в протокол IPv6 по сравнению с IPv4, сводятся к следующему:

1) произведено расширение адресации;
2) изменена спецификация формата заголовков;
3) введена возможность задания нескольких заголовков;
4) улучшена поддержка расширений и опций;
5) введена возможность пометки потоков данных;
6) добавлена идентификация и защита частных обменов.

Рассмотрим эти изменения более детально.

Расширение адресации. В IPv6 длина адреса расширена до 128 битов (против 32 в IPv4), что позволяет обеспечить больше уровней иерархии адресации, увеличить число адресуемых узлов, упростить автоконфигурацию. Для расширения возможности групповой маршрутизации в адресное поле введено субполе "scope" (группа адресов). Определен новый тип адреса –

серверный групповой ("anycast address"), который используется для посылки запросов клиента любой группе серверов. Групповая адресация предназначена для использования с набором взаимодействующих серверов, чьи адреса не известны клиенту заранее.

Адресное пространство IPv6 распределяется комиссией по стандартным числам в Интернет IANA *(Internet Assigned Numbers Authority)*. IANA делегирует права выдачи IP-адресов региональным сервис-провайдерам, субрегиональным структурам и организациям. Отдельные лица и организации могут получить адреса непосредственно от регионального распределителя или сервис-провайдера.

Спецификация формата заголовков. Некоторые поля заголовка IPv4 исключаются или делаются опционными. Таким образом, снижаются издержки, связанные с обработкой заголовков пакетов. Это позволяет уменьшить влияние расширения длины адресов в IPv6.

Наличие нескольких заголовков. Дейтаграмма может содержать один основной (базовый) и несколько вспомогательных заголовков, за которыми следуют данные.

Улучшенная поддержка расширений и опций. Изменение кодирования опций IP-заголовков позволяет облегчить переадресацию пакетов, ослабляет ограничения на длину опций, и делает более доступным введение дополнительных опций в будущем.

Возможность пометки потоков данных. Она позволяет помечать пакеты, принадлежащие определенным транспортным потокам, для которых отправитель запросил определенную процедуру обработки, например, нестандартный тип, вид услуг или обработку данных в реальном масштабе времени. Это сделано для осуществления возможности передачи по сети с коммутацией пакетов видео- и аудиоинформации.

Идентификация и защита частных обменов. В IPv6 введена спецификация идентификации сетевых объектов или субъектов, для обеспечения целостности данных и при желании защиты частной информации.

Основной заголовок протокола IPv6 содержит меньше информации, чем его предшественник, однако в два раза длиннее заголовка пакета IP. Формат пакета показан на рисунке 4.10. Большую часть заголовка занимают адреса отправителя и получателя, длина каждого из которых составляет 16 байт. Остальные поля имеет следующее назначение: **Версия** (4 бита) – номер версии Интернет-протокола (для IPv6= 6).

Класс трафика (8 битов) определяет класс пути, который используется для выбора маршрутизатора. Задавая класс пути можно добиться того, чтобы сеть создала маршрут с минимальной задержкой распространения, что очень важно, например, при передаче видео- или аудиоинформации.

Рисунок 4.10 – Формат заголовка пакета IPv6

Метка потока (20 битов) служит для использования с новыми приложениями, которые гарантируют заданное качество обслуживания. Метка может использоваться для того, чтобы ассоциировать дейтаграмму с определенным сетевым маршрутом.

Размер поля данных (16 битов) – код длины поля данных в октетах, которое следует сразу после заголовка пакета. Если код равен нулю, то длина поля данных записана в специальном поле данных *jumbo*, которое в свою очередь хранится в поле опций.

Следующий заголовок (8 битов) идентифицирует тип заголовка, который следует непосредственно за IPv6 заголовком. Если дейтаграмма содержит дополнительный заголовок, то в этом поле указывается тип такого заголовка. В случае наличия в пакете только основного заголовка в поле "Следующий заголовок" указывается код протокола верхнего уровня (TCP или UDP). Для обозначения протоколов верхнего уровня используются те же значения, что и в протоколе IP.

Предельное число шагов (8 битов) соответствует полю "Время жизни" IP-пакета. Оно уменьшается на 1 в каждом узле, через который проходит пакет. При достижении предельного числа шагов нулевого значения пакет удаляется.

4.2.3. Способы адресации IPv6

Протоколом IPv6 предусмотрено использовать три типа адресов.

1. **Индивидуальный** (*unicast*) адрес соответствует единственному компьютеру Пакет, посланный по индивидуальному адресу, доставляется интерфейсу, указанному в адресе получателя.

2. **Адрес набора интерфейсов** (*anycas*t) соответствует группе компь-

ютеров, имеющих одинаковый адресный префикс (это означает, что они находятся в одной и той же сети). Пакет, отправленный по этому типу адреса, доставляется одному из интерфейсов, указанному в адресе, который находится ближе всего к отправителю (в соответствии с мерой, определенной протоколом маршрутизации).

3. **Групповой** (*multicast*) соответствует многим компьютерам, принадлежащих разным узлам. Пакет, посланный по групповому адресу, доставляется всем интерфейсам, заданным этим адресом.

В IPv6 не существует широковещательных адресов, их функции переданы групповым адресам. В адресе IPv6 допускается использовать все нули и все единицы, если только не оговорено исключение.

Индивидуальные (*уникастные*) адреса отличаются от групповых значением старшего октета: значение FF (11111111) идентифицирует групповой адрес; любые другие значения свидетельствуют о том, что адрес индивидуальный. Наборы адресов (*anycast*-адреса) берутся из пространства индивидуальных адресов, и синтаксически неотличимы от них.

Существует три стандартные формы для представления IPv6 адресов в виде текстовых строк.

1. **Основная форма** имеет вид x:x:x:x:x:x:x:x, где “x” шестнадцатеричные 16-битовые числа. При этом разрешается не писать начальные нули в каждом из конкретных полей, но в каждом поле должна быть, по крайней мере, одна цифра (за исключением случая, описанного в пункте 2).

Например: caf4:defc:ba98:4758:fbdc:632f:4d7e:f3c2 или 2175:0:0:0:6:400:df0C:851b.

2. **Специальная форма** принята по причине того, что из-за метода записи некоторых типов IPv6 адресов, они часто содержат длинные последовательности нулевых бит. Для того чтобы сделать запись адресов, содержащих нулевые биты, более удобной, имеется специальный синтаксис для удаления лишних нулей. Применение записи вида "::" указывает на наличие групп из 16 нулевых битов. Комбинация "::" может появляться только при записи адреса. Последовательность "::" разрешается также использовать для удаления из записи начальных или завершающих нулей в адресе. Например: вместо последовательности ff01:0:0:0:0:0:0:43 можно применять ff01::43.

3. **Альтернативная форма** записи, которая более удобна при работе с IPv4 и IPv6; она имеет вид x:x:x:x:x:x:d.d.d.d, где 'x' шестнадцатиричные 16-битовые коды адреса, а 'd' десятичные 8-битовые, составляющие младшую часть адреса (стандартное IPv4 представление). Например: 0:0:0:0:0:0:172.3.47.12 или 0:0:0:0:0:facd:64.137.35.44.

IPv6 адреса всех типов ассоциируются с интерфейсами, а не узлами. Так как каждый интерфейс принадлежит только одному узлу, индивидуальный (уникастный) адрес интерфейса может идентифицировать узел. IPv6 ин-

дивидуальный адрес соотносится только с одним интерфейсом. Одному интерфейсу могут соответствовать много IPv6 адресов различного типа (индивидуальный адрес хоста, широковещательные и групповые). Существует два исключения из этого правила.

1. Индивидуальный адрес может приписываться нескольким физическим интерфейсам, если приложение рассматривает эти несколько интерфейсов как единое целое при представлении его на уровне Интернет.

2. Маршрутизаторы могут иметь ненумерованные интерфейсы (например, интерфейсу не присваивается никакого IPv6 адреса) для соединений по схеме точка-точка, чтобы исключить необходимость вручную конфигурировать и объявлять эти адреса. Адреса не нужны также для маршрутизаторов, соединенных по схеме точка-точка, если эти интерфейсы не используются в качестве точки отправления или назначения при посылке IPv6 дейтаграмм.

Специфический тип IPv6 адресов идентифицируется начальными битами адреса. Поле переменной длины, содержащее эти начальные биты, называется *префиксом формата* FP (*Format Prefix*).

Индивидуальный адрес хоста вида 0:0:0:0:0:0:0:1 называется адресом обратной связи. Он может использоваться для посылки IPv6 дейтаграмм самому себе. Его нельзя использовать в качестве идентификатора интерфейса. Адрес обратной связи не должен применяться в качестве адреса отправителя в IPv6 дейтаграммах, которые посылаются за пределы узла. IPv6 дейтаграмма с адресом обратной связи в качестве адреса места назначения также не может быть послана за пределы узла.

4.3. Протоколы транспортного уровня UDP и TCP

4.3.1. Назначение и разновидности протоколов транспортного уровня

Протоколы транспортного уровня предназначены для обеспечения непосредственного информационного обмена между двумя пользовательскими процессами. Для организации информационного взаимодействия на транспортном уровне кроме задания сетевого адреса абонента должен быть указан и ***номер порта*** процесса. В данном случае порт является виртуальным интерфейсом транспортного уровня.

В качестве протоколов транспортного уровня в сети Интернет используются два протокола:

- протокол передачи пользовательских дейтаграмм **UDP** (*User Datagram Protocol*);

- протокол управления передачей **TCP** (*Transmission Control Protocol*).

Протокол UDP реализует негарантированную доставку сообщений в сети Интернет. Протокол может быть использован в тех приложениях, которые либо не нуждаются в таком качестве, либо гарантия доставки обеспечивается другими средствами. Примерами приложений, использующих протокол UDP, являются службы TELNET и TFTP. Протокол TCP применяется для создания надежного информационного обмена на транспортном уровне в сетях Интернет.

Протокол пользовательских дейтаграмм передает сообщение в виде отдельных независимых информационных блоков – *дейтаграмм*. Каждая порция данных (*запись*), которую приложение передает модулю UDP, воспринимается последним, как *отдельное сообщение*. UDP формирует из него дейтаграмму, длина которой соответствует размеру полученной порции данных. Таким образом, протокол UDP при передаче данных сохраняет границы сообщений. При дейтаграммной связи нет необходимости устанавливать и разрывать соединения, а также управлять потоком. Протокол доставки дейтаграмм *прост в реализации*, однако, не обеспечивает гарантированной и безошибочной доставки сообщений.

Существует достаточно много причин, которые могут помешать пакету, передаваемому в сети, успешно достичь станции назначения. Поэтому, если не будут использованы специальные методы для обеспечения гарантированной доставки, принятое сообщение может существенным образом отличаться от переданного.

В протоколе гарантированной доставки решаются следующие задачи:

- реализация потокового обмена;
- установка виртуальных соединений;
- буферизация передачи данных;
- защита от ошибок;
- обмен в режиме полного дуплекса;
- установка таймеров обмена;
- контроль потока данных.

Потоковый обмен протокола характеризуется тем, что протокол TCP рассматривает данные приложения, как *поток байтов* (октетов). TCP делит этот поток на части, равные заданной длине сегмента. Если входной поток меньше длины сегмента, то модуль TCP буферизирует входные данные, стремясь сформировать полноразмерный сегмент. По этой причине протокол TCP не сохраняет границы сообщений. Так, если приложение передало N блоков данных, то это не означает, что модуль TCP отправит N сегментов. Реализация потокового обмена позволяет обеспечить такой режим информационного взаимодействия, когда приемник получает абсолютно ту же после-

довательность байтов, которая была передана отправителем.

Протокол TCP *управляет логическим сеансом связи*. Он устанавливает, поддерживает и закрывает соединение между процессами, проводит по мере необходимости процедуры аутентификации. В процессе информационного обмена через установленное соединение обе стороны контролируют его качество, а при возникновении проблем с передачей данных инициируют процесс разрыва соединения и формируют соответствующие сообщения для протоколов верхних уровней.

Применение буферизации позволяет согласовать скорость информационного обмена в канале связи со скоростью передачи данных приложения пользователя.

В транспортном протоколе TCP осуществляется расчет контрольной суммы для заголовка сегмента и поля данных. Если сегмент прибывает с неверной контрольной суммой, TCP отбрасывает его и подтверждение не генерируется. Получатель ожидает, пока отправитель выждет тайм-аут и осуществит повторную передачу.

TCP не интерпретирует содержимое байтов и не проверяет, происходит ли обмен двоичными данными, символами *ASCII*, *EBCDIC* либо другими символами. Эта интерпретация потока байтов осуществляется приложениями на каждой стороне соединения.

Для обеспечения требования доставки трафика, чувствительного к временным задержкам, в дополнение к буферу может быть реализован дополнительный **механизм проталкивания** "*push*". Использование данного механизма обеспечивает немедленную выдачу содержимого буфера в тот момент, когда в него попадают данные, чувствительные к временным задержкам.

TCP осуществляет контроль потока данных, позволяя источнику посылать данные только в том случае, если получатель имеет возможность поместить их в буфер. Это предотвращает переполнение буферов медленных хостов информацией от быстрых источников.

Для обеспечения гарантированной доставки сообщений протокол TCP использует аппарат *позитивного квитирования* с повторной передачей (решающая обратная связь). Обычно при использовании данной схемы получатель информации посылает специальный сигнал (*квитанцию*) ACK в подтверждение ее получения. Дальнейшее выполнение информационного обмена может быть произведено только в случае, если передающая сторона получит такое подтверждение.

В случае процедуры с ожиданием подтверждения передающая сторона приостанавливает передачу очередного сегмента до получения сигнала подтверждения ACK о приеме предыдущего сегмента (рисунок 4.11).

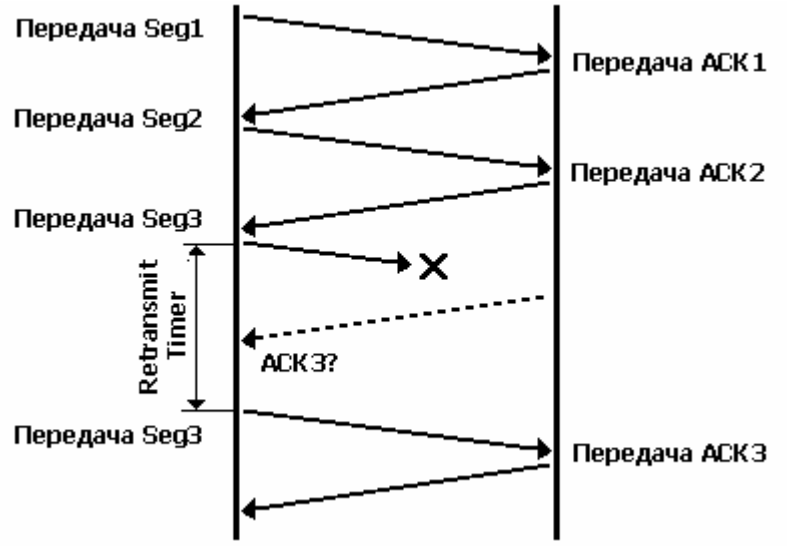


Рисунок 4.11 – Процедура передачи сегментов с ожиданием подтверждения

Интервал ожидания устанавливается равным значению задержки повторной передачи – *Retransmit Timer*. Если в течение этого интервала времени не будет получено подтверждение о приеме переданного сегмента, передача данного сегмента выполняется повторно.

Применение процедуры ожидания квитанции подтверждения (квитирования) не обеспечивает достаточную эффективность использования пропускной способности каналов передачи данных. Около половины времени сеанса связи системы ожидают получения подтверждения. Более эффективной в этом смысле является процедура квитирования с применением скользящего окна, позволяющая передающей стороне отправить несколько сегментов сообщения, не дожидаясь получения подтверждения о приеме.

Максимальное число сегментов, которые передающая сторона может отправить до получения подтверждения приема первого из них, называется ОКНОМ. При использовании механизма окна принимающая сторона может передавать подтверждение на получение сразу нескольких сегментов. В транспортном протоколе TCP процедура скользящего окна реализуется применительно к байтам. **Каждому байту** входного потока присваивается **порядковый номер**.

Для реализации процедур управления процессом передачи применяется три указателя (см. п.1.5.3):

- первый указывает границу между последним байтом, который был передан и получение которого подтверждено, и первым переданным, но неподтвержденным байтом;
- второй – указывает границу между последним переданным байтом,

подтверждение о получении для которого еще не получено, и первым байтом, который может быть выдан в канал, до получения подтверждения о приеме предыдущих переданных байтов;

- третий – указывает границу между последним байтом, который может быть передан до получения подтверждения о приеме предыдущих переданных байтов и остальной частью информационного потока.

Процедура управления потоком заключается в согласовании скорости передачи данных с пропускной способностью канала. Для обеспечения управления потоком в протоколе TCP предусмотрена возможность изменения размера окна. Каждое сообщение подтверждения содержит в себе значение представляемого размера окна – (*window advertisement*), определяющее размер буфера, который может быть использован в текущий момент для приема информации.

Применение скользящего окна для управления информационным потоком делает ненужным реализацию дополнительных механизмов для управления переполнением.

Большинство протоколов приложений используют услуги TCP, среди них Webserver (HTTP-протокол, порт 80), FTP-Server (FTP-протокол, порт 21), Электронная почта (SMTP-протокол, порт 25) и др.

4.3.2. Протокол передачи пользовательских дейтаграмм UDP

Протокол передачи пользовательских дейтаграмм **UDP** (*User Datagram Protocol*) является одним из основных протоколов, расположенных непосредственно над IP. Он предоставляет прикладным процессам транспортные услуги, немногим отличающиеся от услуг протокола IP. Протокол UDP обеспечивает доставку дейтаграмм, но не требует подтверждения их получения. UDP-протокол также не требует соединения с удаленным модулем UDP. К заголовку IP-пакета агент *udp* добавляет поля "Порт отправителя" и "Порт получателя", которые обеспечивают мультиплексирование информации между различными прикладными процессами, а также поля "Длина UDP-дейтаграммы" и "Контрольная сумма", позволяющие проверять целостность данных. Таким образом, если на уровне IP для определения места доставки пакета используется адрес, на уровне UDP – **номер порта**.

Примерами сетевых приложений, использующих UDP-протокол, являются сетевая файловая система NFS, упрощенный протокол передачи файлов TFTP, удаленный вызов процедуры RPC (*Remote Procedure Call*), простой сетевой протокол управления SNMP и доменная служба имен DNS. Широкое применение протокола обусловлено невысокой избыточностью

дейтаграммы и отсутствием необходимости подтверждения получения пакета.

Прикладные процессы и модули UDP взаимодействуют через **UDP-порты**. Эти порты нумеруются, начиная с нуля. Прикладной процесс, предоставляющий некоторые услуги (*сервер*), ожидает сообщений, направленных в порт, специально выделенный для этих услуг. Программа-сервер находится в состоянии ожидания до тех пор, пока какая-нибудь программа-клиент запросит услугу. Например, сервер *snmp* всегда ожидает сообщения, адресованные в порт 161. Если клиент *snmp* желает получить услугу, он посылает запрос в UDP-порт 161 на компьютер, где работает сервер. На каждом компьютере может быть только один агент *snmp*, так как существует только один порт 161. Данный номер порта является общеизвестным, т.е. имеющий фиксированный номер, официально выделенный в сети Интернет для услуг SNMP.

Данные, отправляемые прикладным процессом через модуль UDP, достигают места назначения как единое целое. Например, если процесс-отправитель производит 5 записей в порт, то процесс-получатель должен будет сделать 5 чтений. Размер каждого прочитанного сообщения будет совпадать с размером каждого записанного. Протокол **UDP сохраняет границы сообщений**, определяемые прикладным процессом. Он никогда не объединяет несколько сообщений в одно и не делит одно сообщение на части. Формат UDP-дейтаграммы изображен на рисунке 4.12.

Рисунок 4.12 – Формат UDP-дейтаграммы

Протокол использует заголовок минимального размера (8 байтов). Длина сообщения равна числу байт в UDP-дейтаграмме, включая заголовок. Поле "Контрольная сумма" содержит код, полученный в результате контрольного суммирования UDP-заголовка и поля данных.

Модуль IP передает поступающий IP-пакет модулю UDP, если в заголовке этого пакета указан код протокола UDP. Когда модуль UDP получает дейтаграмму от модуля IP, он проверяет контрольную сумму, содержащуюся в ее заголовке. Если контрольная сумма равна нулю, это означает, что отпра-

витель ее не подсчитывал. ICMP, IGMP, UDP и TCP протоколы имеют один и тот же алгоритм вычисления контрольной суммы. Однако вычисление контрольной суммы для UDP имеет некоторые особенности.

Во-первых, длина UDP-дейтаграммы может содержать нечетное число байтов, в этом случае к ней добавляется нулевой байт, который служит лишь для унификации алгоритма и никуда не выдается.

Во-вторых, при расчете контрольной суммы для UDP и TCP добавляются 12-байтные псевдозаголовки, содержащие IP-адреса отправителя и получателя, код протокола и длину дейтаграммы (рисунок 4.13). Как и в случае IP-дейтаграммы, если вычисленная контрольная сумма равна нулю, в соответствующее поле будет записан код 65535 (обратный код 0). При совпадении контрольной суммы проверяется порт назначения, указанный в заголовке дейтаграммы. Если прикладной процесс подключен к этому порту, то прикладное сообщение, содержащиеся в дейтаграмме, становится в очередь к прикладному процессу для прочтения. В остальных случаях дейтаграмма отбрасывается.

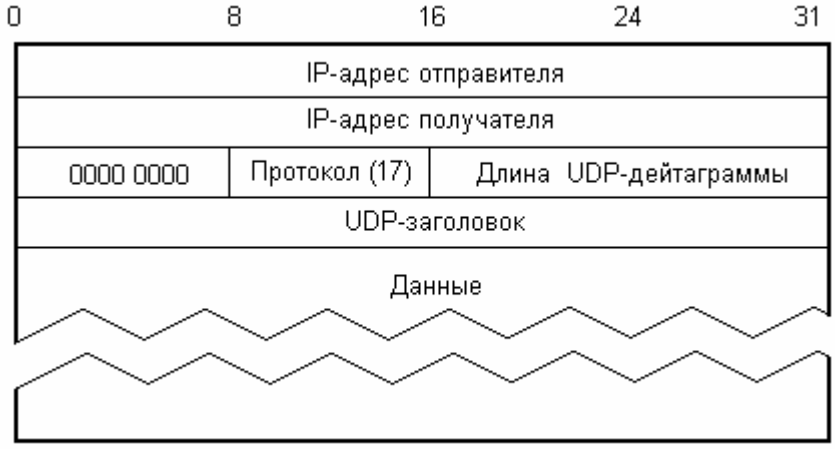


Рисунок 4.13 – Псевдозаголовок, используемый при расчете контрольной суммы

В случае поступления дейтаграмм быстрее, чем их успевает обрабатывать прикладной процесс, при переполнении очереди сообщений поступающие дейтаграммы отбрасываются модулем UDP.

В связи с тем, что максимальная длина IP-дейтаграммы равна 65535 байтам, максимальная протяженность информационного поля UDP-дейтаграммы составляет 65507 байтов (20 байтов минимальная длина IP-заголовка и 8 байтов UDP-заголовка). Обычно на практике большинство систем работает с UDP-дейтаграммами с длиной не более 8192 байтов.

4.3.3. Протокол с установлением виртуальных соединений TCP

Протокол **TCP** (*Transmission Control Protocol*) в отличие от UDP осуществляет доставку отрезков сообщений, называемых *сегментами*, в виде байтовых потоков с установлением соединения. Перед тем как какая-либо сторона предполагает послать данные другой, между ними должно быть установлено соединение. Протокол TCP применяется в тех случаях, когда требуется гарантированная доставка сообщений. Он использует контрольные суммы пакетов для проверки в них ошибок и освобождает прикладные процессы от необходимости тайм-аутов и повторных передач с целью обеспечения необходимой достоверности. Для отслеживания подтверждения доставки в TCP реализуется алгоритм "скользящего" окна.

Максимальный размер сегмента MSS (*Maximum Segmet Size*) – это самый большой отрезок данных, который TCP пошлет на удаленную сторону. В процессе установления соединения каждая сторона может объявить свой MSS. IP-дейтаграмма, осуществляющая передачу TCP-сегмента, обычно на 40 байтов больше, в связи с тем, что 20 байт отводится под TCP- заголовок и 20 байтов под заголовок IP.

Модуль TCP также, как и UDP, выполняет функции мультиплексора/демультиплексора между прикладными процессами и IP-модулем. При поступлении пакета в модуль IP он будет передан в TCP- или UDP-модуль, в зависимости от кода, записанного в поле протокола данного IP-пакета. Формат сегмента (пакета) TCP изображен на рисунке 4.14.

Каждый TCP сегмент содержит **номера портов** (*port number*) отправителя и получателя, с помощью которых идентифицируются отправляющее и принимающее приложения. Эти два значения вместе с IP-адресом источника и получателя в IP-заголовке уникально идентифицируют каждое соединение.

Комбинация IP-адреса и номера порта называется **сокетом** (*socket*). Пара сокетов, содержащая IP-адрес клиента, номер порта клиента, IP-адрес сервера и номер порта сервера, указывает две конечные точки, однозначно идентифицирующие каждое TCP-соединение.

Номер последовательности SN (*sequence number*) – порядковый номер первого октета в поле данных сегмента среди всех октетов потока данных для текущего соединения, т.е., если в сегменте пересылаются октеты с 2001-го по 3000-й, то SN=2001. Если во время фазы установления соединения в заголовке сегмента активизирован бит SYN, то в поле SN записывается начальный номер ISN (*Initial Sequence Number*). Номер первого октета данных, посылаемых после завершения фазы установления соединения, равен ISN+1. Номер последовательности представляет собой 32-битное беззнаковое число, переходящее через 0 по достижению значения $(2^{32} – 1)$.

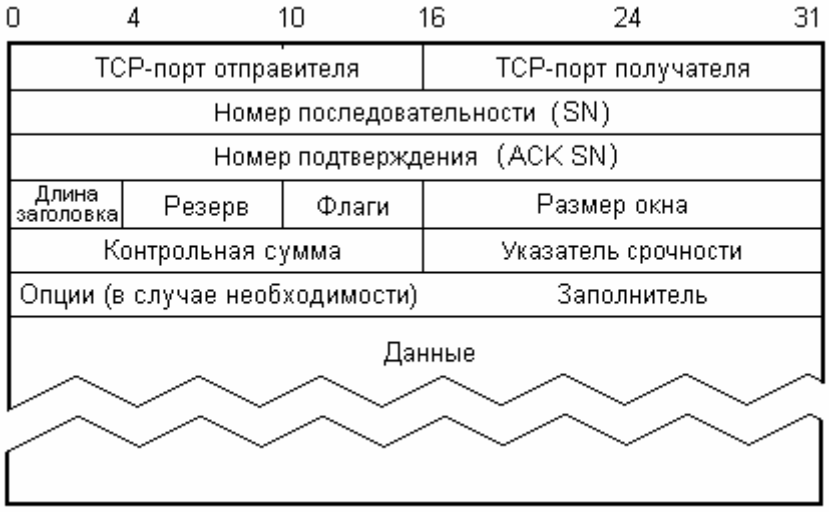


Рисунок 4.14 – Формат протокола TCP

Номер подтверждения ACK SN (*acknowledgment number*) – это следующий номер байта в последовательности, который ожидает получить отправитель сигнала подтверждения. Обратите внимание, что в TCP-протоколе нумеруются не сегменты, а байты потока, поступающего от приложения. В связи с тем, что каждый байт, участвующий в обмене, пронумерован, то номер подтверждения – это номер последнего успешно принятого байта данных, увеличенный на 1. Поле ACK SN принимается в рассмотрение только в случае, если имеется сигнал подтверждения (флаг ACK установлен в единицу).

Длина заголовка (*header length*) занимает 4 бита и указывает длину заголовка в 32-битных словах. Введение этого поля в заголовок объясняется переменной длиной поля опций. Максимальная длина заголовка 60 байтов. Без опций стандартный размер заголовка составляет 20 байтов.

Флаги (*Flags*) занимают 6-разрядное поле, биты которого имеют следующее значение:

- **URG** (*urgent pointer*) – указатель срочности показывает, что поле "Указатель срочности" должно приниматься во внимание;
- **ACK** (*acknowledgment*) указывает получателю, что номер подтверждения необходимо принять в рассмотрение;
- **PSH** (*Push*) - получатель должен передать приложению данные этого сегмента немедленно;
- **RST** (*Reset*) – сбросить соединение; при получении сегмента с установленным флагом RST соединение ликвидируется, недоставленные данные уничтожаются;

- **SYN** – запрос на установление соединения;
- **FIN** (*finish*) означает, что отправитель заканчивает посылку данных.

В сегменте одновременно могут быть установлены несколько флагов.

Размера окна (*window size*) – это количество байт, начинающееся с указанного в поле номера подтверждения, которое приложение готово принять. Этот параметр служит для управления потоком данных TCP. Очевидно, что 16-битовое поле ограничивает размер окна в 65535 байтов.

Контрольная сумма (*checksum*) охватывает собой весь TCP-сегмент, т.е. TCP-заголовок и TCP-данные. Это обязательное поле, которое должно быть рассчитано и сохранено отправителем, а затем проверено получателем. Контрольная сумма TCP рассчитывается так же, как контрольная сумма UDP, с использованием псевдозаголовка.

Указатель срочности используется для задания длины данных, которые размещаются в начале поля данных сегмента. Он указывает смещение последнего октета со срочными данными относительно первого октета в сегменте. Например, в сегменте передаются октеты с 2001–го по 3000–й, при этом первые 100 октетов являются срочными данными. В этом случае в указателе срочности будет находиться число 100.

Поле "**Опции**" имеет переменную длину; может отсутствовать или содержать одну опцию либо список опций, реализующих дополнительные услуги протокола TCP. Максимальная длина опций равна 40 байтам. Наиболее распространенными опциями являются: "Максимальный размер сегмента", "Масштабирование окна", "Выборочные подтверждения разрешены" и др. Поле опций состоит из октета, определяющего тип опции, затем может следовать октет с длиной опции и октеты с данными для опций.

Максимальное значение размера сегмента **MSS** зависит от ряда факторов. Если IP-адрес назначения "не локальный", MSS обычно устанавливается по умолчанию – 536. Некоторые операционные системы объявляют MSS, равный 1460, когда обе стороны находятся на одной сети *Ethernet*. В процессе установления соединения каждая сторона может объявить свой MSS. В общем случае, чем больше MSS, тем лучше, до тех пор, пока не происходит фрагментация.

Для установления TCP-соединения необходима посылка трех сегментов с получением подтверждения (рисунок 4.15). Такую процедуру называют трехразовым рукопожатием (*three-way handshake*). Она осуществляется следующим образом.

- *Сегмент* 1: Запрашивающая сторона А отправляет с установленным флагом синхронизации (SYN=1) сегмент с указанием номера порта удаленного компьютера Б, к которому станция А хочет подсоединиться, и начальный номер последовательности **ISN** (*Initial Sequence Number*). В данном примере ISN =4712, начиная с которого будут отсчитываться отправляемые

октеты данных. Первый сегмент не содержит поля данных, а служит только для установления соединения.

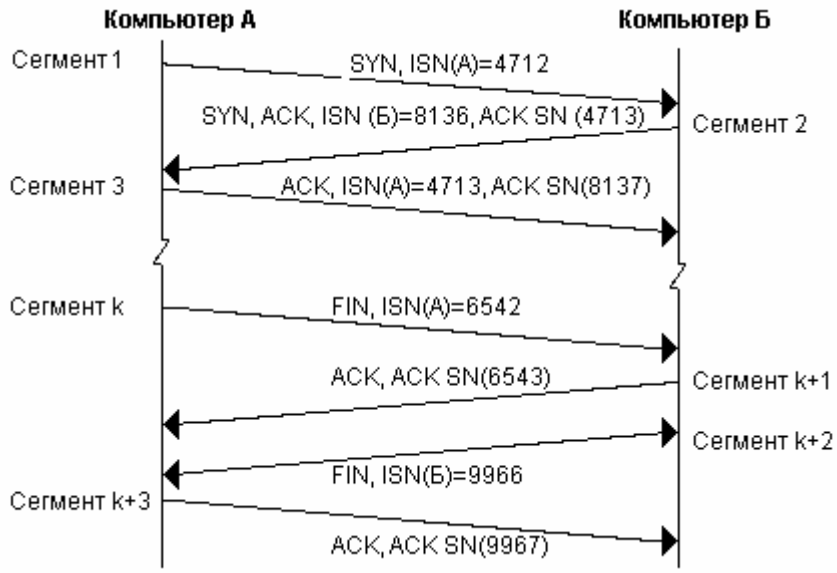


Рисунок 4.15 – Временная диаграмма установления и разрыва соединения

- *Сегмент* 2: Станция Б откликается TCP-сегментом, в заголовке которого установлен бит АСК, подтверждающий установление соединения с компьютером А и установлен номер подтверждения ACK SN, равный ISN(A) +1=4713, свидетельствующий о готовности Б к получению данных от компьютера А, начиная с номера 4713. В связи с тем, что протокол TCP обеспечивает полнодуплексную передачу данных, то компьютер Б в этом же сегменте устанавливает бит SYN, означающий запрос начала сеанса для передачи данных от Б к А, и передает свой исходный номер последовательности (в нашем примере ISN(Б)=8136). Полезных данных этот сегмент тоже не содержит.
- *Сегмент* 3: Станция А посылает сегмент в ответ на сегмент, полученный от компьютера Б. В связи с тем, что соединение считается установленным (получено подтверждение от Б), то станция А может включать в свой сегмент данные, нумерация которых начинается с ISN(A) +1. Данные нумеруются по количеству отправленных октетов. В заголовке этого же сегмента устанавливается бит АСК, подтверждающий установление соединение Б→А.

На временной диаграмме показан также процесс разрыва соединения, которое произошло на этапе, когда станции А и Б успели передать по 1830 байтов каждая. Процедура разрыва соединения инициируется посылкой сегмента с установленным флагом FIN и номером последовательности. Разрыв завершается передачей противоположной станций пакета подтверждения.

Как отмечалось выше, в процессе посылки сегмента SYN с целью открытия соединения активная сторона указывает исходный номер последовательности ISN для этого соединения. Это значение берется с 32-битного счетчика, который увеличивается на единицу каждые 4 микросекунды. Счетчик номера последовательности устанавливается в 0 при начальной инициализации TCP-модуля, и после переполнения снова начинает счет с нуля. Очевидно, что при установлении очередного соединения ISN будет каждый раз разным. Использование номеров последовательностей позволяет контролировать очередность поступления сегментов. Пакеты с нарушенной очередностью изменения ISN отбрасываются, а передающая сторона повторяет сегменты, не дошедшие до узла назначения.

После установки соединения обе стороны могут начать обмен пакетами данных. Источник посылает данные потребителю и ждет от него подтверждений об их получении. При этом каждый поступивший без ошибок сегмент данных подтверждается посылкой сегмента с соответствующим номером правильно принятых октетов. Затем источник снова посылает данные и т.д., пока сообщение не закончится. Получатель определяет, что сообщение закончилось по появлению сегмента с установленным флагом FIN, что означает "нет больше данных".

В дуплексном TCP-соединении данные могут передаваться в любом направлении независимо от противоположного направления. Поэтому при завершении соединения каждое направление должно быть закрыто независимо от другого. Для того чтобы разорвать соединения требуется посылка четырех сегментов. Процесс закрытия состоит в том, что обе стороны должны после завершения обмена данными послать сегмент с установленным флагом FIN. В момент обнаружения модулем TCP в принимаемом сегменте установленного в "1" флага FIN, он уведомляет приложение о том, что удаленная сторона разрывает соединение и прекращает передачу данных в этом направлении.

Сторона, которая первой закрывает соединение, т.е. отправляющая первой FIN, осуществляет так называемое *активное закрытие*, а другая сторона, принявшая этот FIN, осуществляет *пассивное закрытие*. Обычно, одна сторона выполняет активное закрытие, а другая – пассивное. Однако активное закрытие могут реализовать и обе стороны. Этот случай и изображен на временной диаграмме.

Передача данных на транспортном уровне с использованием протокола TCP может осуществляться в интерактивном (*диалоговом*) и неинтерактивном (*пакетном*) режимах. TCP способен обрабатывать оба типа данных, однако при передаче разных типов данных используются различные алгоритмы.

Рассмотрим как осуществляется передача данных при вводе интерак-

тивной команды на примере соединения приложения *Rlogin*. Это приложение обычно посылает от клиента серверу по одному символу данных. Каждое нажатие оператором клавиши генерирует один пакет данных. Причем, от клиента серверу в пакете посылается только 1 байт данных. Передаваемые пакеты имеют стандартный размер 41 байт: 20 байтов IP-заголовок, 20 байтов TCP-заголовок и 1 байт данных. В ответ на этот пакет сервер *Rlogin* отправляет свой пакет, в котором отражает эхом символ, переданный клиентом (рисунок 4.16).

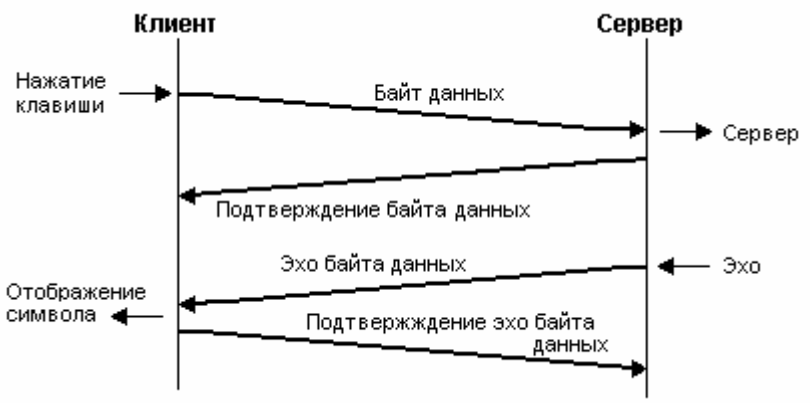


Рисунок 4.16 – Временная диаграмма передачи сегментов в интерактивном режиме

Таким образом, на одно нажатие клавиши генерируется 4 сегмента: (1) интерактивный ввод символа от клиента, (2) подтверждение получения символа от сервера, (3) эхо-передача сервером введенного символа и (4) подтверждение на эхо от клиента. Обычно сегменты 2 и 3 объединяются, т.е. подтверждение введенного символа отправляется вместе с эхо. В приложении *Telnet* предусмотрена опция, которая позволяет отправить целую строку ввода от клиента серверу. Тем самым уменьшается загрузка сети.

Маленькие пакеты, называемые **тиниграммами** (от англ. *tiny* – маленький) обычно не создают проблем для локальных сетей, так как большинство их из-за высокой пропускной способности не перегружается. Однако поток таких пакетов может привести к перегрузке глобальной сети. Для предотвращения перегрузки Дж. Наглом был предложен простой и эффективный алгоритм, который получил название ***алгоритм Нагла***.

Суть алгоритма состоит в том, что первый пакет с одним символом данных отправляется обычным образом. До прибытия подтверждения на прием этого символа данные, поступающие от источника, не передаются, а накапливаются в буфере. После получения подтверждения на первый символ модуль TCP отправляет накопленные в буфере символы в одном сегменте. Очевидно, что при этом появляется задержка от момента нажатия клавиши,

до момента поступления символа на сервер.

Достоинство этого алгоритма заключается в том, что он сам настраивает темп передачи: чем быстрее придет подтверждение, тем быстрее будут отправлены данные. В медленных глобальных сетях, где задержка получения подтверждения достаточно большая, компьютер-источник отправляет пакеты с большим количеством символов. Благодаря этому снижается избыточность передаваемых сегментов и повышается эффективная скорость передачи данных.

В ряде случаев задержка, возникающая при использовании алгоритма Нагла, недопустима, например, передача координат перемещения манипулятора "мышь". Для устранения этого недостатка предусмотрена команда отключения алгоритма Нагла.

В случае передачи данных в неинтерактивном режиме в передаваемом сегменте размещается группа символов, количество которых определяется максимальным размером сегмента (например, MSS=1024 байта). При этом в канал передается несколько сегментов без ожидания подтверждения правильности приема предыдущих сегментов. Количество переданных без ожидания подтверждения сегментов определяется шириной окна, задаваемого в поле заголовка TCP-сегмента.

4.3.4. Протокол динамической конфигурации сетевых компьютеров DHCP

Протокол динамической конфигурации хоста **DHCP** (*Dynamic Host Configuration Protocol*) предназначен для автоматической настройки параметров стека TCP/IP сетевого компьютера в момент его загрузки или по требованию пользователя. Этот протокол организован по принципу ***клиент-сервер***: клиент запрашивает информацию, делая широковещательный запрос в сеть, а сервер сообщает ее клиенту в ответ. Протокол DHCP использует транспортный протокол UDP для передачи сообщений между клиентом (порт 68) и сервером (порт 67). Протокол динамической настройки параметров имеет смысл применять в сетях с большим количеством компьютеров. Если сеть состоит всего лишь из нескольких компьютеров, проще дать им фиксированные IP-адреса, чем заниматься настройкой DHCP.

Серверы DHCP используются для динамического назначения IP-адресов, а также для сообщения клиентам DHCP такой настроечной информации, как маска сети, адрес сервера имен и т.д. Обязательно выделяется только **адрес** и **маска**, все остальные параметры назначаются в зависимости от настроек сервера и клиента DHCP.

Протокол DHCP позволяет клиенту передавать серверу желаемое имя и адрес компьютера. В большинстве случаев выдача IP-адреса привязана к MAC-адресу. Компьютер, идентифицированный в DHCP по MAC-адресу, не получает выданный IP навсегда: адрес сдается ему в "аренду" (lease) на некоторое время. Если до истечения срока аренды абонент не подтвердил желание пользоваться адресом и дальше (не послал повторный DHCP-запрос), адрес считается незанятым. Но когда компьютер подключается к сети после долгого перерыва, сервер DHCP сначала просматривает "арендную историю" на предмет того, какой IP этому абоненту уже выдавался. Если этот IP не занят, то будет выдан именно он. И только когда к сети подключится совсем новый абонент (а все адреса уже когда-нибудь кому-то выдавались) среди них будет выбран и отдан в аренду новичку тот, который дольше всех оставался невостребованным. После подключения новой станции к сети, в которой может быть несколько DHCP-серверов, станция (клиент DHCP) отправляет широковещательное сообщение DHCPdiscover "Поиск адреса". В это сообщение клиент может включить желаемые параметры конфигурации (IP-адрес, срок аренды и т.п.). Все DHCP-серверы сети отвечают на этот запрос предложением DHCPoffer с перечнем предлагаемых сетевых адресов. Если в течение определенного времени не поступило ни одного предложения, то клиент начинает процесс заново.

Получив предложение, клиент на основании своих настроек решает принять предложение определенного сервера или первое поступившее (если никаких настоек нет). В это время серверы DHCP резервируют предложенные адреса. Затем клиент извещает соответствующий сервер о принятом решении посылкой сообщения DHCPrequest, в котором указывается идентификатор данного сервера и параметры конфигурации. Сообщение DHCPrequest регистрируют все DHCP-серверы. Те из них, которые не обнаруживают своего идентификатора в этом сообщении разблокируют невыбранные адреса. Избранный DHCP-сервер передает клиенту подтверждение с дополнительными конфигурационными параметрами. Выбранный же клиентом сервер закрепляет за клиентом сетевой адрес, устанавливает срок аренды и подтверждает это посылкой сообщения DHCPack. Если же сервер по какой-либо причине не может удовлетворить клиента (например, пакет DHCPrequest задержался и сервер уже выдал адрес другому клиенту), то сервер отвечает сообщением-отказом DHCPnack.

После завершения работы клиент может освободить занимаемый адрес путем отправления серверу сообщения DHCPrelease.

4.4. Маршрутизация в *IP*-сетях

4.4.1. Общие принципы маршрутизации в объединенных сетях

Объединенная компьютерная сеть состоит из множества физических сетей, связанных посредством специальных компьютеров, называемых маршрутизаторами. Каждый маршрутизатор напрямую подключается как минимум к двум сетям. Компьютер пользователя (*хост*) обычно подключен только к одной сети. В процессе доставки IP-дейтаграмм до конечного получателя могут участвовать не только специализированные маршрутизаторы, но и хосты, за которыми работают пользователи. На рисунке 4.17 показан вариант сети Б, в которой находятся два маршрутизатора М1 и М3, каждый из которых может обеспечить доставку дейтаграмм от хоста Б1 к хосту А1 сети А. В этом случае, несмотря на то, что хост Б1 имеет только одно сетевое подключение, он должен самостоятельно осуществить маршрутизацию к хосту А либо через маршрутизатор М1 или М3.

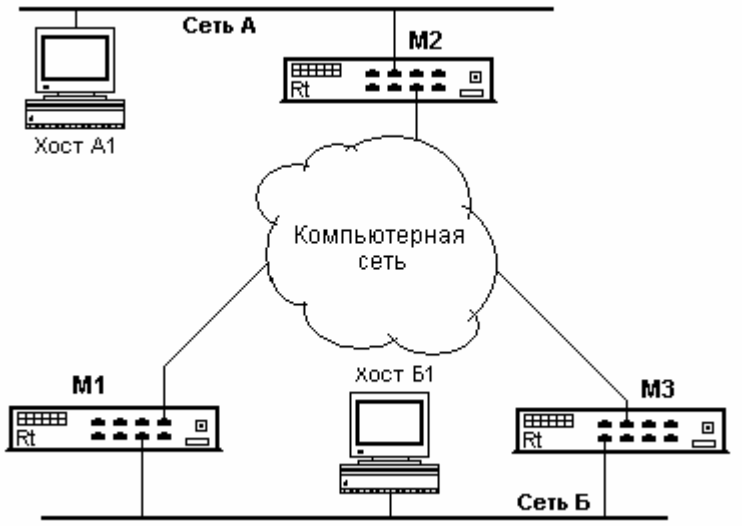


Рисунок 4.17 – Схема определения маршрута хостом и маршрутизаторами

В сетях с коммутацией пакетов под термином **маршрутизация** (*routing*) понимается процесс выбора пути, по которому должны передаваться пакеты к конечному получателю. Сведения, которые используются для выбора пути следования пакетов до конечного получателя, называются маршрутной информацией.

Программы маршрутизации должны анализировать загрузку сети, длину дейтаграммы, а также запрошенный в ее заголовке тип обслуживания, и на основе полученных данных выбирать оптимальный маршрут. Однако

большинство программ маршрутизации не обладают столь развитой логикой и выбирают маршрут следования дейтаграммы, исходя из ряда сделанных допущений о его минимальной длине.

Существует два способа маршрутизации: **прямая** (*direct delivery*) и **непрямая** (*indirect delivery*) доставка дейтаграмм. Прямая доставка происходит при передаче дейтаграммы от источника до получателя, находящихся в одной физической сети. Выполняя непосредственную доставку дейтаграммы, источник использует процедуру ARP и определяет физический адрес станции назначения.

Непрямая доставка производится в случае, когда конечный получатель дейтаграммы располагается в другой физической сети. При этом отправитель пересылает дейтаграмму ближайшему узлу маршрутизации, выполняющему ее дальнейшую доставку до конечного получателя. В сети Интернет непрямая доставка применяется намного чаще прямой.

Чтобы определить, находится ли получатель в одной физической сети с отправителем, последний должен выделить номер сети из IP-адреса получателя и сравнить его с номером сети, выделенным из собственного IP-адреса. Совпадение номеров означает, что дейтаграмма может быть послана получателю напрямую.

В объединенной сети прямая доставка выполняется на заключительном этапе пересылки любой дейтаграммы от узла к хосту, независимо от того, через какое количество сетей и промежуточных маршрутизаторов она прошла. Последний из маршрутизаторов, находящийся на пути следования дейтаграммы и подключенный к одной физической сети с конечным получателем, выполняет ее прямую доставку до получателя. Таким образом, прямую доставку дейтаграмм следует рассматривать как частный случай общего процесса маршрутизации, выполняемого на всем этапе следования дейтаграммы от отправителя до конечного получателя. При прямой доставке дейтаграмма не проходит через промежуточные узлы маршрутизации.

Процесс непрямой доставки является намного сложнее по сравнению с прямой доставкой дейтаграмм, поскольку отправителю нужно определить адрес ближайшего маршрутизатора, находящегося в одной с ним сети, которому должна быть послана дейтаграмма.

В случае если одному из компьютеров сети нужно отправить дейтаграмму другому компьютеру, он инкапсулирует ее в канальный кадр и пересылает его по физической сети ближайшему маршрутизатору своей сети, так как к каждой физической сети подключен как минимум один маршрутизатор. Приняв дейтаграмму, маршрутизатор с помощью своего программного обеспечения извлекает ее из сетевого кадра и передает на обработку программе маршрутизации протокола IP. После этого программа маршрутизации выполняют поиск адреса следующего маршрутизатора, находящегося на

пути следования пакета до конечного получателя. Как только будет определен IP-адрес следующего узла маршрутизации, дейтаграмма снова помещается в сетевой пакет и пересылается этому узлу по соответствующему участку физической сети. Этот процесс повторяется многократно до тех пор, пока дейтаграмма не дойдет до физической сети, в которую включен получатель. Далее дейтаграмма отправляется конечному получателю методом прямой доставки.

Обычно маршрутизация дейтаграмм в объединенной сети выполняется с помощью специальных **таблиц межсетевой маршрутизации** (*Internet routing table*), которые иногда называют **таблицами IP-маршрутизации** (*IP routing table*). В них хранится информация о возможных путях следования дейтаграмм и способах их достижения. Поскольку в процесс маршрутизации вовлечены как компьютеры пользователя, так и сетевые маршрутизаторы, то такие таблицы должны храниться на каждом компьютере объединенной сети, независимо от выполняемых ею функций. В момент отправки дейтаграммы с компьютера пользователя запущенная на нем программа маршрутизации (она является частью программ поддержки протокола IP) с помощью таблицы маршрутизации определяет узел сети, которому следует послать эту дейтаграмму. В связи с тем, что в объединенной сети на межсетевом уровне происходит доставка пакетов от одной подсети к другой, таблица маршрутизации должна содержать только префиксы адреса сети, а не полные IP-адреса всех ее компьютеров.

Использование выделенного из адреса получателя вместо полного адреса узла *префикса*, идентифицирующего сеть, повышает эффективность маршрутизации и сокращает размер соответствующих таблиц. Кроме того, подобный подход позволяет скрыть информацию о структуре сети, сосредоточить данные о конкретных узлах в рамках локальной среды, к которой они подключены. Обычно в таблице маршрутизации содержатся пары значений (*B, N*), где *B* представляет IP-адрес сети получателя, а *N* является IP-адресом "следующего" по порядку маршрутизатора, расположенного на пути движения пакетов до сети назначения. Маршрутизатор N называется ближайшей **точкой перехода** (*next hop*), а сам способ хранения в таблице маршрутизации адреса ближайшей точки перехода для каждого получателя получил название маршрутизации на шаг вперед (*next-hop routing*). Таким образом, в таблице маршрутизации, хранящейся на узле *N*, содержатся данные о пути следования дейтаграмм от узла *N* до ближайшей точки перехода в направлении сети получателя. Следует заметить, что маршрутизатор *N* не располагает данными о полном маршруте дейтаграммы к конечному получателю.

Размер таблицы маршрутизации узла *N* зависит от количества физических сетей в объединенной сети. Увеличение количества элементов в этой таблице происходит только в случае подключения к объединенной сети но-

вых подсетей. Однако на размер таблицы маршрутизации и на ее содержимое не оказывает влияния количество индивидуальных компьютеров, подключенных ко всем сетям.

Выше отмечалось, что в сетях TCP/IP маршрутизация выполняется на основе адреса сети, а не полных адресов отдельных ее узлов. Тем не менее, в большинстве реализаций протокола IP предусмотрена возможность выполнения маршрутизации для адресов отдельных узлов. Это позволяет сетевым администраторам более точно распределять потоки данных в сети, тестировать ее отдельные участки сети, а также управлять правами доступа к сетевым ресурсам. Возможность задать индивидуальный маршрут дейтаграмм до конкретного компьютера особенно полезна при отладке сетевых подключений или проверке таблиц маршрутизации.

Алгоритм маршрутизации представляется в следующем виде.

1: Извлечь IP-адрес (Dest*ID*) места назначения из дейтаграммы.

2: Выделить IP-адрес сети назначения (Net*ID*).

3: ЕСЛИ Net*ID* соответствует какому-либо адресу данной подсети, выполнить прямую доставку дейтаграммы по этому адресу.

4: ИНАЧЕ, ЕСЛИ Net*ID* присутствует в маршрутной таблице, то послать дейтаграмму на маршрутизатор, указанный в таблице.

5: ИНАЧЕ, ЕСЛИ описан маршрут по умолчанию, то послать дейтаграмму к стандартному маршрутизатору, адрес которого берется из таблицы.

6: ИНАЧЕ выдать сообщение об ошибке маршрутизации.

Интернет, как уже неоднократно отмечалось, является всемирной сетью, которая состоит из многих независимых сетей. К наиболее крупным составным частям объединенной сети относятся так называемые автономные системы. **Автономной системой AS** (*Autonomous Systems*) называют группу сетей и маршрутизаторов (*routers*), объединенных общей политикой маршрутизации и находящихся под управлением одного административного органа, например Укртелекома. Под *политикой маршрутизации* понимают административное регулирование маршрутов путем внесения ограничений на передачу и использования маршрутной информации.

AS являются автономными относительно распределения маршрутной информации, т.е. информация о достижимости IP-адресов и префиксов IP-адресов (маршрутизаторов) не передается другим автономным системам. Следует заметить, что автономная сеть в свою очередь может состоять из нескольких сетей, однако управление ими всегда осуществляется единым административным органом. Если маршрутная политика автономной системы позволяет передавать через свои сети транзитный трафик других AS, то такая автономная система называется *транзитной*.

Для определения маршрута внутри AS применяется **внутренние протоколы** маршрутизации **IGP** (*Interior Gateway Protocols*). Наиболее распространенными протоколами внутренней маршрутизации являются протоколы **RIP** (*Routing Information Protocol*) и **OSPF** (*Open Shortest Path First*). В старых системах IGP иногда все еще используется протокол HELLO. Фирмой Cisco был разработан протокол **IGRP** (Interior Gateway Routing Protocol) альтернативный протоколу RIP. Затем Cisco представила улучшенный вариант IGRP – протокол **EIGRP** (Enhanced Interior Gateway Routing Protocol), свободный от основного недостатка протоколов векторов расстояний – зацикливанию маршрутов.

Каждая AS имеет однозначный AS-номер. Эти номера распределяются международным информационным центром NIC (*Network Information Center*). Организация NIC в свою очередь подразделяется на региональные отделения (по странам).

Автономные системы, объединенные между собой при помощи внешних маршрутизаторов, и образуют глобальную сеть Интернет (рисунок 4.18).

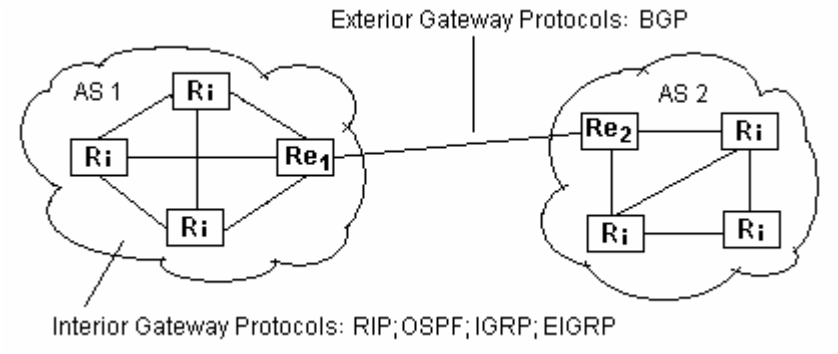


Рисунок 4.18 – Область действия внутренних и внешних протоколов маршрутизации

Изначально, по терминологии Интернет, внутренние и внешние маршрутизаторы получили название **шлюз** (*Gateway*). Эти названия часто используются и по настоящее время как синонимы маршрутизаторам. Внешние маршрутизаторы называют также **пограничными** (*Border*) маршрутизаторами.

Два маршрутизатора, которые обмениваются информацией о маршрутах, называются внутренними соседями в том случае, если они принадлежат к одной автономной системе, и внешними, если они принадлежат к различным автономным системам. На рисунке 4.18 маршрутизаторы Ri являются внутренними для автономных систем AS1 и AS2 соответственно. Re_1 и Re_2 совмещают функции внешнего и внутреннего маршрутизаторов. Маршрутизатор Re_1 представляет для автономной системы AS2 маршруты к сетям, которые находятся в автономной системе AS1. Аналогичную функцию выпол-

няет маршрутизатор Re_2 по отношению к маршрутам AS1.

Информационное взаимодействие между компонентами различных автономных систем может быть выполнено только через специальную область, которая предназначена для интеграции всей системы в целом (рисунок 4.19). Такая область, называется *магистральной* или **опорной сетью** (*Backbone Area*).

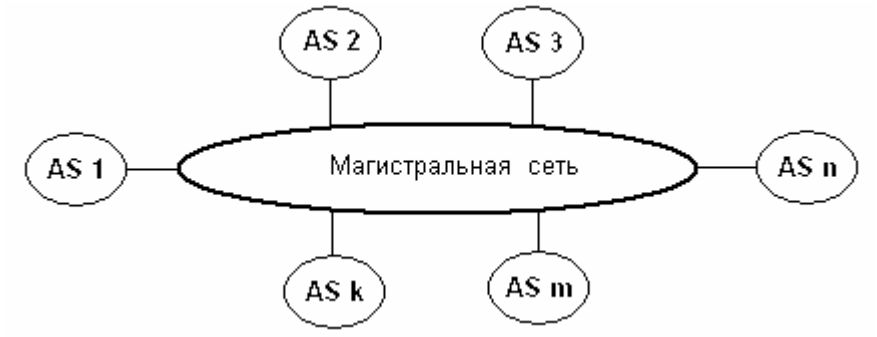


Рисунок 4.19 – Взаимодействие автономных сетей через магистральную сеть

Магистральная сеть в свою очередь может иметь различную топологию (кольцевую, многосвязную и пр.). Такая сеть строится на основе высокоскоростных маршрутизаторов и линий связи. Маршрутизация IP-пакетов в глобальной сети Интернет может осуществляться только при возможности обмена маршрутной информацией между внешними маршрутизаторами. Для этого был разработан протокол **внешней маршрутизации EGP** (*Exterior Gateway Protocol*). В соответствии с этим протоколом автономные системы сообщают глобальной сети, какие IP-префиксы они используют. В качестве внешнего протокола EGP в настоящее время применяется протокол **BGP4** (*Border Gateway Protocol*) четвертой версии.

Следует подчеркнуть, что протоколы маршрутизации не осуществляют маршрутизацию дейтаграмм. Маршрутизация в любом случае производится модулем IP согласно записям в таблице маршрутов. Протоколы маршрутизации на основании тех или иных алгоритмов динамически редактируют таблицу маршрутов, т.е. вносят и удаляют записи. При этом часть записей может вноситься администратором вручную. Управление маршрутизатором осуществляется путем подачи администратором команд с компьютера, подключенного к консольному порту маршрутизатора или удаленным способом посредством службы Telnet. В последнем случае команды передаются на IP-адрес одного из портов маршрутизатора.

4.4.2. Дистанционно-векторный протокол RIP

Дистанционно-векторный протокол маршрутизации **RIP** (*Routing Information Protocol*) предназначен для сравнительно небольших и относительно однородных сетей. Для нахождения оптимального пути используется алгоритм Белмана-Форда (см. п.1.4.3). Маршрут в данном алгоритме характеризуется **вектором расстояния** до места назначения (*место назначение* – направление вектора; *метрика* – модуль вектора). Предполагается, что каждый маршрутизатор является отправной точкой нескольких маршрутов до сетей, с которыми он связан. Описания этих маршрутов хранится в специальной маршрутной таблице. Таблица маршрутизации RIP содержит по одной записи на каждый обслуживаемый компьютер (на каждый маршрут). Запись должна включать в себя следующее:

1) IP-адрес места назначения;

2) метрику маршрута (от 1…15 – число шагов (*hops*) до места назначения);

3) IP-адрес ближайшего маршрутизатора (*gateway*) по пути к месту назначения;

4) таймеры (счетчики времени) маршрута.

Каждый маршрут содержит счетчики тайм-аута и "сборщика мусора". Счетчик тайм-аута сбрасывается в ноль в момент коррекции или инициализации маршрута. Если с момента последней коррекции прошло 3 мин или получено сообщение, что расстояние равно 16, то маршрут закрывается, однако запись о нем не стирается до наступления времени "сборки мусора".

Функции RIP-модуля в маршрутизаторе состоят в рассылке, получении и обработке векторов расстояний до IP-сетей, находящихся в области действия протокола. Каждый маршрутизатор один раз в 30 секунд посылает всем смежным узлам, с которыми он непосредственно связан, сообщение об изменении своей **маршрутной таблицы** (*routing update*). Маршрутизатор-получатель просматривает таблицу изменений, и если в таблице присутствует новый путь или сообщение о более коротком маршруте, либо произошли изменения длин пути, то эти изменения фиксируются получателем в своей маршрутной таблице. Если сетевой узел в течение определенного времени не получил маршрутное сообщение от какого-либо соседа, то это воспринимается как авария, и маршрут данного направления исключается из таблицы.

По протоколу RIP сообщения инкапсулируются в UDP-дейтаграммы, при этом передача осуществляется через порт 520. В качестве метрики RIP использует **число шагов** (*хопов*) до цели. Если между отправителем и приемником расположено три маршрутизатора (*gateway*), то считается, что между ними 4 шага. Такой вид метрики не учитывает различий в пропускной

способности или загруженности отдельных сегментов сети. Применение вектора расстояния не может гарантировать оптимальность выбора маршрута, поскольку, например, два шага по сегментам сети Ethernet обеспечат большую пропускную способность, чем один шаг через последовательный канал на основе интерфейса *RS*-232C.

Формат сообщения протокола RIP имеет вид, показанный на рисунке 4.20. Поле "**Команда**" определяет вид маршрутизационного сообщения. Имеется шесть кодов команд: 1– Запрос на получение частичной или полной маршрутной информации; 2 – Отклик, содержащий информацию о расстояниях из маршрутной таблицы отправителя; 3 – Включение режима трассировки; 4 – Выключение режима трассировки; 5-6 – Зарезервированы для внутренних целей фирмы *Sun Microsystem*. Команды 3 и 4 применялись в ранних версиях протокола.

Рисунок 4.20 – Формат пакета протокола маршрутизации RIP

Поле "**Версия**" для RIP равно 1 (для RIP-2 двум). Поле "**Набор протоколов сети *i*** " определяет набор протоколов, которые используются в соответствующей сети (для Интернет это поле имеет значение 2). Поле "**Стоимость до сети *i*** " содержит целое число шагов (от 1 до 15) до данной сети. В одном маршрутном сообщении может присутствовать информация о 25 маршрутах.

При реализации RIP можно выделить следующие режимы.

Инициализация – определение всех функционирующих интерфейсов путем посылки запросов, получение таблиц от других маршрутизаторов. Часто используются широковещательные запросы.

Получен запрос. В зависимости от типа запроса адресату высылается полная таблица маршрутизации или проводится индивидуальная обработка.

Получен отклик. Проводится коррекция таблицы маршрутизации (удаление, исправление, добавление).

Регулярные коррекции. Каждые 30 секунд вся или часть таблицы маршрутизации посылается всем соседним маршрутизаторам. Могут посылаться и специальные запросы при локальном изменении таблицы.

Неотъемлемой частью протокола маршрутизации RIP является реализация правила "расщепленный горизонт" (*Split horizon*). Оно предназначено для предотвращения появления циклических маршрутов в сети. В соответствии с этим правилом маршрутизатор разделяет свои маршруты на столько групп, сколько у него имеется активных интерфейсов. Причем, обновления для маршрутов, которые были получены через *k*-й интерфейс, не должны передаваться через этот же интерфейс.

Использование процедуры *Split horizon* позволяет избежать появления зацикленного маршрута у двух шлюзов. Однако возможно возникновение ситуации, когда в циклическом маршруте участвуют три шлюза. Использование процедуры *Split horizon* не сможет предотвратить появление такой петли, поскольку сообщения о маршруте поступают не от того маршрутизатора, которому передаются сообщения модификации. Следовательно, эта петля будет разорвана только в случае, когда метрика циклического маршрута достигает заданного максимального значения.

Протокол RIP достаточно прост в эксплуатации и конфигурации, за что он и получил широкое распространение. Однако ему присущ ряд недостатков.

1. RIP не работает с адресами подсетей. Если нормальный 16-битовый идентификатор хоста класса B не равен 0, RIP не может определить является ли ненулевая часть подсетевым ID, или полным IP-адресом.

2. RIP требует много времени для восстановления связи после сбоя в маршрутизаторе (минуты). В процессе установления режима возможно зацикливание.

3. Число шагов важный, но не единственный параметр маршрута, да и 15 шагов становится ограничением для современных сетей.

Новой версией RIP, которая в дополнение к широковещательному режиму поддерживает **групповую адресацию** (мультикастинг), является протокол RIP-2. Он передает информацию о масках сетей, позволяя работать с бесклассовой адресацией. На рисунке 4.21 показан формат сообщения для протокола RIP-2. Поле "**Маршрутный демон**" служит идентификатором резидентной программы-маршрутизатора, а поле "**Метка маршрута**" используется для поддержки внешних протоколов маршрутизации, сюда записываются коды автономных систем.

Рисунок 4.21 – Формат сообщений протокола RIP-2

4.4.3. Протокол маршрутизации с учетом состояния линий

Альтернативой протокола RIP для внутренней маршрутизации сетей стал протокол **OSPF** (*Open Shortest Path First*). Он гораздо сложнее RIP-протокола, однако OSPF может функционировать в сетях любой сложности и не имеет ограничений, характерных для RIP. Время, используемое на построение таблиц маршрутизации и загрузки сети служебной информацией, в среднем меньше по сравнению с тем, что потребовал бы RIP для такой же системы. Если между узлами сети существуют несколько маршрутов с одинаковыми или близкими по значению метриками, то протокол OSPF позволяет **разделять часть трафика** по этим маршрутам обратно пропорционально значениям метрик. Например, если имеются два пути с метриками 1 и 2, то две трети трафика будет направлено по первому из них, а оставшаяся треть – по второму. Такая способность протокола позволяет равномерно загружать направления сети и уменьшать среднее время задержки дейтаграмм. Кроме этого, переходные процессы в OSPF завершаются быстрее, чем в RIP.

OSPF представляет собой **протокол учета состояния маршрута**. Маршрутизация выполняется по *алгоритму Дийкстры*. В качестве метрики используется коэффициент качества обслуживания QoS (*Quality of Service*). Каждый маршрутизатор обладает полной информацией о состоянии всех интерфейсов всех маршрутизаторов (узлов коммутации) автономной системы. Протокол OSPF реализован программным модулем – **демоном маршрутизации** *gated*, который поддерживает также RIP и внешний протокол маршру-

тизации BGP. Качество обслуживания (*QoS*) характеризуется следующими параметрами:

- пропускной способностью канала;
- задержкой (временем распространения пакета);
- числом дейтаграмм, стоящих в очереди для передачи;
- загрузкой канала;
- требованиями безопасности;
- типом трафика;
- числом шагов до места назначения;
- возможностями промежуточных связей (например, многовариантность достижения адресата);
- надежностью передачи пакетов.

Доминирующими являются три характеристики: **задержка**, **пропускная способность** и **надежность**. На практике чаще всего метрика связи в OSPF определяется как количество секунд, требуемых для передачи 100 Мбит по каналу, через который проложен маршрут. Например, метрика сети на основе 10BASE-T *Ethernet* равна 10, метрика канала модемной связи со скоростью 56 кбит/с составляет 1785, а канала со скоростью 100 Мбит и выше равна 1.

Для облегчения управления сетями и обеспечения возможности их расширения протокол OSPF позволяет администрации сетевого центра разделить свою сеть и маршрутизаторы на **области** *(areas)*. Все области изолированы друг от друга. Это означает, что сведения о топологической схеме области можно скрыть от других зон. В результате несколько групп компьютеров, находящихся в пределах одного сетевого центра, могут использовать протокол OSPF для маршрутизации пакетов между собой. При этом каждая из групп может независимо от других областей изменять топологию своей внутренней сети. Для транспортных целей OSPF применяет IP непосредственно, т.е. не привлекая протоколы UDP или TCP. OSPF имеет свой код в протокольном поле IP-заголовка. Код типа обслуживания TOS (*type of service*) в IP-пакетах, содержащих OSPF-сообщения, равен нулю, значение типа обслуживания TOS здесь задается в самих пакетах OSPF.

Маршрутизация в протоколе OSPF определяется IP-адресом и типом сервиса. В связи с тем, что протокол не требует инкапсуляции пакетов, существенно облегчается управление сетями с большим количеством мостов и сложной топологией (исключается циркуляция пакетов, сокращается транзитный трафик). Автономная система может быть поделена на **отдельные области**, каждая из которых становится объектом маршрутизации, а внутренняя структура снаружи не видна. Этот прием позволяет значительно сократить необходимый объем маршрутной базы данных. В OSPF используется термин **магистральная сеть** (*backbone*), обозначающий среду для комму-

никаций между выделенными областями. Протокол OSPF работает лишь в пределах автономной системы.

В стеке протоколов TCP/IP протокол OSPF находится непосредственно над протоколом IP, его код равен 89. Поэтому если значение поля "Протокол" IP-дейтаграммы равно 89, то данные дейтаграммы являются сообщением OSPF и передаются OSPF-модулю для обработки. Соответственно размер OSPF сообщения ограничен максимальным размером дейтаграммы.

При передаче OSPF-пакетов фрагментация не желательна, но не запрещается. Для передачи статусной информации OSPF использует широковещательные сообщения *Hello*. Повышение безопасности обеспечивается авторизацией процедур. OSPF-протокол требует резервирования двух групповых адресов: 224.0.0.5 – предназначен для обращения ко всем маршрутизаторам, поддерживающим этот протокол; 224.0.0.6 – служит для обращения к специально выделенному маршрутизатору. Любое сообщение OSPF начинается с 24-октетного заголовка (рисунок 4.22):

Рисунок 4.22 – Формат заголовка сообщений для протокола маршрутизации OSPF

Поле "**Версия**" определяет версию протокола (= 2). Поле "**Тип**" идентифицирует функцию сообщения, в частности: 1– Hello (используется для проверки доступности маршрутизатора); 2 – Описание базы данных (топология); 3 – Запрос состояния канала; 4 – Изменение состояния канала; 5– Подтверждение получения сообщения о статусе канала.

Поле "**Длина пакета**" указывает длину блока в октетах, включая заголовок. "**Идентификатор области**" – 32-битный код, задающий область, которой данный пакет принадлежит. Все OSPF-пакеты ассоциируются с той или иной областью. Большинство из них не преодолевает более одного шага. Пакеты, перемещающиеся по виртуальным каналам, помечаются идентификатором опорной (магистральной) области (*backbone*) 0.0.0.0.

Поле "**Контрольная сумма**" содержит проверочную сумму IP-пакета, включая поле типа идентификации. Поле "**Тип идентификации**" может принимать значения 0 при отсутствии контроля доступа, и 1 при его наличии. В дальнейшем функции поля предполагается расширить.

Для взаимообмена данными между соседними маршрутизаторами протокол OSPF использует сообщения типа *Hello.* Важную функцию в этих сообщениях выполняет однобайтное поле "Опции", служащее для объявления состояния канала и описания базы данных Структура пакетов этого типа показана на рисунке 4.23.

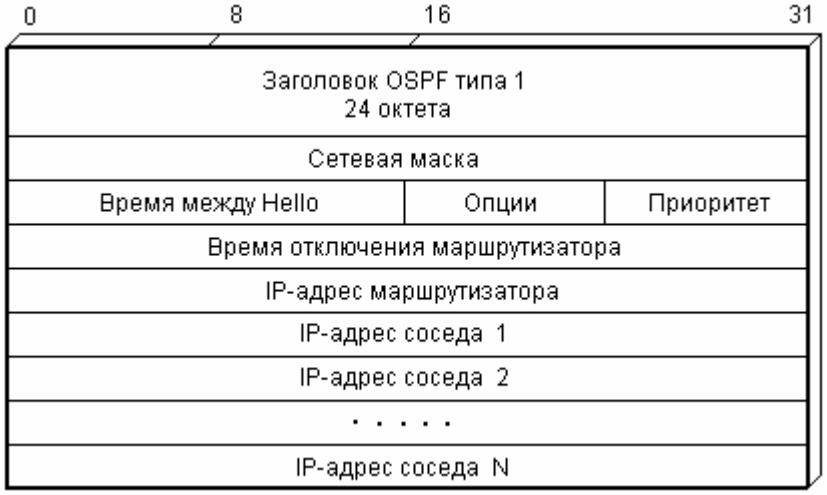


Рисунок 4.23 – Формат сообщения Hello в протоколе OSPF

Особую роль в поле "Опции" играют младшие биты E и T: Бит **E** характеризует возможность внешней маршрутизации и имеет значение только в сообщениях типа *Hello*, в остальных сообщениях этот бит должен быть обнулен. Если E=0, то данный маршрутизатор не будет посылать или принимать маршрутную информацию от внешних автономных систем. Бит **T** определяет сервисные возможности маршрутизатора. Если T=0, это означает, что маршрутизатор поддерживает только один вид услуг (тип сервиса *TOS*=0) и он не пригоден для маршрутизации с учетом вида услуг. Такие маршрутизаторы, как правило, не используются для транзитного трафика.

Поле "**Сетевая маска**" сообщения *Hello* соответствует маске подсети данного интерфейса. Например, если интерфейс принадлежит сети класса B и третий байт адреса служит для выделения нужной подсети, то сетевая маска будет иметь вид *0xFFFFFF00*.

Поле "**Время между Hello**" содержит значение времени в секундах, между сообщениями *Hello*. Поле "**Опции**" характеризует возможности, которые предоставляет данный маршрутизатор. Поле "**Приоритет**" задает уровень приоритета маршрутизатора, используемый при выборе резервного (*backup*) маршрутизатора. Если приоритет равен нулю, данный маршрутизатор никогда не будет реализован в качестве резервного. Поле "**Время отключения маршрутизатора**" определяет временной интервал в секундах, по истечении которого "молчащий" маршрутизатор считается вышедшим из

строя. IP-адреса маршрутизаторов, записанные в последующих полях, указывают место, куда следует послать данное сообщение. Поля "**IP-адрес соседа *k***" образуют список адресов соседних маршрутизаторов, откуда за последнее время были получены сообщения *Hello*.

Маршрутизаторы обмениваются сообщениями из баз данных OSPF, чтобы инициализировать, а в дальнейшем актуализовать свои базы данных, характеризующие топологию сети. Обмен происходит в режиме клиент-сервер. Клиент подтверждает получение каждого сообщения. Формат пересылки записей из базы данных изображен на рисунке 4.24.

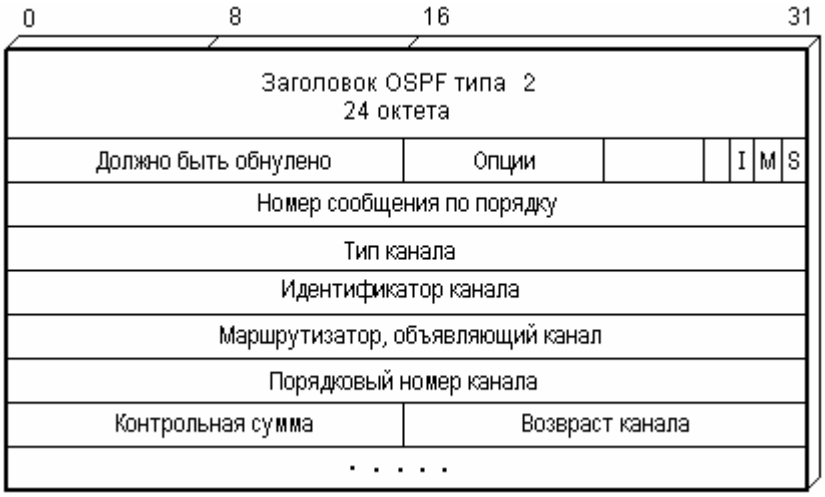


Рисунок 4.24 – Формат OSPF-сообщений о маршрутах

Поля, начиная с "**Тип канала**", повторяются для каждого описания канала. Если размер базы данных велик, ее содержимое может пересылаться по частям. Для реализации этого используются биты **I** и **M**. Бит I устанавливается в 1 в стартовом сообщении, а бит M принимает единичное состояние для сообщений, которые являются продолжением. Бит **S** определяет, кем послано сообщение (S=1 для сервера, S=0 для клиента, этот бит иногда имеет имя MS).

Поле "**Номер сообщения по порядку**" служит для контроля пропущенных в процессе обмена информацией блоков. Первое сообщение содержит в этом поле случайное целое число M, последующие: M+1, M+2,...M+L. Поле "**Тип канала**" может принимать значения, приведенные в таблице 4.2. Поле "**Идентификатор канала**" указывает вид идентификатора, в качестве которого может быть IP-адрес маршрутизатора или сети. Маршрутизатор, объявляющий канал, определяет адрес этого маршрутизатора. Поле "**Порядковый номер канала**" позволяет маршрутизатору контролировать порядок прихода сообщений и их потерю. Поле "**Возраст канала**" задает время в секундах с момента установления связи. После обмена сообщениями с соседя-

ми маршрутизатор может выяснить, что часть данных в его базе устарела. Он может послать своим соседям запрос с целью получения свежей маршрутной информации о каком-то конкретном канале связи. Сосед, получивший запрос, высылает необходимую информацию.

Таблица 4.2 – Коды типов состояния каналов (LS)

LS–тип	Описание объявления о маршруте
1	Описание каналов маршрутизатора, т.е. состояния его интерфейсов.
2	Описание сетевых каналов. Это перечень маршрутизаторов, непосредственно связанных с сетью.
3 или 4	Сводное описание каналов, куда входят маршруты между отдельными областями сети. Информация поступает от пограничных маршрутизаторов этих областей. Тип 3 приписан маршрутам, ведущим к сетям, а тип 4 характеризует маршруты, ведущие к пограничным маршрутизаторам автономной системы.
5	Описания внешних связей автономной системы. Такие маршруты начинаются в пограничных маршрутизаторах AS.

Маршрутизаторы посылают широковещательные (или групповые) сообщения об изменении состояния своих непосредственных связей. Сообщения об изменениях маршрутов могут быть вызваны следующими причинами:

1) возраст маршрута достиг предельного значения;
2) изменилось состояние интерфейса;
3) произошли изменения в маршрутизаторе сети;
4) произошло изменение состояния одного из соседних маршрутизаторов;
5) изменилось состояние одного из внутренних маршрутов (появление нового, исчезновение старого и т.д.);
6) изменение состояния межзонного маршрута;
7) появление нового маршрутизатора, подключенного к сети;
8) вариация виртуального маршрута одним из маршрутизаторов;
9) возникли изменения одного из внешних маршрутов;
10) маршрутизатор перестал быть пограничным для данной автономной системы (например, перезагрузился).

Маршрутизатор, получивший OSPF-пакет, посылает подтверждение его приема. Возможно подтверждение одним пакетом получения нескольких объявлений о состоянии линий. Адресом места назначения этого пакета может быть индивидуальный маршрутизатор, их группа или все маршрутизаторы автономной системы.

Объявление сетевых связей относится к типу 2 кодов состояния каналов. Сообщения посылаются для каждой транзитной сети в автономной сис-

теме. Транзитной считается сеть, которая имеет более одного маршрутизатора и административная политика автономной системы позволяет передавать через свои сети транзитный трафик других AS. Сообщение о сетевых связях должно содержать информацию обо всех маршрутизаторах, подключенных к сети, включая тот, который рассылает эту информацию. Расстояние от сети до любого подключенного маршрутизатора равно нулю для всех видов сервиса (TOS), поэтому поля TOS и метрики в этих сообщениях отсутствуют.

Информация об адресатах в пределах автономной системы передается LS-сообщениями типа 3 и 4. Для IP-сетей используется тип 3. В этом случае в качестве идентификатора состояния канала применяется IP-адрес сети. Если же адресатом является пограничный маршрутизатор данной AS, то используется LS-сообщение типа 4, а в поле идентификатора состояния канала записывается OSPF-идентификатор этого маршрутизатора.

Объявления внешних маршрутов относится к пятому типу. Эта информация рассылается пограничными маршрутизаторами. Сведения о каждом внешнем адресате, известном маршрутизатору, посылаются независимо. Такой вид описания используется и для **маршрутов по умолчанию**, для которых идентификатор состояния канала устанавливается равным **0.0.0.0**.

Маршрутная таблица OSPF включает следующие поля:

- IP-адрес места назначения и маску;
- тип места назначения (сеть, граничный маршрутизатор и т.д.);
- тип функции (возможен набор маршрутизаторов для каждой из функций TOS);
- область (описывает область, связь с которой ведет к цели; возможно несколько записей данного типа, если области действия граничных маршрутизаторов перекрываются);
- тип пути (характеризует путь как внутренний, межобластной или внешний, ведущий к автономной системе AS);
- цена маршрута до цели;
- очередной маршрутизатор, куда следует послать дейтаграмму;
- объявляющий маршрутизатор (используется для межобластных обменов и для связей автономных систем друг с другом).

К преимуществам протокола маршрутизации OSPF следует отнести:

1) возможность применения для любого получателя нескольких маршрутных таблиц, по одной на каждый вид IP-операции;

2) каждому интерфейсу присваивается безразмерная цена, учитывающая пропускную способность, время транспортировки сообщения; собственная цена (коэффициент качества) может быть присвоена любой IP-операции;

3) при существовании эквивалентных маршрутов OSFP распределяет поток равномерно по этим маршрутам;

4) поддерживается адресация подсетей (разные маски для разных мар-

шрутов);

5) при связи точка-точка не требуется IP-адрес для каждого из конечных интерфейсов;

6) применение групповой рассылки вместо широковещательных сообщений снижает загрузку значительной части сегментов.

Недостаток протокола OSPF состоит в том, что трудно получить информацию о предпочтительности каналов для узлов, поддерживающих другие протоколы, или протоколы со статической маршрутизацией. Кроме того, OSPF является только внутренним протоколом.

4.4.4. Протоколы внешней маршрутизации

Маршрутизация между автономными системами осуществляется **пограничными** (*Border*) **маршрутизаторами**, таблицы маршрутов которых составляются с помощью протоколов внешней машрутизации EGP (*Exterior Gateway Protocol*). Сложность внешней маршрутизации состоит в том, что у разных автономных систем возможно использование различных метрик и ограничений на прокладку путей, что не позволяет разработать эффективный алгоритм маршрутизации. Кроме того, при расчете маршрутов протоколы внешней маршрутизации должны учитывать не только топологию сети, но и *политические ограничения*, вводимые администрацией автономных систем на маршрутизацию через свои сети трафика других автономных систем. Например, пакеты (из соображений безопасности) не должны проходить транзитом через некоторые автономные системы, хотя путь к получателю по такому маршруту является более коротким или экономичным. Необходимые ограничения вынуждены вноситься на каждом отдельном маршрутизаторе администратором сети при начальной конфигурации вручную или с помощью скриптов.

В настоящее время известны два типа протоколов внешней маршрутизации: **BGP** (*Border Gateway Protocol*) и **EGP** (*Exterior Gateway Protocol*). Протокол внешней маршрутизации EGP характеризуется следующими признаками::

- использование механизма установления отношений между маршрутизаторами;
- применение маршрутизаторами EGP специального механизма для определения статуса своих партнеров по протоколу;
- периодический обмен информацией между маршрутизаторами EGP о достижимости сетей посредством передачи сообщений об обновлениях маршрутов.

В процессе установления взаимных отношений, а также для выполнения

обмена маршрутной информацией, маршрутизаторы EGP передают специальные сообщения с подтверждением их корректного приема. В зависимости от ситуации, передаются сообщения, содержащие следующее:

- информацию о соседях (*Neighbor Acquisition Messages*);
- сведения о достижимости соседей (*Neighbor Reachability Messages*);
- запрос данных о состоянии маршрута (*Poll Request Messages*);
- сведения об изменении маршрута (*Routing Update Messages*).

Внешние маршрутизаторы могут находиться в активном или пассивном состоянии. В активном состоянии маршрутизатор периодически посылает сообщения *Hello* вместе с обновлениями маршрутов и ожидает ответа от соседа. Если маршрутизатор находится в пассивном режиме, он может использовать содержимое поля STATUS принимаемого сообщения Hello для определения состояния соседа вместо того, чтобы периодически опрашивать его. Обычно оба соседних маршрутизатора находятся в активном состоянии.

Протоколу EGP свойственен ряд существенных недостатков.

1. Маршрутизатор EGP представляет только один путь до каждой сети. Это делает невозможным использование процедур динамического перераспределения нагрузки между параллельными каналами.
2. Маршрутизатор EGP не поддерживает внеклассовые сети.

Протокол пограничных шлюзов BGP (*Border Gateway Protocol*) относится к более новым, по сравнению с протоколом EGP, протоколам внешней маршрутизации автономных систем. Хотя оба этих протокола построены по примерно одинаковой схеме, однако протокол BGP имеет ряд существенных преимуществ по отношению к EGP.

Отличительная особенность протокола BGP заключается в использовании **маршрутно-векторной маршрутизации** (*path-vector routing*). Это связано с тем, что у разных автономных систем могут быть различные метрики и ограничения. Поэтому при маршрутно-векторном способе не используют метрику маршрутизации. Маршрутизаторы просто обмениваются информацией о том, к каким сетям у них имеется доступ и какие автономные системы нужно пересечь, чтобы достичь места назначения. Маршрутизатор взаимодействует с другими маршрутизаторами по протоколу BGP только в том случае, если администратор явно указал при конфигурации, что эти маршрутизаторы являются его соседями.

Протокол BGP применяется для передачи информации о внутренних маршрутах между автономными системами. Он может быть использован для определения:

- маршрутов, которые соединяют данную автономную систему с одной или несколькими другими автономными системами (*Inter-autonomous system routing*);
- маршрута внутри автономной системы, когда несколько маршрутиза-

торов участвуют в процессе определения маршрута (*Intra-autonomous system routing*);

- маршрутов, проходящих через автономную систему, которая не участвует в процессе BGP (*Pass-through autonomous system*).

С целью обеспечения информационного обмена маршрутизаторы BGP используют сообщения стандартной формы. Для передачи таких сообщений в протоколе BGP предусматривается применение транспортного протокола **TCP** с **портом 179**. Сообщения BGP передаются в следующих случаях:

- начало сеанса (*Open*), используется для установки соседских отношений с другим маршрутизатором;
- для периодической проверки состояния соседа (*Keep Alive*);
- при изменении содержания таблицы маршрутов автономной системы (*Updat*e);
- при возникновении аварийной ситуации (*Notification*).

Каждое сообщение BGP состоит из заголовка и ряда специфических полей. Заголовок имеет фиксированную длину (19 байтов). Шестнадцать из них занимает **маркер**, два – **длина сообщения** и один байт – **тип** сообщения. В поле маркера может быть помещена информация, необходимая для выполнения операции аутентификации абонента. Отправитель может заполнить это поле значением, которое будет использоваться как часть механизма аутентификации, позволяющая получателю проверить подлинность отправителя. Если установление подлинности абонента не требуется, маркер заполняется единицами. В поле "Длина сообщения" помещается размер сообщения (вместе с заголовком), выраженный в байтах, а в поле "Тип" заносится код сообщения. Установлено четыре типа сообщений.

Первое сообщение, которое передается маршрутизатором BGP после установления соединения – TCP-сообщение *OPEN*. При успешном его прохождении противоположный маршрутизатор должен откликнуться сообщением KEEPALIVE ("Еще жив"). После этого возможны любые другие сообщения BGP.

После получения сообщения OPEN BGP-маршрутизатор должен задать величину времени сохранения взаимодействия маршрутизаторов – интервал времени между последовательными сообщениями об обновлении или подтверждении взаимодействия. Обычно выбирается меньшее из полученного в сообщении OPEN значения и собственного аналогичного параметра, определенного при конфигурации системы (в диапазоне 0…3 с). Время сохранения регламентирует максимальный интервал в секундах между сообщениями KEEPALIVE и UPDATE или между двумя UPDATE-сообщениями.

Каждому узлу в рамках BGP приписывается 4-октетный BGP-идентификатор. Он задается при инсталляции и идентичен для всех интерфейсов локальной сети. Если, например, два узла установили два канала свя-

зи друг с другом, то согласно правилам должен будет сохранен канал, начинающийся в узле, BGP-идентификатор которого больше. Предусмотрен механизм разрешения проблемы при равных идентификаторах.

Однобайтный код идентификации позволяет организовать систему доступа. Если он равен нулю, маркер всех сообщений заполняется единицами, а поле идентификационных данных имеет нулевую длину. При неравном нулю коде идентификации определяется процедура доступа и алгоритм вычисления кодов поля маркера. Для передачи маршрутной информации между BGP-шлюзами об изменении маршрутов используется сообщения типа UPDATE . Этот тип сообщения позволяет проинформировать об одном новом маршруте или объявить о закрытии группы маршрутов, причем объявление об открытии нового и закрытии старых маршрутов возможно в пределах одного сообщения.

4.4.5. Бесклассовая междоменная маршрутизация CIDR

Выше было показано, что Интернет-адреса разделены на пять классов (см. п. 4.1.3), из которых для адресации хостов применяются три: A, B и C. Недостатком классовой системы адресации является неравномерное использование адресного пространства внутри класса. Так, максимально возможное количество сетей класса B равно $2^{14} \approx 17\ 000$, а для сетей класса C оно составляет 2^{21}, т.е. более двух миллионов. В связи с быстрым ростом сети Интернет и спроса на адреса класса B адресное пространство этого класса практически распределено, хотя во многих случаях пространство адресов используется не эффективно. Например, если потребителю требуется построение сети с 860-тью хостами, то ему может быть выделен либо один адрес класса B, либо четыре адреса класса C. В первом случае из 2^{16} возможных адресов компьютеров будет применяться только восьмая часть. Остальные же адреса (свыше 63 тыс.) никем другим не могут быть использованы.

Выделение потребителю группы адресов класса C вместо одного адреса класса B является более рациональным, так как в этом случае невостребованными окажутся лишь 1024–860=164 адреса, что вполне приемлемо. Однако это приведет к увеличению количества записей в таблицах маршрутизаторов в четыре раза. Стремительный рост таблиц магистральных маршрутизаторов в начале 90-х годов XX столетия с ростом количества компьютеров, подключенных к Интернету (сотни тысяч записей маршрутов), уменьшало быстродействие маршрутизаторов и приводило к сбоям в их работе. Для решения этих проблем в 1993 г. была разработана технология бесклассовой междоменной маршрутизации CIDR (*Classless Inter-Domain Routing*). Термин "бесклассовая" используется потому, что решения о выборе маршру-

тов принимаются на основе масок, накладываемых на полный 32-битный IP-адрес. При этом не существует различия между адресами класса A, B или C.

В соответствии с технологией CIDR адреса выделяются централизованно поставщику Интернет-услуг (*провайдеру*) либо организации, планирующей создание крупной сети. Причем, каждому провайдеру (организации) вместо одного набора адресов класса B, выделяется *подряд* несколько блоков адресов класса *C*. За счет этого существенно экономится пространство адресов класса *B*. Количество адресов класса C выделяется достаточным для нумерации всех сетей, которые организация планирует в будущем подключить к Интернету. При таком подходе адреса всех сетей каждого поставщика услуг имеют общую группу битов в старшей части адреса – *префикс*. В связи с этим маршрутизация на магистралях объединенной сети может производиться **на основе префиксов**, а не полных адресов сетей. Группирование адресов способствовало уменьшению объема таблиц в маршрутизаторах всех уровней в несколько раз и повышению пропускной способности сети Интернет.

Технология CIDR позволяет заменить традиционное использование классов адресов протокола IP на обобщенный **сетевой префикс**. Вместо того чтобы по первым двум или трем битам идентифицировать класс IP-адреса, маршрутизаторы для определения границ между номером сети и номером хоста в IP-адресе выделяют сетевой префикс, задаваемый маской, которая рассылается вместе с адресом. В связи с этим при использовании технологии CIDR маршрутная информация рассылается (*анонсируется*) совместно с сетевым префиксом. Длина сетевого префикса помогает определить число старших битов, соответствующих номеру сети в записи таблицы маршрутизации.

Провайдер выделяет каждому клиенту из своего набора адресов подсеть переменной длины. Таким образом, выделенный блок адресов может быть разбит произвольным образом на **префикс сети** и **суффикс хоста**. Несмотря на то, что группе подключенных к одной сети компьютеров могут быть назначены последовательные адреса, диапазон этих адресов не обязательно должен принадлежать одному из предопределенных классов. Предложенный метод разделения дает системному администратору определенную свободу при назначении адресов, поскольку он может точно определить количество битов, занимаемых префиксом IP-адреса.

Поскольку для определения блоков адресов в CIDR нужно указать два значения – начальный адрес и маску, – была предложена специальная сокращенная форма записи, позволяющая компактно выразить эти два значения. Официально ее назвали *формой записи CIDR (CIDR notation),* однако чаще всего ее называют *записью через косую (slash notation)*. В сокращенной форме записи вначале указывается начальный адрес блока, представленный

в точечной десятичной форме, а затем через косую черту – длина маски в битах, выраженная целым десятичным числом.

Предположим, организации выделен непрерывный блок IP-адресов в количестве 2048, начиная с адреса 128.211.168.0. Пусть для адресации хостов должно использоваться 11 младших битов этого адреса. В таблице 4.3 приведены значения первого и последнего адресов диапазона, представленные в точечной десятичной и в двоичной форме.

Таблица 4.3 – Границы IP-адресов диапазоном 2048 хостов

Граница адреса	IP-адрес	Двоичный эквивалент
Нижняя	128.211.168.0	10000000 11010011 10101000 00000000
Верхняя	128.211.175.255	10000000 11010011 10101111 11111111

Для определения блока адресов в CIDR, показанного в таблице 4.3, необходимо задать 32-разрядное значение начального адреса блока и 32-разрядную маску. В CIDR-маске рассматриваемого примера необходимо установить в единицу первые 21 бит, т.е. 11111111 11111111 11111000 00000000. Это означает, что граница раздела между префиксом и суффиксом будет проходить по 21 биту, префикс занимает первые 21 бит двоичного числа, а суффикс – последние 11 битов. В сокращенной форме записи CIDR блок адресов, приведенный в таблице 4.3, будет выглядеть так: 128.211.168.0/21. Здесь запись /21 означает, что длина маски префикса равна 21 биту.

При использовании CIDR провайдер, получивший в свое распоряжение набор IP-адресов, может выделить из этого набора каждому из своих клиентов блок адресов требуемого размера и указать им соответствующие сетевые маски. Если провайдеру предоставлен набор адресов, CIDR-маска которого имеет длину *N* битов, то он может выделить своим клиентам блок адресов любой длины, однако CIDR-маска этого блока должна быть больше *N* битов. Например, если провайдеру выделен набор адресов 128.211.0.0/16, он вполне может предоставить одному из своих клиентов блок из 2048 адресов 128.211.168.0/21, как показано в таблице 4.3. При запросе от представителя малого предприятия, которому нужно два-три IP-адреса, чтобы подключить несколько компьютеров к Интернету, провайдер может выделить ему другой блок адресов 128.211.176.212/30, диапазон которых приведен в таблице 4.4. Очевидно, что использование битовой маски переменной длины позволяет более гибко выделять блоки IP-адресов, чем это было при использовании классовой системы адресации.

Таблица 4.4 – Границы IP-адресов диапазоном 4 хоста

Граница адреса	IP-адрес	Двоичный эквивалент
Нижняя	128.211.176.212	10000000 11010011 10110000 11010100
Верхняя	128.211.176.215	10000000 11010011 10110000 11010111

Технология CIDR в настоящее время встроена в магистральные Интернет-маршрутизаторы, работающих с протоколом BGP, а обычные хосты в локальных сетях ее не поддерживают.

4.5. Протокол передачи управляющих сообщений ICMP

4.5.1. Назначение и формат управляющих сообщений

В процессе обмена информацией в объединенной IP-сети возможно появление ошибок передачи, отказов аппаратного и программного обеспечения, возникновение условий и ситуаций, требующих принятия определенных мер. Для реализации механизма реагирования на такие ситуации разработан **протокол передачи управляющих сообщений ICMP** (*Internet Control Message Protocol*). На данный протокол возлагаются только функции информирования об особых случаях в сети, а не локализация и устранение причин, которые привели к возникновению аномальных ситуаций. При обнаружении тех или иных проблем промежуточные маршрутизаторы или конечные станции генерируют сообщения ICMP того или иного типа, указывая в них код ошибки, и передают отправителю исходного пакета. ICMP выполняет следующие функции:

- передает отклик на пакет или эхо на отклик;
- контролирует время жизни дейтаграмм в системе;
- реализует переадресацию пакета;
- выдает сообщения о недостижимости адресата или о некорректности параметров;
- формирует и пересылает временные метки;
- выдает запросы и отклики для адресных масок и другой информации.

Для передачи сообщений протокола ICMP по объединенной сети используются IP-дейтаграммы обычного формата. Сообщение ICMP в данном случае помещается (инкапсулируется) в поле данных IP-дейтаграммы (рисунок 4.25).

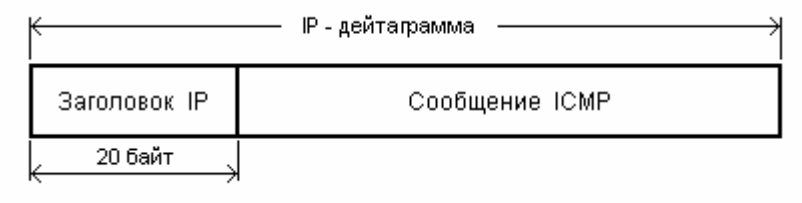


Рисунок 4.25 – Инкапсуляция ICMP-сообщения в IP-дейтаграмму

Сообщение ICMP состоит из заголовка и собственно информационно-управляющего сообщения (рисунок 4.26).

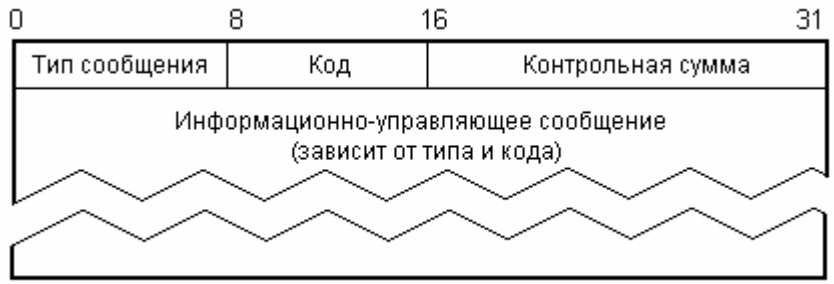


Рисунок 4.26 – Формат ICMP-сообщения

Заголовки всех сообщений ICMP имеют примерно одинаковый формат. Заголовок может занимать до 8 байтов (два 32-х разрядных слова). В нем размещается идентификатор типа сообщения ICMP. Существует 255 различных значений поля "Тип сообщения", которые указывают на конкретный тип ICMP-сообщения. В настоящее время используется порядка 25 типов. Остальные зарезервированы для будущего применения. Для детализации некоторых типов сообщений введено поле "Код" сообщения. Контрольная сумма вычисляется суммированием всех полей, начиная с поля "Тип". При формировании проверочной суммы значение поля "Контрольная сумма" полагается равным 0. Информационно-управляющая часть сообщения ICMP не имеет фиксированной длины, поэтому размер данного поля определяется только типом сообщения.

4.5.2. Типы управляющих сообщений

Сообщения ICMP можно условно разделить на *парные* и *непарные*. Парные сообщения состоят из двух компонентов – **запрос** (*Request*) и **ответ** (*Reply*). Сообщение типа "Ответ" высылается станцией назначения только в ответ на полученное от источника сообщение типа "Запрос". К сообщениям

такого типа относятся "Эхо запрос/ответ". Непарные сообщения формируются асинхронно при возникновении какой либо проблемы при передаче дейтаграммы, и передаются в адрес источника данной дейтаграммы. К сообщениям подобного типа относятся сообщения "Место назначения недоступно" и "Подавление источника".

Следует заметить, что сообщение об ошибке ICMP никогда не генерируется в ответ на следующие ситуации:

1) ICMP сообщение об ошибке (однако, ICMP-сообщение об ошибке может быть сгенерировано в ответ на ICMP-запрос);

2) дейтаграмму, направляющуюся на широковещательный IP-адрес или групповой адрес IP;

3) дейтаграмму, посылаемую широковещательным запросом на канальном уровне;

4) фрагмент, который не является первым.

5) дейтаграмму, адрес источника которой не указывает на конкретный хост; это означает, что адрес источника не может быть нулевым, *loopback* адресом, широковещательным или групповым адресом.

Данные правила введены с целью предотвращения лавинообразного роста количества широковещательных сообщений, который может произойти, если ICMP сообщения об ошибках будут отправляться в ответ на широковещательные пакеты.

В таблице 4.5 приведены числовые значения поля "Тип ICMP" и пояснение типов сообщений, соответствующих этим значениям. Поясним значения некоторых сообщений. Сообщение "**Место назначения недоступно**" принадлежит к непарным сообщениям ICMP. Его формат показан на рисунке 4.27. Оно формируется в случае, если требуемый сетевой ресурс недоступен для запрашивающей его станции. В поле "**Код**" сообщения "**Место назначения недоступно**" размещается код, который соответствует типу запрошенного недоступного сетевого ресурса или конкретизирует причину, из-за которой этот ресурс недоступен в данном случае. Возможные также значения поля "Код": 0 – сеть недоступна; 1 – хост недоступен; 3 – порт недоступен; ... 12 – хост недоступен для данного типа обслуживания.

Сообщения данного типа могут быть сформированы как станцией назначения ("Код"=2 и 3), так и одним из промежуточных маршрутизаторов – шлюзов ("Код"=0,1,6 и т.д.). При этом в качестве адреса источника указывается IP-адрес узла, который обнаружил проблему. Например, сообщение "**Хост недоступен**" может быть сформировано последним маршрутизатором, который пытается доставить сообщение до хоста по непосредственно подключенной сети. Для того, чтобы станция-источник смогла правильно интерпретировать диагностическое сообщение, в тело сообщения "Место назначения недоступно" помещается заголовок и первые 8 байтов исходной

дейтаграммы.

Сообщения типа 3 с кодом 3 "**Порт недоступен**" отправляется источнику, если протокол UDP в процессе приема дейтаграммы обнаружил, что порт назначения не соответствует порту, который обслуживается каким–либо процессом, либо порт закрыт для доступа.

Таблица 4.5 – Сообщения протокола ICMP

Тип сообщения	Сообщение
0	Эхо-отклик (*Echo Reply*)
3	Место назначения не достижимо (*Destination Unreachable)*
4	Подавление источника (*Source Quench*)
5	Перенаправление (*Redirect*)
8	Эхо-запрос (*Echo Request*)
9	Объявление маршрутизатора (*Router Advertisement*)
10	Запрос к маршрутизатору (*Router Solicitation*)
11	Время истекло (*Time Exceeded*)
12	Проблемы с параметрами (*Parameter Problem*)
13	Запрос временной метки (*Timestamp Request*)
14	Отклик с временной меткой (*Timestamp Reply*)
15	Информационный запрос (*Information Request*)
16	Информационный отклик (*Information Reply*)
17	Запрос маски адреса (*Address Mask Request*)
18	Ответ с маской адреса (*Address Mask Reply*)

Рисунок 4.27 – Формат управляющего сообщения "Место назначения недоступно"

Сообщение **"Время истекло"** относится к непарным сообщениям ICMP. Оно формируется в том случае, если в процессе передачи дейтаграммы истекло допустимое время её существования в сети или на хосте.

Сообщение "**Проблемы с параметрами**" также принадлежит к непар-

ным сообщениям ICMP. Такое сообщение формируется в ситуации, если в процессе обработки заголовка дейтаграммы на хосте или маршрутизаторе были обнаружены некорректные аргументы, делающие невозможным дальнейшее перемещения дейтаграммы. В этом случае дейтаграмма уничтожается, а в адрес её источника передается сообщение *Parameter Problem* (рисунок 4.28).

Рисунок 4.28 – Вид управляющего сообщения "Проблема с параметром"

В поле "Код" данного сообщения размещается признак типа диагностической информации. В том случае, если в этом поле находится код "0", значение поля "Счетчик" (*Pointer*) сообщения *Parameter Problem* соответствует номеру байта в заголовке исходного сообщения, который не может быть адекватно интерпретирован. Например, значение *Pointer*=1, в данном случае указывает на возникновение проблемы с интерпретацией поля "Тип" исходного сообщения.

Значение поля Код=1 формируется в ситуации, когда причина, по которой данная дейтаграмма не может продолжать перемещение по сети, заключается в несоответствии запрашиваемых параметров установленным требованиям. Такими требованиями могут быть, в частности, требования по обеспечению безопасности.

Сообщение **"Сдерживание источника"** принадлежит к непарным сообщениям ICMP. Оно формируется в случае возникновения в процессе передачи дейтаграммы угрозы перегрузки. Источник, получив от станции назначения или одной из промежуточных станций такое сообщение, должен уменьшить скорость информационного обмена в указанном направлении.

При передаче сообщения "Сдерживание источника" в качестве адреса назначения указывается IP-адрес источника первичного сообщения. Для того чтобы станция-источник смогла правильно интерпретировать диагностическое сообщение, в тело сообщения *Destination Unreachable* помещается заголовок и первые 8 байтов исходной дейтаграммы.

Сообщение **"Изменение маршрута"** (*Redirect)* относится к непарным сообщениям ICMP. Оно формируется в случае, если при получении дейтаграммы шлюз обнаруживает, что для её передачи был выбран неудач-

ный маршрут. На рисунке 4.29 показан пример использования сообщения *Redirect* для изменения неоптимального маршрута.

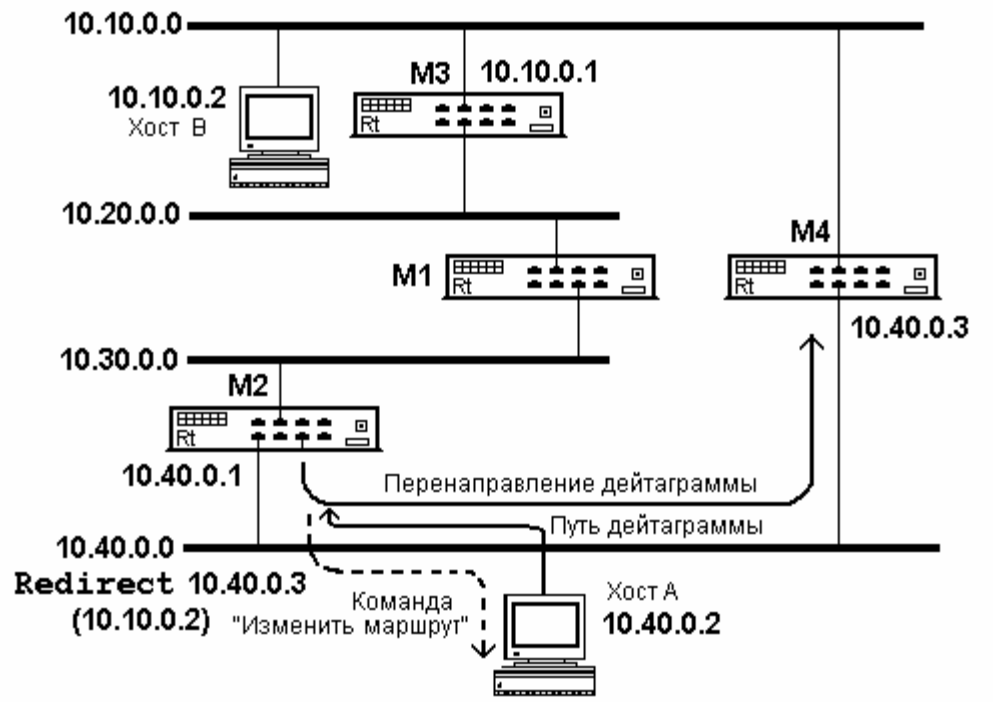


Рисунок 4.29 – Пример использования сообщения «Изменение маршрута»

На схеме хост A (10.40.0.2) отправляет дейтаграмму в направлении хоста B (10.10.0.2), используя для этого в качестве шлюза маршрутизатор M2. После того, как маршрутизатор M2 получает дейтаграмму, он определяет, что данная дейтаграмма адресована в направлении сети 10.10.0.0. Кратчайший маршрут достижения этой сети для маршрутизатора M2 проходит через маршрутизатор M4, который в данном случае подключен к тому сегменту сети, из которого была получена принятая дейтаграмма.

Маршрутизатор M2 перенаправляет дейтаграмму по направлению M4, одновременно формирует командное сообщение ICMP *Redirect* (штрихоая стрелка), в котором он рекомендует хосту A впредь для передачи дейтаграмм в направлении сети 10.10.0.0 использовать в качестве шлюза маршрутизатор M4. Команды переадресации маршрутизатор посылает только хосту и никогда другим маршрутизаторам. В заголовке сообщения ICMP *Redirect* размещается IP-адрес шлюза, рекомендуемый для использования с целью достижения сетевого ресурса, указанного в исходной дейтаграмме и тип маршрута, который должен быть изменен по предложению источника сообщения ICMP.

Сообщения типа "**Эхо**" (Echo) принадлежат к парным сообщениям ICMP. Они применяются для определения возможности достижения и статуса компонентов сети Интернет. Любой компонент сети, получивший адресованное ему сообщение *Echo Request*, формирует ответное сообщение *Echo Reply* в адрес источника полученного сообщения.

Это сообщение используется в простых утилитах ***Ping***, которые посылают одно сообщение *Echo Request* в адрес назначения и указывают на получение ответного сообщения *Echo Reply*. Более сложные утилиты формируют несколько последовательных сообщений *Echo Request* и измеряют значение интервала времени, разделяющего момент передачи этих сообщений от момента приема соответствующих ответных сообщений.

Сообщения *Echo Request* и *Echo Reply* имеют одинаковый формат (рисунок 4.30) и отличаются только содержимым поля "Тип".

Рисунок 4.30 – Вид управляющего сообщения "Эхо запрос"

Значение поля Тип = 0 соответствуют сообщению Echo Reply, а Тип = 8 – Echo Request. В поле "Код" сообщений *Echo Request/Reply* устанавливается значение 0. Поля "Идентификатор" (обычно это идентификатор процесса) и "Номер последовательности" (увеличивается на 1 при посылке каждого пакета) служат для того, чтобы отправитель мог связать в пары запросы и отклики.

Сообщения "**Запрос и ответ временной метки**" (*Timestamp Request/Reply*) принадлежат к парным сообщениям ICMP. Они предназначаются для обеспечения взаимной синхронизации счетчиков времени у различных компонентов сети Интернет.

Поскольку компоненты Интернета не имеют общего управления, значения индивидуальных датчиков времени у каждого из компонентов могут существенно отличаться. Для обеспечения взаимной синхронизации таких датчиков используются сообщения *Timestamp Request/Reply*. Сообщения "Запрос" и "Ответ" имеют одинаковый формат и отличаются только содержимым поля "Тип". Значение Тип = 14 соответствуют сообщению *Timestamp*

Reply, а Тип = 13 – *Timestamp Request* (рисунок 4.31).

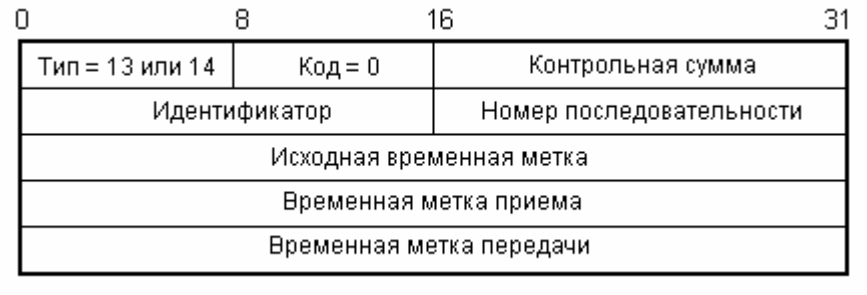


Рисунок 4.31 – Управляющее сообщение для запроса и приема ответа временной метки

В полях "Идентификатор" и "Номер последовательности" помещаются кодовые комбинации, которые позволяют станции назначения установить соответствие между переданными и полученными сообщениями.

В поле сообщения "Исходная временная метка" (*Originate Timestamp*) отправитель помещает состояние своего датчика времени в момент отправки управляющего сообщения. В момент получения этого сообщения станция назначения помещает в поле "Временная метка приема" (*Receive Timestamp*) отсчет собственного датчика времени. Значения этих полей отвечающая станция копирует в соответствующие поля возвращаемого сообщения *Timestamp Reply*.

Поле "Временная метка передачи" (*Transmit Timestamp*) ответного сообщения формируется в момент отправки этого сообщения в адрес отправителя исходного запроса. Значение поля "Временная метка передачи" в данном случае будет соответствовать значению счетчика времени *Timestamp Request*-сервера в момент отправки сообщения.

Получив сообщение *Timestamp Request*, отправитель запроса может определить величину задержки распространения между ним и сервером и, таким образом, сформировать поправку к своему датчику времени. Значения временных меток, возвращаемых по соответствующим запросам, измеряется в миллисекундах, которые прошли с полуночи в формате UTC (Универсальное согласованное время – *Coordinated Universal Time*. В старых руководствах UTC называется Среднее время по Гринвичу – *Greenwich Mean Time*.).

Использование описанного метода синхронизации датчиков времени не может обеспечить достаточной точности в связи с тем, что задержки распространения сигналов в прямом и обратном направлениях могут существенно отличаться и изменяться во времени. Поэтому для обеспечения надежной взаимной синхронизации датчиков времени должны быть использованы несколько сессий синхронизации.

При подключении к сети хост может не знать адреса своего маршрутизатора. Для получения этой информации он отправляет широковещательно или по групповому адресу "всем маршрутизаторам" сообщение-запрос типа 10 "Запрос к маршрутизатору". В ответ на это маршрутизаторы должны ответить сообщением типа 9 "Объявление маршрутизатора". Маршрутизаторы могут также, не дожидаясь запросов, периодически рассылать объявления о маршрутах (сообщение типа 9) по групповому адресу "всем хостам". Такие объявления содержат адреса одного или нескольких маршрутизаторов, снабженных значениями приоритета для каждого маршрутизатора.

4.5.3. Протокол ICMPv6

С появлением протокола IPv6 был разработан протокол передачи управляющих сообщений 6-й версии ICMPv6, получивший номер 58. Протокол ICMPv6 отличается от ICMPv4 только теми параметрами, которые напрямую связаны со структурой IP-пакета. В частности, изменились идентификатор ICMPv6, определяющий тип заголовка (*Next Header*), коды типов сообщений ICMPv6, добавились параметры для типов сообщений, работающих с вложенными заголовками и т. д. Подробное описание ICMPv6 можно найти в RFC-1885.

Таблица 4.6 – Сообщения об ошибках в протоколе ICMPv6

Тип ошибки	Сообщение
1	Место назначения не достижимо (*Destination Unreachable*). Более точная спецификация места назначения задается кодом ошибки.
2	Пакет слишком большой (*Paket to Big*). Пакет является слишком большим для сегмента сети, так как он превышает максимально допустимое значение MTU. Это сообщение посылается только в случае установки флага "Запрет фрагментации".
3	Превышение времени (*Time Exceeded*). Лимит хопов достиг нулевого значения и дейтаграмма отброшена.
4	Проблема с параметром (*Parameter Problem*). В заголовке пакета имеются ошибки и он не может быть корректно обработан.

Заголовок протокола ICMPv6 аналогичен заголовку его предшественника. Все типы сообщений ICMPv6 разделены на две группы: **сообщения об ошибках** и **информационные сообщения**. Сообщения об ошибках имеют номера от 0 до 127, а информационные – от 128 до 255. Кроме этого, новая версия ICMP имеет возможность обеспечивать широковещательный режим, располагает способностью автоматического распределения адресов, конфигурации маршрутизаторов и обнаружения соседей. Основные сообщения об ошибках в ICMPv6 приведены в таблице 4.6, а информационные сообщения – в таблице 4.7.

Таблица 4.7– Информационные сообщения протокола IPv6

Тип ошибки	Сообщение
128	Эхо-запрос (*Echo Request*).
129	Эхо-отклик (*Echo Reply*). Совместно с Эхо-запросом используется для определения пропускной способности сети и состояния удаленного узла IPv6.
130	Запрос групповой принадлежности (*Group Membership Query*).
131	Сообщение групповой принадлежности (*Group Membership Report*).
132	Сокращение групповой принадлежности *(Group Membership Reduction*). Выход из мультикастной группы.
133	Запрос маршрутизатора (*Router Solicitation*).
134	Объявление маршрутизатора (*Router Advertisement*).
135	Запрос соседей (*Neighbour Solicitation*). Запрос MAC–адреса соседних хостов сети.
136	Объявление соседей (*Neighbour Advertisement*). Выдача MAC-адреса интерфейса. Представляет своего рода ARP-ответ.
137	Перенаправление (*Redirect*). Маршрутизатор задает изменение маршрута.

4.6. Организация сервисных служб в сети Интернет

4.6.1. Служба терминального доступа *Telnet* и *Rlogin*

Первой службой, реализованной в сети Интернет, была *Telnet*. В этой службе используется взаимодействие компьютеров по схеме "клиент-сервер". Работа в *Telnet* позволяет пользователю (*клиенту*) устанавливать сеансы с удаленными компьютерами (*серверами*) и работать в режиме удаленного терминала, т.е. работать на чужом компьютере, используя свою клавиатуру и дисплей, как будто бы они непосредственно были подключены к этому компьютеру. В *Telnet-протоколе* используется принцип "**сетевого виртуального терминала" NVT** (*Network Virtual Terminal*). NVT представляет собой мнимый базовый терминал, обладающий свойствами многих терминалов различных типов. Фактические свойства физического терминала с помощью протокола *Telnet* эмулируются виртуальным терминалом.

Службе выделен **порт 23**. Для работы с удаленным компьютером устанавливается TCP-соединение между клиентом и портом 23 сервера (рисунок 4.32). Затем пользователь индентифицирует и аутентифицирует себя путем ввода своего имени и пароля.

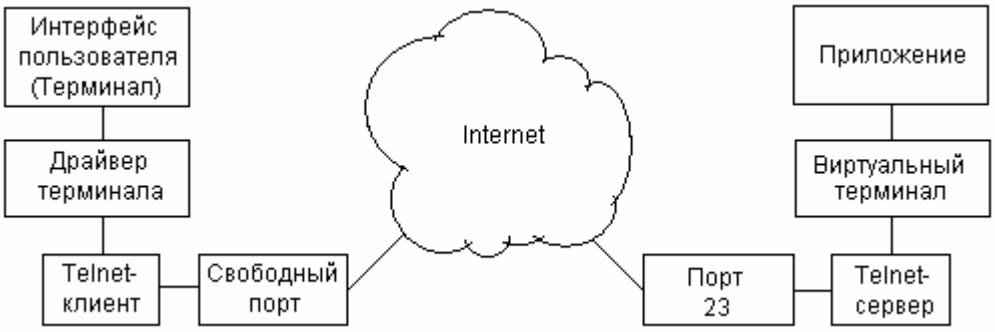


Рисунок 4.32 – Схема доступа к другому компьютеру посредством *Telnet*

Собственно сам *Telnet*-сервер не производит какой-либо аутентификации клиента, но при открытии соединения с клиентом сервер выдает свое имя и строку "login". Это стандартный запрос системы Unix, который выдается при попытке входа пользователя с любого терминала, реального или виртуального. Пользователь, получив этот текст, вводит свое имя. *Telnet*-сервер вызывает программу *login*, передавая ей имя пользователя. Последняя в свою очередь запрашивает у пользователя пароль и проверяет его по файлу */etc/shadow*. Если пароль верен, то пользователю предоставляется право работать с сервером. Во время процедуры аутентификации пароль отсы-

лается в незашифрованном виде в текстовом формате. Такой вид передачи не исключает возможность перехвата пароля злоумышленником с последующим несанкционированным доступом на сервер.

Клиентская программа *telnet* (MS Windows – *telnet.exe*) входит в поставку всех современных операционных систем. В качестве сервера используется программа *in.telnetd*, являющаяся составной частью всех разновидностей ОС Unix.

Кроме вводимых пользователем символов, протокол *Telnet* позволяет в том же самом потоке передавать команды, управляющие параметрами сеанса связи. Признаком команды является октет с десятичным значением 255, символически обозначаемый *IAC (Interpret as Command)*. После этого октета следует один октет с кодом команды управления. Команда может также включать одну из опций. Основная область применения команд – переговоры клиента и сервера о дополнительных функциях соединения. Обычно переговоры осуществляются в начале сеанса, но они могут иметь место и в процессе работы. В качестве опций наиболее часто используются опция включения и отключения режима "Эхо" (сервер посылает все полученные от пользователя символы обратно через сеть на терминал), опция изменения размера окна своего терминала.

Протокол **Rlogin** (*Remote login*) выполняет ту же задачу, что и *Telnet* – через сеть подключает к удаленному серверу виртуальный терминал. Однако авторизация в этом случае выполняется самим сервером Rlogin. Это сделано для осуществления возможности входа пользователям, работающим на заслуживающих доверия клиентских компьютерах, в операционную систему компьютера-сервера без предъявления пароля.

Передача данных между *Rlogin*-клиентом и сервером выполняется посредством протокола TCP на стандартный порт сервера 513. Клиент может подсоединяться с любого порта. Протокол *Rlogin* реализуется стандартными программами, входящими в состав ОС Unix: клиентом *rlogin*, сервером *in.rlogind*. Для установления соединения достаточно набрать строку: % *rlogin* имя_сервера.

После установления соединения с сервером клиент посылает четыре строки, заканчивающиеся нулевым октетом. Первая стока состоит только из нулевого октета, во второй указывается имя пользователя на хосте-клиенте, в третьей – имя пользователя на хосте-сервере, в четвертой задается тип терминала клиента и скорость передачи по каналу.

Сервер откликается нулевым октетом, подтверждающим авторизацию пользователя, и сеанс переходит в фазу передачи данных. Как и в *Telnet*, протокол *Rlogin* предусматривает возможность помещения в поток данных управляющих команд.

4.6.2. Служба передачи файлов FTP

Служба предназначена для обмена данными между двумя компьютерами под управлением пользователя. **Протокол FTP** (*File Transfer Protocol*) основывается на TCP-виртуальном соединении. Наряду с FTP имеется еще и упрощенный вариант этого протокола – TFTP (*Trivial File Transfer Protocol*), который базируется на протоколе дейтаграммной службы UDP.

В *FTP* также используется взаимодействие компьютеров по схеме "клиент-сервер". Особенностью протокола является разделение управляющего и информационного соединений (рисунок 4.33). Применение *FTP-протокол*а предполагает установку двух различных соединений. Одно соединение через порт 21 предназначено для передачи команд управления, а второе через порт 20 служит для обмена данными. Номер порта клиента может быть произвольным.

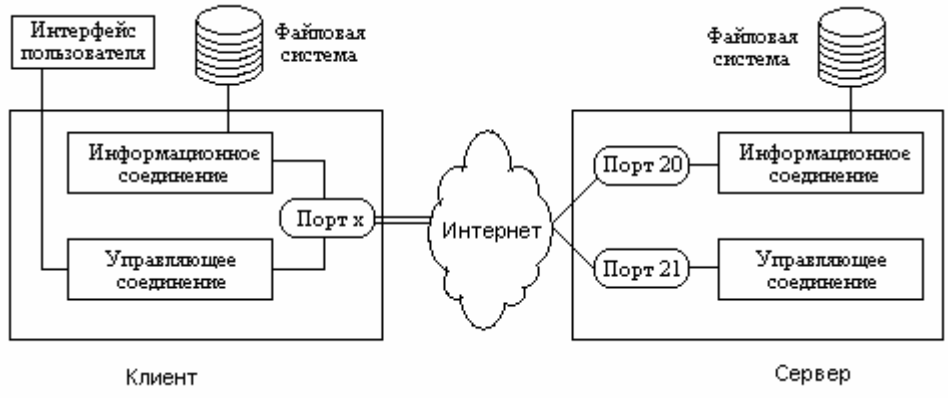


Рисунок 4.33 – Схема обмена данными по протоколу FTP

Для того чтобы обеспечить возможность управления взаимодействующими компьютерными системами с различными *FTP*-серверами, были разработаны стандартные команды протокола *FTP*. Они не совпадают с командами интерфейса пользователя. Для их трансляции используется программа клиент-*FTP*.

FTP-сессия протекает следующим образом. Клиент инициализирует установление управляющего соединения с сервером на порт 21 и передает по нему команды управления. Сервер отправляет ответы на команды тоже по управляющему соединению. Команды клиента задают параметры передачи данных, например, тип данных, режим передачи, структуру данных и выполняемые операции (читать, записывать или сохранять). После задания параметров начинается процесс передачи данных по информационному соединению. Завершив процедуру установления соединения, программа "клиент"

требует от пользователя ввести пользовательское имя. Сервер, получив идентификатор пользователя *Userid*, дополнительно запрашивает пароль. Ответы сервера выдаются в форме трехзначного числа и короткого комментария.

С помощью службы *FTP* пользователь может посредством передачи команд со своего компьютера инициализировать обмен данными между двумя другими компьютерами сети. Для этого пользователь инициирует установление соединения с двумя серверами А и В и устанавливает, например, сервер В на передачу данных компьютеру А. После получения сообщения о завершении обмена пользователь разрывает соединение.

Следует заметить, что после установления соединения за установку канала передачи данных между своим портом 20 и любым другим портом клиента отвечает сервер. Это дает возможность злоумышленнику перехватить запрос, представиться сервером и вставлять свои данные в передаваемый поток. Для защиты от нарушителей к стандартному протоколу *FTP* был добавлен **пассивный режим**, при котором клиент сам инициирует создание соединения для передачи данных через TCP-порт 20. Необходимость введения пассивного режима было введено из-за применения в компьютерных сетях защитных экранов (брандмауэров) в связи с тем, что в защищенных сетях *FTP*-сервер не может активизировать соединение с клиентом.

4.6.3. Всемирная информационная служба WWW

Всемирная гипермедиа информационная служба *WWW (World Wide Web* – всемирная паутина, сокращенно ***Web)*** объединяет многие Интернет-службы под одной оболочкой. Она является в настоящее время основной службой получения гипермедиа-информации. Термин гипермедия обозначает мультимедиальный текст, т.е. документ всевозможных видов: текст, неподвижные изображения, звук и живое видео. Документ может содержать *ссылки,* ведущие к другим связанным документам, расположенным географически в различных пунктах. Путем выбора гиперссылки можно перемещаться от одного документа к другому, не задумываясь, на каком сервере они располагаются. Концептуально служба Web состоит из большого набора документов, называемых *Web-страницами,* доступ к которым можно получить из любой точки глобальной сети Интернет. Каждая Web-страница представляет собой гипермедиа-документ.

Начиная с 2000 года, трафик службы Web практически полностью вытеснил трафики других протоколов. На его транспортировку затрачивается большая часть пропускной способности глобальной сети Интернет.

Служба *WWW* реализована на основе клиент-серверной технологии, при которой используются два типа сообщений: запросы от клиента (*браузера*) к серверу и ответы сервера клиенту. Предполагается, что в ответ на запрос сервер отправляет клиенту некоторый квант информации, называемый *контентом*. В простейшем случае контент ответа представляет собой содержание файла, расположенного на диске сервера. Служба *WWW* базируется на следующих четырех компонентах:

- прикладном протоколе HTTP *(HyperText Transfer Protocol),* являющимся базовым для обмена данными;
- языке гипертекстовой разметки HTML *(HyperText Markup Language)*, на котором составлены гипермедиа-документы;
- схеме адресации, использующей унифицированные указатели информационных ресурсов URL *(Uniform Resource Locator*);
- оболочке пользователя (браузере), применяемой для доступа к ресурсам *WWW.*

Гипертекстовый транспортный протокол HTTP разработан специально для работы в гипертекстовой системе и служит для связи *WWW*-сервера с одной из *WWW*-клиентских программ (Web-браузером). Он является весьма простым и надежным расширяемым протоколом. Протокол HTTP работает на уровне приложения. Его данные инкапсулируются в формат сегмента транспортного протокола TCP. Доступ к серверу осуществляется через стандартный порт 80, а порт клиента может быть любой. Схема взаимодействия клиента с сервером изображена на рисунке 4.34.

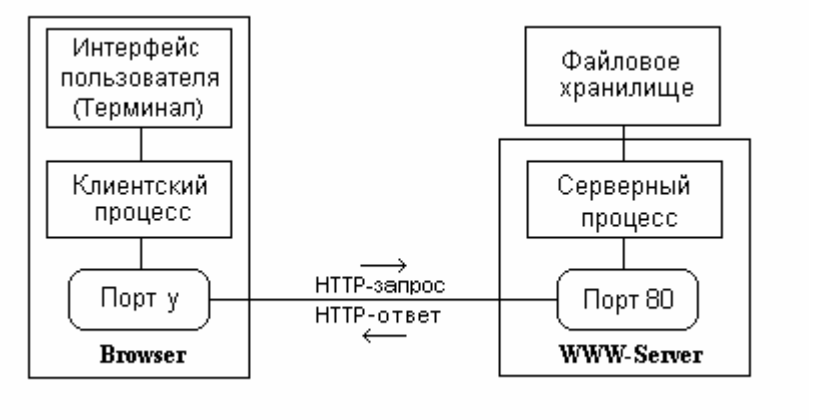


Рисунок 4.34 – Схема обмена по протоколу HHTP

После установки сеанса связи с сервером одна из сторон (браузер) должна отправить HTTP-запрос, в ответ на который другая сторона (сервер) присылает данные. Каждый HTTP-запрос является независимым. Сервер не сохраняет список предыдущих запросов или предыдущих сеансов работы с

клиентом. Протокол HTTP позволяет браузеру и серверу согласовывать параметры сеанса связи, например, набор символов, который нужно использовать при передаче. При этом отправитель сообщает серверу набор поддерживаемых им параметров, а получатель возвращает те из них, которые он может принять.

Язык гипертекстовой разметки HTML используется для составления WWW-документов. HTML-документ является обычным текстовым файлом, содержащим кроме собственно текста внедренные в него команды. Такие команды называются *дескрипторами,* или *тэгами.* Они предназначены для обработки браузерами и управляют отображением страницы на экране монитора. HTML позволяет включать ссылки на другие документы в любом месте текста.

Унифицированный указатель на ресурс URL имеет следующий формат: Тип протокола://Адрес хоста: Номер порта/ Путь к ресурсу/Аргументы запроса. Например, URL ресурса, доступного через протокол HTTP, имеет вид:

http://www.minisoft.ru:8080/dir/slovar.html

Унифицированный указатель на ресурс URL применяется браузером для формирования запроса к HTTP-серверу. Имя сервера с помощью службы доменных имен DNS преобразуется в IP-адрес. Затем производится соединение с этим адресом и указанным портом и посылается запрос с указанием **идентификатора ресурса URI** (*Uniform Resource Identifier*), имени WWW-сервера, информации о пароле и пользователе (если требуется). URI содержит путь к ресурсу и параметры обмена. В приведенном выше примере URI имеет вид: /dir/slovar.html.

Существует две формы URL: абсолютная и относительная. *Абсолютная* форма URL содержит полную спецификацию документа, приведенную выше. При *относительной* форме URL адрес сервера не указывается. Относительная форма применяется только в случае, когда соединение уже установлено и сервер задан неявно.

Браузер является клиентской программой просмотра файлов, позволяющей пользователю работать с *WWW*-серверами. Браузер просматривает не только текстовые, но и гипертекстовые документы. Если документ полностью передан по сети от *WWW*-сервера программе-браузеру, то браузер сохраняет его в своем кэше. При повторном обращении к этому документу, браузер проверяет сначала нахождение его в кэше, и в случае наличия связывается с сервером и проверяет дату последней модификации документа. Если документ на сервере не новее документа в кэше, то браузер выводит его на экран потребителю. Это существенно увеличивает скорость работы. В настоящее время наиболее распространенными являются браузеры *Microsoft Internet Explorer* и *Netscape Navigator*. Весьма перспективным явля-

ется браузер *Opera for Windows*, отличающийся малыми размерами, скоростью загрузки HTML документов, как из Интернета, так и с локального диска, универсальностью в загрузке и отображении веб-страниц.

Между браузером и *WWW*-сервером могут быть расположены промежуточные серверы, так называемые **прокси-серверы** (*Proxy-Server*). В этом случае браузер отправляет свой запрос прокси-серверу, указывая в запросе абсолютный URL требуемого ресурса. Дальнейшее обслуживание запроса возлагается на прокси-сервер, который возвращает браузеру ответ с контентом или с кодом ошибки.

Прокси-сервер выполняет обычно две функции: *контроль доступа* пользователей к WWW и *кеширование контентов*. Настройки прокси–сервера позволяют блокировать доступ к определенным адресам, делить пользователей на группы с различным приоритетом обслуживания запросов, выполнять другие (контрольные и учетные) функции.

Кэширование контентов применяется для повышения эффективности работы Web-системы. Оно позволяет уменьшить время ожидания ответа пользователем и сократить сетевой трафик за счет устранения ненужных пересылок. Одно из самых очевидных преимуществ кэширования заключается в промежуточном хранении Web-страниц. При первом обращении к Web-странице ее копию сохраняют на своих локальных дисках браузер, или прокси-сервер, либо тот и другой. При обращении клиента за определенным ресурсом проверяется, имеется ли этот ресурс в кэше прокси-сервера. В случае наличия информации в кэше она выдается клиенту, а при ее отсутствии прокси-сервер производит запрос к WWW-серверу, на котором расположен запрашиваемый ресурс.

Чтобы обеспечить корректную работу системы, в протокол HTTP включены средства явной поддержки прокси-серверов. В протоколе точно определено, как прокси-сервер должен обрабатывать запросы, как должны интерпретироваться заголовки прокси-серверами, как броузер согласовывает параметры с прокси-сервером и как последний согласовывает параметры с Web-сервером. Кроме того, несколько типов http-заголовков было создано специально для использования прокси-серверами.

4.6.4. Служба доменных имен DNS

На техническом уровне для идентификации сетевых устройств в компьютерных сетях применяются 32-битовые целые числа, называемые адресами протокола IP. Для облегчения запоминания и применения сетевых адресов удобнее использовать их символические обозначения, так называемые

доменные имена. Преобразование доменных имен в IP-адреса выполняется специальной службой **DNS** (*Domain Name System*).

Доменные имена состоят из символьных полей, разделенных точками. Такие имена построены по иерархическому принципу. Крайнее правое поле обозначает домен верхнего уровня. Далее, справа налево, следуют поддомены в порядке иерархической вложенности. Крайнее левое поле обозначает имя компьютера пользователя. Максимальная длина доменного имени 255 символов, причем каждое поле имени не может превышать в длину 63 символа. Доменами верхнего уровня являются двухбуквенные национальные домены (.ua, .ru, .de и т.д.) или трехбуквенные домены сетей определенной области деятельности организаций (.com, .edu, .net и др.)

Механизм преобразования доменных имен основывается на группе независимых, но взаимодействующих между собой компьютерных систем, называемых *серверами имен (name servers).* Сервер имен — это программа, запущенная на компьютере, исполняющем роль сервера, которая преобразует имена доменов в IP-адреса. Очень часто эта программа запускается на специализированном компьютере. В этом случае сама ЭВМ является сервером имен. Клиентская часть программы называется *распознавателем имен (name resolver).* В процессе преобразования имени она обращается к одному или нескольким серверам имен.

Серверы доменных имен образовывают древовидную структуру, в которой каждый сервер отвечает за определенную зону – свою часть дерева доменных имен, хранит соответствующие базы данных и отвечает на запросы. При этом вышестоящие по дереву серверы содержат информацию об адресах нижестоящих серверов, что обеспечивает связность дерева. Таким образом, вышестоящие серверы делегируют нижестоящим полномочия по обслуживанию определенной зоны домена. Одним из способов повышения эффективности трансляции имен в адреса является кэширование, то есть хранение в оперативной памяти (*кэше*) имен-адресов, которые использовались последнее время особенно часто. Каждый DNS-сервер, кроме таблицы соответствия имен, содержит ссылки на DNS-серверы своих поддоменов. Эти ссылки связывают отдельные DNS-серверы в единую службу DNS. Ссылки представляют собой IP-адреса соответствующих серверов.

При необходимости произвести какое-либо из DNS-преобразований хост обращается к своему локальному серверу DNS, адрес которого устанавливается в настройках стека TCP/IP хоста. Обращение происходит, как правило, по протоколу UDP на порт 53. Если локальный DNS-сервер не может выдать ответ на поступивший запрос по причине отсутствия адресной информации в его базе данных или в кэше предыдущих запросов, то он обращается к одному из внешних корневых серверов.

Схема такого взаимодействия программы пользователя с серверами

имен показана на рисунке 4.35.

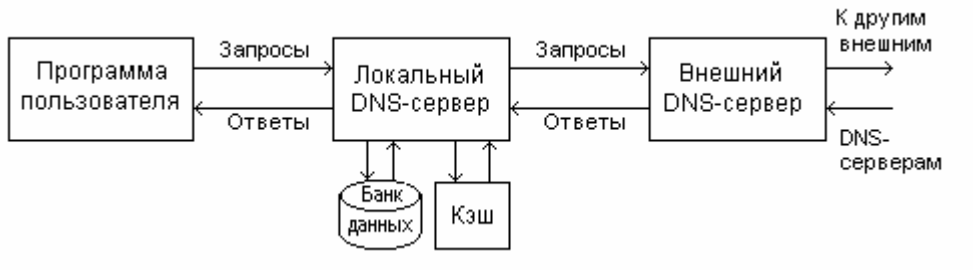


Рисунок 4.35 – Схема взаимодействия с серверами имен

Если у внешнего сервера нет в кэше требуемых данных, он обращается к другим внешним DNS-серверам, т.е. к корневому серверу доменной системы. На самом деле в реальной сети таких серверов несколько, а их базы синхронизированы друг с другом, поэтому обращение может осуществляться к любому из них.

4.6.5. Электронная почта в объединенной сети

Электронная почта, наряду со службами FTP и Telnet, является одной из самых первых сервисных служб, реализованных в компьютерных сетях. Электронная почта – своеобразный аналог обычной почты, в которой письма пересылаются от одного почтового адреса к другому. Адрес электронной почты имеет вид: **vs_chernega@mail.ru.** Слева от символа @ (эт) располагается имя почтового ящика пользователя, а справа – обычное доменное имя компьютера или целой сети (как в нашем случае).

Процесс доставки электронной почты существенно отличается от других уже рассмотренных способов использования объединенной сети. Во всех описанных ранее службах сетевые протоколы отсылают пакеты напрямую получателям. При этом отдельные сегменты передаются повторно, если в течение установленного интервала времени получатель не подтвердит прием. Однако при пересылке электронной почты учитываются ситуации, когда удаленная станция временно недоступна по той или иной причине. Для решения такой проблемы в системах электронной почты используется метод *буферизации* сообщений *(spooling)*. Во время отправки письма по электронной почте, система помещает его копию во временный буфер (spool). Вместе с сообщением в буфер также помещается служебная информация: идентификационные параметры отправителя и получателя, адрес компьютера-получателя, а также время, на протяжении которого копия должна храниться в буфере. Затем система начинает передачу данных удаленному компьютеру

в фоновом режиме, что позволяет компьютеру-отправителю выполнять другие вычислительные операции.

Фоновый процесс передачи электронной почты относится к клиентским процессам. Сначала он использует систему доменных имен, для преобразования имени компьютера-получателя в IP-адрес, а затем пытается создать TCP-соединение с сервером электронной почты, запущенным на компьютере получателя. Если соединение удается установить, копия сообщения пересылается удаленному серверу, который сохраняет ее в своем буфере. Получив от почтового сервера подтверждение получения сообщения и сохранения его в буфере, клиент удаляет локальную копию сообщения из собственного буфера. Если в процессе передачи не удается создать TCP-соединение или соединение внезапно обрывается, фиксируется время доставки сообщения и работа клиента завершается. Периодически (обычно каждые 30 минут) клиентский фоновый процесс просматривает область буферной памяти, проверяя, есть ли недоставленные сообщения. Обнаружив такое сообщение или помещенное пользователем новое исходящее сообщение, фоновый процесс пытается его доставить. Если по истечении установленного интервала времени (например, трех дней) программа пересылки электронной почты не сможет доставить сообщение получателю, она возвращает его отправителю вместе с сообщением об ошибке.

Для расширения возможности взаимодействия систем электронной почты, соответствующие стандарты семейства протоколов TCP/IP разделены на две группы. Одна группа протоколов определяет формат сообщений электронной почты, другая — детали обмена сообщениями электронной почты между двумя компьютерами. Разделение стандартов электронной почты на две независимые группы позволяет создать **почтовые шлюзы**, соединяющие системы доставки электронной почты сторонних производителей и стандартные почтовые системы объединенной сети TCP/IP. При этом в обеих системах используется один и тот же формат сообщений.

Каждое текстовое сообщение электронной почты состоит из двух частей: заголовка и тела, разделенных пустой строкой. В стандарте семейства протоколов TCP/IP для сообщений электронной почты определен строгий формат заголовков, а также семантическая интерпретация каждого его поля. Формат тела сообщения задается самим отправителем. В частности, в стандарте определено, что заголовки имеют текстовый формат и размещаются в отдельных строках. Строка заголовка в свою очередь состоит из *ключевого слова*, после которого следует двоеточие, а затем — *значение*. Некоторые ключевые слова являются обязательными, другие — необязательными, а остальные — необрабатываемыми. Например, в области заголовка должна находиться строка, определяющая получателя. Эта строка начинается из ключевого слова **To**: (*Кому:*), после которого указывается адрес электронной

почты получателя сообщения. В строке, начинающейся со слова **From**: (*От:*), указывается адрес электронной почты отправителя. При желании отправитель может установить адрес, по которому нужно отсылать ответы (т.е. указать, что ответы необходимо посылать по другому адресу, а не по адресу отправителя). Если в заголовке присутствует строка, начинающаяся со слов **Reply-to**: (*Обратный адрес*:), то в ней находится адрес, по которому следует пересылать ответы. В случае отсутствия такой строки, в качестве адреса для пересылки ответов используется информация, содержащаяся в строке "From:".

Стандартизация формата сообщений электронной почты облегчает их обработку и передачу через разнотипные почтовые серверы. Выбор простого текстового формата для заголовка сообщений электронной почты позволяет использовать его в различных системах. При этом исчезает проблема выбора стандарта для двоичного представления чисел, а также необходимость преобразования стандартного представления в представление, используемое в локальной системе.

В Интернете для работы с электронной почтой применяются прикладные протоколы SMTP, POP или IMAP, которые базируются на протоколах объединенной сети TCP/IP.

Протокол *SMTP* (*Simple Mail Transfer Protocol*) – простой протокол передачи почты, является стандартным протоколом для передачи сообщений между почтовыми серверами сети Интернет. Взаимодействие между клиентом и сервером осуществляется с помощью команд, посылаемых в виде ASCII-строк.

Сначала клиент устанавливает с сервером надежное потоковое соединение и ожидает, пока сервер не отправит сообщение типа 220 – READY FOR MAIL, свидетельствующее о его готовности к приему электронной почты. Если сервер перегружен, он может временно задержать отправку сообщения 220. Получив сообщение 220, клиент отсылает команду HELO. Признаком конца команды является символ конца строки. В ответ сервер предоставляет свою идентификационную информацию.

Как только взаимодействие установлено, отправитель может переслать одно или несколько сообщений электронной почты, закрыть соединение или послать серверу запрос на обмен ролями отправителя и получателя. Последняя команда необходима для того, чтобы можно было обменяться сообщениями, следующими в обратном направлении. Получатель должен подтвердить прием каждого сообщения. Он также может досрочно закрыть соединение с сервером или оборвать текущую передачу сообщения.

Сеанс передачи электронной почты начинается с команды MAIL, посредством которой клиент идентифицирует отправителя сообщения, пересылая серверу информацию из поля "FROM:", в чей адрес следует пересылать

возможные сообщения об ошибках. При этом сервер подготавливает внутренние структуры данных для получения нового сообщения электронной почты, и в ответ на команду MAIL отсылает сообщение с кодом 250. Сообщение 250 означает, что процесс подготовки к получению почты прошел успешно. Полный ответ сервера состоит из текста "250 ОК".

Завершив выполнение команды MAIL, отправитель с помощью нескольких команд идентификации получателя RCPT идентифицирует адресатов сообщения электронной почты. Получатель должен подтвердить каждую команду RCPT путем отсылки ответного сообщения 250 ОК или сообщения об ошибке 550, содержащего текст No such user here *(пользователь не найден)*.

После того как на все команды RCPT будут получены сигналы подтверждения, отправитель выдает команду DATA. Этим действием он информирует получателя о своей готовности к отсылке полного сообщения электронной почты. На команду DATA получатель отвечает сообщением 354 – "начать прием почты" (*Start mail input*) и указывает последовательность символов, которые будут использоваться в качестве признака окончания сообщения электронной почты. Такая последовательность состоит из пяти символов: возврат каретки, перевод строки, точка, возврат каретки и перевод строки (<CR+LF+. +CR+LF>).

После отправки всех сообщений, предназначенных для определенного почтового сервера, клиент может выдать команду TURN, чтобы изменить направление передачи сообщений по каналу связи. При подаче такой команды сервер отвечает текстом 250 ОК и берет на себя управление каналом связи. После обмена ролями сторона соединения, которая раньше была сервером, отсылает клиенту сообщения электронной почты, находящиеся в очереди на отправку. Завершить сеанс связи может та из сторон, которая в настоящий момент управляет взаимодействием (т.е. клиент). Для этого используется команда QUIT. Другая сторона отвечает строкой 221, означающей согласие на завершения сеанса. Затем обе стороны по очереди закрывают TCP-соединение.

Почтовый офисный протокол POP (*Post Office Protocol)* дает пользователю доступ к пришедшим к нему на почтовый сервер электронным сообщениям. Протокол POP позволяет осуществлять связь между компьютером пользователя и почтовым сервером, на котором зарегистрирован почтовый ящик пользователя и выполняет пересылку всех сообщений, адресованных пользователю с сервера на его компьютер. В настоящее время применяется третья версия офисного протокола – POP3.

Для получения почты пользователь вызывает программу клиента протокола POP3, которая устанавливает TCP-соединение с POP3-сервером, запущенным на компьютере, где находится удаленный почтовый ящик. Для

аутентификации сеанса связи пользователь отправляет *регистрационное имя* и *пароль*. Когда процесс аутентификации успешно пройден, клиентская программа посылает серверу команды на доставку копии одного или нескольких сообщений и стирания их из удаленного почтового ящика. Сообщения хранятся и передаются в виде текстовых файлов, представленных в стандартном формате "822".

Следует заметить, что на почтовом сервере должны быть запущены две почтовые программы. Первая – сервер SMTP – принимает предназначенную пользователю почту и помещает полученные сообщения в его почтовый ящик. Вторая – сервер POP3 позволяет пользователю извлекать сообщения из почтового ящика и удалять их. Чтобы обеспечить корректную работу системы, оба сервера должны согласованно использовать почтовый ящик. Поэтому если сообщение поступает через SMTP-сервер, в то время как пользователь извлекает сообщения через сервер протокола POP3, содержимое почтового ящика не искажается.

Протокол доступа к сообщениям в сети Internet IMAP4. Версия 4 протокола доступа к сообщениям в сети Internet *IMAP 4 (Internet Message Access Protocol)* является альтернативой протоколу POP3, в основу работы которого положен такой же общий принцип. Подобно протоколу POP3, в протоколе IMAP4 определено абстрактное понятие, называемое *почтовым ящиком* (mailbox). Почтовые ящики размещаются на том же компьютере, что и сама программа сервера. Аналогично протоколу POP3, пользователь запускает программу клиента протокола IMAP4, которая связывается с сервером для выборки своих сообщений. Однако в отличие от POP3 протокол IMAP4 дает возможность пользователю динамически создавать, удалять или переименовывать почтовые ящики.

В протоколе IMAP4 расширены функциональные возможности по выборке и обработке сообщений. Пользователь может получить информацию о сообщении или проанализировать поля его заголовка, не извлекая сообщение целиком. Кроме того, пользователь может выполнить поиск заданной строки и извлечь из сообщения определенные части. Такая возможность особенно удобна при работе по низкоскоростным коммутируемым каналам связи, поскольку пользователю не нужно загружать с сервера лишние данные.

Для обеспечения передачи по электронной почте данных, представленных не в ASCII-формате, разработан протокол многоцелевых расширений электронной почты в сети Интернет – ***MIME*** *(Multipurpose Internet Mail Extensions*. Стандарт MIME не вносит изменений в протоколы SMTP или POP3 и не заменяет их. В протоколе MIME определены алгоритмы кодирования двоичных данных и представления их в формате ASCII. Это позволяет пересылать получателю файлы произвольных типов в обычном сообщении

электронной почты. Для этого в каждое сообщение протокола MIME помещают служебную информацию, в которой указывается тип передаваемых данных и используемый алгоритм кодировки. Информация протокола MIME помещается в заголовок электронной почты формата "822". В MIME-заголовках указывается используемая версия протокола MIME, тип передаваемых данных и метод кодирования, применяемый для преобразования данных в формат ASCII.

4.7. Выводы по разделу

1. Объединенная сеть Интернет (*Interwork* или *Internet*) представляет собой совокупность независимых компьютерных сетей, соединенных между собой маршрутизаторами (шлюзами). Взаимодействие отдельных разнородных подсетей осуществляется на основе единых межсетевых протоколов TCP/IP. Для пользователей объединенная сеть выглядит как единая виртуальная сеть, к которой подключены все компьютеры.

2. Обмен информацией между подсетями осуществляется пакетами без привлечения прикладных программ. Взаимодействие между подсетями происходит на сетевом уровне, в функции которого входит передача пакетов между конечными узлами.

3. Для обеспечения универсального взаимодействия между компьютерами объединенной сети разработано семейство протоколов TCP/IP, в котором протоколы расположены по слоям (уровням). Такая совокупность протоколов получила название "стек протоколов". Стек протоколов TCP/IP отличается от стека модели OSI и содержит всего лишь четыре уровня: приложений, транспортный, межсетевой и доступа к сети.

4. Обмен данными между приложениями различных компьютеров сети осуществляется отдельными блоками. На прикладном уровне данные, подлежащие передаче, представлены в форме прикладного сообщения. Прикладное сообщение на транспортном уровне делится на части, которые в зависимости от вида транспортного протокола называется TCP-сегментом (протокол TCP) или UDP-дейтаграммой. На сетевом уровне блок сообщения получил название IP-пакет, а на уровне доступа – кадр, фрейм или блок.

5. При движении блока данных сверху вниз по стеку протоколов на каждом уровне к нему добавляется заголовок соответствующего уровня, т.е. происходит инкапсуляция данных. В процессе движения кадра по стеку снизу вверх осуществляется декодирование и удаление заголовков. При декодировании заголовков определяется протокол последующего верхнего уровня и передача ему инкапсулированных данных.

6. В объединенных сетях применяются два вида адресов: физический (адрес сетевого адаптера) и сетевой IP-адрес. Последний состоит из адреса сети и адреса сетевого компьютера или маршрутизатора.

7. Межсетевые 32-битовые IP-адреса делятся на 5 классов. Первые три класса различаются количеством номеров сетей и хостов, четвертый класс позволяет адресовать пакет группе хостов, а пятый является резервным.

8. Обеспечение установления соответствия между сетевыми IP-адресами и аппаратными MAC-адресами возлагается на протокол ARP.

9. Для отделения друг от друга редко взаимодействующих компьютеров сети одной и той же организации их разделяют на подсети. Адресация подсети реализуется за счет части битов адреса хоста. Выделение адреса подсети осуществляется с помощью маски подсети, содержащей все единицы в битовых полях адреса сети и подсети.

10. Передача блоков данных между узлами сети производится согласно протоколу IP. Минимальной независимой единицей передачи информации является межсетевая дейтаграмма (МД). IP-протокол реализует негарантированную доставку пакетов, а также производит их маршрутизацию.

11. Номинальная длина МД равна 576 байтам, из них 64 выделено под заголовок, а 512 – под поле данных. В поле заголовка IP-дейтаграммы размещаются IP-адреса отправителя и получателя, а также поля, указывающие версию протокола, длину заголовка и полную длину МД, возможность ее фрагментации, протокол близлежащего верхнего уровня, время жизни пакета и др. Максимальная длина МД содержит 65535 октетов.

12. Объединенная сеть Интернет состоит из отдельных подсетей, составные части которой могут иметь различную максимальную допустимую величину блока передачи (MTU). Поэтому в функции IP-модуля входит фрагментация длинных МД при необходимости передачи таких дейтаграммм через подсети с малой допустимой длиной кадров.

13. Сборка отдельных фрагментов в исходную дейтаграмму выполняется только на последнем этапе, поскольку отдельные фрагменты могут перемещаться по сети различными путями. Порядок сборки определяется указателем фрагмента. В случае необходимости фрагментация дейтаграммы может быть запрещена, о чем в поле флагов ставится соответствующая пометка.

14. Основным недостатком существующей четвертой версии IP-протокола – сравнительно малое адресное пространство, задаваемое 32-битовым полем. Этот недостаток ликвидирован в шестой версии IPv6, в котором длина адреса увеличена до 128 битов.

15. В заголовке IPv6 содержится меньше информации по сравнению с IP, однако, его дейтаграмма может включать несколько заголовков. Кроме того, в заголовке введены новые поля, задающие качество обслуживания,

защиту частной информации и др.

16. Обеспечение передачи данных между пользовательскими процессами регламентируются протоколами транспортного уровня TCP и UDP. Протокол TCP создает сквозной виртуальный канал с получателем, реализует потоковую безошибочную доставку пользовательских сообщений, осуществляя подтверждение каждого правильно принятого сегмента. В случае неполучения такого подтверждения отправитель по истечении тайм-аута повторяет неподтвержденный сегмент. TCP-протокол производит регулирование темпа потока данных, предотвращая перегрузку сети.

17. Протокол пользовательских дейтаграмм UDP передает прикладное сообщение в форме дейтаграмм без предварительного установления соединения. Доставка дейтаграмм при этом не гарантируется.

18. Оба протокола транспортного уровня осуществляют мультиплексирование и демультиплексирование пакетов по конкретным приложениям, номера портов которых задаются в заголовках пакетов.

19. Для выбора оптимального пути, по которому будут передаваться пакеты получателю, используется процедура маршрутизации. Если отправитель и получатель относятся к одной физической сети, то выполняется прямая маршрутизация, а при расположении получателя в другой физической сети осуществляется непрямая маршрутизация.

20. Сеть Интернет состоит из множества независимых автономных систем (AS), находящихся под контролем одного административного органа. Определение маршрутов в пределах одной AS регламентируется внутренними протоколами маршрутизации, к которым относятся протоколы RIP и OSPF.

21. Автономные системы объединяются между собой посредством внешних (пограничных) маршрутизаторов, взаимодействующих между собой через магистральную (опорную) сеть под управлением протокола внешней маршрутизации (BGP).

22. Протокол внутренней маршрутизации RIP относится к дистанционно-векторным протоколам с использованием алгоритма поиска оптимального пути Беллмана-Форда. Функции RIP-модуля в маршрутизаторе состоят в рассылке, получении и обработке векторов расстояний до IP-сетей, находящихся в области действия протокола. Вектор расстояния содержит адрес местонахождения и метрику. Описание маршрутов хранится в таблице маршрутизации. Достоинство протокола – простота конфигурации и эксплуатации; недостаток – ограничение количества шагов маршрута (15), большие затраты времени на восстановление связи после сбоя.

23. Протокол внутренней маршрутизации OSPF относится к протоколам учета состояния линий связи. Поиск оптимального пути находится по алгоритму Дийкстры. В качестве метрики используется коэффициент каче-

ства обслуживания, определяемый параметрами: пропускная способность канала, задержка распространения пакета, число шагов до получателя и др.

24. К преимуществам OSPF относится возможность функционирования протокола в сетях любой сложности и отсутствие ограничений, характерных для RIP. Время, используемое на построение таблиц маршрутизации и загрузки сети служебной информацией в среднем меньше, по сравнению с тем, что потребовал бы RIP для такой же системы. Если между узлами сети существуют несколько маршрутов с одинаковыми или близкими по значению метриками, то протокол OSPF позволяет распределять трафика по этим маршрутам; недостаток протокола состоит в том, что он сложнее RIP.

25. Протокол внешней маршрутизации BGP применяется для определения маршрутов, соединяющих одну автономную систему с другой, а также маршрутов, проходящих через автономную систему, не участвующей в процессе BGP, используя политику маршрутизации. Отличительной особенностью протокола является применение маршрутно-векторной маршрутизации, при которой не используется метрика маршрутизации.

26. Для более рационального использования адресного пространства и сокращения записей в таблицах маршрутизации была разработана технология бесклассовой адресации CIDR. Она позволяет провайдерам сети Интернет выделять своим клиентам непрерывный блок IP-адресов нужного размера, причем количество адресов в блоке всегда кратно степени двух. Для определения блока адресов задается его начальный адрес и маска, которая указывает границу раздела между префиксом сети и суффиксом хоста.

27. Для информирования узлов о возникновении нарушений работы и ошибок в сети предназначен протокол передачи управляющих сообщений ICMP. При обнаружении проблемных ситуаций маршрутизаторы или конечные станции отправляет ICMP-сообщение определенного типа с указанием кода ошибки. Передача ICMP-сообщения выполняется посредством обычной IP-дейтаграммы. Протокол транспортного уровня (TCP), получая сообщения ICMP об ошибках в сети, может выполнять те или иные действия для преодоления возникших проблем.

28. На основе сети Интернет организовано ряд сервисных служб, позволяющих осуществлять удаленный терминальный доступ (*Telnet, Rlogin*), передачу файлов (*FTP*), работу с всемирной гипермедиа информационной службой *WWW*, осуществлять поиск информации, пользоваться услугами электронной почты и др.

29. Для более углубленного изучения вопросов построения и настройки сети, управления и тестирования Вам следует обратиться к специальной литературе, в частности [3, 10, 11, 18, 19, 24, 32, 36].

4.8 Контрольные вопросы

1. С помощью каких технических средств осуществляется объединение разнородных сетей в единую сеть Интернет?
2. На каком уровне происходит взаимодействие подсетей объединенной сети, и каковы преимущества такого взаимодействия?
3. Проведите сравнительную характеристику стеков протоколов TCP/IP и OSI.
4. Поясните процедуру инкапсуляции данных, и с какой целью она введена в компьютерных сетях?
5. Каким образом маршрутизатор определяет класс адреса поступившей межсетевой дейтаграммы?
6. Что такое "частная сеть", и каким образом осуществляется адресация компьютеров в таких сетях?
7. Поясните необходимость и способ реализации процедуры преобразования Интернет-адресов.
8. С какой целью выполняют разбиение сетей на подсети, и какова роль маски сети?
9. На основании каких признаков осуществляется сборка межсетевых дейтаграмм на оконечном узле в правильной последовательности?
10. Для какой цели в IP-дейтаграмму введено поле дополнительных услуг?
11. Как и для чего на сетевом уровне производится фрагментация и дефрагментация дейтаграмм?
12. Какова цель введения протокола IPv6 и в чем заключается существенное отличие в форматах пакетов IPv6 и IPv4?
13. В чем состоит отличие протоколов транспортного уровня UDP и TCP?
14. Каким образом поступившее в хост сообщение попадает в конкретное приложение?
15. Какой смысл применять протокол UDP, не гарантирующий доставку пакетов, если есть протокол TCP, гарантирующий их доставку?
16. В чем состоит различие потоковой и дейтаграммной передачи сообщений?
17. С какой целью в транспортном протоколе TCP введен механизм проталкивания?
18. Поясните назначение каждого из флагов заголовка TCP-сегмента.
19. Поясните суть и необходимость процедуры "трехразового рукопожатия", используемую в процессе установления TCP-соединения.
20. Каким образом может быть реализована автоматическая настройка параметров стека протоколов TCP/IP и в каких случаях ее целесообразно применять?

21. В каких случаях хост должен самостоятельно выполнять маршрутизацию?
22. Какие данные располагаются в таблице маршрутизации, и на основании чего осуществляется их модификация?
23. Для какой цели применяются протоколы внутренней и внешней маршрутизации?
24. По какой причине был введен протокол внутренней маршрутизации OSPF, если для аналогичных задач уже применялся протокол RIP?
25. Какая часть компьютерной сети может быть отнесена к автономной системе и какова необходимость выделения таких систем?
26. Раскройте особенности внешней маршрутизации и дайте сравнительную характеристику протоколов, реализующих этот вид маршрутизации.
27. С какой целью введена бесклассовая адресация, и в чем состоит ее суть и преимущества?
28. По какой причине при использовании бесклассовой адресации маршрутная информация рассылается вместе с сетевым префиксом и маской сети?
29. Каковы функции реализует протокол передачи управляющих сообщений ICMP?
30. Что представляет собой сетевой виртуальный терминал, и какие технические средства его реализуют?
31. Расскажите о назначении и функционировании службы FTP?
32. Как осуществляется доступ к серверу всемирной информационной службы WWW?
33. С какой целью введена служба доменных имен DNS и как она функционирует?
34. Расскажите об особенностях функционирования электронной почты. Зачем на почтовом сервере запускаются две почтовые программы?

Раздел 5
ГЛОБАЛЬНЫЕ СЕТИ СВЯЗИ

Почему нельзя передавать данные по телефонным каналам без модемов? Каким образом в xDSL-технологии удается передавать данные по кабелям связи городской телефонной сети общего пользования со скоростью в десятки раз выше, чем это осуществлялось посредством телефонных модемов? Как удается совместить по скорости передачи цифровые каналы европейской и американской систем уплотнения? Можно ли дополнительно повысить скорость передачи сигналов по оптическим линиям? Что означает "интегральное обслуживание" в цифровых сетях? Почему в высокоскоростных сетях передачи данных ослаблены мероприятия по защите от ошибок? Чем вызвано существенное уменьшение длины блока в сетях ATM? На эти и другие вопросы Вы найдете ответы в данном разделе.

5.1. Общая характеристика глобальных сетей

Глобальные компьютерные сети служат для объединения территориально рассредоточенных компьютеров, находящихся на значительном удалении друг от друга, либо в различных городах и странах. Хронологически они появились раньше локальных. Глобальные сети очень многое унаследовали от других, гораздо более старых и распространенных сетей связи - телефонных. Так как прокладка высококачественных линий связи на большие расстояния обходится очень дорого, то в первых глобальных сетях использовались уже существующие каналы, изначально предназначенные для других целей. В течение многих лет глобальные сети строились на основе **телефонных каналов тональной частоты**, позволяющих в каждый момент времени вести передачу только одного разговора в аналоговой форме.

Прогресс глобальных компьютерных сетей во многом определялся прогрессом телефонных сетей. С конца 60-х годов XX века в телефонных сетях стала все чаще применяться передача речи в цифровой форме, что привело к появлению высокоскоростных цифровых каналов, соединяющих АТС и позволяющих одновременно передавать по парам проводов десятки и сотни разговоров. Была разработана специальная технология **плезиохронной цифровой иерархии** PDH (*Plesiochronous Digital Hierarchy*). Технология PDH поддерживала скорости до 140 Мбит/с. Для реализации этой технологии были созданы цифровые системы передачи данных, базирующиеся на

многоканальных системах передачи - аппаратуре уплотнения линий связи с импульсно-кодовой модуляцией (ИКМ). Сеть, состоящая из многоканальных систем передачи на основе типовых физических линий и каналов, сетевых узлов распределения и коммутации сигналов и предназначенная для доставки сообщений между ее любыми абонентами, называется **первичной** сетью. Первичная сеть является фундаментом, на котором строятся национальные системы телекоммуникации. Первичная сеть "Укртелеком" состоит из магистральных и зоновых линий связи. **Магистральные линии** связывают области, крупные города и стыкуются с международными линиями. **Зоновый уровень** обеспечивает внутриобласную связь и имеет выход на магистральные линии. В современных первичных сетях используются цифровые систем передачи, а в качестве линий связи применяются электрические и оптические кабели, а также тракты радиорелейной и спутниковой связи.

Наличие такой инфраструктуры способствовало появлению **глобальных цифровых сетей,** представляющих собой совокупность узлов коммутации и высокоскоростных цифровых каналов связи, расположенных на территории региона (области, страны, континента или всего земного шара), и предназначенных для обеспечения услуг связи большому количеству абонентов, расположенных в пределах региона. Глобальные цифровые сети называют также *территориальными* или *региональными* сетями. Англоязычное название глобальной сети – *Wide Area Networks*, **WAN**.

Типичными абонентами глобальной сети являются локальные сети предприятий и корпораций, расположенных в разных городах страны, которым требуется взаимный обмен информацией. Пользоваться услугами сети могут и отдельные компьютеры. Крупные компьютеры класса мейнфрейм обычно обеспечивают доступ к корпоративным данным, в то время как персональные ЭВМ используются для доступа к корпоративным данным и публичной информации Интернет. Глобальные сети характеризуются очень высокой стоимостью, которая обусловлена стоимостью линейных сооружений и узлов коммутации и распределения информации.

Глобальные сети используются преимущественно как **транзитная транспортная система**, обеспечивающая доставку сообщений между абонентами сети. Такие сети предоставляют в основном услуги трех нижних уровней эталонной модели взаимодействия открытых систем *OSI*. Однако, в последнее время количество услуг верхнего уровня, предоставляемыми глобальными сетями, постоянно растет. Сюда в первую очередь относится доступ к гипертекстовой информации, широковещательная рассылка видео- и аудиоинформации и др. В результате глобальные и локальные сети постепенно сближаются за счет взаимопроникновения технологий разных уровней – от транспортных до прикладных.

При создании высокоскоростных сетей передачи данных, речи, видео и

мультимедиа в территориальных и крупных корпоративных сетях все чаще применяются *выделенные цифровые каналы* первичной сети связи, созданные на основе новых коммуникационных технологий, таких как **цифровая синхронная иерархия *SDH*** (*Synchronous Digital Hierarchy*) и технология **плотного волнового мультиплексирования** DWDM (*Dense Wave Division Multiplexing*). SDH расширила диапазон скоростей цифровых каналов до 10 Гбит/с, а технология спектрального мультиплексирования DWDM позволила организовать на оптических линиях связи цифровые каналы со скоростью до сотен гигабит в секунду.

Высокоскоростные цифровые каналы непосредственно соединяют маршрутизаторы, размещаемые на границе локальных сетей отделений предприятий и корпораций. Преимуществом выделенных каналов является гарантированная пропускная способность и минимальная временная задержка. Однако компьютерные сети многих предприятий не в состоянии на 100% загрузить выделяемые им дорогостоящие каналы и владельцы сетей фактически оплачивают неиспользуемое время.

Проблема недозагрузки каналов может быть устранена путем разделения пропускной способности цифровых каналов первичной сети за счет коммутации каналов или коммутации пакетов. Коммутация каналов была реализована в цифровых сетях интегрального обслуживания ***ISDN*** (*Integrated Services Digital Network*), а коммутация пакетов – в сетях с ретрансляцией кадров ***Frame Relay*** и асинхронных сетях передачи сообщений ***ATM*** (*Asynchronous Transfer Mode*).

5.2. Аналоговые телефонные сети

5.2.1. Структура и особенности построения сети

Первые телефонные сети были аналоговыми, так как в них абонентское устройство (телефонный аппарат) преобразовывало звуковые колебания, являющиеся аналоговыми сигналами, в колебания электрического тока. В настоящее время в телефонных сетях все чаще применяется передача речевых сообщений в цифровой форме с последующим мультиплексированием пользовательских каналов по времени.

Типичная структура аналоговой телефонной сети показана на рисунке 5.1. Сеть состоит из узлов коммутации, роль которых выполняют автоматические телефонные станции АТС. В пределах одного населенного пункта АТС соединяются между собой по принципу "каждый с каждым" посредством кабельных линий. Линии связи между АТС получили название **соеди-**

нительные линии (СЛ). Соединительные линии являются четырехпроводными. Телефонные аппараты, расположенные в окрестности одной из АТС, подключаются к ней с помощью двухпроводных линий, называемых **абонентскими линиями** (АЛ).

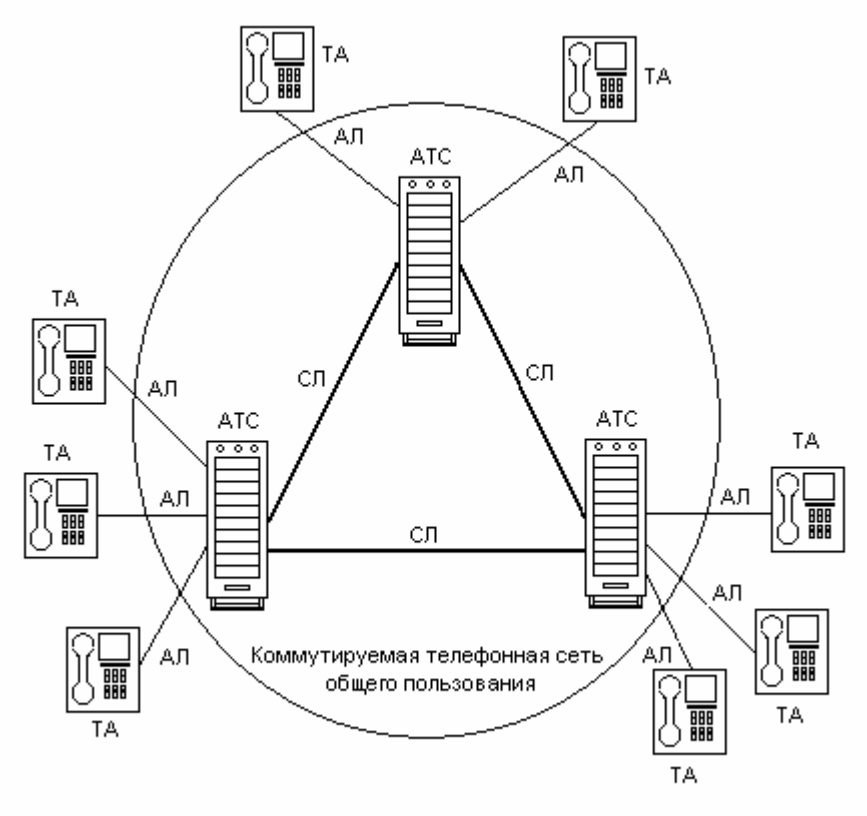


Рисунок 5.1 – Структура аналоговой телефонной сети

Отличительной особенностью коммутируемой телефонной сети является обязательная процедура предварительного установления соединения между абонентскими устройствами. Эта процедура выполняется на основе **протокола сигнализации**, который регламентирует посылку управляющих сигналов установления соединения. Управляющие сигналы в аналоговых телефонных сетях передаются по тем же линиям, что и разговорные токи.

В общем случае процедура установления соединения включает следующие действия.

1. Посылка телефонным аппаратом (ТА) на собственную оконечную АТС сигнала запроса вызова (замыкание шлейфа абонентской линии). АТС отыскивает свободный линейный искатель и посылает вызывающему ТА непрерывный зуммер – сигнал приглашение набора номера, а в случае отсутствия свободного искателя АТС выдает в АЛ сигнал "Занято" (короткие гудки).

2. Получив сигнал разрешения к набору, ТА передает на свой комму-

тационный узел номер вызываемого абонента. Номер абонента состоит из группы десятичных цифр. Первые (2 или 3) цифры обозначают номер АТС, в которую включен вызываемый абонент, а последующие (3 или 4) – номер абонентского ТА. Передача номера в АТС с импульсным набором осуществляется путем размыкания и замыкания шлейфа АЛ с частотой 10 Гц. Количество размыканий линии соответствует цифре номера. В АТС с тональным набором каждая цифра номера передается комбинацией двух гармонических сигналов тонального диапазона.

3. По первым цифрам номера АТС выбирает нужную СЛ, соединяющую ее с оконечной станцией вызываемого абонента. Затем АТС проверяет занятость вызываемого абонента. Если абонент свободен, ему посылается сигнал вызова ("Звонок"). Этот же сигнал транслируется вызывающему абоненту. После снятия вызываемым абонентом микротелефонной трубки, подача сигналов вызова прекращается. Наступает фаза передачи информационных сигналов, в процессе которой вызывающий и вызываемый ТА обмениваются голосовыми сообщениями. Передача сообщений между телефонными аппаратами осуществляется в дуплексном режиме.

4. Разрыв соединения осуществляется по инициативе одного из телефонных абонентов, который кладет трубку ТА в предназначенное для нее место, воздействуя тем самым на контакты рычажного переключателя. При этом происходит размыкание шлейфа абонентской линии. Искатели телефонной станции, участвовавшие в данном соединении, возвращаются в исходное состояние.

Для более эффективного использования пропускной способности линий связи осуществляют их уплотнение. В аналоговых телефонных сетях применяется аппаратура уплотнения с **частотным разделением каналов** (ЧРК). В зарубежной литература частотное разделение каналов называют **частотным мультиплексированием FDM** (*Frequency Division Multiplexing*). Напомним, что под **каналом связи** понимается тракт передачи сигналов, образованный аппаратурой уплотнения путем использования части ресурсов линии связи. В качестве ресурсов линии используется ее полоса пропускания (в системах с частотным разделением каналов) либо время ее занятия (в системах с временным разделением). Аппаратура уплотнения с ЧРК образует иерархию каналов, включающую стандартный канал тональной частоты (ТЧ) с полосой пропускания 0,3…3,4 кГц; первичный широкополосный канал (ПШК) с полосой пропускания 48 кГц и каналы более высокой ступени иерархии. Параметры этих каналов приведены в п.2.1.4

5.2.2. Компьютерные сети на основе аналоговых модемов

В аналоговых телефонных сетях составной канал между абонентами имеет полосу пропускания от 300 до 3400 Гц. Цифровые данные представляются импульсами постоянного тока, спектр которых начинается с нулевой частоты (постоянной составляющей), т.е. телефонный канал является "непрозрачным" для цифровых сигналов. Для возможности передачи двоичных импульсов постоянного тока по такому каналу необходимо перенести их спектр в полосу прозрачности канала. Как известно, перенос спектра осуществляется с помощью процедуры модуляции, в результате которой выполняется умножение спектра исходного сигнала на частоту вспомогательного генератора. Область расположения спектра модулированного сигнала определяется частотой вспомогательного генератора, которая называется **несущей**. Для восстановления спектра исходного сигнала на приемной стороне выполняется процедура демодуляции. Реализация этих процедур осуществляется в модеме, содержащем в своем составе **Мо**дулятор и **ДЕМ**одулятор. Кроме модуляции и демодуляции модем обеспечивает побитную и поблочную синхронизацию, процедуру установления соединения, электрическое сопряжение с линией или каналом связи. В отечественной литературе модем получил название "Устройство преобразования сигналов", сокращенно – УПС. Цепи сопряжения модема носят название "стык". Стандартными интерфейсами сопряжения модема с компьютером и телефонной линией связи являются соответственно стык С2 (аналогичный американский стандарт RS-232C) и стык С1. Схема построения компьютерной сети на основе коммутируемой телефонной сети общего пользования (ТФОП) показана на рисунке 5.2.

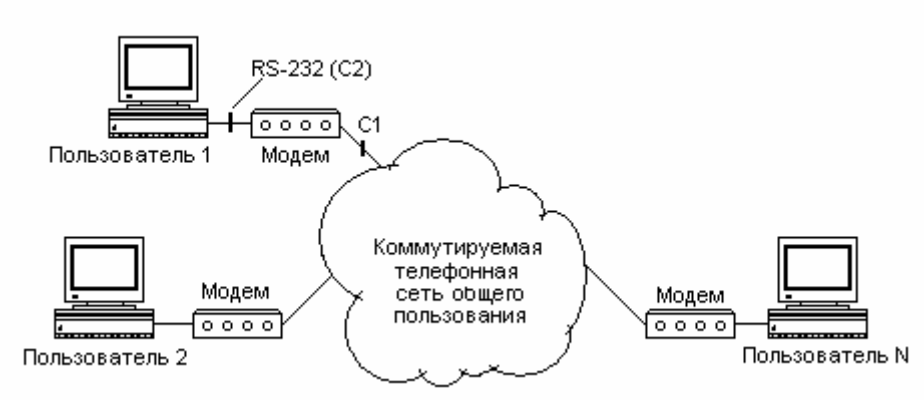


Рисунок 5.2 – Компьютерная сеть на основе модемов

Для телефонной сети модемы являются терминальными устройствами, которые, как и телефоны, выполняют стандартную процедуру вызова або-

нента путем замыкания/размыкания шлейфа линии связи. Максимальная скорость, обеспечиваемая современными модемами на канале тональной частоты, равна 56 кбит/с. Скорость передачи данных ограничивается не только за счет малой ширины полосы пропускания канала тональной частоты, но и по причине относительно высокого уровня помех в канале связи.

Обобщенная структурная схема модема с основными цепями стыка изображена на рисунке 5.3.

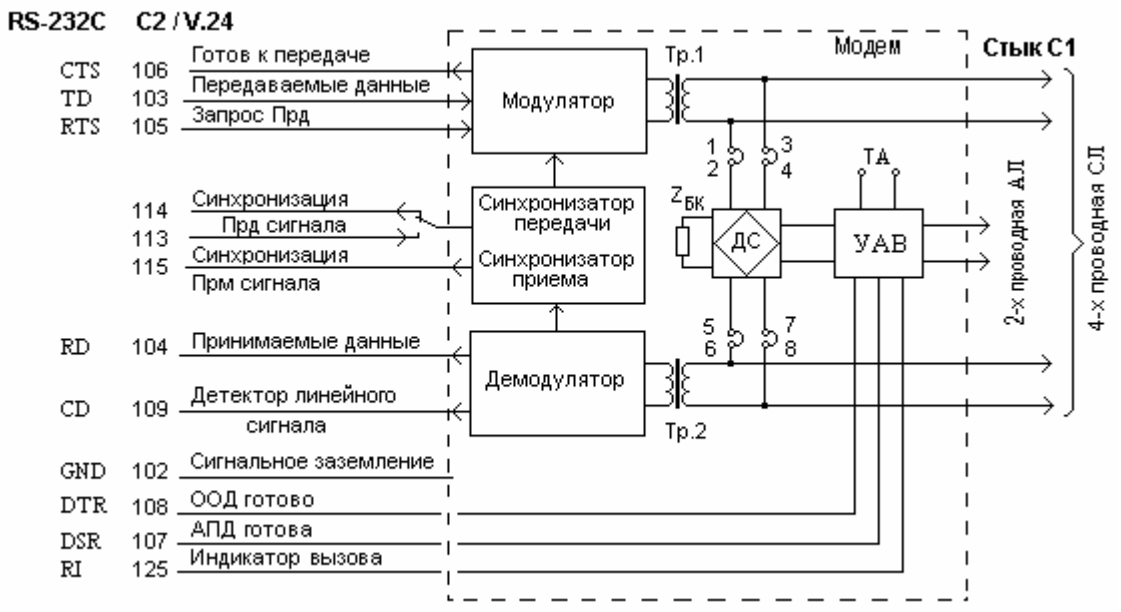


Рисунок 5.3 – Структурная схема модема с цепями сопряжения

Для подключения абонента к компьютерной сети ему могут быть предоставлены коммутируемый или некоммутируемый (выделенный) канал связи. Коммутируемый канал имеет двухпроводное окончание и соединяет абонента с ближайшей коммутационной станцией двухпроводной абонентской линией (АЛ), по которой сообщения могут передаваться в обоих направлениях. Выделенные каналы оканчиваются в помещении абонента в виде 2-х или 4-х проводной линии. В этом случае связь модема с каналом ТЧ осуществляется 2-х или 4-х проводной соединительной линией (СЛ).

Модем, подключенный к 2-х проводной линии, передает или принимает данные по одной и той же паре проводов, а работающий с 4-х проводной линией передает сообщение по одной паре, а принимает по другой. Многие современные модемы могут работать с 2-х или 4-х проводным окончанием канала. Для этого в их состав включается дифференциальная система (ДС), обеспечивающая разделение направлений передачи, осуществляя переход с двухпроводного окончания на 4-х проводное. Гальваническая развязка цепей

передающей и приемной частей и согласование с линией связи осуществляется с помощью трансформаторов Тр.1 и Тр.2. На рисунке 5.3 показана схема подключения линейной части такого модема. При работе с 2-х проводным каналом точки 1-2, 3-4 и 5-6, 7-8 соединяются перемычками, которые удаляются при наличии 4-х проводного окончания.

Ввод и вывод данных, а также управление модемом осуществляется по цепям стыка С2 (рекомендация МККТТ V.24). На рисунке показано соответствующее обозначение цепей стыка по американскому стандарту RS-232C. Следует заметить, что почти все современные модемы имеют только цепи стыка RS-232C. Это объясняется конструктивными особенностями модема, состоящими в том, что собственно модем, устройство защиты от ошибок и устройство автоматического вызова выполнены на одной печатной плате или в одной микросхеме и все необходимые цепи сопряжения располагаются внутри платы либо кристалла модема. Управление модемом осуществляется не уровнями напряжений на цепях стыка, а путем подачи команд по линии 103 (TxD).

Временные диаграммы обмена сигналами по цепям стыка RS-232C/C2 изображены на рисунке 5.4.

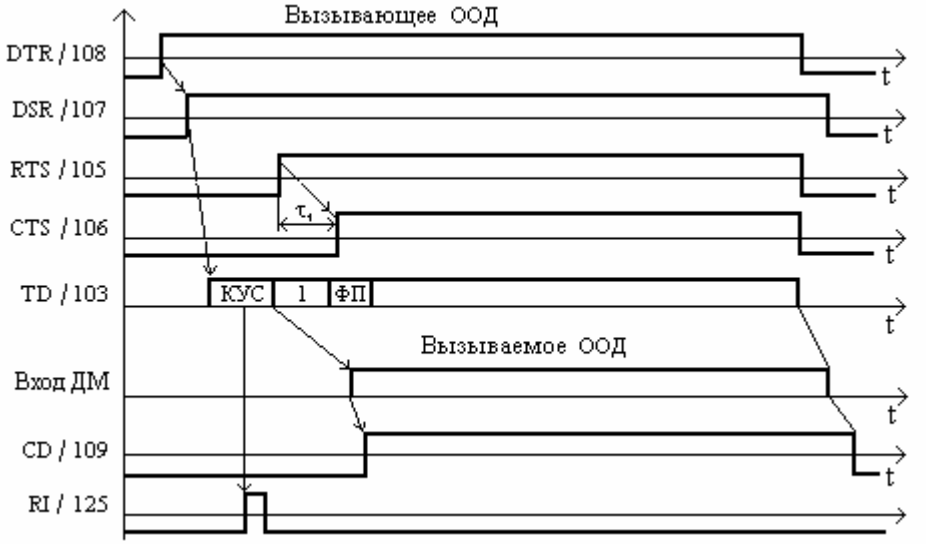


Рисунок 5.4 – Временные диаграммы работы модема на стыке RS-232C/V.24

Рассмотрим более подробно назначение основных цепей стыка и их взаимодействие с блоками модема при двухпроводной схеме подключения модема. При необходимости передачи данных вызывающее оконечное оборудование данных (ООД) включает цепь DTR/108 "ООД готово" (*Data Terminal Ready*). По этому сигналу устройство автоматического вызова модема (УАВ) переключает двухпроводную абонентскую линию с телефонного

аппарата (ТА) на вход модема. Завершение процесса переключения модем индицирует включением цепи DSR/107 "АПД готова" (*Data Set Ready*). Затем оконечное оборудование данных (ООД) по линии TD/103 "Передаваемые данные" (*Transmit Data*) выдает команды на установление соединения (КУС) с вызываемым устройством. Последовательность КУС включает команды занятия линии, проверки наличия сигнала АТС о готовности к установлению соединения, передачу номера вызываемого устройства. После завершения передачи номера АТС устанавливает физическое соединение между устройствами и подает на вход вызываемого ООД сигнал вызова. Устройство автоматического вызова приемного модема регистрирует этот сигнал и включает цепь RI/125 "Индикатор вызова" (*Ring Indicator*). Включение цепи RI/125 заставляет модем "снять трубку" и выдать ответный аналоговый сигнал в абонентскую линию. После приема ответного сигнала удаленного модема вызывающее ООД включает цепь RTS/105 "Запрос передачи" (*Request To Send*). Модулятор модема использует этот сигнал для разрешения выдачи аналогового сигнала передатчика (*несущей*) в линию.

Через небольшую задержку модем переводит цепь CTS/106 "Готов к передаче" (*Clear To Send*) из положения "Выключено" в состояние "Включено". Длительность задержки τ_1 включения цепи CTS/106 определяется типом модема и находится в пределах от десятков до сотен миллисекунд. В течение промежутка времени между включением цепей RTS и CTS аналоговые сигналы поступают в линию связи. Структура этого сигнала зависит от типа модема и в большинстве случаев, как для асинхронных, так и для синхронных модемов состоит из сплошных логических единиц. В некоторых типах модемов применяются специальные синхронизирующие последовательности, используемые как для целей синхронизации, так и для настройки корректора частотных характеристик канала связи.

При обнаружении сигнала несущей частоты детектор уровня приемника удаленной станции переводит состояние цепи CD/109 "Детектор принимаемого сигнала канала данных" (*Carrier Detected*) из состояния "Выкл" (– 6 В) в положение "Вкл" (+6В). При этом между моментом обнаружения несущей и включением цепи CD/109 вносится задержка (аналогично имеется задержка на выключение цепи CD/109).

Во внутренней схеме демодулятора состояния цепи CD/109 используется для разрешения выдачи потребителю по цепи RD/104 *(Receive Data*) принимаемых данных. Во избежание ошибок цепь CD/109 блокирует выход линии 104 при снижении уровня несущей ниже допустимого значения. Задержка τ_1 включения сигнала обнаружения несущей и разрешения цепи приема данных обеспечивает защиту (при отсутствии несущей) от кратковременных выбросов помех в канале, которые могут вызвать появление ложных сигналов в цепи приема данных RD/104. В современных модемах

задержка включения / выключения цепи CD/109 находится в пределах 10…30 мс.

5.2.3. Архитектура модемов для телефонных сетей

Архитектура модемов для телефонных каналов связи регламентируется рекомендациями международного консультативного комитета по телефонии и телеграфии - **МККТТ**, агл.**CCITT** (*International Consultative Committee for Telegraphy and Telephony*) V.21…V.92. С 1994 г. этот комитет получил название ITU-T *(International Telecommunication Union*) – международный союз телекоммуникаций. На основе этих рекомендаций разрабатываются национальные стандарты.

С целью обеспечения требуемой помехоустойчивости в системах передачи данных на канальном уровне применялись устройства защиты от ошибок (**УЗО**), которые выполнялись в виде самостоятельных устройств, соединяемых с модемами стандартными цепями стыка. Для установления соединения на коммутируемых каналах совместно с модемами использовалось устройство автоматического вызова (**УАВ**), которое также выполнялось в виде автономного блока. Совокупность модема, вызывного устройства и УЗО в отечественной литературе получило название "Аппаратура передачи данных" (**АПД**). С появлением специализированных сверхбольших интегральных схем (СБИС) аппаратура передачи данных стала выполняться в виде нескольких микросхем, размещенных на плате в одном корпусе. Это устройство продолжали называть модемом, хотя, строго говоря, оно является аппаратурой передачи данных. В настоящее время эти термины практически повсеместно используются как синонимы.

Модемы, используемые для передачи данных по аналоговым каналам связи, могут передавать сигналы в дуплексном и полудуплексном режимах, синхронным или асинхронным способом. При передаче сигналов по телефонной сети общего пользования канал связи разделяется на два одинаковых подканала, используемые для дуплексной передачи, либо на основной и вспомогательный подканалы – для полудуплексной. Вспомогательный подканал служит для асинхронной передачи сигналов обратной связи (*квитирования*), используемого для подтверждения правильности приема данных в обратном направлении с приемной станции на передающую. Он занимает более узкую полосу частот по сравнению с основным каналом, а стандартная скорость передачи по нему установлена 75 бит/с.

Простейшим модемом для передачи данных по коммутируемому телефонному каналу является Модем-300 (в отечественных стандартах – УПС-0.3 ТФ), выполненный в соответствии с рекомендацией МККТТ V.21. В мо-

деме применяется частотная модуляция, обеспечивающая довольно высокую помехоустойчивость при минимальных затратах на аппаратную реализацию устройства. Передача данных может вестись одновременно в двух направлениях, для чего канал ТЧ разделяется на два подканала со средними частотами 1080 и 1750 Гц. Отклонение частоты (*девиация*) от средней при передаче логического 0 или 1 составляет ± 100 Гц. Распределение частот модема показано на рисунке 5.5

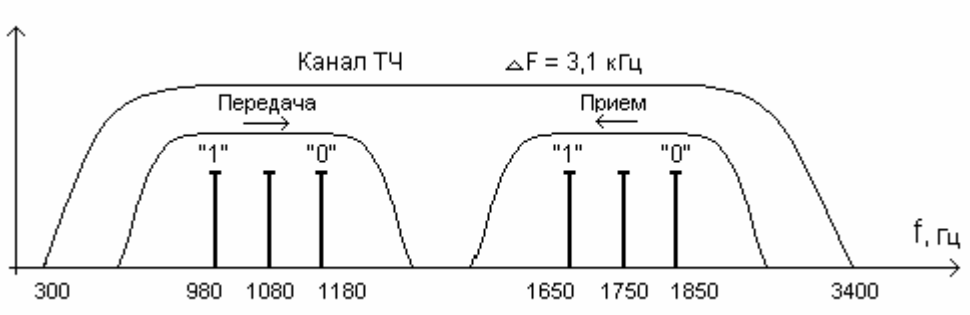


Рисунок 5.5 – Схема распределения частот в подканалах УПС-0,3 ТФ

Увеличение скорости передачи приводит к расширению спектра передаваемых сигналов. Поэтому для передачи данных со скоростью 1200 бит/с используется большая часть полосы коммутируемого канала связи, и передача данных может выполняться только полудуплексным способом.

Для асинхронной и синхронной передачи сигналов по коммутируемому телефонному каналу в полудуплексном режиме МККТТ разработало рекомендацию V.23 по построению Модема-1200 (УПС-1.2 ТФ). В соответствии с этой рекомендацией полоса телефонного канала разделяется на два подканала: основной и вспомогательный (рисунок 5.6). Основной подканал используется для передачи данных, а вспомогательный - для сигналов квитирования. В основном и вспомогательных подканалах применяется частотная модуляция. При передаче данных со скоростью 1200 бит/с средняя частота сигнала выбрана равной 1700 Гц, а девиация ± 400 Гц. В случае ухудшения условий передачи модем может быть переключен на пониженную скорость 600 бит/с, при этом средняя частота снижается до 1500 Гц, а девиация – до ± 200 Гц. Скорость модуляции по вспомогательному подканалу установлена 75 бод, а частоты для передачи 0 и 1 соответственно равны 390 и 450 Гц.

Дальнейшее повышение скорости передачи данных при неизменной скорости модуляции может быть обеспечено за счет увеличения количества значащих позиций модулированного сигнала. Это свойство используется в модемах, выполненных согласно рекомендации V.22. Модем позволяет вести дуплексную передачу по двухпроводному коммутируемому или выделенно-

му каналу тональной частоты со скоростью 1200 бит/с. Для этого телефонный канал разделяется на два подканала с одинаковыми полосами пропускания. Передача по каждому из них осуществляется методом 4-х позиционной фазоразностной модуляции (**ФРМ**) со скоростью 600 бод с несущими частотами соответственно 1200 и 2400 Гц.

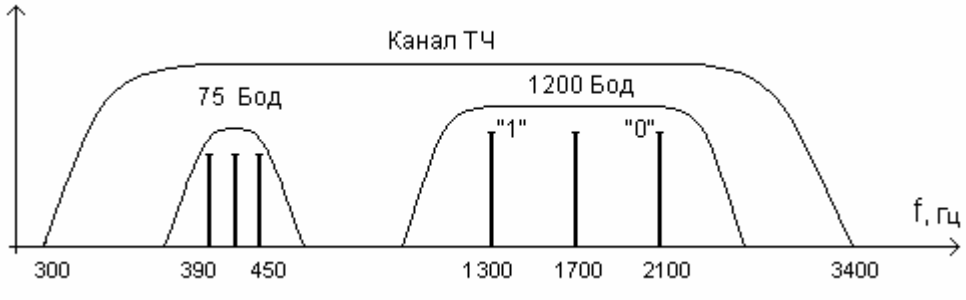


Рисунок 5.6 - Схема распределения частот в подканалах УПС-1,2 ТФ

При 4-х позиционной модуляции последовательности битов, поступающих от источника, объединяются по два (*дибиты*). Дибитам соответствуют разности фаз двух соседних посылок сигнала, согласно фазовой диаграмме, изображенной на рисунке 5.7,а. На рисунке 5.7,б показана форма сигнала на выходе передатчика УПС при использовании 4-х позиционной фазоразностной модуляции.

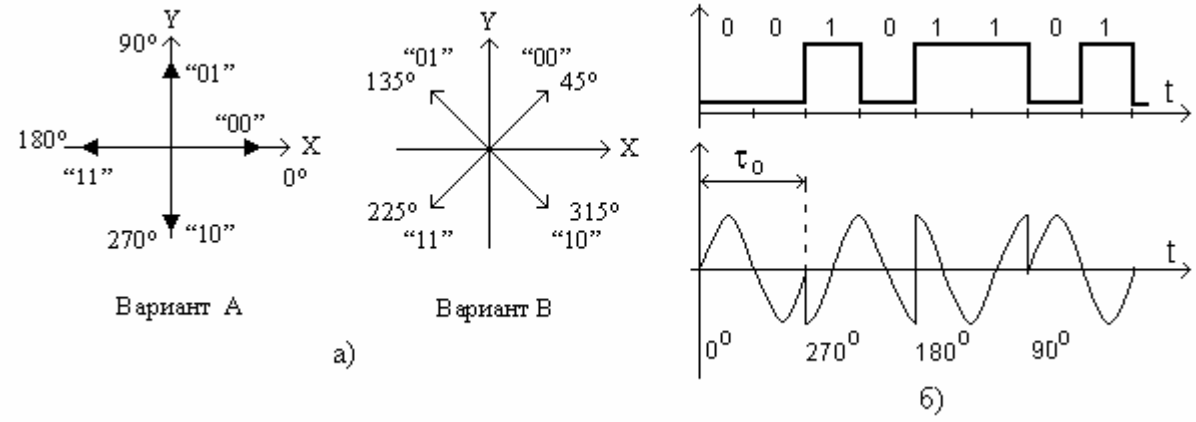


Рисунок 5.7 – Векторные (а) и временные (б) диаграммы сигналов с ФРМ

Передача данных с такой скоростью может быть осуществлена только при точной корректировке частотных характеристик канала связи, для чего в модеме устанавливается автоматический адаптивный детектор.

Рекомендация V.22 bis регламентирует построение дуплексного модема с частотным разделения каналов и с применением квадратурной амплитудной модуляции (**КАМ**) в каждом из подканалов. При этом используются

несущие частоты F_{K1}=1,2 кГц и F_{K2}=2,4 кГц. Скорость манипуляци в каждом из подканалов B=600 бод. В схему модема введены дополнительно блоки скремблера и дескремблера, и адаптивный корректор частотных характеристик линии связи. *Скремблер,* устанавливаемый на входе передатчика, осуществляет преобразование поступающей от ООД информации в псевдослучайную последовательность. *Дескремблер* производит на выходе приемника обратное преобразование, восстанавливая исходную последовательность. Скремблирование позволяет увеличить количество изменений значащих позиций сигнала при передаче последовательностей, где эти изменения отсутствуют. Операция скремблирования / дескремблирования осуществляется делением на образующий полином последовательности, поступающей от источника на передающей стороне и с выхода демодулятора – на приемной. Вид полинома задается стандартом согласно типу модема и скорости передачи.

В последующих разработках скорость передачи данных по двухпроводному каналу ТЧ в дуплексном режиме была повышена сначала до 2400 бит/с, а затем постепенно увеличена еще более чем на порядок. Повышение скорости удалось поднять за счет увеличения позиций сигналов.

Рекомендация V.32 bis регламентирует параметры дуплексного модема с эхоподавлением и модуляцией с *треллис-кодированием* **ТСМ** (*Trellis coded modulation*). Треллис-кодирование представляет собой комбинацию фазоразностной модуляции с решетчатым кодированием и декодированием по алгоритму Витерби (см. пп.2.6.4). Такой вид кодирования повышает помехоустойчивость передачи данных, снижая требуемое отношение сигнал/помеха на входе приемника на 3…6 дБ. Пространство сигналов расширяется вдвое путем добавления к информационным битам еще одного, образованного посредством сверточного кодирования над частью информационных битов. Расширенная таким образом группа подвергается АФМ. При демодуляции применяется алгоритм Витерби, с использованием критерия максимального правдоподобия, позволяющий выбрать из сигнального пространства наиболее достоверную эталонную точку за счет введения избыточности и знания предыстории декодирования и тем самым определить значение информационных битов. Частота несущей в модеме $F_н$ = 1800 Гц, а скорость модуляции 2400 бод. Имеются режимы 16-, 32-, 64- и 128-ТСМ. Соответственно возможно обеспечение скорости 7,2 кбит/с, 9.6 кбит/с, 12 кбит/с и 14,4 кбит/с.

Рекомендация V.33 определяет исполнение дуплексного модема с амплитудно-фазовой модуляцией и применением решетчатого кодирования. Модем обеспечивает дуплексную передачу по 4-х проводному выделенному каналу. Несущая частота $F_н$=1,8 кГц, а скорость манипуляции B=2,4 кбод. Используются режимы модуляции 64- и 128-ТСМ. Информационная ско-

рость соответственно равняется 12 и 14,4 кбит/с.

Рекомендация V.34 регламентирует модем с передачей данных со скоростями до 28,8 кбит/с. При этом достигается почти верхний предел, теоретически допустимый для существующих телефонных линий. Основная скорость модуляции составляет по прежнему 2400 бод. С использованием встроенных алгоритмов компрессии модемы V.34 могут достигать эффективной скорости более 100 кбит/с. Отличительные особенности модема V.34 следующие:

- возможность выбора оптимальной несущей частоты в диапазоне от1600 до 1959 Гц вместо фиксированной частоты 1800 Гц;
- применение 64-, 32- и 16-позиционного кода с решетчатым сверточным кодированием;
- использование новой формы линейной адаптивной коррекции - *предкоррекция*, которая оказывает влияние не только на принимаемый, но и на передаваемый сигнал путем компенсации неравномерности АЧХ. Перед началом передачи осуществляется измерение АЧХ, для чего используется тест-сигнал в диапазоне 150…3750 Гц. Затем производится коррекция частотных характеристик;
- применение нелинейного кодирования, деформирущего расположение внешних точек для достижения неоднородного распределения вероятности ошибки, что повышает устойчивость к нелинейным искажениям.

Модемы V.34 являются мультистандартными (от V.21 до V.34) и поддерживают более шести видов модуляции. Они являются совместимыми не только с большинством дуплексных модемов, но и с факсимильными аппаратами, оборудованием сжатия речи, работающим на цифровых (ИКМ) каналах со скоростью 64 кбит/с, а также поддерживают видеотелефон.

В процессе установления соединения в соответствии с протоколом V.8 модем V.34 определяет список функциональных возможностей, являющихся общими для него и удаленного модема другого типа. После этого тестирующим сигналом определяются характеристики телефонного канала. Затем передатчик и приемник модемов начинают работать на минимальной скорости, наращивая ее до максимально возможной. Вся процедура установления соединения и настройки модемов занимает до 5 секунд.

В модемах V.34 используется процедура помехозащитного кодирования V.42, а для сжатия данных предусмотрена процедура компрессии, основанная на алгоритме Лемпеля-Зива-Уелча (LZW) V.42 bis, которая работает в паре с процедурой V.42. При определенных структурах сообщений коэффициент компрессии может достигать 4, что при технической скорости передачи 28,8 кбит/с обеспечивает эффективную скорость 115,2 кбит/с. Модем V.34 реализует кроме 2400 бод и другие обязательные к применению модуляционные скорости: 3000, 3200 бод и дополнительные скорости: 2743, 2800,

3429 бод.

Рекомендация V.90 регламентирует построение модема для передачи данных по каналам телефонной связи со скоростью до 56 кбит/с. Передача данных по городской телефонной сети со скоростью 56 кбит/с возможна только в случае, если районная АТС какого-либо Интернет-провайдера цифровая и если телефонные каналы связи, соединяющие эту АТС с районной АТС абонента сети, являются каналами ИКМ-системы с временным уплотнением (т.е. тоже цифровыми). Важнейшей особенностью технологии 56К является прямое взаимодействие цифрового модема с цифровым окончанием конкретного канала ИКМ-системы связи (минуя индивидуальный абонентский кодек- ЦАП/АЦП). При этом, в обратном направлении связи в канале нет ни одного АЦП (рисунок 5.8), и следовательно, ничто не порождает шум квантования. За счет этого появилась реальная возможность увеличить скорость передачи в этом направлении.

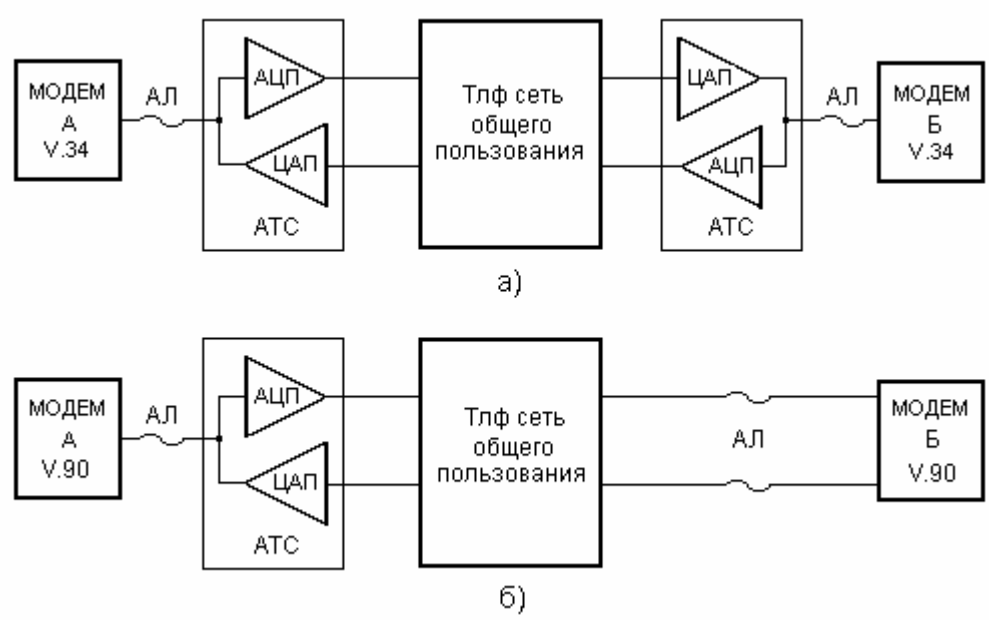


Рисунок 5.8 – Схема передачи данных с модемами V.34 и V.90

Использование модема V.90 позволяют абоненту телефонной сети "перекачивать" информацию с повышенной скоростью до 56 кбит/с входящий поток (*Downstream*), отсылая свою информацию с “ привычной “ скоростью не более 33,6 кбит/с (исходящий поток – *Upstream*). При этом, скорость до 56 кбит/с входящего потока обеспечивается только при обращении к провайдерам Интернет, которые имеют возможность и согласны применить специальную аппаратуру и модемы, рассчитанные для работы в этой технологии, а также специальный вид взаимодействия с их районными цифровыми АТС. Для соединений же типа абонент-абонент по коммутируемой телефонной се-

ти общего пользования технологии 56К непригодны и, следовательно, работа будет возможна только на "обычных" скоростях, т.е. не выше 33,6 кбит/с.

Рекомендация V.92 внесла несколько модификаций в модем V.90. Устройство может передавать данные со скоростью 56 кбит/с на условиях, оговоренных выше. В нем применяется новый протокол сжатия данных V.44. В модем V.92 введены три новых процедуры, обеспечивающие дополнительные услуги пользователям.

Процедура *МОН* *(Modem-on-Hold)* — особенность, позволяющая пользователю при поступления входящего звонка временно приостановить обмен данными, не разрывая соединения, а по окончании разговора вернуться к приостановленной сессии. МОН рассчитана на длительность второго соединения в несколько минут, что уже вполне достаточно для нормального разговора. К тому же клиент и сервер могут заранее задавать время, на которое линия на сервере будет установлена в состояние Hold, до того как сервер посчитает это обрывом связи.

Процедура быстрого вызова *QC* *(Quick Connect)* позволяет уменьшить время установки соединения в части измерения параметров линии связи. Типовой процесс соединения включат четыре стадии:

1) установление физического соединения путем дозвона клиентского оборудования до серверного;

2) выполнение процедуры измерения параметров канала связи, выбор оптимальной скорости соединения, компенсация неравномерности АЧХ канала, подавление эхосигналов аналоговой петли, компенсация прямой и обратной задержки прохождения и пр.;

3) установка цифрового соединения с коррекцией ошибок по протоколу V.42 и, наконец, соединение по протоколу PPP или SLIP.

Суммарное время всех этапов длится 25…30 секунд. Наибольшее время занимает вторая стадия. Процедура быстрого вызова QC предусматривает сохранение параметров настройки (аналоговых — АЧХ и характеристик подавления эхосигналов, а также цифровых — задержки прохождения сигнала и т. п.) клиентского модема в энергонезависимой памяти при первом соединении с использованием процедуры тестирования канала по стандарту V.90. При последующих соединениях клиентский модем загружает предыдущие установки и тестирует только тональный ответ модема провайдера. Если условия связи близки к предыдущим, то процесс тестирования канала пропускается, если нет — снова выполняется стандартная процедура соединения V.90.

Процедура *V.PCM Upstream* — увеличение скорости передачи данных от пользователя к провайдеру до 48 кбит/с. При этом предполагается применение ИКМ в обоих каналах. В модеме V.92 применен новый алгоритм сжатия данных по протоколу V.44, в связи с чем степень сжатия повышается в

среднем на 26% по сравнению с протоколом V.42bis, который использовался в предыдущих версиях модемов.

Как уже отмечалось в начале текущего подраздела, современный модем представляет собой аппаратуру передачи данных, выполняющей кроме переноса спектра сигналов данных в область полосы пропускания канала также процедуры сжатия информации и защиты ее от ошибок. В современном модеме практически все процедуры кодирования, модуляции и коррекции сигналов осуществляются программным способом на основе одного или нескольких микропроцессоров.

Типовая структурная схема микропроцессорного модема изображена на рисунке 5.9. В ее состав входит постоянное запоминающее устройство (ROM) и энергонезависимая оперативная память с произвольным доступом (NVRAM – *Non-Volatile RAM*). В постоянную память записывается фирменное программное обеспечение и интерпретатор команд. Энергонезависимая память содержит буфер ввода/вывода (128…256 байт), а также хранит конфигурационные профайлы модема, телефонные номера и т.д.

Процедуры коррекции, модуляции и демодуляции сигналов, синхронизации по единичным элементам выполняет быстродействующий цифровой сигнальный процессор. ПЗУ сигнального процессора выполняется либо на кристалле процессора, или в виде микросхемы ОЗУ, в которую программа обработки сигналов загружается из ПЗУ контроллера модема.

Входной сигнал проходит через полосовой фильтр (ПФ), усиливается программно управляемой схемой усилителя с автоматической регулировкой уровня и поступает на аналого-цифровой преобразователь (АЦП). В модемах применяются 14…16-разрядные АЦП, а дискретизации сигналов осуществляется с частотой от 7,2 до 9,6 кГц. Преобразования выходных сигналов из цифровой формы в аналоговую выполняет цифро-аналоговый преобразователь (ЦАП), разрядность и частота дискретизации которого обычно совпадает с аналогичными параметрами АЦП. Сглаживающий полосовой фильтр формирует спектр выходного сигнала в соответствии с заданными требованиями к линейному сигналу.

Дифференциальная система ДС выполняется часто на основе трансформатора со средней точкой (см. рисунок 2.8). В более дорогих модемах разделение сигналов передачи и приема осуществляется сигнальным процессором способом эхо-компенсации (рисунок 2.9). Линейный трансформатор служит для гальванической развязки входных цепей модема с телефонной линией и обеспечения входного сопротивления по переменному току, которое должно составлять 600 ±100 Ом.

Защита входной цепи модема со стороны линии связи осуществляется путем параллельного включения варистора, сопротивление которого скачкообразно уменьшается при напряжении свыше 200 В.

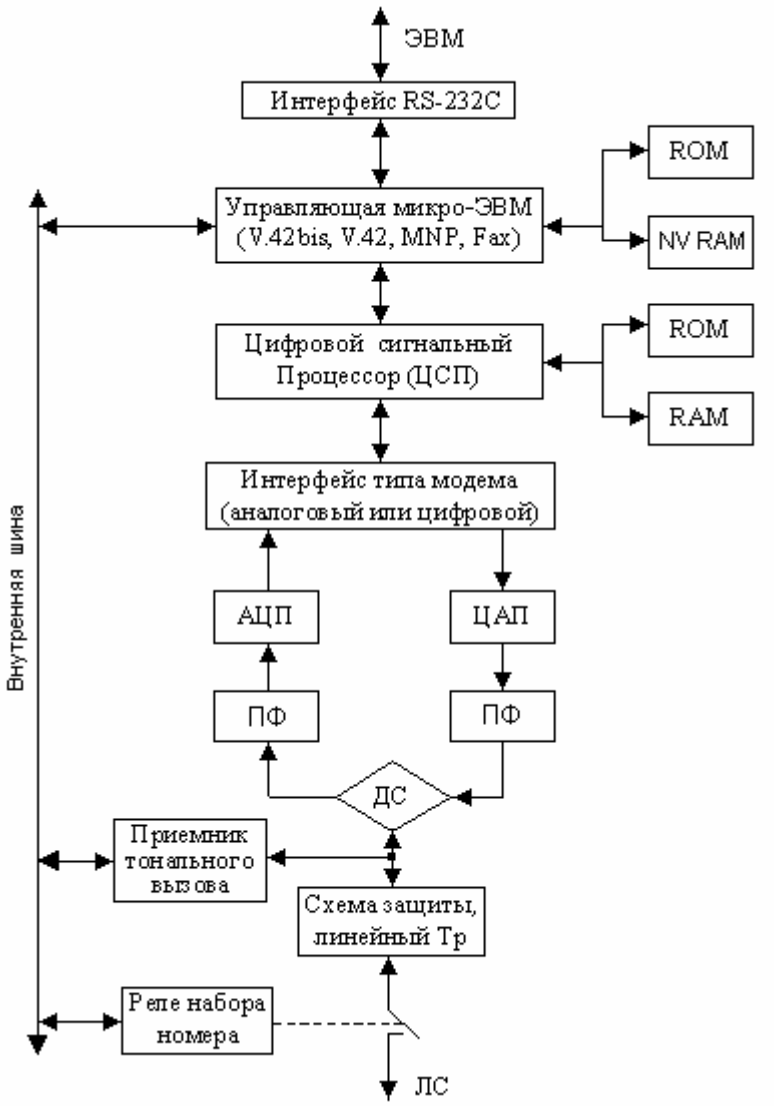


Рисунок 5.9 – Типовая структура микропроцессорного модема

Варистор, кроме того, обеспечивает входное сопротивление модема по постоянному току, равное 300 Ом. Цепь по постоянному току нужна для "удержания линии" коммутирующими элементами АТС на время сеанса связи ("снятия трубки"). Вторая ступень защиты реализуется подсоединением встречно включенных стабилитронов параллельно вторичной обмотке линейного трансформатора (см. рисунок 2.8,б).

Передача сигналов установления соединения (набора номера) осуществляется традиционным способом путем замыкания и размыкания контактами электромагнитного реле шлейфа абонентской линии связи.

5.2.4. DSL-модемы в компьютерных сетях

Для обеспечения эффективной и качественной работы в сети Интернет требуется пропускная способность канала порядка 1,5 Мбит/с. В то же время передача данных по телефонному каналу связи, который используют подавляющее большинство абонентов компьютерной сети, ограничивается полосой пропускания канала и не может практически превысить 115 кбит/с (с учетом стандартной процедуры сжатия). Причем, низкая пропускная способность всего тракта передачи определяется участком между абонентом и узлом (провайдером) сети, так как между узлами (серверами) сети используются преимущественно высокоскоростные каналы связи. Проблема последнего участка тракта передачи называется проблемой "последней мили".

Примечательным является то, что абонентская телефонная линия связи между пользователем и узлом обладает достаточно широкой полосой пропускания, способной обеспечить передачу дискретных сигналов со скоростью несколько мегабит в секунду на расстояние 3…15 км. Однако каналообразующее оборудование телефонной сети общего пользования (ТФОП) использует только часть пропускной способности линии для образования канала ТЧ и ограничивает полосу пропускания телефонного канала на уровне 0,3…3,4 килогерц.

Суть технологии **xDSL** состоит в том, чтобы использовать физические линии телефонной сети для передачи цифровых сигналов, обходя при этом "узкое горлышко" – оборудование линейного тракта ТФОП. Технология **DSL** (*Digital Subscibe Line* – цифровая абонентская линия) обеспечивает высокоскоростные сетевые соединения по обычным местным телефонным линиям путем применения для передачи данных импульсов постоянного тока или специальных методов модуляции сигналов данных. Достигаемые при этом скорость и верность передачи ранее были доступны лишь на волоконно-оптических линиях связи.

Технология xDSL предоставляет пользователям высокоскоростную среду передачи данных между узлами сети по обычной медной телефонной паре. Важной особенностью нового поколения абонентских DSL-устройств является наличие в них встроенного дополнительного узла – частотного разделителя, роль которого выполняет фильтр нижних частот (ФНЧ). Такой узел обеспечивает передачу сигналов аналоговой телефонии в нижней части полосы частот линии связи (рисунок 5.10).

Название **xDSL** – общее обозначение семейства технологий по созданию цифровых абонентских линий. Имеется ряд разновидностей несовместимых между собой xDSL-технологий:

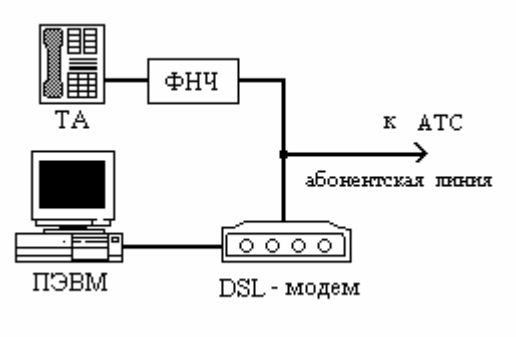


Рисунок 5.10 – Схема подключения компьютера к цифровой абонентской линии

- HDSL – высокоскоростная (*High-bit-rate*) DSL;
- SDSL – симметричная (*Symmetric*) DSL;
- ADSL – асимметричная (*Asymmeric*) DSL;
- RADSL – адаптивная по скорости абонентская линия (*Rate-Adaptive*) DSL;
- UADSL – универсальная асимметричная (*Universal Asymmetric*) DSL;
- VDSL – сверхскоростная (*Very bith-bit rate*) DSL;
- IDSL – ISDN-цифровая абонентская линия.

Симметричная цифровая абонентская линия (SDSL) использует симметричную технологию, при которой данные в обоих направлениях передаются с одинаковой скоростью, в частности, до 2048 кбит/с. SDSL чаще всего применяется на магистральных линиях сетей передачи данных.

Высокоскоростная цифровая абонентская линия (HDSL) представляет собой разновидность симметричной линии и работает на скоростях 1,5…2 Мбит/с в обоих направлениях.

В *асимметричной цифровой абонентской линии* (ADSL) цифровой поток от узла к пользователю (*нисходящий*) более интенсивный, чем обратный (*восходящий*). Эти линии целесообразно использовать для высокоскоростного доступа к Интернету и для осуществления видеоконференц-связи. Нисходящие потоки передаются со скоростью от 1,5 до 6 Мбит/с, восходящие – от 64 до 640 кбит/с.

Адаптивная по скорости абонентская линия (RADSL) – разновидность ADSL. Это одна из самых последних разновидностей xDSL-технологий, позволяющая передавать нисходящие потоки со скоростью от 640 кбит/с до 7 Мбит/с, и восходящие – от 128 до 1500 кбит/с. Ее основное преимущество - способность автоматически подбирать наиболее подходящую скорость передачи в соответствии с уровнем помех в линии.

Универсальная асимметричная цифровая абонентская линия (UADSL) – другая разновидность ADSL. Предназначена для обеспечения доступа в

Интернет. По конструкции представляет собой внутреннюю интерфейсную плату, вставляемую в слот компьютера и поддерживающая технологию *plug and play*. Максимальная скорость передачи 1,5 Мбит/с. Стоимость таких устройств не превышает стоимости обычных модемов для коммутируемых линий.

Сверхскоростная цифровая абонентская линия (VDSL) позволяет передавать нисходящие потоки со скоростями до 51 Мбит/с, а восходящие – от 1,5 до 2,3 Мбит/с на расстояния 100…300 метров. VDSL ориентирована на поддержку передачи данных в АТМ-сетях.

ISDN-цифровая абонентская линия (ISDL) предназначена для передачи данных в цифровых сетях интегрального обслуживания (ISDN) со скоростью 128 кбит/с.

Принцип организации связи при использовании ADSL-технологии иллюстрируется рисунком 5.11.

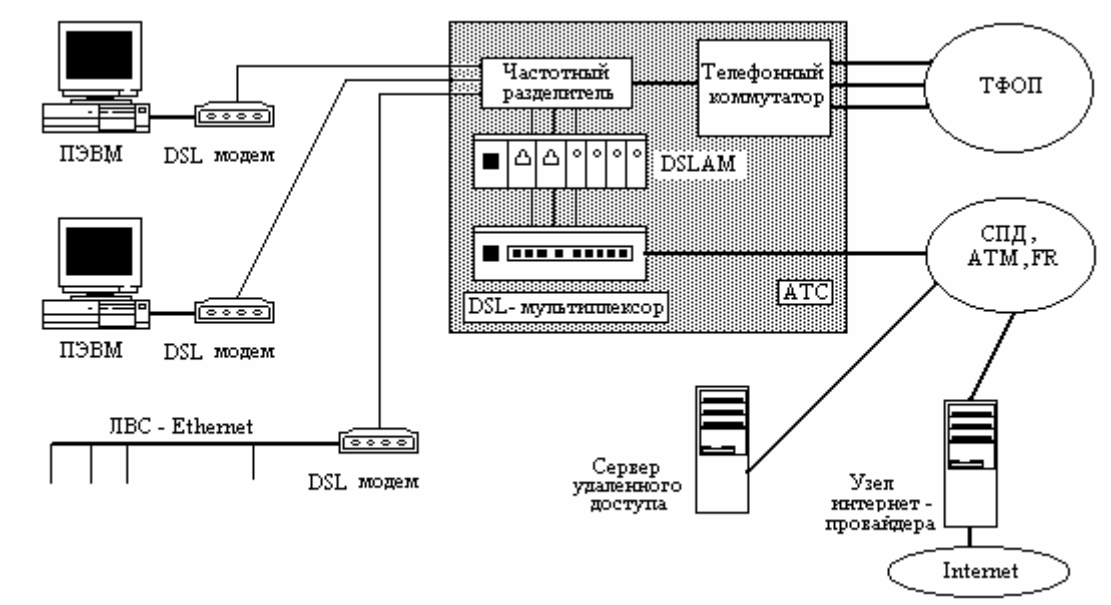


Рисунок 5.11 – Схема организации связи при ADSL-технологии

Услуги ADSL организуется с помощью модема ADSL, расположенного у пользователя и модуля (стойки) модемов ADSL на автоматической телефонной станции (АТС) или у провайдера сети. Эта стойка называется DSL *Access Module* (**DSLAM**). Практически все DSLAM оснащаются портом локальной сети Ethernet, что позволяет использовать на узлах обычные концентраторы, коммутаторы и маршрутизаторы. На участке между DSL-модемом и DSLAM циркулируют три потока: высокоскоростной поток к ПЭВМ, менее интенсивный с запросами ПЭВМ к серверам сети и двунаправленный

речевой (в стандартном диапазоне канала ТЧ).

Реализация цифровых каналов осуществляется путем использования DSL-модемов, DSL-мультиплексоров и другого вспомогательного оборудования. Обход оборудования телефонных каналов ТФОП реализуется путем применения частотного разделителя, который представляет собой фильтр нижних частот (ФНЧ), отделяющий высокоскоростной цифровой поток данных от низкочастотных аналоговых сигналов.

Принципы построения модемов с xDSL рассмотрим на примере построения модема ADSL, регламентируемого рекомендацией ITU-T G.992.1. Особенностью построения модема ADSL является то, что наряду с асимметричной скоростью передачи (к абоненту – до 6,144 Мбит/с и от абонента – до 640 кбит/с), в нем используется способ многоканальной передачи с ортогональным разделением сигналов OFDM (см. п.2.4.5). В качестве ортогональных сигналов применяются гармонические функции

$$\sin k\omega_0 t \text{ и } \cos k\omega_0 t \text{ при } k = 1, 2, \ldots, N; \text{ и } 0 < t < T; \omega_0 = 2\pi F_0, \quad (5.1)$$

ортогональные на интервале $\tau_0 = 1/F_0$.

Длительность тактового интервала $T_{ти}$ выбирается несколько большей, чем интервал ортогональности τ_0. Их разность составляет защитный временной интервал $\tau_з$ между последовательно передаваемыми единичными элементами группового сигнала, вводимого для повышения защищенности OFDM от интерференционных помех. Система ортогональных сигналов (5.1) имеет существенное преимущество по сравнению с другими: во-первых, высокая концентрация энергии k-го сигнала в области частот от $(k-1)\omega_0$ до $(k+1)\omega_0$ и быстрое убывание энергии вне этого диапазона и, во-вторых, существование эффективных методов реализации алгоритмов их модуляции и демодуляции.

Каждая из пар сигналов (5.1) образует двумерную систему координат. Независимая модуляция каждой из ортогональных составляющих передаваемыми информационными сигналами реализует соответствующее КАМ-созвездие (КАМ – *квадратурная амплитудная модуляция*). На передающей стороне n пар несущих ($0 < n < N - 1$) одновременно и независимо модулируются передаваемыми на i-м тактовом интервале информационными сигналами. Модулированные сигналы суммируются, порождая групповой сигнал $S_{гр}$, который на i-м тактовом интервале описывается выражением

$$S_{гр}(t - iT) = \sum_{k=0}^{N-1} A_{ki} \cos k\omega_0 (t - iT) + B_{ki} \sin k\omega_0 (t - iT). \quad (5.2)$$

Значения амплитуд A_{kj} и B_{ki} определяются передаваемой информацией.

Количество несущих задается адаптивно в зависимости от скорости и направления (вверх либо вниз) передачи и характеристик канала. Демодуляция группового сигнала (5.2) выполняется вычислением коэффициентов корреляции принимаемого колебания с опорными сигналами (5.1). При цифровой реализации ADSL-модемов операции модуляции (5.2) и демодуляции выполняются методами быстрого дискретного преобразования Фурье (ДПФ).

Существует два исполнения модема: станционное и абонентское. Структурная схема алгоритмов преобразования сигналов в передатчике станционного модема ADSL (ATU-C) изображена на рисунке 5.12.

Функционально передатчик ATU-C и передатчик абонентского модема ATU-R реализуются по одним и тем же алгоритмам. Существующие между ними незначительные различия обусловлены разными скоростями передачи "вниз" и "вверх", различным числом используемых несущих сигналов, а также режимом работы "ведущий" – "ведомый".

Рассмотрим назначение блоков передатчика, а также содержание выполняемых алгоритмов. Так как принятый вид многочастотной модуляции OFDM вносит значительные задержки в передачу сигналов (определяются длительностью тактового интервала и нормируются), то Рекомендацией G.992.1 предусмотрены специальные меры по ограничению задержки сигналов, вносимой ADSL-системой.

С этой целью передаваемые данные разбиваются в передатчике на два потока: один обрабатывается в так называемом **быстром буфере**, а второй – в **перестановочном**.

Соответствующие биты данных, обрабатываемые в быстром или перестановочном буфере, также называются быстрыми или перестановочными. Для выполнения требований по задержке быстрые и перестановочные данные обрабатываются по-разному: для быстрых данных длина блока кодирования меньше, отсутствует операция сверточной перестановки данных. Нормативная задержка быстрых данных составляет до 2 мс, а перестановочных — до 20 миллисекунд.

Блок мультиплексирования объединяет до четырех симплексных каналов (AS0…AS3) с суммарной скоростью передачи до 6,144 Мбит/с и до трех дуплексных каналов (LS0…LS2) с суммарной скоростью до 640 кбит/с, синхронизированных тактовой частотой 4 кГц, а также сигналами управления, администрирования и эксплуатации в два отдельных потока данных: *быстрый* и *перестановочный*. Каждый из потоков подвергается независимому кодированию циклическим кодом с образующим полиномом седьмой степени.

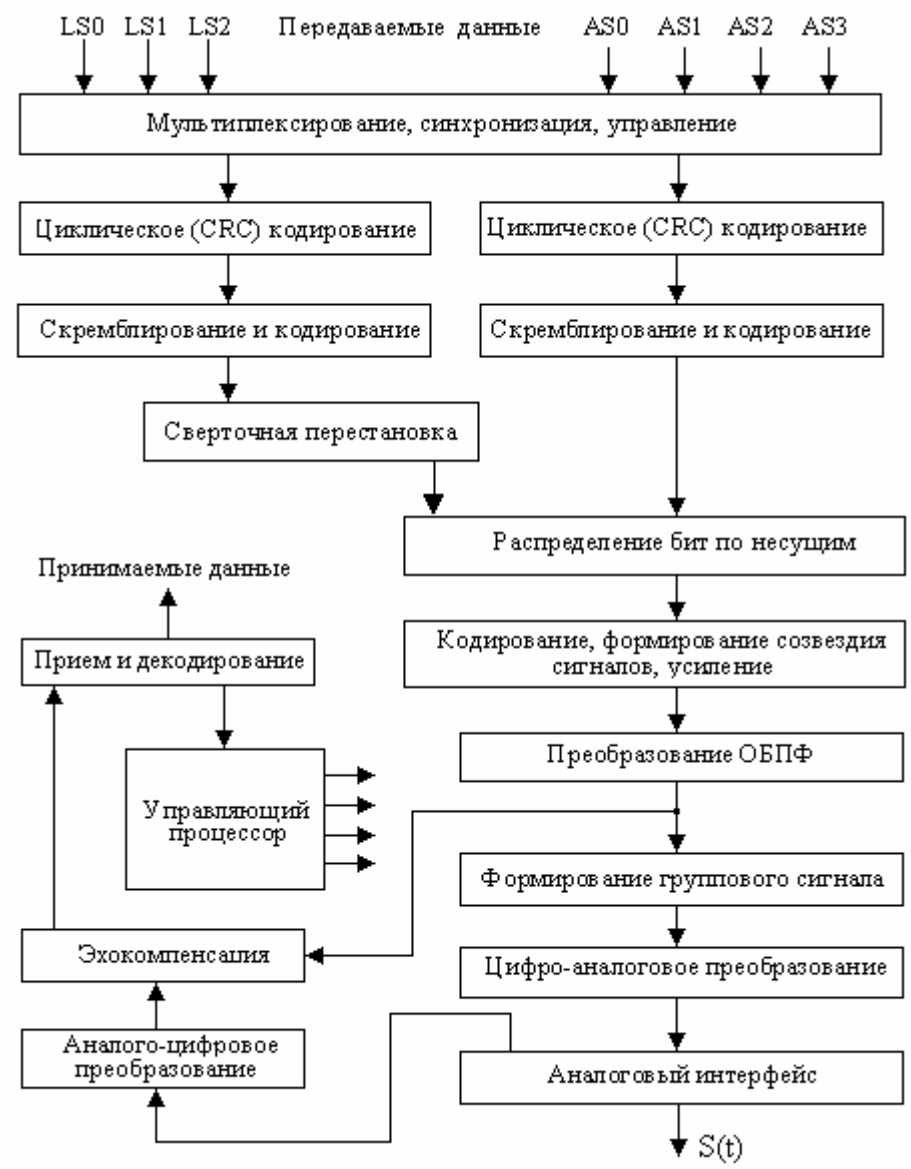


Рисунок 5.12 – Структурная схема передающей части ADSL-модема

Остаток от деления передается на противоположную сторону и служит для обнаружения ошибок при декодировании принятого сигнала. Циклическое кодирование является внутренней функцией ADSL-модема, которое введено для оценки качества образуемого канала передачи (оценки вероятности ошибки при передаче информации).

Затем быстрые и перестановочные кодированные данные независимо скремблируются с целью придания передаваемым сигналам свойств статистической независимости. Далее быстрые и перестановочные данные кодируются кодом Рида-Соломона, одним из наиболее эффективных линейных кодов, корректирующих ошибки. Кодированию подвергаются не биты, а байты.

Используемые параметры кода позволяют исправлять до 64 одиночных ошибок в блоке максимального размера 119 000 битов (минимальный блок составляет 530 битов).

После этого, с целью декорреляции порождаемых при передаче ошибок, данные только перестановочного потока подвергаются специальной операции сверточной перестановки. По определенному закону на передаче изменяется порядок следования данных, а на приеме восстанавливается изначальный порядок следования. В случае появления пакета ошибок в принятой последовательности, в результате обратной сверточной перестановки ошибки распределяются равномерно по последовательности и характеризуются как одиночные. Эта процедура повышает эффективность коррекции ошибок линейным кодом.

Результирующие последовательности данных быстрого и перестановочного буферов объединяются в кадр, передаваемый в течение длительности единичного элемента группового сигнала ADSL. Общее число бит в кадре, распределение их между быстрым и перестановочным буферами изменяются в соответствии с условиями передачи. Двоичные данные передаваемого кадра распределяются по несущим (каналам) передатчика, определяя значения амплитуд A_{ki} и B_{kj} в соответствии с (5.2). При этом адаптивно задается не только вид созвездия КАМ (число битов по каждой координате), но и коэффициент усиления в каждом канале.

Значение коэффициента усиления и вид созвездия в приемнике определяются в процессе тестирования канала, осуществляемого на этапе инициализации модема. При инициализации передатчик станционного модема передает по всем каналам немодулированные несущие. Приемник удаленной станции, принимая эти сигналы, измеряет их уровень, с помощью специальных алгоритмов вычисляет коэффициент усиления и вид сигнального созвездия для каждого канала и отправляет эту информацию на передающую сторону. В результате число битов передаваемой информации распределяется так, что скорость передачи данных по каналу связи максимизируется. Если какой-либо канал не может быть использован для передачи данных, он выключается. Энергия и количество передаваемых битов для различных каналов перераспределяются в зависимости от отношения сигнал/шум в рабочей части полосы частот.

Максимальное число несущих, которое может быть использовано для передачи "вниз", равно 255, "вверх" — 31. Практически число несущих при фильтровом способе разделения сигналов встречных направлений передачи для модема ATU-C достигает порядка 200. Несущие с номерами 64 и 16 (276 и 69 кГц) рекомендованы в качестве пилот-сигналов для передачи синхронизирующей информации соответственно "вниз" и "вверх". Для повышения помехозащищенности ADSL-модема рекомендуется использовать решетчатый код с шестнадцатью состояниями.

В результате на выходе кодирующего устройства формируется дискретный комплексный вектор размерностью 256, который преобразуется процессором обратного ДПФ (ОДПФ) в отрезок дискретного группового сигнала, содержащий 512 отсчетов. В блоке формирования группового сигнала полученная последовательность отсчетов дополняется первыми 32-мя отсчетами, соответствующими периодическому продолжению группового колебания на защитный интервал. Дополненный групповой сигнал последовательно считывается из буфера с тактовой частотой 2,208 МГц и преобразуется в аналоговую форму посредством ЦАП. Аналоговый интерфейс обеспечивает фильтрацию выходного колебания и согласование с линией связи. В соответствии с Рекомендацией ITU-T разделение сигналов встречных направлений передачи может осуществляться либо с помощью фильтров, либо эхокомпенсатором. Структурная схема приемной части модема изображена на рисунке 5.13.

На вход приемника поступает сумма аналогового сигнала S(t) передатчика удаленной станции и эхо-сигнала $S_э(t)$ собственного передатчика. Эта смесь предварительно усиливается, фильтруется и преобразуются в цифровую форму в блоке АЦП. Фильтрация призвана ограничить полосу частот принимаемого группового сигнала ADSL диапазоном 26...1104 кГц с целью подавления помех от сигналов аналогового телефонного аппарата и высокочастотных помех вне рабочего диапазона частот. Коэффициент усиления этой цепи близок к единице.

Аналого-цифровой преобразователь имеет не менее 12-ти разрядов, что определяется необходимым динамическим диапазоном преобразования, обусловленным требованиями эхокомпенсации. Эхокомпенсатор (ЭК) представляет собой адаптивный фильтр (см. рисунок 2.9), включаемый между выходом передатчика и входом приемника параллельно дифсистеме, разделяющей входные и выходные системы приемной части. Коэффициенты фильтра адаптируются таким образом, чтобы его частотная характеристика повторяла передаточную функцию цепи, по которой сигнал передатчика поступает на вход своего приемника. В результате на входы вычитающего устройства эхокомпенсатора поступают близкие по форме сигналы с выхода фильтра и дифсистемы, которые вычитаются. Одно из требований, выдвигаемых системой ЭК, заключается в равенстве частот дискретизации сигналов передатчика и приемника каждого из модемов до момента компенсации эхосигнала. Условием же оптимального приема является когерентность принимаемых и опорных сигналов. На вход корректора поступает дискретизированный сигнал $S(q\tau)$, $q = 0, 1, 2, \ldots$, представляющий собой групповой сигнал ADSL, прошедший через канал связи.

Реализация корректора в частотной области предполагает выполнение операций разбиения входной дискретной последовательности на секции длительности L отсчетов (блок секционирования), вычисление спектра полученных отрезков сигнала, умножение его на передаточную функцию корректора канала, выполнение обратного ДПФ и "сшивание" полученных отрезков

сигнала. Для первоначальной настройки корректора передается настроечный (известный на приеме) сигнал. Настроечная последовательность используется для адаптации параметров корректора канала.

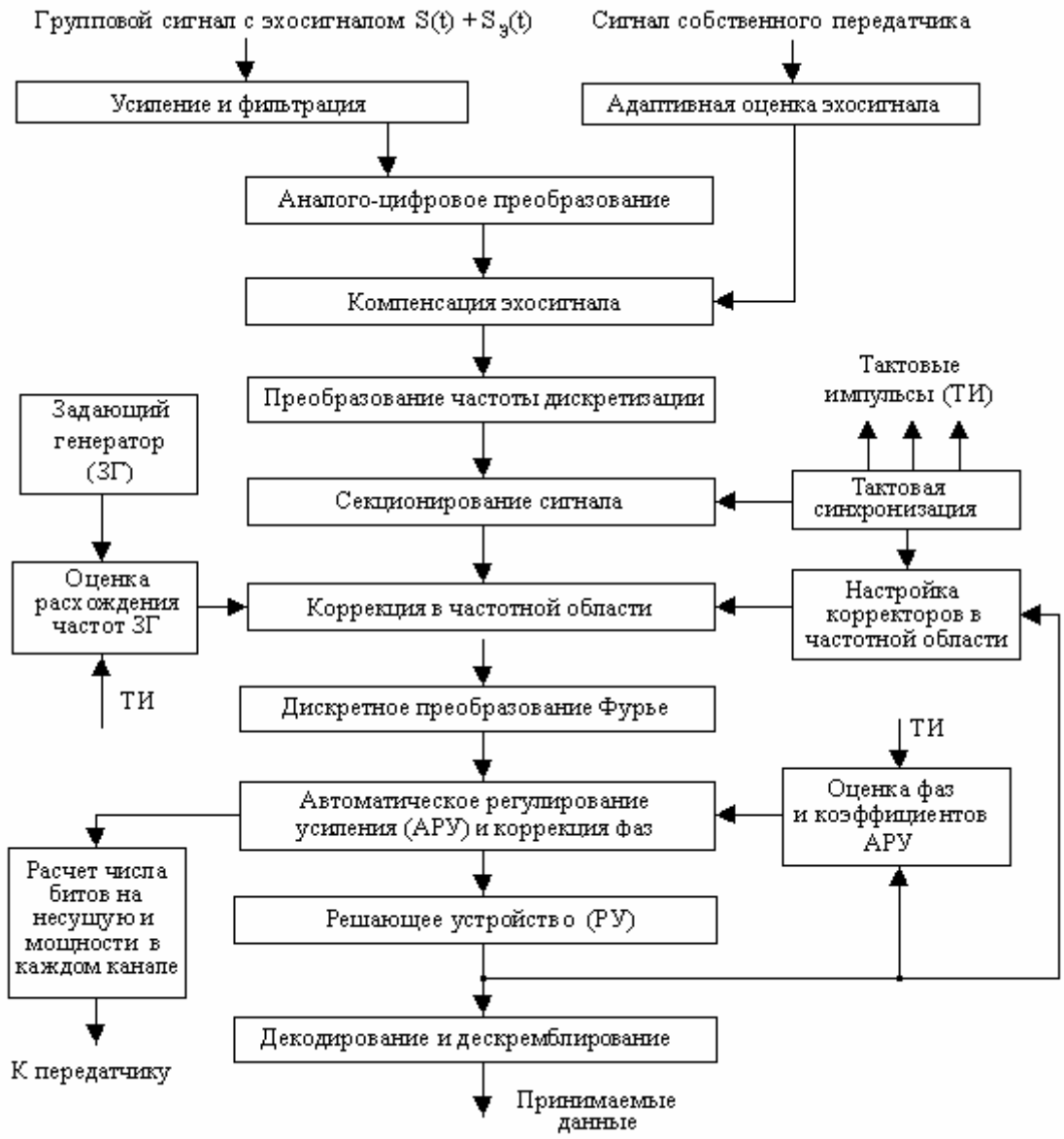


Рисунок 5.13 – Структурная схема приемной части ADSL-модема

Методы коррекции линейных искажении АЧХ и ФЧХ канала связи в ADSL не гарантируют идеальность частотных характеристик в пределах полосы частот каждого индивидуального канала системы передачи. Это приводит к тому, что демодулированные сигналы отличаются от сигналов, сформированных передатчиком как по амплитуде, так и по фазе. Для устранения этой неопределенности, вносимой каналом связи, с целью реализации оптимального

приема в каждом канале ADSL по специальным алгоритмам оцениваются отклонения коэффициента усиления и фазы, а затем они компенсируются. Поэтому демодулятор ADSL наряду с корректором частотных характеристик канала связи содержит также компенсатор линейных искажений индивидуальных сигналов, включаемый на выходе блока ДПФ демодулятора. Практически алгоритм компенсации реализуется в виде двух независимых цепей автоматического регулирования: отдельно коэффициента усиления (АРУ) и отдельно фазы. Каждому из принимаемых векторов сигнала решающее устройство ставит в соответствие двоичный кодовый символ. Кодовые символы всех каналов объединяются в общую последовательность, которая подвергается операциям декодирования, дескремблирования и разделения на стандартизованные потоки. Эти операции являются обратными по отношению к тем, что применялись в передатчике ADSL.

Как уже отмечалось, система передачи многочастотных ортогональных сигналов способна распределять число бит передаваемой информации таким образом, что скорость передачи информации по каналу связи максимизируется. На этапе инициализации системы приемник оценивает отношения сигнал/помеха в каждом канале модема с помощью специальных алгоритмов. Затем на основании этих данных рассчитывает оптимальный спектр передаваемого группового сигнала и оптимальное распределение передаваемой по каналам информации. Результаты этих расчетов посылаются на передатчик.

Схема распределения частотного диапазона линии связи при использовании ADSL-модема показана на рисунке 5.14.

Рисунок 5.14 – Использование частотного диапазона линии связи в ADSL

Рекомендации ITU-T регламентируют параметры передаваемых системами передачи сигналов и алгоритмы их формирования, не ограничивая возможности разработчиков в области применения алгоритмов обработки сигналов на приеме.

5.2.4. Модемные протоколы передачи файлов

Наиболее часто применяемой функцией сетевого программного обеспечения является передачи файлов. Она осуществляется с помощью специальных протоколов передачи файлов. Выбор и использование протокола передачи файлов может производиться пользователем в явном виде, как это делается в терминальных программах, так и в неявном, например, в игровых программах, поддерживающих модемную связь. Основными задачами протоколов передачи файлов являются:

- обеспечение безошибочной передачи данных;
- управление потоком передаваемых данных;
- передача вспомогательной информации;
- защита соединения.

Задача безошибочной передачи данных по каналам связи с помехами и по сегодняшний день остается одной из основных в компьютерных сетях. Для защиты информации передаваемые данные разбиваются на блоки (кадры) определенной длины, и в каждый из них включается проверочная комбинация для обнаружения ошибок. Эта комбинация формируется по определенному правилу на основе передаваемых информационных битов блока. На приемной стороне производится повторное вычисление проверочной комбинации по тому же правилу и сравнение ее с принятой. При совпадении проверочных комбинаций принимающая сторона посылает подтверждение правильного приема блока, а при несовпадении – запрос на повторную передачу данного блока.

Перед непосредственной передачей файла необходимо установить соединение на уровне канала данных (уровень 2 модели ВОС), передать информацию о имени файла, его размере, дате последней его модификации и т.п., а после передачи - произвести разъединение канала данных. Все это осуществляется при помощи вспомогательной служебной информации, передаваемой по каналу связи.

В последние годы в функции протоколов передачи файлов включают защиту соединения, например проверку пароля. Среди протоколов, рассчитанных на отсутствие аппаратной защиты от ошибок можно выделить широко распространенные протоколы Xmodem, Xmodem-CRC, Xmodem- 1K, Ymodem, Kermit и ряд других.

Если же применяются модемы с аппаратной коррекцией ошибок (поддерживающие протоколы типа MNP или V.42), то предпочтительней использовать протоколы передачи файлов типа Ymodem–g и Zmodem. В этом случае исключается потеря времени на повторный запрос данных, передаваемых с ошибками. Протоколы типа Zmodem допускают оба варианта приме-

нения.

Известны специализированные протоколы, предназначенные для определенных служб и сетей, такие как SEALink, Telnet, Compuserve Quik B. Практически все они являются модификациями протокола Xmodem. Рассмотрим подробнее наиболее распространенные протоколы передачи файлов.

Протокол **Xmodem**, благодаря широкому использованию в справочных службах и введению в недорогие связные программы для PC, стал фактически стандартом для связи между персональными компьютерами.

Компьютер-источник начинает передачу файла только после приема от удаленного компьютера знака **NAK** (*Negative AcKnowledge*), представляющего собой последовательность 0010101 в кодировке ASCII. Принимающая станция передает эту последовательность несколько раз, пока не начинается передача собственно файла. Однако если передано 9 знаков NAK, а передача файла не началась, процесс прерывается.

После приема знака NAK источник передает знак начала блока **SOH** (*Start Of Header*), код 01h, два номера блока (сам номер и его двоичное дополнение до единицы), блок данных из 128 байтов и контрольную сумму блока CS (*Check Sum*). Блоки нумеруются по модулю 256. Контрольная сумма в 1 байт представляет собой остаток от деления на 255 суммы значения кодов ASCII знаков, входящих в блок данных. Формат передаваемого кадра протокола Xmodem показан на рисунке 5.15.

Принимающий компьютер тоже вычисляет контрольную сумму и сравнивает ее с принятой. Если сравниваемые значения различны, либо прошло 10 секунд, а прием блока не завершен, принимающий компьютер посылает передатчику знак NAK, означающий запрос на повторную передачу последнего блока. Если блок был принят правильно, приемник передает подтверждение правильности приема знаком **ACK** (*Acknowledge,* кодовая комбинация 16h). В случае, если последующий блок не поступил в течении 10 с, то передача знака ACK повторяется несколько раз, пока блок не будет принят правильно. После девяти неудачных попыток передачи блока связь прерывается. Перерыв в передаче блока свыше 10 секунд считается перерывом связи.

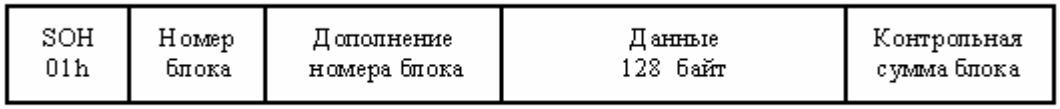


Рисунок 5.15 – Формат блока протокола XModem

В протоколе используется двукратная передача номера блока. Это исключает повторную выдачу получателю одного и того же блока из-за потери

подтверждающего сообщения. Компьютер-получатель контролирует уникальность номеров принимаемых блоков. Если блок ошибочно передан повторно, он сбрасывается. После успешной передачи всех блоков данных передающая станция посылает знак завершения передачи **EOT** (*End Of Transmission*), код которого 04h. Этот знак сообщает об окончании передачи файла.

Преимущество данного протокола над другими заключается в его доступности для разработчиков программных средств, простоте реализации на языках высокого уровня, малом объеме приемного буфера (256 байтов) и возможности передачи не только символьных (в кодах ASCII), но и исполняемых файлов (*.com, *.exe). Последнее возможно благодаря тому, что конец файла определяется подсчетом переданных байтов и использованием вместо знака файлового маркера (Ctrl-Z) специального знака завершения. Вероятность необнаруженной ошибки при передаче данных по этому протоколу несколько ниже, чем при обычной асинхронной проверке паритета, где она равняется 0,05.

К основным недостаткам протокола Xmodem можно отнести низкую производительность, обусловленную в основном использованием механизма автопереспроса с ожиданием, большую вероятность необнаруженных ошибок, необходимость задания имени файла при приеме и относительно большой объем передаваемой служебной информации. Последующие модификации протокола Xmodem были направлены на устранение этих и некоторых других недостатков.

Протокол **Xmodem-CRC** представляет собой модификацию протокола Xmodem, в котором обнаружение ошибок производится с использованием циклического кода. Длина проверочной последовательности составляет 16 битов. Благодаря этому гарантируется обнаружение практически всех двойных и одиночных ошибок, всех нечетных ошибок, всех пакетов ошибок длиной до 16 битов, а также всех 17-битовых ошибок с вероятностью 0,999969 и более длинных пакетов ошибок с вероятностью 0,99984.

В начале соединения вместо знака NAK приемная сторона по обратному каналу передает последовательность знаков “**c**” (63h). Если передатчик не поддерживает протокол Xmodem-CRC, он игнорирует эти знаки. Не получив ответа на передачу 3-х знаков ”c”, приемник переходит на работу к протоколу Xmodem и передает знак NAK.

Протокол **Xmodem-1К** представляет собой модификацию протокола Xmodem-CRC с блоками длиной 1024 байтов. Использование блоков длиной 1 Кбайт позволяет снизить задержки при передаче файлов в сетях с коммутацией пакетов, где длина пакета, как правило, равна 1023 байтам либо кратна ей. Кроме того, по сравнению с обычным протоколом Xmodem, уменьшена относительная доля заголовков в большем объеме передаваемой инфор-

мации.

Для передачи приемнику сообщения об увеличении длины передаваемого блока вместо знака SOH (01h) в начале блока ставится знак **STX** (02h). Номер блока, передаваемый во втором и в третьем байтах, увеличивается с передачей очередного блока на единицу независимо от его длины. Формат блока показан на рисунке 5.16. Передатчик не должен изменять длину блока (128 или 1024 байтов) до тех пор, пока не будет принят знак ACK для текущего блока. Игнорирование этого ограничения может привести к пропуску ошибок.

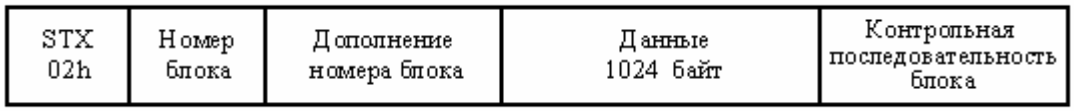


Рисунок 5.16 – Формат блока протокола Xmodem-1K

При использовании блоков по 1024 байтов возможно увеличение длины передаваемого файла до значения, кратного 1024. Блоки длиной 1024 байта могут применяться при групповой или одиночной передаче файлов. Для сохранения совместимости передаваемых данных с протоколом Xmodem-CRC необходимо использовать циклический код с CRC-16.

Затем появился более совершенный протокол **Ymodem,** представляющий собой протокол Xmodem-CRC, в котором реализована групповая передача кадров и решающая обратная связь с ожиданием подтверждения. Все программы, реализующие Ymodem**,** должны выполнять следующие функции:

- передавать информацию о имени и пути файла в блоке 0 в виде строки знаков ASCII, завершающейся знаком NUL (0h);
- использовать эту информацию на приемной стороне в качестве имени и пути принятого файла, если иная реализация не оговорена специально;
- выполнять проверку CRC-16 при приеме знаков “c”, в противном случае использовать 8-битовую контрольную сумму;
- принимать любую комбинацию из 128 или 1024-байтных блоков внутри каждого принимаемого файла;
- обеспечивать возможность переключения длины блоков в конце передачи файла и в случае частых повторных передач;
- передающая программа не должна изменять длину неподтвержденного блока;
- передавать в конце каждого файла знаки EOF до 10 раз, пока не будет принят знак ACK;
- обозначать конец сеанса связи нулевым именем пути.

Коммуникационные программы, в которых не реализованы все перечисленные функции, несовместимы с протоколом Ymodem. Выполнение этих минимальных требований, однако, не гарантирует надежной передачи файлов в условиях сильных помех. Протокол Ymodem устраняет некоторые недостатки протокола Xmodem, в основном сохраняя его простоту.

Как и в случае передачи одного файла, приемник инициирует групповую передачу путем посылки знака ”c”. Передатчик открывает файл и передает номер 0. Для групповой передачи требуются только имена файлов. С целью обеспечения совместимости “снизу вверх” все неиспользуемые байты блока 0 должны иметь нулевое значение.

Имя файлов передается как строка кодов ASCII, завершаемая знаком NUL. Этот формат имени файла используется в функциях, ориентированных на операционные системы типа MS-DOS, и в функции *fopen* библиотеки Си. В имя файла не включены пробелы. Обычно передается только само имя без префикса, имя диска источника не передается. Если передатчик не поддерживает передачу знаков в обоих регистрах, имя передается в строчном регистре. Если в имя файла включен каталог, его название должно ограничиваться знаком “/”.

Обозначения длины файла и каждого последующего поля произвольны. Длина файла представляется в блоке как десятичная строка, задающая количество байтов в файле. В нее не должны входить знаки EOF или другие символы, используемые для заполнения последнего блока. Если передаваемый файл увеличивается во время передачи, то параметр ”длина файла” должен иметь значение, соответствующее максимально ожидаемому размеру или не передаваться вовсе.

Дата модификации файла является необязательным параметром, имя и длина файла могут передаваться без передачи даты модификации. Протокол Ymodem допускает возможность введения других полей заголовка без нарушения совместимости со своими прежними версиями. Оставшаяся часть блока устанавливается в 0. Это важно для сохранения совместимости ”снизу вверх”.

Если блок имени файла принят с ошибкой, необходим запрос на повторную передачу. По умолчанию приемник использует процедуру исправления ошибок с решающей обратной связью (РОС) и циклическим кодом CRC-16. Прием блока с именем файла, успешно открытого для записи, подтверждается знаком ACK. Если файл не может быть открыт для записи, то приемник прерывает передачу с помощью знака отмены CAN (CANcel).

Далее приемник инициирует передачу содержимого файлов в соответствии с протоколом Xmodem-CRC. После того как содержимое файла получено, приемник запрашивает имя следующего файла. Передача источником нулевого имени файла может означать, что групповая передача завершена, и

что запрошенные у передатчика файлы не могут быть открыты для чтения.

В настоящее время разработаны методы, обеспечивающие передачу данных с очень высокими скоростями и малой вероятностью ошибок. Эти методы реализованы в высокоскоростных модемах и некоторых коммуникационных программах. Применение таких методов позволяет достичь скорости передачи, близкой к теоретически возможной.

Вариант **g** протокола Ymodem обеспечивает высокую эффективность передачи данных за счет использования РОС без ожидания подтверждения. При использовании протокола **Ymodem-g** приемник инициирует групповую передачу посылкой знака “**g**” вместо ”c”. Передатчик, распознавший этот символ, прекращает ожидание обычных подтверждений по каждому переданному блоку и передает последовательные блоки на полной скорости с использованием метода управления потоком XON/XOFF. В соответствии с этим протоколом при возникновении перегрузки передатчик по линии TxD выдает команду XOFF (DC3h) оконечному оборудованию о необходимости остановки информационного потока. Восстановление передачи информационного потока ООД-источником инициируется посылкой ему команды XON (DC1h) по линии RxD.

Прежде чем передавать файл, передатчик ожидает поступления последовательности знаков “g”, а в конце передачи каждого файла – подтверждающего знака ACK. Такой способ синхронизации позволяет приемнику открывать и закрывать файлы в нужное время.

При обнаружении ошибки, в случае использования протокола Ymodem-g, приемник прерывает передачу и отвечает последовательностью служебных символов CAN.

Введение протокола Ymodem-g позволило значительно повысить скорость передачи данных в каналах, защищенных от ошибок, т. е. при использовании модемов со встроенными протоколами защиты. Это достигнуто за счет отказа от переспроса принятых с ошибками блоков, а при обнаружении ошибок передача файла прерывается. Для повышения быстродействия в последующих модификациях протокола Ymodem применен так называемый “оконный” алгоритм, при котором информационные блоки передаются подряд, без ожидания подтверждения правильного приема некоторого числа блоков.

Наиболее совершенным и распространенным протоколом, используемым в большинстве связных программ, является протокол **Zmodem**. Он совместим с протоколами Xmodem и Ymodem, устраняет их недостатки и имеет ряд следующих преимуществ:

- применяется динамическая адаптация к качеству канала связи посредством изменения в широких пределах размера передаваемых блоков;
- введена возможность возобновления прерванной передачи файла с то-

го места, на котором произошел сбой;

- повышена достоверность передачи благодаря использованию 32-разрядной проверочной комбинации циклического кода CRC;
- имеется возможность отключения функции контроля ошибок передаваемых блоков при использовании модемов с аппаратной коррекцией ошибок.

Протокол позволяет программно инициировать передачу файлов или передавать команды и (или) модификаторы другим программам. Названия файлов достаточно ввести только один раз. Выбор файлов реализуется с помощью меню. При групповых передачах возможно задание файлов одной маской. Организация передачи осуществляется путем введения минимального количества команд с клавиатуры. При передаче файлов посылается кадр ZRQINIT, который инициирует автоматический прием файлов. Протокол Zmodem может эмулировать режим протокола Ymodem, если процесс на удаленном компьютере не поддерживает Zmodem. С момента начала сеанса связи протокол Zmodem защищает передаваемые данные циклической проверочной комбинацией, состоящей из 16 или 32 битов (CRC-16 или CRC-32). Использование 32-битной проверочной комбинации позволяет уменьшить вероятность необнаруженных ошибок не менее чем на 5 порядков.

5.2.5. Протоколы подключения к сети Интернет через телефонные каналы

Для подключения к сети Интернет посредством модема через телефонный канал и создания соединения с использованием протокола TCP/IP на канальном уровне применяются два протокола: **SLIP** (*Serial Line Internet Protocol*) и **PPP** (*Point to Point Protocol*). Логически протоколы SLIP и PPP находятся между модемным портом компьютера и стеком TCP/IP (см. рисунок 4.3).

Установление соединения по SLIP – это наиболее простой способ подключения сетевого компьютера к сети Интернет. SLIP обеспечивает инкапсуляцию IP-пакетов в кадры, пригодные для передачи последовательным способом по каналу связи. SLIP не предоставляет возможности адресовать данные, обозначать типы кадров, определять и корректировать ошибки, сжимать информацию. Отсутствие этих возможностей делает протокол чрезвычайно простым в реализации, и поэтому он еще до сих пор сохранил свою популярность, однако он обладает рядом недостатков и не является официальным стандартом для Интернета.

Протокол **CSLIP –** это **SLIP со сжатием.** Для ограничения кадра в SLIP применяются следующие два символа: END (код C0h) и ESC (код DBh).

Символ END передается компьютером в конце каждого пакета данных. Знак ESP используется для обозначения символов пакета данных, совпадающих с кодами END и ESC, т.е. введение символа ESC обеспечивает прозрачность передачи.

Временная диаграмма формирования кадров SLIP из IP-пакетов изображена на рисунке 5.17. Вставка END в начале кадра приводит к игнорированию его, что позволяет исключить влияние помех в линии связи. Однако на этом заканчиваются способности SLIP определять ошибки в линии связи. Протокол возлагает защиту от ошибок на вышележащие протоколы.

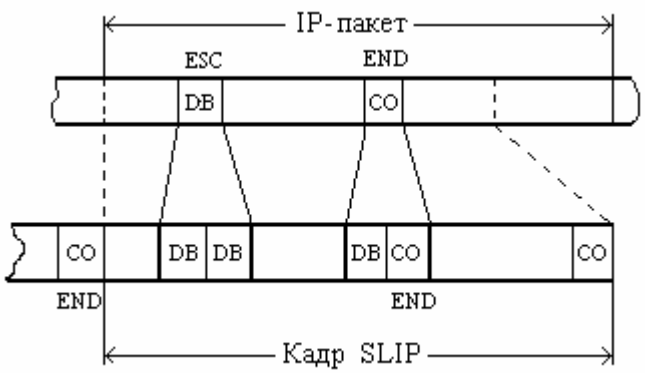


Рисунок 5.17 – Формат кадра протокола SLIP

Протокол CSLIP позволяет сжимать только заголовки TCP/IP-пакетов для сегментов данных TCP. Сами данные пакетов остаются неизменными. Суть сжатия состоит в том, что на протяжении TCP-соединения около половины заголовка остается неизменной. Поэтому передатчик выдает в канал только изменения в заголовках.

Протокол PPP (*Poin- to-Point Protocol*). В нем устранены недостатки, присущие протоколу SLIP. Одной из целей протокола возможность передачи по одному каналу связи данных для нескольких сетевых протоколов. В протоколе PPP реализованы следующие процедуры:

- инкапсуляция данных, позволяющая на одном и том же канале использовать различные сетевые протоколы;
- управление соединением LCP (*Link Control Protocol*), реализуемое в программном обеспечении для установки, конфигурации и тестирования соединения;
- управление сетью с использованием протоколов NCP (*Network Control Protocol*), позволяющее PPP-соединению применять протоколы различных сетевых уровней. В PPP-протоколе реализуется формат протокола канального уровня по рекомендации MKKTT X.25, который изображен на рисунке 5.18.

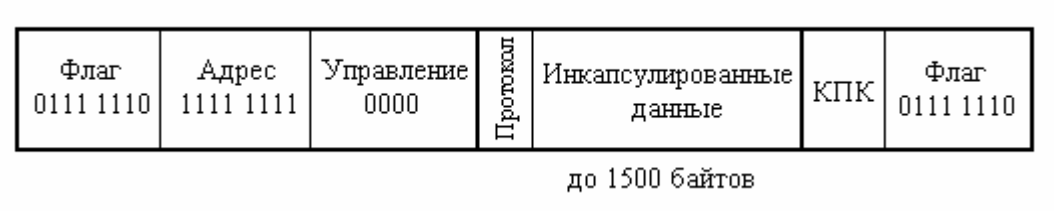


Рисунок 5.18 – Формат кадра протокола PPP

Для обеспечения "прозрачности" передачи при появлении в поле данных флаговой комбинации 7E (01111110) она заменяется двумя символами: ESC=7D (01111101) и флаговой комбинацией, у которой инвертирован 6-й бит, т.е. комбинацией 5E (01011110). Если же в поле данных появится ESC-символ, то после него вставляется ESC-комбинация, в которой проинвертирован 6-й бит, т.е. 01011101 (5Dh). Таким образом, вместо комбинации 7E передается последовательность 7D 5E, а вместо 7D – 7D 5D.

С целью обеспечения передачи сообщений различных протоколов в начале поля данных кадра выделено два байта-указателя типа используемого протокола, задающего до конца кадра тип данных. Так, например, указатель C021H индицирует, что кадр содержит данные протокола управления соединением *LCP*. Это означает, что информация, полученная PPP из такого кадра, будет использоваться им самим, а не передаваться куда-либо дальше по сети. Значение указателя 8021H свидетельствует о присутствии данных для протокола управления сетью *NCP*. Указатель 0021H определяет, что инкапсулированные данные предназначены для TCP/IP сети.

В связи с тем, что младшая часть указателя для всех типов кадров одинакова, в процессе конфигурации соединения может быть передан только один байт. Можно сократить заголовок еще больше, если при конфигурации исключить адресное и управляющее поля.

Функции протокола PPP по управлению соединением. До того, как два сетевых объекта начнут обмениваться данными по PPP, им необходимо правильно настроить канал и проверить его состояние. Для этого используется протокол управления соединением **LCP** (*Link Control Protocol*). Затем для установки ряда параметров на сетевом уровне в работу включается протокол **NCP** (*Network Control Protocol*). Процесс установки соединения состоит из ряда фаз.

Фаза неактивности. PPP находится в состоянии ожидания сигнала готовности линии передавать или принимать данные. Например, сигнал на выходе цепи модема (109) DC – *наличие несущей*, свидетельствует, что модем "дозвонился" и соединился с удаленной станцией. PPP приступает к фазе установления соединения.

Фаза установления соединения. Получив сигнал готовности от фазового уровня PPP, используя протокол LCP, переходит в фазу установки соединения. В составе PPP различают три типа пакетов: *конфигурации соединения, окончание сеанса и управления соединением.* Пакеты конфигурации предназначены для установления и настройки канала передачи. Пакеты управления соединением используются LCP для обслуживания и отладки установленного соединения PPP. Пакеты окончания сеанса служат для завершения сеанса связи. Пакеты конфигурации бывают следующих типов:

- "конфигурация–запрос";
- "конфигурация–подтверждение";
- "конфигурация–неподтверждение";
- "конфигурация–отказ".

Для открытия соединения PPP должен послать пакет "конфигурация–запрос". Это требование является обязательным для любой реализации протокола PPP. Поле данных этого пакета содержит список желательных вариантов настройки соединения. Компьютер с установленным модулем PPP, получивший пакет "конфигурация–запрос", обязан в ответ послать пакет "конфигурация–подтверждение", если список параметров устраивает его полностью. Поле данных подтвержденного пакета содержит точную копию принятого.

Если PPP не в состоянии выполнить какой-либо пункт желательной настройки соединения, он должен ответить пакетом "конфигурация-неподтверждение". Поле данных этого пакета содержит список неподдерживаемых вариантов, а также может включать в список дополнительные варианты настройки соединения. Аналогичным образом оба участника соединения PPP продолжают обмениваться пакетами до тех пор, пока не согласуют конфигурацию. Модуль PPP, обслуживающий неизвестный ему вариант настройки, содержащийся в пакете-запросе, обязан ответить пакетом "конфигурация–отказ". "Отказавшие" варианты дальнейшему рассмотрению не подлежат.

Параметрами и функциями конфигурации, в частности, являются следующие.

1. ***Максимальная длина принимаемого блока*** **MRU** (*Maximum Receive Unit*). По умолчанию MRU=1500 байт. В принципе, может передаваться пакет любой длины, но не более установленного MRU.

2. ***Качество соединения***. Осуществление наблюдения за числом потерянных или ошибочно принятых пакетов.

3. Образование шлейфа для передачи самому себе пакета (Loop-back). Для задания этого режима используется специальная кодовая комбинация, называемая "магическое число".

4. Сжатие полей адреса и управления.

Фаза управления сетью. На протяжении этой фазы устанавливается ряд параметров сетевого уровня. Используя протокол NCP, модуль PPP может "на ходу" открыть новый или завершить обмен по старому сетевому протоколу.

Фаза прекращения соединения. Она служит для завершения соединения. Закрытие всех протоколов в PPP не должно приводить к разрыву соединения. Сетевое программное обеспечение должно само указать PPP на необходимость прерывания соединения.

5.3. Цифровые выделенные каналы связи глобальных сетей

5.3.1. Плезиохронная цифровая иерархия PDH

Первые цифровые сети были разработаны для обеспечения передачи телефонного трафика по высокоскоростным магистральным каналам. В связи с использованием цифровых технологий качество передачи речевых сообщений по телефонным каналам значительно возросло. Существенно снизились затраты на эксплуатацию цифровых телекоммуникационных систем.

В аппаратуре уплотнения с ИКМ речевые сообщения оцифровываются с частотой 8 кГц с использованием 8 битов на отсчет. В результате каждый абонентский канал формирует поток битов со скоростью 8×8000 = 64 кбит/с, который поступает на мультиплексор аппаратуры уплотнения. Цифровой канал со скоростью передачи 64 кбит/с назван **основным цифровым каналом**. В первичной цифровой ступени преобразования (ЦСП) многоканальной аппаратуры мультиплексор объединяет битовые потоки группы абонентских каналов в так называемый **кадр** (*frame*), выдаваемый побитно на выход аппаратуры уплотнения. Групповой поток битов получил название **первичный цифровой поток**. Скорость группового потока зависит от количества объединяемых каналов. В американских и японских системах уплотнения объединяется 24 канала и формируется результирующий **поток T1**, который кроме 24-х канальных интервалов содержит один бит синхронизации. Таким образом, результирующая скорость первичного потока T1 равняется ((24×8)+1)×8000=1544 кбит/с.

В европейской системе в первичной ЦСП в кадр объединяется 30 абонентских основных цифровых каналов и два служебных. В результате скорость первичного цифрового **потока E1** составляет 32×8×8000=2048 кбит/с.

Для обеспечения потребностей в более высоких скоростях передачи образована ступенчатая иерархия скоростей. Чем выше ступень иерархии, тем мощнее цифровой поток, т.е. тем выше его скорость. К системам переда-

чи, стоящим в самом низу иерархической лестницы, относится цифровой поток **T1** североамериканской системы уплотнения и **E1** – европейской. Данные по скоростям передачи для различных систем построения аппаратуры уплотнения приведены в таблице 5.1.

Таблица 5.1 – Скорости передачи трех системы цифровой иерархии

Уровень цифровой иерархии	Скорости передач, соответствующие различным системам цифровой иерархии		
	Американская	Японская	Европейская
0	64	64	64
1	1544	1544	2048
2	6312	6312	8448
3	44736	32064	34368
4	---	97728	139264

Стандарты построения и группирования каналов в европейских странах несколько отличались от стандартов, принятых в США и в Японии. Иерархия скоростей в цифровых каналах связи, принятая в Европе и в Америке, показана на рисунке 5.19. Как видно из этого рисунка, формирование потоков на последующих цифровых ступенях передачи также отличается как по скорости, так и по количеству мультиплексируемых каналов.

Изображенная иерархия скоростей при объединении цифровых потоков получила название **плезиохронная цифровая иерархия PDH** (*Plesiochronous Digital Hierarchy*). Скорости цифровых потоков одной и той же ступени иерархии, но образуемых ЦСП, расположенными на различных станциях сети, могут несколько отличаться друг от друга в пределах допустимой нестабильности частот задающих генераторов. Именно поэтому рассматриваемая иерархия ЦСП называется плезиохронной (почти синхронной). Наличие нестабильности задающих генераторов требует принятия специальных мер при объединении исходных потоков в поток более высокой ступени иерархии, что заметно усложняет эксплуатацию первичной сети связи в целом и снижает ее качественные показатели

В связи с тем, что скорости от разных каналов не всегда совпадают, для синхронизации потоков добавляют нужное число битов в каналы с меньшей скоростью, осуществляя тем самым *выравнивание* скоростей. Такой способ выравнивания получил название *плезиохронного* (почти синхронного) **PDH**. В плезиохронных цифровых системах передачи используется принцип временного разделения каналов, поэтому правильное восстановление исходных сигналов на приеме возможно только при синхронной и синфазной работе генераторного оборудования на передающей и приемной станциях.

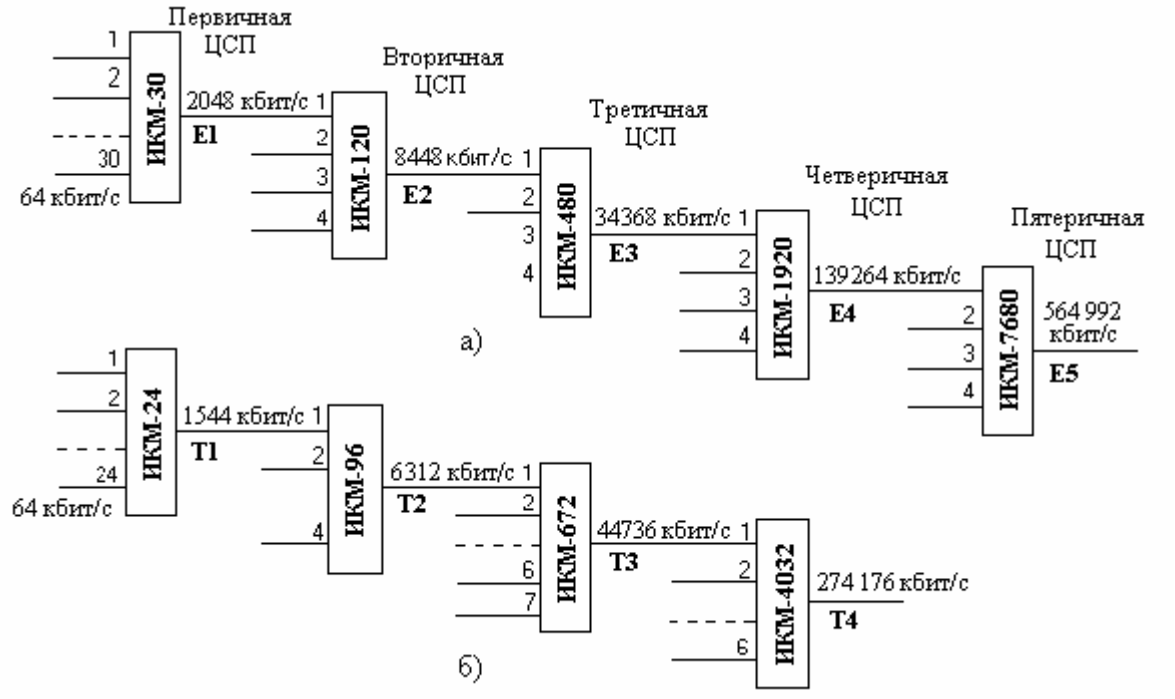


Рисунок 5.19 –Иерархия скоростей передачи в цифровой сети:
а) европейская и б) американская системы

Для нормальной работы плезиохронных систем передачи должны быть обеспечены следующие виды синхронизации:

- тактовая синхронизация – поддерживает равенство скоростей обработки цифровых сигналов в линейных и станционных регенераторах, кодеках и других устройствах ЦСП, осуществляющих обработку сигнала с тактовой частотой F_T;
- цикловая синхронизация – обеспечивает разделение и декодирование кодовых групп цифрового сигнала и распределение декодированных отсчетов по соответствующим каналам в приемной части аппаратуры;
- сверхцикловая синхронизация – выполняет на приеме распределение сигналов управления и взаимодействия (СУВ) по соответствующим телефонным каналам. СУВ представляют собой набор сигналов, управляющих работой АТС (набор номера, ответ, отбой, разъединение и пр.)

Нарушение хотя бы одного из видов синхронизации приводит к потере связи по всем каналам цифровой системы передачи.

Технология PDH является достаточно эффективной для цифровой телефонии, однако для передачи данных она оказалась недостаточно гибкой. Она не позволяет извлекать на промежуточных узлах сети потоки данных со скоростью 64 кбит/с или 2 Мбит/с, входящие в групповой поток со скоростью 140 Мбит/с, без полного демультиплексирования и удаления выравни-

вающих битов. Учитывая недостатки PDH, международный комитет ITU-T принял решение разработать для волоконно-оптических сетей единую синхронную цифровую иерархию SDH (*Synchronous Digital Hierarchie*). Разработка велась на основе существующей в США аналогичной технологии передачи цифровых сигналов по оптическим линиям, получившая название **SONET** (*Synchronous Optical Network*).

5.3.2. Синхронная цифровая иерархия SDH

Синхронная цифровая иерархия SDH (*Synchronous Digital Hierarchy)* – это принцип построения цифровых систем передачи, использующих мультиплексирование цифровых потоков, но со значительно большей базовой скоростью передачи, чем в PDH, и ***синхронизацией*** всего каналообразующего и передающего оборудования от общего задающего генератора. Технология синхронной цифровой иерархии SDH разработана для создания надежных транспортных сетей, позволяющих гибко формировать цифровые каналы широкого диапазона скоростей — от единиц мегабит до десятков гигабит в секунду. Основная область применения технологии SDH — **первичные сети операторов связи**. В настоящее время на магистральной сети Украины используется оборудование синхронной цифровой иерархии STM-4 – STM-16. Иногда такие сети строят и крупные предприятия, и организации, имеющие разветвленную структуру подразделений и филиалов, покрывающих большую территорию, например, в сетях предприятий энергетического комплекса или железнодорожных компаний.

Каналы SDH относятся к классу полупостоянных. Формирование канала происходит по инициативе оператора сети SDH. Пользователи же лишены такой возможности, в связи с чем каналы SDH обычно применяются для передачи достаточно устойчивых во времени потоков. Из-за полупостоянного характера соединений в технологии SDH чаще используется термин **кросс-коннект** (cross-connect), а не коммутация.

Сети SDH относятся к классу сетей с коммутацией каналов, использующих синхронное мультиплексирование с разделением времени TDM (*Time Division Multiplexing*), при котором информация от отдельных абонентов адресуется относительным временным положением внутри составного кадра, а не явным адресом, как это происходит в сетях с коммутацией пакетов. Каналы SDH обычно применяют для объединения большого количество периферийных менее скоростных каналов, работающих по технологии плезиохронной цифровой иерархии. Сети SDH обладают многими достоинствами, главными из которых можно назвать следующие.

1. *Гибкая иерархическая схема* мультиплексирования цифровых потоков разных скоростей. Это позволяет вводить в магистральный канал и выводить из него пользовательскую информацию любого поддерживаемого технологией уровня скорости, не демультиплексируя поток в целом. Схема мультиплексирования стандартизована на международном уровне, что обеспечивает совместимость оборудования разных производителей.

2. *Высокая устойчивость к отказам сети* за счет автоматической реакции оборудования на такие типичные отказы, как обрыв кабеля, отказ порта, выход из строя мультиплексора или отдельного адаптера связи, направления трафика по резервному пути или перехода на резервный модуль. Переключение на резервный путь происходит очень быстро (обычно в течение 50 мс).

3. *Мониторинг и управление сетью* на основе информации, встроенной в заголовки кадров.

4. *Высокое качество транспортного обслуживания* для трафика любого типа — речевого, видео и данных.

Техника мультиплексирования с временным разделением каналов, лежащая в основе SDH, обеспечивает трафику каждого абонента гарантированную пропускную способность, а также низкий и фиксированный уровень задержек. Характерным признаком SDH-иерархии является используемый способ мультиплексирования (рисунок 5.20. Основой (*базисом*) способа мультиплексирования является "синхронный транспортный модуль **STM-1**" (от агл. *Synchronous Transport Module*) со скоростью передачи 155,52 Мбит/с. Синхронный транспортный модуль STM-1 представляет собой кадр длительностью 125 мкс, состоящий из 2430 байтов. Отсюда вытекает, что скорость передачи кадра должна быть $(2430\times8) / (125 \cdot 10^{-6}) = 155{,}52$ Мбит/с. В случае битовых скоростей ниже и выше STM-1 – транспортной скорости 155,52 Мбит/с применяются различные способы мультиплексирования.

Для скоростей, меньших STM-1-скорости, пропускная способность STM-1 режима подразделяется иерархически в так называемые *контейнеры* различной величины. **Контейнер** представляет собой структуру одного из плезиохронных сигналов (например, поток со скоростью 2,048 Мбит/с). Имеется набор из различных типоразмеров контейнеров (таблица 5.2). В каждом контейнере находится управляющий заголовок POH (*Path Overhead)*, который позволяет транспортировать данные через узлы SDH-сети. Размеры контейнеров выбраны таким образом, что они подходят для передачи существенно различающихся региональных PDH-сигналов.

В противоположность к PDH в сети SDH возможен селективный доступ к отдельным сигналам без демультиплексирования общего транспортного потока. Это достигается посредством байт-ориентированного мультиплексирования и благодаря непосредственной адресации начала каждого из

контейнеров.

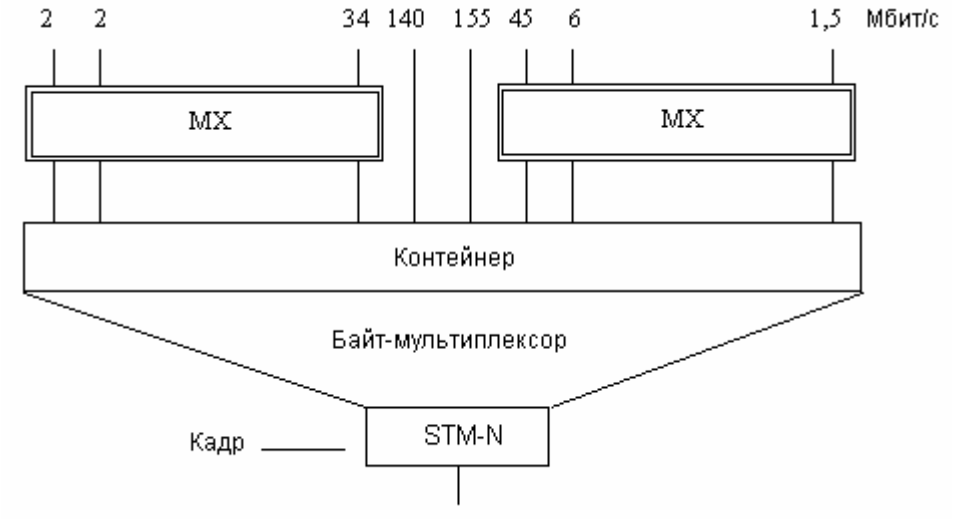


Рисунок 5.20 – Схема мультиплексирования в SDH

В контейнере имеются **указатели**, определяющие текущее положение контейнера в структуре более высокого уровня. Это позволяет мультиплексору находить положение пользовательских данных "на лету", без полного демультиплексирования, как это производится в PDH.

Таблица 5.2 – Параметры типоразмеров контейнеров STM

Обозначение контейнера	Скорость передачи, Мбит/с	Соответствующая скорость PDH, Мбит/с
C1.1	1,648	1,544
C1.2	2,224	2,048 и 1,544
C2	6,832	6,312
C3	48,384	44,734 и 34,368
C4	149,760	139,264

Кадр STM-1 обычно представляют в виде матрицы, состоящей из 270 столбцов и 9 строк. Элементами матрицы являются байты, передача которых осуществляется последовательно: строка за строкой. Первые 9 байтов каждой строки занимает заголовок строки, 260 байт отводится для пользовательских данных и один байт выделен под заголовок тракта. Последний позволяет контролировать соединение "из конца в конец". Служебная часть служит непосредственно для обозначения начала временного кадра, указания структуры потока и контроля качества передаваемого сообщения, а также содер-

жит управляющую информацию, обеспечивающую транспортировку STM по сети.

Для скоростей выше 155,52 Мбит/с *N* синхронных транспортных модулей мультиплексируются в новый транспортный модуль SDH-N без ограничения сверху. Так, например, при *N*=4 транспортная скорость достигает 622,08 Мбит/с, при *N*=16 она возрастает до 2488,32 Мбит/с. Во всех иерархических способах мультиплексирования, в том числе в SDH, гибкость битовой скорости и эффективность передачи ограничены, в том случае, если обслуживаются различные скорости. В частности, полезные сигналы со скоростью, превышающей пропускную способность одного контейнера, должны быть переданы в следующем контейнере. Так, например, поток со скоростью 34 Мбит/с должен быть размещен в контейнере 48 Мбит/с (С3). А это ведет к снижению эффективности использования канала.

Основным функциональным модулем сетей SDH является **синхронный мультиплексор** (SMUX). В нем имеется несколько портов для цифровых иерархий PDH и SDH (рисунок 5.21), например, порты PDH на 1,5; 2; 34 и 45 Мбит/с и порты SDH на 155 Мбит/с (STM-1) и на 622 Мбит/с (STM-4).

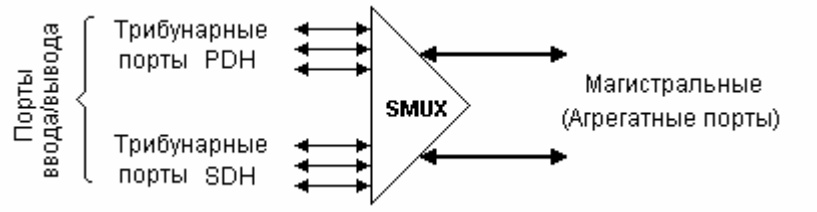


Рисунок 5.21 – Мультиплексор синхронной цифровой иерархии SDH

Порты мультиплексоров подразделяются на пользовательские порты ввода/вывода (т.н. **трибунарные** порты) и магистральные (**агрегатные** порты). Мультиплексоры SDH выполняют как функции собственно мультиплексора, так и функции устройств терминального доступа, позволяя подключать низкоскоростные каналы PDH иерархии непосредственно к своим входным портам. Они являются универсальными и гибкими устройствами, позволяющими кроме мультиплексирования выполнять задачи коммутации, концентрации и регенерации. Это представляется возможным в силу модульной конструкции SDH мультиплексора SMUX, при которой выполняемые функции определяются лишь возможностями системы управления и составом модулей, включенных в спецификацию мультиплексора.

Для построения SDH-сетей первых двух уровней SDH-иерархии со скоростью передачи 155 и 622 Мбит/с – наиболее широко используется кольцевая топология (рисунок 5.22).

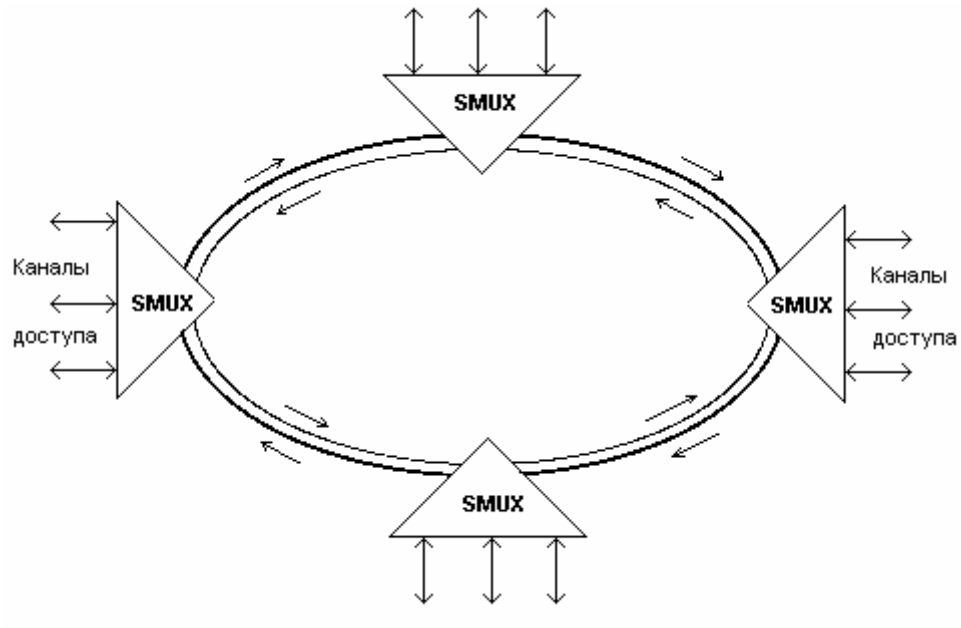


Рисунок 5.22 – Схема построения сети SDH первого уровня

Основное преимущество этой топологии – легкость организации резервирования типа 1+1 (один основной тракт и один резервный) благодаря наличию в синхронных мультиплексорах SMUX двух пар оптических каналов приема/передачи, дающих возможность формирования двойного кольца со встречными потоками.

5.3.3. Цифровые сети спектрального мультиплексирования WDM

Системы спектрального мультиплексирования **WDM** (*Wave Division Multiplexing*) основаны на способности оптического волокна одновременно передавать свет различных длин волн (цветов) без взаимной интерференции. Каждая длина волны представляет отдельный оптический канал в волоконно-оптической линии связи (ВОЛС). Существуют различные оптические методы для объединения нескольких каналов в одном волокне, а затем выделения их в нужных точках сети. Одна из схем мультиплексирования показана на рисунке 5.23. На входе системы передачи канальные оптические сигналы с помощью призмы объединяются в один групповой сигнал, передаваемый по ВОЛС. На выходе с помощью аналогичной призмы эти сигналы разделяются. Число каналов на входе и выходе может достигать 32, а в отдельных случаях и более.

В технологии WDM нет многих ограничений и технологических труд-

ностей, свойственных системам с временным разделением каналов TDM. Для повышения пропускной способности, вместо увеличения скорости передачи в едином составном канале, как это реализовано в технологии TDM, в технологии WDM увеличивают число каналов (длин волн), используемых в системах передачи. Теоретически возможна передача в любом диапазоне длин волн, однако практические ограничения оставляют для использования в системах WDM узкий диапазон в окрестности длины волны 1550 нм. Но даже этот диапазон предоставляет огромные возможности для передачи данных.

Рисунок 5.23 – Мультиплексирование сигналов с разделением каналов по длине волны

Рост пропускной способности при использовании технологии WDM осуществляется без дорогостоящей замены оптического кабеля. Применение технологии WDM позволяет сдавать в аренду не только оптические кабели или волокна, но и отдельные длины волн, т.е. реализовать концепцию "виртуального волокна". По одному волокну на разных длинах волн можно одновременно передавать самые разные приложения: кабельное телевидение, телефонию, трафик Интернет, “видео по требованию” и т.д. Как следствие этого, часть волокон в оптическом кабеле можно использовать для резерва. Применение технологии WDM позволяет исключить дополнительную прокладку оптических кабелей в существующей сети.

Первые устройства WDM позволяли организовать на одном волокне от 4 до 16 каналов, каждый из которых поддерживал передачу сигналов синхронной цифровой иерархии SDH/SONET со скоростью 2,5 Гбит/с. На сегодняшний день технология WDM позволяет организовать на одном волокне каналы с разницей длин волн между соседними каналами всего в единицы нанометра (1 нм), что называется плотным волновым мультиплексированием **DWDM** (*Dense WDM*). Развитие технологии DWDM позволило создать коммерческие сети, в которых по отдельным волокнам создаются более сотни независимых оптических каналов, а также сети с передачей сигналов в обоих направлениях в одном и том же оптическом волокне.

Своими успехами технология DWDM во многом обязана достижениям в разработке усилителей на оптической среде. В таком усилителе оптические

сигналы усиливаются без преобразования в электрические и обратно, что дает возможность создавать сети передачи данных высокой протяженности при значительной экономии электронных компонентов или вообще без них.

Система DWDM структурно во многом похожа на традиционную систему с частотным уплотнения с FDM. Сигналы разных длин волн (частот), генерируемые одним или несколькими оптическими передатчиками, объединяются мультиплексором в многоканальный составной оптический сигнал, который далее распространяется по оптическому волокну. Стандартный частотный интервал между каналами составляет 100 ГГц (около 0,8 нм по длине волны). При этом для передачи данных применяется 41 волна. Имеются предложения по стандартизации частотного плана с расстоянием 50 ГГц (около 0,4 нм) и даже 25 ГГц. При разнесении частот с шагом 50 ГГц используется 81 волна. Реализация частотных каналов с шагом 50 или 25 ГГц предъявляет очень жесткие требования к оборудованию DWDM, особенно в случаях, если волна транспортирует сигналы со скоростью 10 Гбит/с.

При большой протяженности тракта передачи на линии связи устанавливается один или несколько оптических повторителей. Демультиплексор принимает составной сигнал, выделяет из него исходные каналы разных длин волн и направляет их на соответствующие фотоприемники. На промежуточных узлах некоторые каналы могут быть добавлены или выделены из составного сигнала посредством мультиплексоров ввода/вывода или устройств кросс-коммутации.

Система DWDM в общем случае состоит из одного или нескольких лазерных передатчиков, мультиплексора, одного или нескольких оптических усилителей, мультиплексоров ввода/вывода, оптического волокна (кабеля), демультиплексора и соответствующего числа фотоприемников, а также электронного оборудования, которое обрабатывает передаваемые данные в соответствии с используемыми протоколами связи, и системы сетевого управления.

На линиях связи большой протяженности в настоящее время используются скорости передачи 2,5 Гбит/с (STM-16) и 10 Гбит/с (STM-64). Скорость передачи в сетях связи городского и регионального масштабов обычно намного меньше.

Следует отметить, что сама технология DWDM предоставляет только каналы передачи данных. Кодирование передаваемой на каждой волне информации относится к аппаратуре, использующей эти каналы, в частности SDH или 10Gigabit Ethernet.

5.4. Цифровая сеть интегрального обслуживания ISDN

5.4.1. Назначение и общая характеристика сетей

Технология **ISDN** (*Integrated Services Digital Network*) появилась в середине 70-х годов прошлого столетия. Основополагающие положения содержатся в рекомендациях МККТТ I.122, I.430 и I.431. Разработчики ISDN ставили целью соединить жилые дома и предприятия проводными линиями связи в единую интегральную сеть и передавать по ней единым цифровым способом всевозможные сообщения.

Сеть интегрального обслуживания ISDN относится к сетям с коммутацией каналов. Цифровые сети ISDN можно использовать для решения широкого класса задач, а именно:

- цифровой телефонии;
- передачи данных;
- объединения удаленных локальных сетей;
- доступа к глобальным сетям (Интернет);
- передачи видео, графической информации, звука, данных, чувствительных к задержкам.

Оконечным устройством сети может быть цифровой телефонный аппарат, отдельный компьютер с установленной ISDN-платой, файловый или специализированный сервер, мост или маршрутизатор.

Преимущества сетей ISDN по сравнению с аналоговыми телефонными сетями общего пользования состоят в следующем:

1) обеспечивается полностью цифровая связь, обладающая высокой помехоустойчивостью;
2) достигается сравнительно высокая скорость передачи различных видов информации;
3) реализуется широкий набор функций для телефонии, высокое качество звука (переадресация звонков, идентификация линии, конференции нескольких абонентов, внутренняя сокращенная нумерация, функция автоответчика и т.д.);
4) осуществляется быстрый набор номера (менее 1с);
5) широко распространяется во многих странах.

Недостатки ISDN сети вызваны преимущественно проблемами совместимости оборудования различных производителей, сложностью модернизации центральных коммутаторов, Необходимостью больших финансовых вложений.

Основу иерархии составляет **цифровой канал** со скоростью передачи **64 кбит/с**, называемый в ISDN B-каналом. По таким каналам может переда-

ваться речь, данные и изображения. 30 цифровых каналов (плюс два канала для служебных целей) европейской системы (24 – американской и японской) объединяются в первичную цифровую ступень преобразования (ЦСП) с суммарной скоростью передачи 2048 (1544) кбит/с. Затем четыре первичных цифровых ступеней преобразований группируются в одну вторичную со скоростью 8 448 кбит/с и т.д.

Стандартное подключение линий ISDN осуществляется по интерфейсам BRI (*Basic Rate Interface*) или PRI (*Primary Rate Interface*). Служба **BRI** (ISDN 2) обеспечивает 2 дуплексных B-канала по 64 кбит/с и один служебный D-канал с пропускной способностью 16 кбит/с. Компьютер, подключенный к сети ISDN, может совместно использовать оба B-канала для пересылки данных со скоростью 128 кбит/с.

При подключении крупных организаций применяется служба **PRI** (ISDN 30), предоставляющая 30 B-каналов по 64 кбит/с с суммарной пропускной способностью 2048 кбит/с. Кроме этого имеется специальный D-канал с пропускной способностью 64 кбит/с. Возможно предоставление такого количества каналов, которое требуется заказчику (например, 4, 6, ...).

Канал D (*Delta*) – служебный, служащий для передачи управляющих сигналов. Один канал обслуживает 2 или 30 B-каналов. Он обеспечивает возможность быстрой генерации и сброса вызовов, а также передачу информации о поступающих вызовах, в том числе о номере обращающегося к сети абонента.

Физическим уровнем интерфейса BRI является витая телефонная пара, которая работает в дуплексном режиме передачи данных, так называемый U-интерфейс.

5.4.2. Подключение пользовательского оборудования к сети

В ISDN различают два типа оконечного (терминального) оборудования (*Terminal Equipment*): TE-1 и TE-2. Первый совместим с ISDN, а под TE-2 подразумевается любой другой тип ООД с собственным протоколом. Поэтому для подключения TE-2 к сети ISDN необходим **терминальный адаптер** (*Terminal Adaptor*) TA. Устройства TE и TA соединяются с помощью стандартного R-интерфейса. Состав, расположение и обозначение стандартных интерфейсов между различными устройствами абонентского оборудования и линией связи с узлом коммутации ISDN показан на рисунке 5.24. Устройства TE-1 и TA ISDN-совместимы и могут быть подключены к **сетевому терминатору** (*Network Terminator*) интегральной сети NT-2 или NT-1. Терминатор NT-1 представляет собой мультиплексор, который обычно подключается к первичному цифровому каналу. NT-2 является устройством канального или

сетевого уровня, которое выполняет функции концентрации интерфейсов и их мультиплексирование.

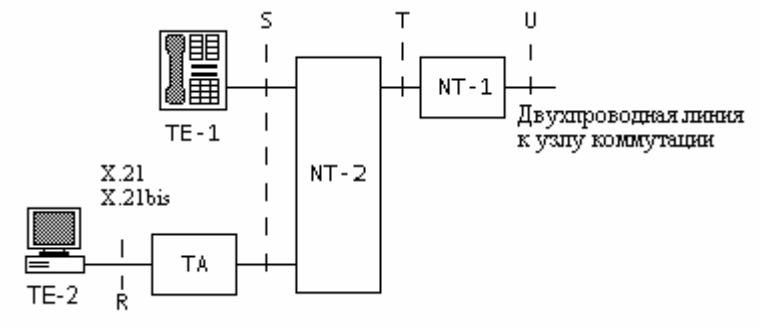


Рисунок 5.24 – Состав интерфейсов и абонентского оборудования сети ISDN

Наличие этого оборудования не является обязательным, в отличие от NT-1. Терминатор NT-1 может быть также снабжен соединителем с основным (базовым) цифровым каналом.

Основными функциями терминального адаптера TA являются:

1) согласование R-интерфейса с B- или D-каналом;
2) преобразование формата данных;
3) реализация функций 2-го и 3-го уровней эталонной модели ВОС;
4) управление потоком данных и контроль состояния R-интерфейса;
5) передача и прием управляющих сигналов, поступающих через D-канал.

Схема связи между различным оконечным оборудованием в сети ISDN показана на рисунке 5.25.

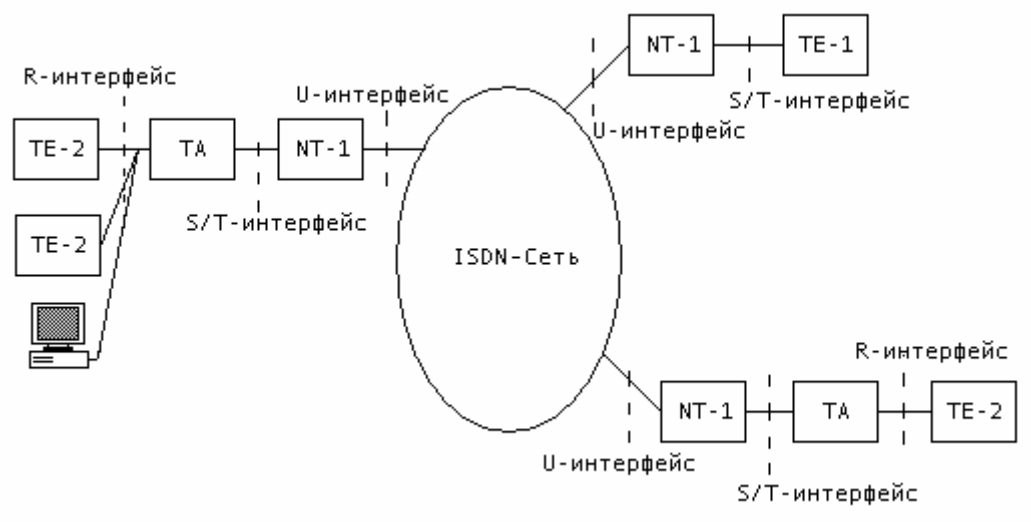


Рисунок 5.25 – Схема подключения оконечного оборудования в сети ISDN

Адресация в ISDN. Технология ISDN разрабатывалась в качестве ос-

новы всемирной телекоммуникационной сети, позволяющей связывать как телефонных абонентов, так и абонентов других компьютерных глобальных сетей. Основное назначение ISDN – передача телефонного трафика. Поэтому за основу адреса ISDN был взят формат международного телефонного плана номеров, описанный в стандарте ITU-T E.163. Однако этот формат был расширен для поддержки большего числа абонентов и для использования в нем адресов других сетей, например X.25. Стандарт адресации в сетях ISDN получил номер E.164, он предусматривает до 55 десятичных цифр в номере.

При вызове абонентов из сети, не относящейся к ISDN, их адрес может непосредственно заменять адрес ISDN. Например, адрес абонента сети X.25, в которой используется система адресации по стандарту X.121, может быть целиком помещен в поле адреса ISDN, но для указания, что этот адрес стандарта X.121, ему должно предшествовать поле префикса, в которое помещается код стандарта адресации, в данном случае стандарта X.121.

5.4.3. Взаимодействие на сетевом уровне

Процессом передачи информации между узлами, как было сказано в первом разделе, управляет система сигнализации по общему каналу SS7 (см.п.1.5.4). Сигнальная система SS7 была разработана первоначально для целей внутреннего мониторинга и управления коммутаторами в сети ISDN. Для передачи информации системы сигнализации внутри ISDN организована сеть каналов типа D. Все линейные и управляющие сигналы, такие как номер вызываемого абонента, подтверждения принятия данного номера, информация о доступности абонента (занят или свободен), разъединение с указанием конкретного типа отказа и т.д., упаковываются в специальные блоки данных и снабжаются идентификаторами разговорных каналов, исходящих и входящих станций, а также служебной информацией. Эти блоки данных (так называемые сигнальные единицы) передаются по отдельному сигнальному каналу.

Если один из абонентов ISDN-сети является обычным аналоговым абонентом или ему требуется переход на другую систему сигнализации, то многие услуги ISDN для такого абонента становятся недоступными.

Непосредственная передача информации сигнализации осуществляется блоками данных сетевого уровня— кадрами ISDN. Сообщение сетевого уровня ISDN состоит из двух основных частей: *общей* и *специфической*.

Общая часть сообщения начинается с поля *Protocol descriminator*, предназначенного для определения типа протокола, используемого для управления вызовами. Каждому вызову присваивается номер вызова **CRV** (*Call Reference Value*). Это сделано для обеспечения возможности использо-

вания канала D для одновременного управления несколькими вызовами *Message Type*. Данное поле определяет тип процедуры, обеспечивающей непосредственное управление процессом организации и обслуживания вызова.

Специфическая часть пакета ISDN представляет собой поле переменной длины. Информация, размещаемая в данной части соответствующего пакета ISDN, используется для определения конкретных параметров процедуры управления вызовом. В частности, в специфической части пакета, который предназначен для создания вызова, размещаются следующие поля признаков:

- вид запрашиваемого вызова (поток данных, ISDN-телефонный вызов, вызов аналоговой телефонной сети общего пользования);
- скорость передачи данных для запрашиваемого вызова;
- адреса вызываемого и вызывающего абонентов.

Создание вызова происходит в следующем порядке.

1. Передача абонентом сообщения **SETUP**— установление соединения. Адрес (телефонный номер) вызываемого абонента находится в информационном поле специфической части пакета ISDN. Сообщение SETUP передается в направлении ближайшего к данному терминалу коммутатора ISDN.

2. Выдача коммутатором ISDN ответного сообщения **Call Proceeding**. Данное сообщение подтверждает возможность установления соединения и закрепляет за инициируемым вызовом один из каналов B.

3. Подтверждение вызываемым абонентом получение вызова передачей сообщения **Alerting** "Приведение в готовность".

4. Передача сообщения **Connect** (соединение установлено) коммутаторами по сети в направлении терминала-инициатора соединения.

5. Выдача сообщения **Connect Acknowledge,** подтверждающего установлениея соединения.

Завершение вызова осуществляется путем передачи следующих сообщений.

- **Disconnect** – сообщение о разрывании соединения. В информационном поле специфической части пакета ISDN данного типа размещается информация о причине, приведшей к разрыванию соединения.
- **Release** – сигнализация об освобождении ресурсов, которые были использованы при организации вызова.
- **Release Complete** – освобождение завершено.

В настоящее время ISDN повсеместно вытесняет общедоступную коммутируемую телефонную сеть, обеспечивая высокоскоростную связь и более экономичное подключение, чем коммутируемые каналы общего пользования.

С появлением оптических линий связи разработана высокоскоростная

цифровая интегральная сеть B-ISDN (*Broadband - ISDN*).

5.5. Цифровая сеть с коммутацией пакетов X.25

5.5.1. Структура и особенности построения сети

Сети X.25 являются "старейшиной" применяемых пакетных сетей, хотя масштаб использования их быстро падает. Долгое время сети X.25 были единственными доступными сетями с коммутацией пакетов коммерческого типа, в которых давались гарантии коэффициента готовности сети. Важным преимуществом сетей X.25 является то, что они хорошо работают на ненадежных линиях благодаря протоколам с установлением соединения и коррекцией ошибок на двух уровня: канальном и сетевом. Сети X.25 наилучшим образом подходят для передачи трафика низкой интенсивности, характерного для терминалов, и в меньшей степени соответствует более высоким требованиям трафика локальных сетей. Стандарт X.25 не регламентирует внутреннее устройство сети, а только определяет пользовательский интерфейс с сетью. Взаимодействие двух сетей X.25 регламентирует стандарт X.75.

Технология сетей X.25 имеет несколько существенных признаков, отличающих ее от других технологий коммутации пакетов. К ним относятся следующие:

- использование в сети специального устройства сборки/разборки пакетов данных **СбПД** (***PAD*** - *packet assemble/disassemble*); оно предназначено для сборки нескольких низкоскоростных стартстопных потоков байтов от алфавитно-цифровых терминалов в пакеты, передаваемые по сети и направляемые компьютерам для обработки;
- наличие трехуровневого стека протоколов с применением на канальном и сетевом уровнях протоколов с установлением соединения, управляющих потоками данных и исправляющих ошибки;
- ориентация на однородные стеки транспортных протоколов во всех узлах сети; сетевой уровень рассчитан на работу только с одним протоколом канального уровня и не может подобно протоколу IP объединять разнородные сети.

Сеть X.25 состоит из узлов коммутации пакетов **УКП** (англ. **S** – Switch), расположенных в различных географических точках и соединенных высокоскоростными выделенными каналами (рисунок 5.26). Выделенные каналы могут быть как цифровыми, так и аналоговыми.

Асинхронный старт-стопный терминал **T** подключается к сети коммутации пакетов через устройство сборки/разборки пакетов данных и отвечает

рекомендациям X.3, X.28 и X.29. Один СбПД обеспечивает интерфейс для 8, 16 или 24 асинхронных терминалов.

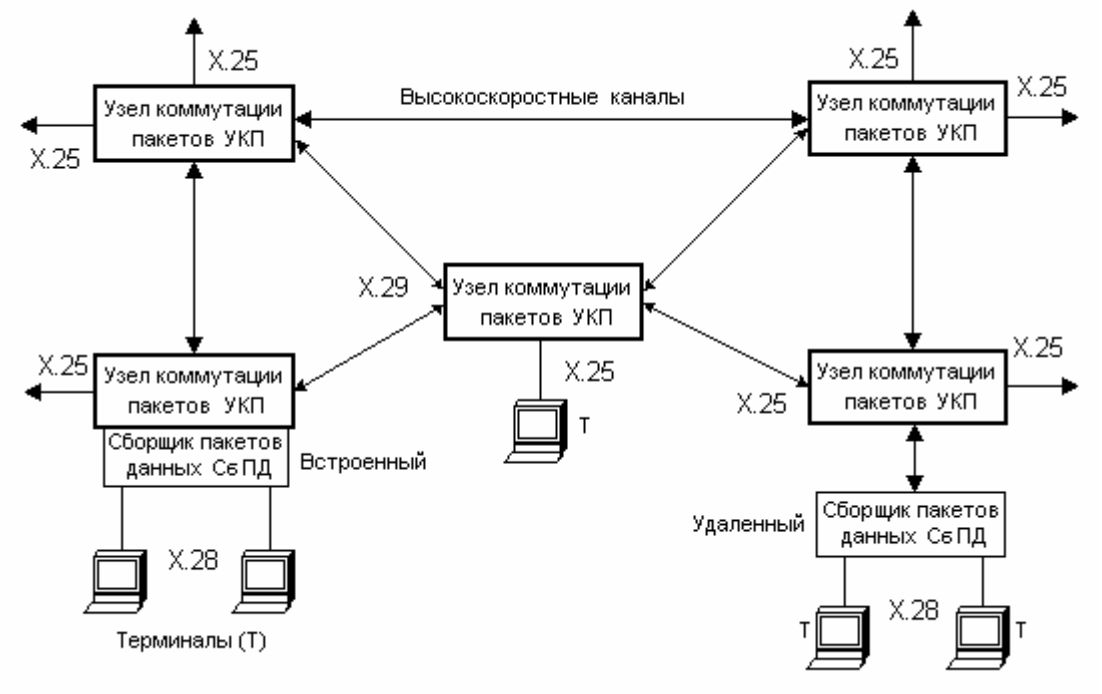


Рисунок 5.26 – Структура сети X.25

Пакет данных состоит обычно из 128 байтов, которые передаются по адресу, содержащемуся в пакете. Длина сетевого пакета может лежать в пределах 64…4096 байтов. Размер пакета также как и величина окна (число пакетов, принимаемых без подтверждения) определяются на фазе установления канала. СбПД конструктивно могут быть выполнены в виде встроенного или удаленного устройства. Встроенное устройство располагается в стойке коммутатора. Терминалы Т получают доступ к встроенному устройству СбПД по телефонной сети с помощью модемов с асинхронным интерфейсом (модемы на рисунке не показаны). Встроенное устройство СбПД также подключается к телефонной сети с помощью нескольких модемов с асинхронным интерфейсом. Удаленное устройство СбПД представляет собой небольшой автономный блок, подключенный к коммутатору через выделенный канал связи X.25. К удаленному устройству СбПД терминалы подсоединяются по асинхронному интерфейсу, обычно для этой цели используется интерфейс RS-232C.

Встроенное устройство СбПД может быть использовано совместно несколькими терминалами, расположенными в различных местах, в то время как удаленное СбПД обслуживает терминалы, расположенные обычно в одном месте. Существенным недостатком встроенного СбПД является отсутст-

вие какого-либо линейного протокола, предусматривающего устранение ошибок в данных, посылаемых от СбПД к терминалу. В удаленном СбПД предусмотрена процедура восстановления ошибочных данных.

Стандарт X.3 определяет основные функции устройства сборки/разборки пакетов данных, к которым относятся следующие:

- сборка символов, полученных от асинхронных терминалов, в пакеты;
- разборка полей данных в пакетах и вывод информации на асинхронные терминалы;
- управление процедурами установления соединения и разъединения по сети X.25 с нужным компьютером;
- обнаружение поступившего от асинхронного терминала сигнала "разрыв соединения" ;
- передача символов по требованию асинхронного терминала, содержащих старт-стопные сигналы и биты паритета;
- редактирование последовательностей команд СбПД.

Адрес в сети X.25 присваивается не терминалу, а порту СбПД, который подключен к коммутатору пакетов X.25 с помощью выделенного канала связи.

Несмотря на то, что задача подключения "неинтеллектуальных" терминалов к удаленным компьютерам возникает сейчас достаточно редко, функции СбПД все еще остаются востребованными. Устройства СбПД часто используются для подключения к сетям X.25 кассовых терминалов и банкоматов, имеющих асинхронный интерфейс RS-232C.

Стандарт X.28 определяет параметры терминала, а также протокол взаимодействия терминала с устройством СбПД. При работе на терминале пользователь сначала проводит некоторый текстовый диалог с устройством СбПД, используя стандартный набор символьных команд. СбПД может работать с терминалом в двух режимах: управляющем и передачи данных. В **управляющем режиме** пользователь с помощью команд может указать адрес компьютера, с которым нужно установить соединение по сети X.25, а также задать некоторые параметры работы СбПД. К ним относятся: выбор специального символа для обозначения команды немедленной отправки пакета, установка режима эхо-ответов от устройства СбПД символов, набираемых на клавиатуре. При этом дисплей не будет отображать символы, набираемые на клавиатуре до тех пор, пока они не вернутся от СбПД. Это – обычный локальный режим работы терминала с компьютером. При наборе комбинации клавиш Ctrl+P СбПД переходит в режим передачи данных и воспринимает все последующие символы как данные, которые нужно передать в пакете X.25 узлу назначения.

В сущности, протоколы X.3 и X.28 определяют процедуру эмуляции терминала. Пользователь с помощью устройства СбПД устанавливает со-

единение с нужным компьютером. После этого он имеет возможность вести диалог с операционный системой этого компьютера. Пользователь может запускать нужные программы и просматривать результаты их работы на своем экране, как и при локальном подключении терминала к компьютеру.

Компьютеры и локальные сети, как правило, подключаются к сети X.25 непосредственно через адаптер X.25 или маршрутизатор, поддерживающий на своих интерфейсах протоколы X.25. Для управления устройствами СбПД в сети существует **протокол X.29**, с помощью которого узел сети может управлять и конфигурировать СбПД удаленно, по сети. При необходимости передачи данных компьютеры, подключенные к сети X.25 непосредственно, услугами СБПД не пользуются, а самостоятельно устанавливают виртуальные каналы в сети и передают по ним данные в пакетах X.25.

5.5.2. Адресация в сетях X.25

Сеть X.25 может работать автономно и не связываться другими сетями. В этом случае в ней разрешается использовать адреса любой длины (в пределах формата поля адреса), а адресам присваиваться произвольные значения. Максимальная длина поля адреса в пакете X.25 составляет 16 байтов. При объединении сети X.25 с другими аналогичными сетями, в ней нужно придерживаться стандартной адресации, которая регламентируется рекомендацией МККТТ X.121.

Адреса X.121 имеют разную длину, которая может доходить до 14 десятичных знаков. Первые четыре цифры адреса называют **кодом идентификации сети** *DNIC (Data Network Identification Code).* Код DNIC поделен на две части: первая часть (3 цифры) определяет страну, в которой находится сеть, а вторая — номер сети X.25 в данной стране. Таким образом, внутри каждой страны можно организовать только 10 сетей X.25. При необходимости пронумеровать больше, чем 10 сетей для одной страны, такой стране дается несколько кодов. Остальные цифры называются **номером Национального терминала** *(National Terminal Number, NTN).* Эти цифры позволяют идентифицировать определенное терминальное устройство в сети X.25.

По стандарту ISO 7498 для нумерации сетей X.25 к адресу в формате X.121 добавляется только один байт префикса, несущий код 36 (использование в адресе только кодов десятичных цифр) или 37 (использование произвольных двоичных комбинаций). Этот код позволяет универсальным коммутаторам, например, коммутаторам сети ISDN, поддерживающим также и коммутацию пакетов X.25, автоматически распознавать тип адреса и правильно выполнять маршрутизацию запроса на установление соединения.

5.5.3. Стек протоколов сети X.25

Стандарт X.25 определяет первые три уровня эталонной модели взаимодействия открытых систем (рисунок 5.27): **физический** (интерфейс X.21), **канальный** (протокол *HDLC*) и **сетевой** (пакетный). Транспортный уровень может быть реализован в конечных узлах, но он стандартом не регламентируется.

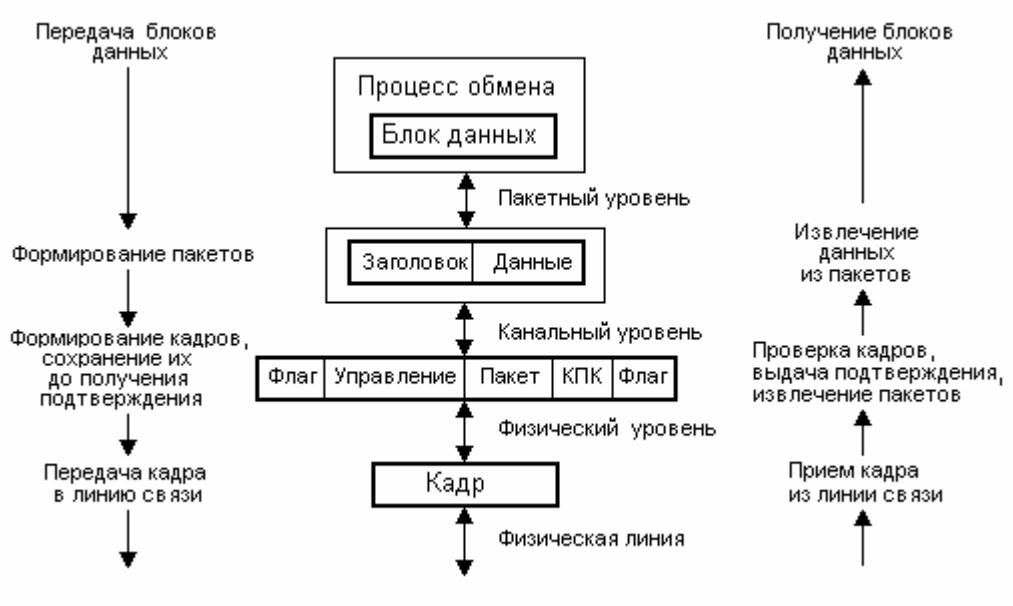


Рисунок 5.27 – Уровни сети X.25

Протокол **физического уровня** канала связи не оговорен, и это дает возможность использовать каналы разных стандартов.

На канальном уровне преимущественно используется балансная процедура доступа к каналу **LAP-B** (*Link Access Procedure Balanced*) на основе асинхронного балансного режима протокола HDLC. Этот протокол обеспечивает сбалансированный режим работы, т.е. оба узла, участвующих в соединении, равноправны. По протоколу LAP-B устанавливается соединение между пользовательским оборудованием (компьютером, IP-маршрутизатором) и коммутатором сети. В протоколе реализуется алгоритм скользящего окна с возможностью автоматической повторной передачи в случае возникновения ошибок в линии. В LAP-B нумеруются кадры, а не байты, как это делается в протоколе TCP. Размер окна имеет фиксированный размер, который установлен равным 8 или 128 кадрам.

Сетевой уровень X.25/3 (в стандарте он назван не сетевым, а пакетным уровнем) реализуется с использованием 14 различных типов пакетов по назна-

чению аналогичных типам кадров протокола LAP-B. Так как надежную передачу данных обеспечивает протокол LAP-B, протокол X.25/3 выполняет только функции маршрутизации пакетов, установления и разрыва виртуального канала между конечными абонентами сети и управления потоком пакетов. После установления соединения на канальном уровне конечный узел должен создать виртуальное соединение с другим конечным узлом сети. Для этого он в кадрах LAP-B посылает пакет запроса *Call Request* протокола X.25. Этот пакет является пакетом сигнализации для сети X.25, которая отличается тем, что режим сигнализации в ней не выделен в отдельный протокол, а представляет собой один из режимов работы общего протокола сетевого уровня X.25/3. Пакет Call Request принимается коммутатором сети и маршрутизируется на основании таблицы маршрутизации, прокладывая при этом виртуальный канал. Начальное значение номера виртуального канала задается в этом пакете пользователем.

По завершению установления виртуального канала конечные узлы обмениваются пакетами данных. Отличие формата пакета данных от формата пакета *Call Request* состоит в отсутствии в нем полей адреса и услуг. Пакет данных не имеет поля, которое бы определяло тип переносимых в пакете данных, т.е. поля, аналогичного полю *Protocol* в *IP*-пакете. Для устранения этого недостатка первый байт в поле данных всегда интерпретируется как признак типа данных.

Коммутаторы, установленные на узлах коммутации сетей X.25, представляют собой гораздо более простые и дешевые устройства по сравнению с маршрутизаторами сетей TCP/IP. Это связано с тем, что они не поддерживают процедур обмена маршрутной информацией и нахождения оптимальных маршрутов, а также не выполняют преобразований форматов кадров канальных протоколов. По принципу работы они ближе к коммутаторам локальных сетей, чем к маршрутизаторам. Однако работа, которую выполняют коммутаторы X.25 над пришедшими кадрами, включает больше этапов, чем при продвижении кадров коммутаторами локальных сетей. Коммутатор X.25 должен принять кадр LAP-B и ответить на него другим кадром LAP-B, в котором подтвердить получение кадра с конкретным номером. При утере или искажении кадра коммутатор должен организовать повторную передачу кадра. Если же кадр LAP-B принят без ошибок, то коммутатор извлекает пакет X.25, на основании номера виртуального канала определяет выходной порт, а затем формирует новый кадр LAP-B для дальнейшего продвижения пакета. В результате производительность коммутаторов X.25 оказывается обычно невысокой — несколько тысяч пакетов в секунду.

5.6. Сеть ретрансляции кадров Frame relay

5.6.1. Назначение и общая характеристика сети

С переходом межсетевых коммуникаций к цифровым и оптоволоконным средам появились новые технологии, которые требуют меньшего уровня контроля ошибок. Одной из перспективных технологий является технология ретрансляции кадров *Frame relay*.

Frame relay **(FR)** – это усовершенствованная технология быстрой коммутации пакетов переменной длины. Разработчики технологии исключили многие функции учета и контроля, используемые в сетях X.25, в связи с отсутствием их необходимости по причине высокой помехозащищенности оптоволоконных линий связи.

Основное отличие протокола сетей ретрансляции кадров от бит-ориентированного протокола канального уровня HDLC заключается в том, что узлы коммутации выполняют только основные функции канального уровня, связанные с получением и дальнейшей передачей (ретрансляцией) кадров. Кадры при передаче через коммутатор не подвергаются преобразованию, из-за чего собственно технология и получила такое название. Другим существенным отличием протокола FR от HDLC является то, что он не предусматривает передачу управляющих сообщений (нет командных или супервизорных кадров, как в HDLC).

На канальном уровне сети Frame relay передача данных между двумя соседними коммутаторами регламентируется протоколом **LAP-F** (*Link Access Procedure for Frame mode*), обозначаемый по рекомендации ITU-T номером **Q.922**.

Для передачи служебной информации используется специально выделенный **канал сигнализации**. Еще одно важное отличие – отсутствие нумерации последовательно передаваемых (принимаемых) кадров. Это связано с тем, что протокол FR не имеет никаких механизмов для подтверждения правильно принятых кадров.

В сетях Frame relay устанавливается **двухточечные** (двухпунктовое) соединение с использованием **постоянного виртуального канала** PVC (*Permanent Virtual Circuit*) для передачи кадров переменной длины. Суть создания виртуального канала состоит в том, что маршрутизация пакетов между коммутаторами сети происходит только один раз на стадии образования виртуального канала. После его создания передача происходит по одному и тому же пути на основании номеров – идентификаторов виртуальных каналов. Данные от источника передаются по цифровой выделенной линии к коммутатору данных сети Frame relay. Далее они проходят по сети FR до

узла назначения.

Высокую скорость в сети Frame relay обеспечивает постоянный виртуальный канал, благодаря чему известен весь маршрут между конечными точками. Поэтому устройства Frame relay избавлены от некоторых постоянных процедур: фрагментации, восстановления, выбора оптимального маршрута. Кроме того, сети FR могут выделять абонентам практически любую скорость передачи. Для передачи по FR-сети необходим совместимый с этой сетью маршрутизатор или мост. Формат кадра сети Frame relay показан на рисунке 5.28. Заголовок длиной 16 или 32 битов содержит идентификатор виртуального канала **ИВК** (DLCI – *Data-Link Connection Identifier*), признак команда/ответ **К/О** (C/R –Command/Response), признак расширения адреса **РА** (EA – *Extended Address*), биты уведомления приемника **БУПП** (FECN – *Forward Explicit Notification*) и источника **БУИП** (BECN – *Backward Explicit Notification*) о перегрузке, а также бит допустимости удаления кадра **ДУК** (DE – *Discard Eligibility*). В начале и конце кадра располагается флаг кадровой синхронизации.

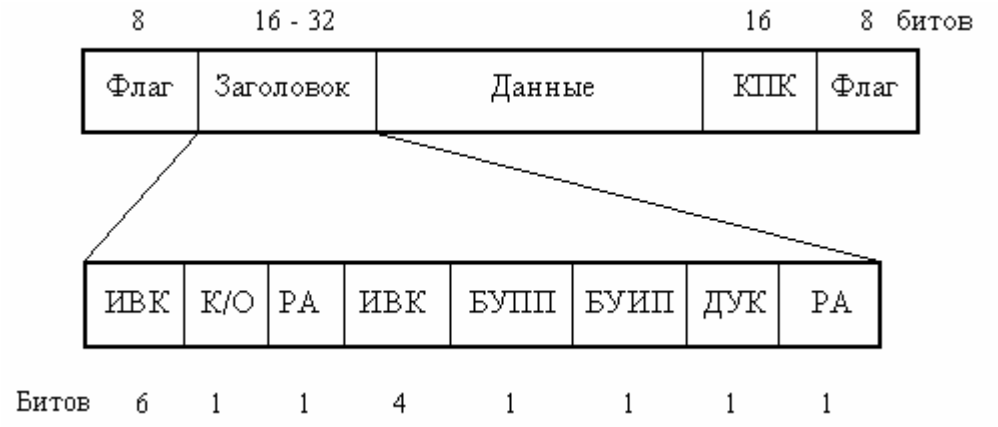


Рисунок 5.28 – Формат кадра сети *Frame relay*

Вместо приоритезации трафика в Frame relay применяется процедура заказа качества обслуживания в процессе установления соединения. Установлен ряд параметров качества, а именно:

- согласованная информационная скорость, с которой сеть будет передавать данные;
- согласованный объем пульсаций скорости;
- дополнительный объем пульсаций.

Большинство процедур обработки и управления передачей информации в Frame relay осуществляется оконечным оборудованием данных с помощью протоколов более высокого уровня. Например, в рамках протокола ретрансляции кадров FR, защита от ошибок ограничивается проверкой кон-

трольной последовательности кадра. Какие-либо механизмы для корректировки ошибочных файлов, например, за счет повторной передачи кадров, данным протоколом не предусматриваются. В случае ошибки сообщение об этом передается протоколам более высокого уровня.

Узлам сети FR разрешено уничтожать искаженные кадры, не уведомляя об этом пользователя. Искаженным считается кадр, которому присущ какой-либо из следующих признаков:

- нет корректного ограничения флагами;
- имеется менее пяти октетов между флагами;
- нет целого числа октетов после удаления бит обеспечения прозрачности передачи;
- обнаружена ошибка в принятом кадре;
- искажено поле адреса (для случая, когда проверка не выявила ошибки в КПК);
- содержится несуществующий идентификатор виртуального канала;
- превышен допустимый максимальный размер кадра (в некоторых вариантах реализации стандартов FR возможна принудительная обработка кадров, превышающих допустимый максимальный размер).

Перераспределение функций между сетевым и канальным уровнями отразились и на структуре информационных блоков. Если в сетях X.25 на канальном уровне информационной единицей является кадр, а на сетевом – пакет, то в сетях FR на этих уровнях используется единая структурная единица – **кадр**. Следует также отметить меньшую избыточность кадров сети Frame relay. Так, при максимальной длине кадра 1024 байта служебная информация составляет 6…8 байтов.

5.6.2. Управление доступом и защита от перегрузок

Управление доступом к сети Frame relay возлагается на интерфейс локального управления **LMI** (*Local Management Interface*). Доступ в сеть FR обеспечивают *порты* FR и FR-*адаптеры* - сборщики/разборщики кадров FR.

Добиться высокой эффективности использования пропускной способности физических линий и каналов связи, а также исключения перегрузок узлов связи и всей сети FR позволяет метод статистического мультиплексирования кадров, в соответствии с которым выполняется следующее:

- постоянное "наблюдение" аппаратурой канала данных (АКД) за потоком заявок от пользователей на передачу сообщений и за текущей загрузкой сети (линий, каналов и узлов связи);
- перераспределение свободного (и высвобождающегося) ресурса про-

пускной способности в соответствии с реальными потребностями абонентов;

- предоставление пользователям каналов информационного обмена, удовлетворяющих их требованиям.

В протоколе ретрансляции кадров FR отсутствует явно выраженное управление потоками для каждой виртуальной цепи. Вместо этого предусмотрены простые *механизмы уведомления о перегрузках*, позволяющие информировать абонента о том, что ресурсы сети приближаются к состоянию перегрузки.

Интерфейс локального управления LMI предусматривает три способа локального управления:

- синхронное симплексное;
- синхронное дуплексное;
- асинхронное.

Для организации **синхронного симплексного управления** используются два вида сообщений: "Запрос состояния" и "Состояние". С помощью этих сообщений интерфейс проверяет целостность соединения, уведомляет о включении или выключении, а также о готовности постоянного виртуального канала PVC.

В процессе реализации процедуры симплексного управления оконечное оборуддование данных ООД периодически запрашивает через интерфейс LMI состояние сети, посылая через определенный временной интервал в сеть сообщение "Запрос состояния" с целью подтверждения целостности соединения. На этот запрос сеть отвечает сообщением "Состояние", содержащим требуемый элемент информации о целостности соединения.

На протяжении процедуры опроса проводится подсчет числа запросов. По достижении определенного числа переданных сообщений "Запрос состояния" ООД запрашивает у сети информацию о так называемом полном состоянии, также используя сообщение "Запрос состояния". Аппаратура канала данных АКД отвечает на него сообщением "Состояние", в котором присутствуют информационные элементы для каждого PVC (если ООД имеет несколько портов). Отсутствие в этом ответе информационного элемента для какого-либо PVC воспринимается терминалом пользователя как отсутствие PVC в интерфейсе "пользователь-сеть". Проверка целостности соединения основана на генерации последовательности специальных пронумерованных кадров и проверке корректности ее передачи. ООД может обнаруживать следующие ошибки:

- прием кадра, информирующего о целостности соединения, с неправильным порядковым номером (не соответствующим порядковому номеру последнего переданного кадра);
- отсутствие приема сообщения "Состояние" по истечении определенного временного интервала после передачи сообщения "Запрос состояния";

- прием кадра с ошибкой в контрольной последовательности.

В этой ситуации виртуальный канал, при опросе которого обнаружены ошибки, исключается из списка активных каналов.

Синхронное дуплексное управление отличается от симплексного только тем, что сообщения "Запрос состояния" и "Состояние" имеют право передавать обе стороны интерфейса.

Главным недостатком синхронных способов управления является потенциальная задержка информирования оконечного оборудования данных ООД или аппаратуры канала данных АКД об изменениях в сетевых виртуальных каналах PVC. Например, при задержке проверки состояния, равной 60 с и скорости передачи 64 кбит/с, пользователь направит в сеть приблизительно 3,5 Мбит данных, прежде чем получит информацию о состоянии постоянного виртуального канала PVC.

Способ **асинхронного управления** позволяет при изменении состояния PVC сети FR сразу передавать стандартные сообщения "Запрос состояния" и "Состояние". Эти сообщения содержат информацию только об отдельных PVC, которые изменили свое состояние. Проверка целостности соединения также основана на генерации последовательности специальных пронумерованных кадров и проверке корректности ее передачи. Асинхронное управление может осуществляться совместно с синхронным, однако, если в сети FR применяются одновременно коммутируемые SVC и постоянные PVC виртуальные каналы, то рекомендуется использовать только асинхронное управление.

Управление потоком в сети ведется как узлами коммутации, так и ООД. Первые проверяют загруженность сети, а вторые собственно и управляют потоком данных. Для контроля над перегрузкой сети используются биты уведомления о перегрузке. При возникновении перегрузки аппаратура канала данных узла АКД отправляет устройствам-адресатам пакет с битом БУПП=1, а узлам, шлющим ему информацию, пакет с битом БУИП=1. Большое число пакетов с такими битами говорит о перегрузке и источник должен снизить частоту посылки пакетов или вовсе ее прекратить. Как правило, реакцией на перегрузку сети является снижение скорости передачи информации. Эта процедура реализуется протоколами более высоких уровней, чем канальный. Устранение перегрузок заключается в удалении части кадров сетевым оборудованием. В первую очередь удаляются кадры с признаком допустимости их удаления. Признак допустимости удаления может устанавливаться как системой, так и пользователем.

Все это позволяет существенно сократить время задержки кадров в узлах коммутации и в сочетании с высокоскоростными каналами обеспечить скорость передачи данных свыше 1,544 Мбит/с, в то время как скорость в сети X.25 не превышает 56 Кбит/с.

5.7. Асинхронная сеть передачи сообщений ATM

5.7.1. Основные принципы технологии ATM

Открытость стека протоколов TCP/IP, его возможности поддержки локальных и глобальных сетей и сегодня способствуют росту его популярности. Однако стандарты на стек протоколов TCP/IP были разработаны достаточно давно. Поэтому он уже не может удовлетворить современным требованиям, например, к передаче аудио- и видеоинформации в реальном времени, так как не имеет возможности резервировать требуемую для приложения часть пропускной способности сети.

Правила обработки информации в соответствии с протоколами TCP/IP одинаковы для всех приложений, будь то электронная почта или видеоконференция, т. е. данные имеют одинаковый приоритет при обработке и передаются с одинаковой скоростью. Кроме того, следует принимать во внимание то, что пересылка пакетов осуществляется в случайном порядке, а это противоречит концепции передачи, критичной к задержкам информации, при которой пакеты должны передаваться в строго определенной последовательности. С целью более полного удовлетворения противоречивых требований к транспортированию разнородной информации, критичной к задержкам в сети, была предложена новая быстродействующая технология построения сетей с коммутацией пакетов ATM.

ATM (*Asynchronous Transfer Mode* – **асинхронный режим передачи**) представляет собой технологию построения и функционирования сети с пакетной коммутацией и высокоскоростным асинхронным режимом передачи. Сеть ориентирована на виртуальное (логическое) соединение и обеспечение большого числа услуг (рисунок 5.29).

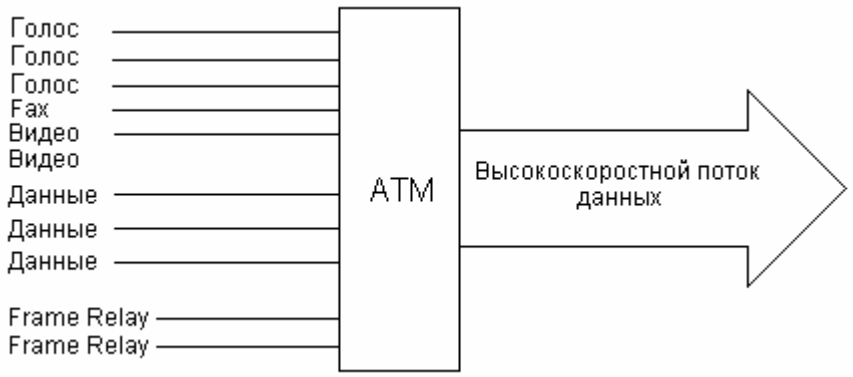


Рисунок 5.29 – Схема мультиплексирования информационных потоков в сети ATM

Технология ATM имеет значительные преимущества по сравнению с протоколом IP: благодаря ориентации на установление логического соединения между абонентами, она гарантирует эффективную и устойчивую работу, становясь идеальным решением для передачи по сети аудио- и видеоинформации. Слово *асинхронный* в названии означает, что тактовые генераторы передатчика и приемника не синхронизованы. Асинхронная передача также не предполагает упорядочивания пакетов по каналам при пересылке.

Технология ATM может использоваться как для построения высокоскоростных локальных сетей, так и магистралей, объединяющих традиционные локальные сети. Одно из главных преимуществ ATM - возможность задавать для различных потоков трафика тот или иной уровень обслуживания **QoS** (*Quality of Service*), определяющий, по существу, степень приоритетности трафика при передаче его по сети.

В ATM-сетях для создания информационных магистралей между передающим и принимающим узлами используются виртуальные каналы (виртуальные цепи). **Виртуальный канал** представляет собой некоторый тракт (путь) передачи между двумя узлами коммутируемой сети, который с точки зрения передачи информации выглядит как выделенное двухточечное соединение, "прозрачное" для пользователя. В ATM-сетях существуют три типа виртуальных каналов: постоянные, коммутируемые и интеллектуальные постоянные виртуальные каналы. Детальнее о них сказано ниже.

ATM осуществляет передачу с высокой скоростью небольших пакетов фиксированной длины через широкополосные каналы связи. По терминологии, принятой в ATM, пакеты называются **ячейками** (*cells*). Этот термин введен, чтобы отличить пакеты ATM от пакетов низкоскоростных сетей. Ячейки имеют два важных преимущества перед кадрами. Во-первых, поскольку кадры имеют переменную длину, каждый поступающий кадр должен буферизироваться, что гарантирует его целостность передачи. В связи с тем, что ячейки всегда имеют одну и те же длину, они требуют меньшей затраты памяти, так как можно предсказать требуемый размер буфера. Во-вторых, все ячейки имеют одинаковую длину, поэтому появление полей пакетов предсказуемо. Их заголовки всегда находятся на одном и том же месте. В результате коммутатор автоматически обнаруживает заголовки ячеек и их обработка происходит быстрее. В сети с трансляцией ячеек размер каждой из них должен быть достаточно мал, чтобы сократить время ожидания (очень важно для телефонии), но достаточно велик, чтобы повысить эффективную скорость передачи данных.

Разработчикам ATM было достаточно трудно найти компромисс между временем ожидания и издержками передачи (снижение эффективной скорости). Они должны были учесть интересы как телефонной отрасли, так и производителей оборудования передачи данных. Телефонистам нужен был

маленький размер ячейки, поскольку голос обычно передается маленькими фрагментами, и уменьшение задержек обеспечивало бы более качественную связь (без пауз). Специалисты по передачи данных требовали большего размера ячеек, поскольку большие файлы обеспечивают более высокую эффективную скорость передачи данных.

В результате дискуссии разработчики нашли компромисс в размере ячейки: **53 байта**, из которых **48** информационные, а **5** – заголовок. Проверочная последовательность отсутствует, так как используются высококачественные каналы связи. Заголовок пакета содержит лишь 5 байтов и предназначен главным образом для определения принадлежности данного пакета определенному виртуальному каналу. Отсутствие контроля ошибок и повторной передачи на физическом уровне приводит к эффекту размножения ошибок. Если происходит ошибка в поле *идентификатора виртуального пути* или *виртуального канала*, то коммутатор может отправить ячейку другому получателю. Таким образом, один получатель не получит ячейку, а другой получит то, что ему не предназначалось.

Для передачи пакетов по сетям АТМ от источника к получателю информации отправитель должен сначала **установить соединение** с получателем информации. Установление соединения в АТМ происходит так же, как и в сети ISDN. Формально эта процедура не является частью АТМ-протокола. Сначала посылается запрос с нулевым номером виртуального пути VPI=0 и пятым номером виртуального канала VCI=5. Если процедура завершилась успешно, то можно начинать формирование виртуального канала. При создании канала возможно использование 6 разновидностей сообщений:

setup – запрос формирования канала;
call proceeding – запрос в процессе исполнения;
connect – запрос принят;
connect ACK – подтверждение получения connect;
release – сообщение о завершении;
release compleate – подтверждение получения сообщения release.

Схема обмена сообщениями при установлении (и разрыве) виртуального соединения показана на рисунке 5.30. Предполагается, что между компьютером-источником и компьютером-адресатом находится два АТМ-коммутатора. Каждый из узлов по пути к месту назначения при получении запроса *setup* откликается, посылая сообщение *call proceeding*. Адрес места назначения указывается в сообщении *setup*.

Преимуществом наличия процедуры установления соединений является гарантированность для данного соединения определенной пропускной способности и запрошенного качества сервиса (допустимое количество потерянных пакетов, допустимое изменение интервалов между ячейками и т.д.). В результате этого сети с установлением соединений могут использо-

ваться для передачи различных видов информации: звука, видео, данных.

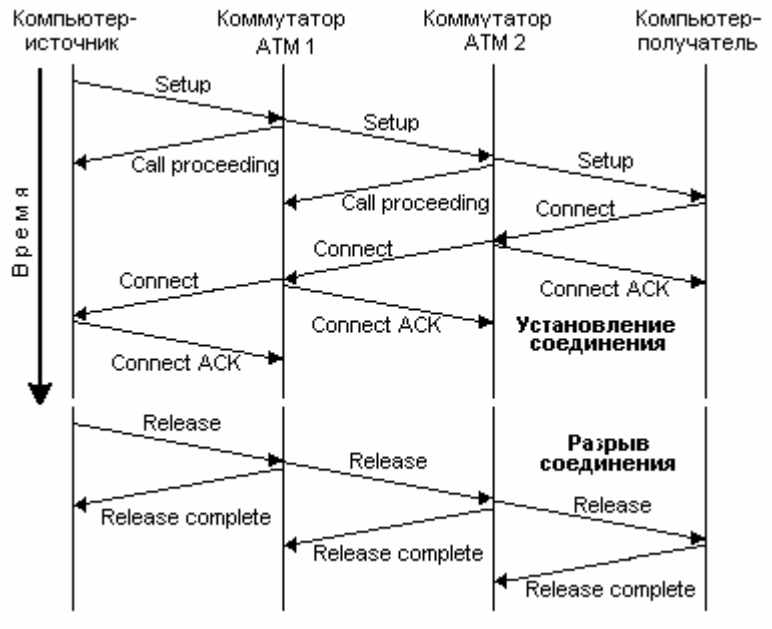


Рисунок 5.30 – Обмен сообщениями при установлении и разрыве виртуального соединения в АТМ

В сети АТМ все источники, такие как рабочие станции, серверы, маршрутизаторы и мосты, подсоединены непосредственно к коммутатору. Когда одно устройство запрашивает соединение с другим, коммутаторы его устанавливают. При этом они определяют оптимальный маршрут (традиционно эту функцию в объединенной сети выполняют маршрутизаторы).

После установления соединения коммутаторы начинают функционировать как мосты, просто пересылая пакеты. Отличие состоит лишь в том, что если мосты отправляют пакеты по всем достижимым адресам, то коммутаторы пересылают ячейки только следующему узлу заранее выбранного маршрута.

5.7.2. Уровни и протоколы сети АТМ

В отличие от общеизвестной 7-уровневой модели взаимодействия открытых сетей АТМ имеет свою собственную модель, разработанную международными организациями по стандартизации (ITU-T и ANSI). Модель АТМ состоит из трех уровней:

1) физического;
2) уровня АТМ;
3) уровня адаптации АТМ.

Эти уровни примерно соответствуют по функциям физическому, канальному и сетевому уровням ВОС. В настоящее время модель ATM не включает в себя никаких дополнительных уровней, которые соответствовали бы более высоким уровням эталонной модели ВОС.

Физический уровень устанавливает способ передачи битов через среду, типы линий связи и скорости передачи. Стандарт ATM не вводит свои спецификации на реализацию физического уровня. Здесь он основывается на технологии SDH/SONET, принимая ее иерархию скоростей. В соответствии с этим начальная скорость доступа пользователя сети – это скорость 155 Мбит/с. Другой скоростью, регламентируемой стандартом, является скорость 622 Мбит/с. При скорости передачи 155 Мбит/с можно использовать не только волоконно-оптический кабель, но и неэкранированную витую пару категории 5. На скорости 622 Мбит/с допустим только волоконно-оптический кабель. Имеются и другие физические интерфейсы к сетям ATM, отличные от SDH/SONET. К ним относятся интерфейсы T1/E1 и T3/E3, распространенные в глобальных сетях, а также интерфейсы локальных сетей, в частности, интерфейс с кодировкой 4B/5B со скоростью 100 Мбит/с (FDDI).

Уровень ATM регламентирует способ передачи сигналов, управление передачей и установление соединения. Функции передачи и управления подобны функциям канального уровня ВОС, а функции соединения ближе всего к функциям маршрутизации, которые определены стандартами модели ВОС для сетевого уровня.

Процедуры уровня ATM регламентируют, как принимать ячейку, сгенерированную на физическом уровне, добавлять 5-байтный заголовок и посылать ячейку уровню адаптации ATM. Они также задают способ установления соединения с конечной станцией.

При установлении соединения различают образование виртуальных путей и виртуальных каналов. **Виртуальный путь** VP (*Virtual Path*)– это путь между двумя коммутаторами, который существует постоянно, независимо от того, установлено ли соединение. По виртуальному пути проходят все сообщения от одного коммутатора к другому. Виртуальный путь объединяет несколько виртуальных каналов, проходящих по одному и тому же направлению на некотором участке сети. Этот параметр дает возможность коммутатору переключать целые группы виртуальных каналов, не затрачивая времени на анализ информации по каждому каналу в отдельности.

Виртуальный канал VC (*virtual channel*) – это тракт передачи, устанавливаемый между двумя станциями на время их взаимодействия. Виртуальный канал представляет собой фиксированный маршрут, состоящий из последовательности номеров портов коммутаторов, через которые проходят все ячейки при данном сеансе связи от одного пользователя к другому. Двухпроводные виртуальные каналы всегда однонаправленные. Для переда-

чи данных в обратном направлении используются другие номера портов.

Виртуальные каналы бывают постоянного включения и коммутируемые. Первый тип называется *постоянным виртуальным каналом* ***PVC*** *(Permanent Virtual Circuit),* а второй — *коммутируемым виртуальным каналом* ***SVC*** *(Switched Virtual Circuit).* Постоянный канал PVC представляет собой выделенную логическую цепь с заранее определенным путем, которая может иметь фиксированную полосу пропускания между двумя конечными точками. PVC используется только для соединений, которые существуют без изменений продолжительное время (несколько недель или лет). Его устанавливает оператор вручную путем ввода с пульта управления команд инициализации коммутатора.

В отличие от канала PVC, коммутируемый виртуальный канал SVC автоматически создается программным обеспечением и разрывается, как только в нем исчезает необходимость. Программное обеспечение узла сети инициирует создание канала SVC путем посылки запроса локальному коммутатору. В запросе указывается ***полный адрес*** удаленного компьютера, с которым необходимо установить канал SVC, а также параметры, определяющие ***требуемое качество соединения*** (например, пропускная способность и величина задержки). Затем узел сети ожидает, пока коммутатор ATM создаст канал и пришлет ответ на запрос.

После установления соединения коммутаторы применяют адресные таблицы, содержащие сведения о том, куда необходимо направлять ячейки. В процессе передачи ячеек по сети ATM используется следующая управляющая информация:

- адрес порта, из которого приходят ячейки;
- специальные значения в заголовках ячейки, называемые идентификаторами виртуального канала и виртуального пути.

Номера виртуальных каналов и виртуальных путей, включаемые коммутатором в заголовки ячеек перед их передачей, указываются в адресных таблицах.

Интеллектуальный постоянный виртуальный канал ATM (ATM *smart permanent virtual circuit*, ***SPVC***) объединяет в себе свойства постоянного и коммутируемого виртуальных каналов. Такой канал, как и постоянный PVC, требует ручного конфигурирования (хотя только на оконечных устройствах). Как и в коммутируемом виртуальном SVC-канале, для каждого сеанса связи с использованием интеллектуального SPVC-канала указывается индивидуальный путь к коммутатору или к тем коммутаторам, через которые должны передаваться данные. Кроме того, как и для постоянных PVC-каналов, операции создания и удаления интеллектуального SPVC-канала не вызывают задержек, поскольку этот канал сконфигурирован заранее. Подобно коммутируемым SVC-каналам, интеллектуальный SPVC-канал отказоустойчив благодаря нали-

чию альтернативных маршрутов.

Уровень адаптации ATM (*ATM Adaptation Layer*, **AAL**). В функции уровня адаптации входят задачи:

- формирование пакетов (ячеек);
- предоставление информации уровню ATM, дающей возможность устанавливать соединение с различным показателем качества услуг;
- предотвращение перегрузки на узлах.

Уровень адаптации состоит из двух подуровней.

Подуровень сегментации и реассемблирования, является нижним подуровнем *AAL.* Он выполняет разбиение (*сегментацию*) сообщения, принимаемого от протокола верхнего уровня, на ячейки ATM и формирует заголовки.

Подуровень конвергенции – верхний подуровень AAL, зависящий от класса передаваемого трафика. В соответствии с протоколом конвергенции выполняется синхронизация между передающими и принимающими узлами, контроль ошибок.

Главным свойством технологии ATM, которое отличает ее от других, является комплексная поддержка параметров качества обслуживания **QoS** (*Quality of Service*). Пользователи сети могут задать различный показатель Выбор для передачи данных любого типа небольшой ячейки фиксированного размера еще не решает задачу совмещения разнородного трафика в одной сети, а только создает предпосылки для ее решения. Для полного решения этой задачи технология ATM практикует заказ пропускной способности и качества обслуживания, аналогичного используемому в технологии Frame relay. Но если сеть Frame relay изначально была предназначена для передачи только пульсирующего компьютерного трафика, то в технологии ATM выделено 4 основных класса трафика, для которых разработаны различные механизмы резервирования и поддержания требуемого качества обслуживания.

Класс трафика, называемый также *классом услуг* (*service class*) качественно характеризует требуемые услуги по передаче данных через сеть ATM. Если приложение указывает сети, что требуется, например, передача голосового трафика, то очевидно, что особенно важными для пользователя будут такие показатели качества обслуживания, как з**адержки** и вариации задержек ячеек, существенно влияющие, на качество переданной информации – голоса или изображения, а потеря отдельной ячейки с несколькими отсчетами сигналов не столь критична, так как, например, воспроизводящее голос устройство может аппроксимировать недостающие отсчеты и качество изменится несущественно. Требования к синхронности передаваемых данных очень важны для многих приложений – не только голоса, но и видеоизображения, и наличие этих требований стало первым критерием для деления трафика на классы.

Другим важным параметром трафика, существенно влияющим на способ передачи его через сеть, является **величина пульсаций** трафика. При разработке технологии ATM выделили два различных типа трафика в отношении этого параметра: трафик с постоянной битовой скоростью **CBR** (*Constant Bit Rate*,) и трафик с переменной битовой скоростью **VBR** (*Variable Bit Rate*). К разным классам отнесены трафики, генерируемые приложениями, использующими для обмена сообщениями протоколы с установлением и без установления соединений. В первом случае данные передаются самим приложением достаточно надежно, как это обычно делают протоколы с установлением соединения, поэтому от сети ATM высокой надежности передачи не требуется. Во втором случае приложение работает без установления соединения и восстановлением потерянных и искаженных данных не занимается, что предъявляет повышенные требования к надежности передачи ячеек сетью ATM. В результате было определено четыре класса трафика, отличающихся следующими качественными характеристиками:

- наличием или отсутствием пульсации трафика, т.е. трафики с постоянной CBR или переменной VBR битовыми скоростями;
- требованием к синхронизации данных между передающей и принимающей сторонами;
- типом протокола, передающего свои данные через сеть ATM с установлением или без установления соединения (только для случая передачи компьютерных данных).

Уровень адаптации определяет 4 категории обслуживания:

- постоянная скорость передачи CBR (*Constant Bit Rate*);
- переменная скорость передачи RT-VBR (*Real Time Variable Bit Rate*) или NRT-VBR (*Non-Real Time Variable Bit Rate*);
- доступная скорость ABR (*Available Bit Rate*);
- неопределенная скорость UBR (*Unspecified Bit Rate*).

Постоянная скорость CBR используется для потребителей, чувствительных к задержкам данных (аудио-, видеоприложения). При этом гарантируется малая задержка. Однако канал используется неэффективно, так как независимо, есть ли передача или нет, канал резервируется постоянно.

При *переменной скорости* резервирование пропускной способности не происходит. Канал используется более эффективно, однако не гарантируется определенная задержка сообщения.

Доступная скорость – допускаются задержки, однако обеспечиваются допустимые длительности задержки и коэффициент потерь. Коэффициент потерь определяет, какой процент ячеек может быть потерян за время передачи.

Неопределенная скорость применяется при передачах по протоколам

TCP/IP, допускающим задержки. В сильно загруженных сетях происходит частое повторение пакетов.

Протоколы уровня адаптации AAL. Уровень адаптации реализован группой протоколов AAL1-AAL5, которые **преобразуют сообщения** протоколов верхних уровней сети ATM в ячейки ATM нужного формата. Функции этих уровней условно соответствуют функциям транспортного уровня модели OSI, например функциям протоколов TCP или UDP. Протоколы AAL при передаче пользовательского трафика работают только в конечных узлах сети, как и транспортные протоколы большинства технологий. Каждый протокол уровня AAL обрабатывает пользовательский трафик определенного класса (таблица 5.3).

Таблица 5.3 – Классы трафиков и протоколы для адаптивного уровня

Класс трафика/ протокол	Синхронизация отправителя и получателя	Частота следования битов	Режим соединения
Класс a / AAL1	необходима	постоянная	с соединением
Класс b / AAL2	необходима	переменная	с соединением
Класс c / AAL3/4 или AAL5	не нужна	переменная	с соединением
Класс d /AAL3/4 или AAL5)	не нужна	переменная	без соединения

На начальных этапах стандартизации каждому классу трафика соответствовал свой протокол AAL, который принимал в конечном узле пакеты от протокола верхнего уровня и заказывал с помощью соответствующего протокола нужные параметры трафика и качества обслуживания для данного виртуального канала.

Протокол **AAL1** предназначен для обеспечения передачи по сетям ATM **трафика с постоянной скоростью** (оцифрованный голос, видеоконференции). Другой протокол уровня адаптации – **AAL3/4** служит для обеспечения передачи по сетям ATM блоков данных. Тип AAL3/4 имеет существенную избыточность (4 байта из 48). По этой причине был введен 5-ый тип – **AAL5**. Этот уровень обеспечивает канал, ориентированный на соединение, с **переменной скоростью обмена** в широковещательном режиме при минимальном контроле ошибок (или вовсе без него). Для передачи по сетям ATM трафика локальных вычислительных сетей наиболее часто используется протокол AAL5.

Несмотря на использование на самом низком уровне ATM небольших ячеек фиксированного размера, протокол AAL5 предоставляет прикладным программам интерфейс, позволяющий пересылать большие по размеру паке-

ты переменной длины. За счет этого интерфейса для прикладных программ создается видимость, что сеть АТМ относится к разряду технологий, не требующих установки соединения между получателями. В частности, в протоколе AAL5 предусмотрено, что в каждом пакете может содержаться от 1 до 65 535 октетов данных.

При передаче данных через АТМ-соединение с помощью протокола AAL5, прикладная программа отправляет блок данных через интерфейс AAL5. При этом программа протокола AAL5 создает хвостовик (*трейлер*) пакета, разделяет пакет на части размером 48 октетов и передает каждую часть по сети АТМ в одной ячейке. На принимающем конце соединения программа протокола AAL5 собирает поступившие ячейки в пакет. Для контроля целостности частей пакета осуществляется циклическое кодирование. В случае отсутствия ошибок блок данных передается прикладной программе. Процесс разделения блока данных на ячейки и их объединение на приемном конце называется *сегментацией и последующей сборкой* ячеек АТМ *(Segmentation And Reassembly,* или *SAR)* . Формат пакета AAL5 показан на рисунке 5.31.

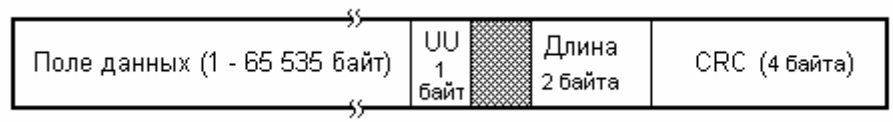


Рисунок 5.31 – Формат пакета по протоколу AAL5

В его состав кроме поля данных входит хвостовик, содержащий поле UU (*user to user*) длиной один байт, служащее для обеспечения мультиплексирования на верхних уровнях; двухбайтное поле указания длины данных; контрольную последовательность блока данных CRC длиной 4 байта. Однобайтовое поле, расположенное между полями *UU* и *Длина,* зарезервировано для использования в будущем.

Протокол AAL5 полностью соответствует принципу разбиения на уровни, поскольку в нем функции сегментации и последующей сборки ячеек отделены от механизма передачи ячеек. Уровень AAL5 относится к классу *сквозных* уровней, поскольку пакеты данных передаются непосредственно от отправителя к конечному получателю. Это означает, что модуль протокола AAL5, запущенный на компьютере получателя, передает прикладной программе получателя точную копию блока, который был получен программой протокола AAL5, запущенной на компьютере отправителя.

Для указания последней ячейки пакета в протоколе AAL5 используется значение младшего бита поля "*тип полезной нагрузки*", расположенного в заголовке ячейки АТМ. Этот бит устанавливается отправителем в единичное

значение при передаче последней ячейки и называется *битом окончания пакета.* Таким образом, программа протокола AAL5 на машине получателя собирает все входящие ячейки до тех пор, пока не обнаружит ячейку с установленным битом окончания пакета. Для описания механизмов, распознающих окончание пакета, в стандарте ATM используется специальный термин — *сходимость (convergence).* Хотя в протоколе AAL5 для определения сходимости проверяется значение только одного бита в заголовке ячейки, в остальных протоколах уровня адаптации ATM могут использоваться и другие механизмы сходимости.

5.7.3. Формат ячейки ATM

Ячейка ATM, как и любой сетевой пакет, состоит из заголовка и поля данных пакета. Для того, чтобы обеспечивать быструю обработку ячеек, их заголовок должен быть относительно коротким. Основная функция заголовка сводится к идентификации виртуального соединения. Для правильного функционирования сети должна быть обеспечена надлежащая защита заголовка. Последствием поражения заголовка ошибкой будет неверная маршрутизация пакета - т.е. однократная ошибка приведет к потере всего пакета. Для того, чтобы уменьшить эту вероятность вводится механизм защиты заголовка по принципу обнаружения или исправления ошибок.

В случае отсутствия защиты от ошибок в заголовке, то, во-первых, данный пакет не был бы доставлен получателю, т.е. имело бы место пропадание пакета в данном виртуальном соединении, а, во-вторых, этот пакет был бы доставлен не тому получателю, и у него произошла бы вставка лишнего пакета. Введение в заголовок функции обнаружения ошибок, т.е. отбрасывания пакета при возникновении в нем ошибки, приводит к тому, что ячейка не будет доставлена ошибочно, а только отброшена. Следовательно, ошибка вызовет только одну потерю. Введение функции исправления ошибок исключает выпадения и вставки пакетов. Именно на этом принципе остановились разработчики процедуры ATM.

Для упрощения реализации кодеков при наличии пакетных ошибок в технологии ATM был введен оригинальный адаптивный метод коррекции ошибок. Его суть состоит в следующем. В нормальном режиме, т.е. когда ошибок нет, заголовок обрабатывается в режиме исправления однократных ошибок. В случае, если декодер обнаружил ошибку и исправил ее исходя из возможностей исправления только одной ошибки, то он сразу переключается в режим обнаружения (но не исправления) ошибок. Это сделано для того, что в том случае, если ошибка была пакетная (а это значит, что исправление было неправильным), пакеты, в которых обнаружена ошибка, будут уничто-

жены. Если же ошибка была все-таки однократная, то следующий пакет скорее всего будет безошибочным, что и отметит механизм обнаружения и вернется в режим с коррекцией ошибок.

Формат ячейки ATM изображен на рисунке 5.32. Первые 4 бита ячейки отведены под поле управления потоком **GFC** (*Generic Flow Control*). Затем следуют идентификатор виртуального пути **VPI** (*Virtual Path Identifier)* и идентификатор виртуального канала **VCI** (*Virtual Circuit Identifier*).

<table>
<tr><th>Бит 8</th><th>Бит 7</th><th>Бит 6</th><th>Бит 5</th><th>Бит 4</th><th>Бит 3</th><th>Бит 2</th><th>Бит 1</th><th>Байты</th></tr>
<tr><td colspan="4">Управление потоком (GFC)</td><td colspan="4">Идентификатор виртуального пути (VPI)</td><td>1</td></tr>
<tr><td colspan="4">Идентификатор виртуального пути VPI (продолжение)</td><td colspan="4">Идентификатор виртуального канала (VCI)</td><td>2</td></tr>
<tr><td colspan="8">Идентификатор виртуального канала VCI (продолжение)</td><td>3</td></tr>
<tr><td colspan="4">Идентификатор виртуального канала VCI (продолжение)</td><td colspan="2">Тип данных (PTI)</td><td colspan="2">Приоритет потери ячейки (CLP)</td><td>4</td></tr>
<tr><td colspan="8">Контроль ошибок в заголовке (HEC)</td><td>5</td></tr>
<tr><td colspan="8" rowspan="4">Данные пакета (Playload)</td><td>6</td></tr>
<tr><td>...</td></tr>
<tr><td></td></tr>
<tr><td>53</td></tr>
</table>

Рисунок 5.32 – Формат ячейки ATM

Как уже упоминалось ранее, при использовании технологии, ориентированной на установку соединения между отправителем и получателем, каждому каналу присваивается уникальный целочисленный идентификатор, который назначается узлом сети при выполнении операций ввода-вывода, а также при закрытии канала. Однако в системах, ориентированных на установку соединения, уникальность такого идентификатора не является глобальной. Другими словами, идентификатор канала является аналогом дескриптора ввода-вывода, который возвращается программой операционной системы при открытии файла.

Как и дескриптор ввода-вывода, идентификатор канала является действительным только при открытом канале. Кроме того, значением идентификатора канала можно воспользоваться только в пределах одного сегмента соединения между коммутаторами. Следует отметить, что идентификаторы одного и того же канала, полученные узлами сети на обоих концах сегмента виртуального канала, обычно отличаются друг от друга. Например, отправи-

тель может использовать для идентификации канала число *15,* в то время как получатель — число *27.* Поэтому при передаче пакета от одного узла сети к другому каждый коммутатор, расположенный по пути прохождения пакета, заменяет в нем номер идентификатора канала на текущий. Весь идентификатор соединения в целом часто называют *идентификатором VPI/VCI.* Эта пара занимает 24 бита: 8-битов для *идентификатора виртуального пути* и *VP* и 16-битов для *идентификатора виртуального канала VCI*.

Индикатор типа данных **PTI** (*Playload Type Indicator*) занимает 3 бита после поля VCI. Цифровые значения индикатора от 0 до 3 указывают на то, что в ячейке передаются данные пользователя, значения 4 и 5 – управляющая информация, а 6 и 7 – зарезервированы.

Поле приоритета потери ячейки **CLP** (*Cell Loss Prioryty*), занимающее последний бит четвертого байта, используется для управления потоком данных. Ячейки с полем CLP=0 являются для сети высокоприоритетными, а ячейки с CLP=0 – низкоприоритетными.

HEC (*Header Error Check*) – поле проверки ошибки заголовка. С его помощью можно восстановить единичную или идентифицировать многократную ошибку заголовка ячейки. Поле содержит проверочные элементы, вычисленные для заголовка ячейки с использованием кода Хемминга. Они позволяют не только обнаруживать, но и исправлять все одиночные ошибки, а также некоторые двойные.

В конце ячейки размещено **поле данных** пакета (*Playload*) размером 48 байтов.

5.7.4. Сетевые интерфейсы и доступ к сети ATM

Сети ATM строятся на основе стандартных блоков, роль которых выполняют специализированные электронные коммутаторы. Хотя сетевой компьютер можно подключить к коммутатору ATM посредством электрического кабеля, в большинстве случаев для повышения скорости передачи данных используют волоконно-оптические линии связи. Как правило, к одному ATM-коммутатору можно подключить ограниченное количество компьютеров. Для создания крупной сети приходится соединять между собой несколько коммутаторов SW (рисунок 5.33,а).

Для компьютера, подключенного к сети ATM, совокупность коммутаторов представляет собой однородную сеть. Поэтому для него создается видимость единой физической сети, к которой подключено большое количество компьютеров.

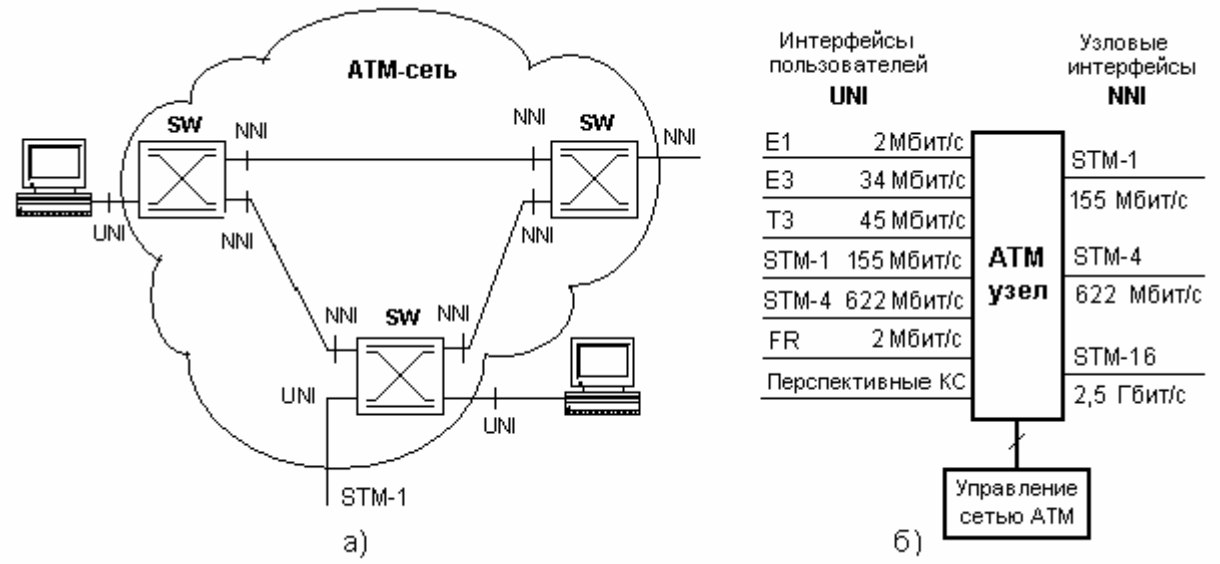


Рисунок 5.33 – Построение компьютерной сети на основе коммутаторов ATM а), интерфейсы ATM-узла, б)

Принцип соединения коммутаторов между собой незначительно отличается от подключения компьютера к коммутатору. В частности, данные между коммутаторами передаются на более высокой скорости, чем между сетевым компьютером и коммутатором; кроме того, при этом используются другие протоколы. Коммутаторы ATM соединяются между собой через узловые интерфейсы **NNI** (*Network Node Interface*). Узловые интерфейсы оснащены преимущественно интерфейсами STM-1 (скорость передачи 155 Мбит/с) или **STM-4** (622 Мбит/с). Постоянно растущий сетевой трафик требует наличия более быстродействующих интерфейсов. В настоящее время уже имеется возможность использовать интерфейсы **STM-16** со скоростью передачи 2,5 Гбит/с.

Оконечные устройства подключаются к узлу ATM через пользовательские интерфейсы **UNI** (*User Network Interface*). У абонентов имеется широкий выбор пользовательских интерфейсов на основе PDH- и SDH-технологий с электрическими и оптическими сигналами в диапазоне скоростей от 2 до 155 Мбит/с (рисунок 5.33,б). Для некоторых применений, например, телевидения с высоким разрешением (HDTV), могут быть предоставлены интерфейсы со скоростью 622 Мбит/с.

Собственно коммутатор ATM состоит из двух коммутаторов: коммутатора виртуальных путей и коммутатора виртуальных каналов. Эта особенность организации ATM обеспечивает дополнительное увеличение скорости обработки ячеек. ATM коммутатор анализирует значения, которые имеют идентификаторы виртуального пути и виртуального канала у ячеек, поступающих на его входной порт и направляет эти ячейки на один из выходных

портов. Определение номера выходного порта коммутатор осуществляет на основании динамически создаваемой таблицы коммутации.

Для передачи дейтаграмм по сети ATM в протоколе IP используется протокол AAL5. Перед отсылкой данных отправитель должен установить с получателем виртуальный канал (постоянный PVC или переменный SVC) и согласовать используемый тип протокола уровня адаптации ATM, например AAL5. В процессе передачи дейтаграммы модулю протокола AAL5, отправитель указывает также пару идентификаторов VPI/VCI, определяющих сам канал. Модуль протокола AAL5 создает трейлер дейтаграммы, разбивает ее на ячейки и передает их по сети. На приемном конце виртуального канала модуль протокола AAL5 выполняет сборку ячеек в пакет данных, вычисляет код CRC и проверяет целостность пакета, извлекает дейтаграмму из пакета и пересылает ее модулю протокола IP.

На практике в протоколе AAL5 используется поле длины пакета, размер которого составляет 16 битов. Это позволяет отсылать в одном пакете до 65 535 (64К) байтов. Несмотря на это в семействе протоколов TCP/IP наложены ограничения на размер дейтаграмм, которые можно посылать по сети ATM. В стандарте максимальный размер IP-дейтаграммы в сетях ATM устанавливается равным 9180 байтов. Как и в любой сетевой технологии, если размер отправляемой дейтаграммы превышает установленный в сети размер максимальной единицы передачи данных MTU, модуль протокола IP разбивает дейтаграмму на фрагменты и пересылает каждый фрагмент модулю протокола AAL5. Таким образом, по умолчанию модуль протокола AAL5 принимает, передает и доставляет дейтаграммы, размер которых не превышает 9180 байтов.

Маршрутизация в ATM отличается от аналогичных процессов в сетях с коммутацией пакетов. Сети ATM в основном ориентированы на соединение. Маршрутизация в сети осуществляется двумя способами. В первом случае, в коммутаторах ATM, выполняющих ретрансляцию ячеек, располагаются локальные адресные таблицы. Ячейки транспортируются по уже выбранному маршруту через коммутаторы ATM в соответствии со значениями идентификаторов виртуального пути и виртуального канала.

Во втором случае, *маршрут* записывается в заголовках блоков данных. Это происходит на основе глобальных адресных таблиц при вводе данных в коммуникационную сеть. Благодаря этому, осуществляется *самомаршрутизация*. Блоки в рассматриваемом случае могут двигаться по различным маршрутам, что имеет много общего с методом передачи дейтаграмм.

5.8. IP-технологии в глобальных сетях

5.8.1. Способы реализации глобальных IP-сетей

В настоящее время существует несколько способов построения глобальных IP-сетей. Первый, самый простой способ, предполагает соединение IP-маршрутизаторов посредством выделенных линий или каналов первичных цифровых глобальных сетей (PDH/SDH/DWDM). Такие сети получили название "чистые" IP-сети (рисунок 5.34). Вариант сети, в которой IP-маршрутизаторы используют каналы, организованные сетью SDH (SONET), получил название **пакетной сети, работающей поверх SONET** (Packet Over SONET, POS).

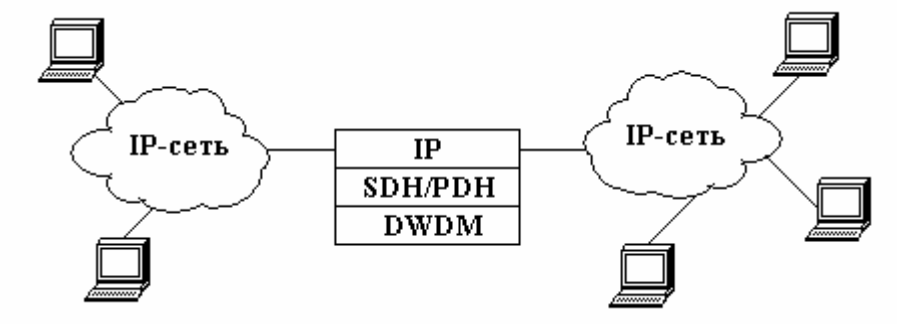


Рисунок 5.34 - Структура "чистой" IP-сети

Для того чтобы маршрутизаторы "чистой" IP-сети могли работать на цифровых каналах, в них должен использоваться один из протоколов канального уровня. Такие протоколы обеспечивают управление потоком данных, взаимную аутентификацию удаленных устройств и регламентируют процедуру согласования параметров обмена данными на канальном и сетевом уровнях. В настоящее время для этих целей применяются преимущественно два типа канальных протоколов: HDLC и PPP (см. пп. 2.5.3 и 5.2.5). В ряде случаев может использоваться устаревший протокол SLIP.

Помимо упомянутых способов транспортировки данных, в глобальных IP-сетях маршрутизаторы на выделенных каналах в последнее время все чаще используют высокоскоростные варианты сети Ethernet: Fast- Gigabit- или 10G- Ethernet. В середине 90-х годов наиболее популярной структурой глобальной IP-сети стала многоуровневая архитектура, в которой под уровнем IP в качестве транспортных магистралей используются сети ATM и Frame relay. Применение на двух уровнях сетей с коммутацией пакетов, использующих разные принципы работы, позволили относительно просто пе-

редавать мультимедийный трафик и широко применять методы обеспечения качества обслуживания QoS.

Дальнейшим шагом на пути интеграции IP-технологии с технологиями виртуальных каналов явилась технология MPLS (*MultiProtocol Label Switching*). Она занимает промежуточное положение между уровнем IP и уровнями таких технологий как ATM, Frame Relay или Ethernet, объединяя их в единую технологию.

Технология MPLS обеспечивает построение магистральных сетей, имеющих практически неограниченные возможности масштабирования, повышенную скорость обработки трафика и гибкость с точки зрения организации дополнительных сервисов. Кроме того, интеграция сети IP и ATM позволила поставщикам услуг не только сохранить средства, инвестированные в оборудование асинхронной передачи, но и получить дополнительную выгоду из совместного использования этих протоколов.

5.8.2. Функционирование IP-сетей поверх ATM/FR

При построении IP-сети поверх сетей ATM или FR между сетевым и канальным уровнями функционирует сеть ATM или Frame Relay. Благодаря более высокой скорости и разветвленной шкале параметров качества обслуживания QoS, в качестве промежуточного слоя преимущественно используется сеть ATM (рисунок 5.35).

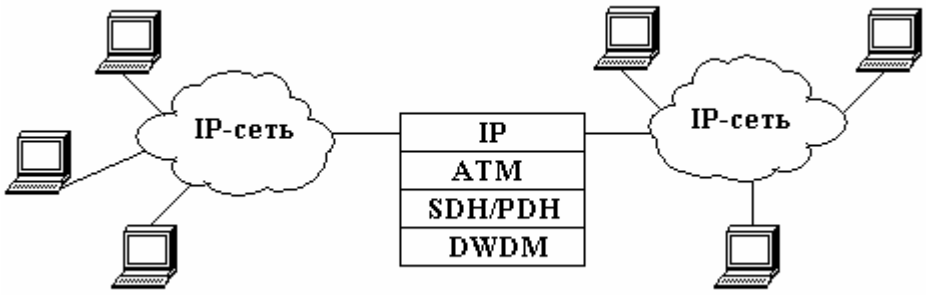


Рисунок 5.35 – Структура IP-сети, наложенной на ATM

В такой сети IP-маршрутизаторы соединяются с коммутаторами ATM. Для этой цели маршрутизаторы IP-сети снабжаются портами, поддерживающими технологию ATM. После установления виртуального соединения между маршрутизаторами IP-сети эти маршрутизаторы могут пользоваться такими соединениями как обычными цифровыми линиями и отправлять данные порту на соседний (по отношению к виртуальному каналу) маршру-

тизатор. Сеть ATM является прозрачной для IP-маршрутизаторов, а сеть IP – **наложенной** (*оверлейной*) по отношению к сети ATM.

Чтобы протокол IP мог корректно работать, в его распоряжении должны быть сведения о соответствии между IP-адресами узлов и адресами виртуальных каналов ATM, с помощью которых достижим соответствующий IP-адрес. Поскольку сеть ATM не поддерживает режим широковещательных запросов, таблица соответствия адресов не может быть создана автоматически. Поэтому администратор IP-сети вынужден вручную выполнить конфигурирование каждого интерфейса маршрутизатора, задавая таблицу соответствия для всех номеров входящих и исходящих виртуальных каналов, соединенных с данным интерфейсом.

Если многослойная сеть IP/ATM должна передавать трафик различных классов с соблюдением параметров QoS для каждого класса, то соседние маршрутизаторы связывают несколькими виртуальными каналами, минимум по одному для каждого класса.

Наложенная IP-сеть может также использовать режим коммутируемых виртуальных каналов (SVC) для передачи IP-трафика. Такой режим целесообразно применять для пульсирующих потоков, существующих в течение небольших промежутков времени. Для того чтобы маршрутизаторы могли использовать режим SVC, необходимо задавать отображение IP-адресов не на номера виртуальных каналов, а на ATM-адреса конечных точек сети ATM, то есть ATM-адреса интерфейсов маршрутизатора. Эта функция разрешения адресов выполняется администратором сети вручную.

5.8.3. MPLS-технология

MPLS (*MultiProtocol Label Switching*) представляет собой сетевую технологию, при которой IP-пакеты передаются по виртуальным каналам сети ATM или FR. Она объединяет технологии виртуальных каналов и коммутации пакетов. В соответствии с этой технологией в IP-пакеты на основе их приоритета добавляется специальный идентификатор – **метка**. Помеченные пакеты передаются по специально созданным LSP-путям коммутации по меткам (*Label Switching Path*). Все передаваемые в MPLS-сети пакеты объединяются в группы (**классы**), перемещаемые по одному и тому же маршруту и с одним и тем же качеством обслуживания. Формируемые в MPLS-сети группы пакетов получили название "**класс эквивалентности продвижения данных**" – FEC (*Forwarding Equivalence Class*). Для каждого класса эквивалентности продвижения данных FEC определяется маршрут через сеть узлов – LSR-маршрутизаторов (*Label Switching Router*). В каждый FEC-класс

включаются пакеты, имеющие одинаковые требования по качеству обслуживания. Технология многопротокольной коммутации с помощью меток MPLS представляет собой сетевую технологию, ориентированную на соединения.

MPLS-сеть (рисунок 5.36) состоит из множества соединенных между собой узлов, поддерживающих технологию MPLS. Узлы такой сети называются "LSR-маршрутизаторы". Эти узлы обладают свойством коммутировать пакеты на основании специальных идентификаторов – ***меток***, добавленных к каждому пакету. Метки определяют поток пакетов между двумя оконечными точками, а в случае групповой рассылки – между оконечным пунктом-источником и группой оконечных пунктов. Кроме LSR-маршрутизаторов в сети установлены **пограничные коммутирующие по меткам маршрутизаторы** LER (*Label switch Edge Routers*). MPLS-сеть может состоять из нескольких независимых MPLS-доменов. Домены соединяются между собой посредством пограничных маршрутизаторов LER.

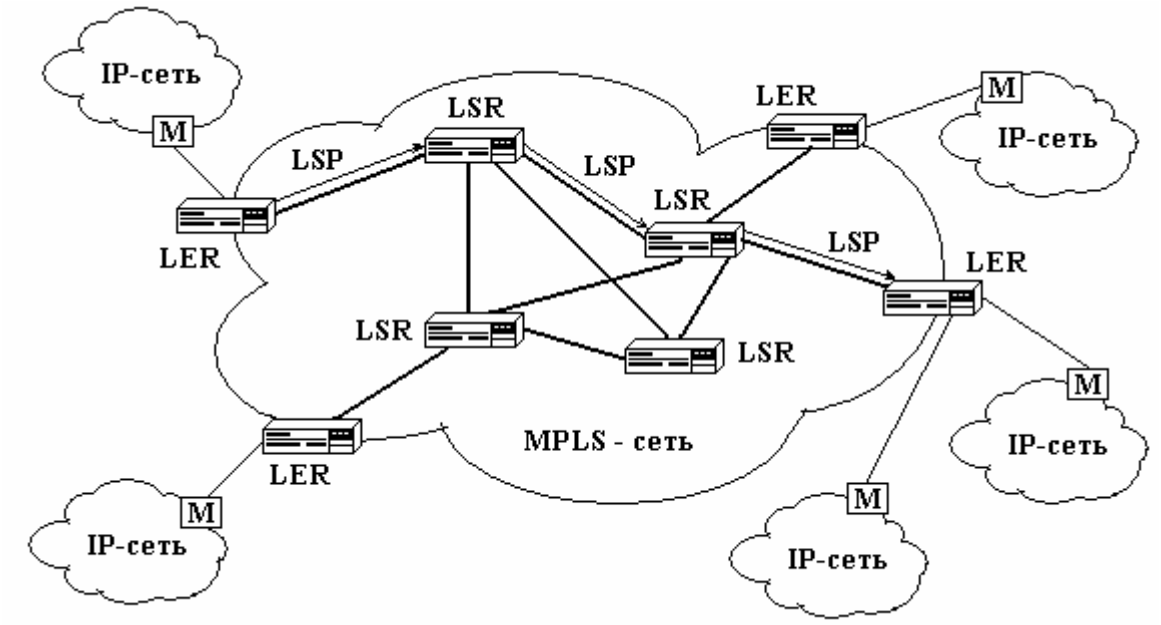


Рисунок 5.36 - Структура MPLS-сети

Соединенные между собой LSR-маршрутизаторы MPLS-сети образуют некоторую разветвленную многоуровневую иерархическую структуру, по которой прокладываются пути коммутации по меткам LSP. Эти пути формируются предварительно в соответствии с топологией межсетевых соединений. Каждый путь представляет собой однонаправленный виртуальный канал. Поэтому для передачи сообщений между двумя LER-узлами устанавливаются два пути коммутации по меткам, т.е. по одному в каждом направле-

нии. Если результирующий путь по MPLS-сети состоит из нескольких участков, относящихся к соответствующим уровням иерархии, то для каждого участка задается своя метка. Группа меток составного пути объединяется в **стек меток**. Детальнее о стеке меток и его использовании сказано ниже.

Для определения путей доставки пакетов LSP и установления параметров качества обслуживания вдоль этих путей применяется либо типовой протокол внутренней маршрутизации (OSPF), используемый для обмена сведениями между узлами о достижимости и маршрутах, либо сетевой администратор явно указывает маршруты и назначает им соответствующие значения меток. В качестве альтернативы для определения маршрута и установления меток между соседними LSR-маршрутизаторами может использоваться специальный протокол распределения меток **LDP** (*Label Distribution Protocol*) либо протокол резервирования ресурсов **RSVP** (*Resource reservation Protocol*). Последний предоставляет возможности разнородным пользователям резервировать сетевые ресурсы под собственные потребности.

Пограничные устройства LER принимают поток пакетов, входящий в MPLS-домен от других сетей в виде стандартных IP-пакетов. LER путем анализа заголовка IP-пакета (IP-адресов отправителя и получателя, идентификаторов IP-протокола, метки потока протокола IPv6 и других полей), добавляют к каждому из пакетов необходимую метку и направляют по соответствующему пути к выходному LER-маршрутизатору через несколько промежуточных LSR-маршрутизаторов. При этом пакет продвигается не на базе IP-адреса, а на основе метки.

Каждый маршрутизатор LSR содержит таблицу, в которой установлено соответствие между записью "входной интерфейс, входная метка" и строкой "префикс адреса получателя, выходной интерфейс, выходная метка". Получая пакет, LSR по номеру интерфейса, на который пришел пакет, и по значению присоединенной к пакету метки определяет для него выходной интерфейс. Причем значение префикса адреса применяется лишь для построения таблицы и в самом процессе коммутации не используется. Старое значение метки заменяется новым, содержавшимся в поле «выходная метка» таблицы, и пакет отправляется к следующему устройству на пути LSP.

Вся операция требует лишь одноразовой идентификации значений полей в одной строке таблицы. Это занимает гораздо меньше времени, чем сравнение IP-адреса отправителя с наиболее длинным адресным префиксом в таблице маршрутизации, которое используется при традиционной маршрутизации.

Как и в других технологиях, использующих технику виртуальных каналов, метки имеют только локальное значение в пределах каждого узла LER и LSR. Поэтому LSR-маршрутизатор удаляет из пакета входную метку

и перед последующей передачей присоединяет к нему соответствующую новую выходную метку.

Выходной узел LER удаляет метку и передает пакет в следующую IP-сеть в стандартном IP-формате. Таким образом, технология MPLS является прозрачной для остальных IP-сетей, соединенных с MPLS-сетью.

Если многослойная сеть IP/ATM должна передавать трафик различных классов с соблюдением параметров QoS для каждого класса, то соседние маршрутизаторы должны быть связаны несколькими виртуальными каналами, по одному для каждого класса. Причем маршрутизаторам должна быть задана политика классификации пакетов, позволяющая отнести передаваемый пакет к определенному классу.

5.8.4. Форматы MPLS-заголовков и стек меток

Заголовки MPLS, добавляемые в кадры канального уровня соответствующих протоколов, содержат четыре поля (рисунок 5.37). Поле "**Метка**" занимает 20 бит и используется для выбора соответствующего пути коммутации по меткам LSP. Затем следует поле, указывающее "**Класс обслуживания**" CoS (*Class of Service*)**.** Оно занимает 3 бита и используется для указания класса трафика, требующего определенного показателя CoS. Следующее однобитовое поле является **указателем дна стека меток S.** Последнее 8-битовое поле определяет **время жизни** пакета. Оно дублирует аналогичное поле IP-пакета.

Рисунок 5.37 – Заголовок кадра MPLS-сети

Последовательность MPLS-заголовков объединяется в стек. Поэтому всегда имеется метка, расположенная в вершине стека, и метка, находящаяся на дне стека. Стек меток позволяет создавать систему объединенных (*агрегированных*) путей LSP с любым количеством уровней иерархии. Кадр, перемещаемый по объединенному пути должен включать столько заголовков

MPLS, сколько уровней иерархии содержит путь. Продвижение MPLS-кадра осуществляется только на основе метки, расположенной в данный момент в вершине стека.

Над метками выполняются следующие операции, задаваемые в поле действий таблицы продвижения: помещение метки в стек "Push", замена текущей метки "Swap" и удаление из стека верхней метки "Pop".

5.8.5. Применение технологии MPLS

В настоящее время существует несколько областей практического применения MPLS-технологии, позволяющие повысить эффективность использования компьютерных сетей. Одно из простейших применений – *ускорение продвижения пакетов* сетевого уровня, перемещающихся по маршрутам, составленных на основе стандартных внутренних шлюзовых протоколов IGP. При использовании технологии MPLS-IGP пути коммутации по меткам прокладываются в соответствии с существующей топологией IP-сетей, связанных MPLS-доменом, и не зависит от интенсивности трафика между этими сетями. Протокол MPLS-IGP ускоряет продвижение пакетов за счет сокращения просматриваемых таблиц продвижения по меткам, так как обычная таблица маршрутизации содержит намного больше записей, чем таблица продвижения. Эффективность функционирования MPLS заметно повышается на крупных магистральных сетях, где маршрутизаторы оперируют с таблицами в несколько десятков тысяч записей.

Другим типовым назначением сетей MPLS является *обеспечение гарантированной средней пропускной способности* на основе применения методов рационального распределения трафика TE (Traffic Engineering). В этом состоит основное отличие технологии MPLS-TE от технологии MPLS-IGP, так как последняя обеспечивает прокладку путей коммутации по меткам только на основе топологии составной сети, а интенсивность трафика при этом не учитывается. Кроме того, в отличие от MPLS-IGP в технологии MPLS-TE пути коммутации по меткам, называемых TE-туннелями, не прокладываются автоматически, а задаются администратором сети. TE-туннели подобны постоянным виртуальным каналам PVC в технологиях ATM и Frame relay.

Третья область применения MPLS-технологии – *построение частных виртуальных сетей* VPN. Создание таких сетей возможно за счет разграничения трафика, без необходимости обязательного шифрования данных. Трафик данного предприятия прозрачно проходит через объединенную сеть, причем можно легко отделять этот трафик от остальных пакетов объединенной сети, предоставляя гарантии производительности и безопастности.

Таким образом, основным преимуществом MPLS-технологии, с точки зрения пользователя, является качество обслуживания QoS, а следующим по важности - упрощение защиты и процедуры доступа к частным виртуальным сетям VPN.

5.9. Выводы по разделу

1. Глобальные сети на начальном этапе развития широко использовали каналы связи общего пользования: телефонные каналы тональной частоты и первичные широкополосные каналы, образованные аналоговой аппаратурой уплотнения с частотным разделением каналов.

2. С развитием и внедрением в системы связи цифровых технологий в глобальных компьютерных сетях повсеместно стали применяться цифровые каналы на основе плезиохронной цифровой иерархии PDH, образованные аппаратурой уплотнения с временным разделением каналов и импульсно-кодовой модуляцией (ИКМ).

3. Глобальные сети используются преимущественно как транзитная транспортная система, обеспечивающая доставку сообщений между абонентами сети. Такие сети предоставляют в основном услуги трех нижних уровней эталонной модели взаимодействия открытых систем *OSI*.

4. При создании высокоскоростных сетей передачи данных, речи, видео и мультимедиа в территориальных и крупных корпоративных сетях в качестве транспортной среды все чаще применяются выделенные цифровые каналы первичной сети связи, созданные на основе новых коммуникационных технологий, таких как цифровая синхронная иерархия *SDH* и технология плотного волнового мультиплексирования DWDM.

5. В аналоговых телефонных сетях составной канал между абонентами имеет ограниченную полосу пропускания. Для передачи двоичных импульсов постоянного тока по такому каналу необходимо перенести их спектр в полосу прозрачности канала. Эта процедура осуществляется путем модуляции несущего колебания. На приемной станции принимаемый сигнал демодулируется. Модуляция и демодуляция сигналов реализуется в модемах.

6. Для соединения модема с компьютером МККТТ был разработан стандартный интерфейс (стык) С2, в состав которого входит около двух десятков цепей. В современных модемах соединение внешнего модема с компьютером осуществляется преимущественно через последовательный интерфейс RS-232C.

7. В состав типового модема входят передатчик и приемник прямого канала и приемо-передатчик обратного (служебного) канала связи. Служебный канал используется для подтверждения (квитирования) правильности приема или запроса повторной передачи блока.

8. Современные телефонные модемы являются мультистандартными и могут работать со скоростями передачи от 300 бит/с до 56 кбит/с, причем, скорость модуляции составляет от 300 до 3000 бод. На низких и средних скоростях в прямом канале применяется двухпозиционная частотная или фазовая модуляция, на скоростях свыше 2400 бит/с используется многопозиционная фазовая или амплитудно-фазовая модуляция. В обратном канале сигналы квитирования во всех типах модемов модулируются по частоте и передаются со скоростью 75 бод. Для повышения помехоустойчивости в модемах используется циклическое и треллис-кодирование. Количество позиций сигнала достигает 512. Во всех типах современных модемов применяется процедура сжатия входных данных.

9. Для повышения скорости на участке "последней мили" – абонентской линии, соединяющей сетевой компьютер с АТС, используется цифровая технология DSL. На АТС устанавливается специальная аппаратура DSL-доступа, позволяющая "обойти" коммутационное и каналообразующее оборудование телефонной станции, сужающее полосу пропускания абонентского тракта до 3,4 кГц. Благодаря этому удается достичь скорости передачи на участке "провайдер-пользователь" в несколько Мбит/с. Существуют асимметричные и симметричные, высокоскоростные и адаптивные DSL-модемы на широкий диапазон скоростей от 1,5 до 51 Мбит/с.

10. В современных DSL-модемах используется цифровая обработка сигналов, коррекция частотных характеристик кабельной линии на передающей и приемной сторонах, комбинирование различных помехоустойчивых способов кодирования (циклическое, Рида-Соломона, перестановочное).

11. Для подключения к сети Интернет посредством модема через телефонный канал и создания соединения с использованием протокола TCP/IP на канальном уровне применяется преимущественно протокол PPP. Он реализует процедуры: инкапсуляции данных, позволяющей на одном и том же канале использовать различные сетевые протоколы; управления соединением LCP, служащей для установки, конфигурации и тестирования соединения; управления сетью с использованием протокола NCP позволяющего PPP-соединению применять протоколы различных сетевых уровней.

12. Для обмена файлами между компьютерами, подключенными к телефонной сети посредством модемов, применяются протоколы передачи файлов, такие как Xmodem, Xmodem-CRC и 1k, Ymodem и Zmodem. Наиболее совершенным является Zmodem. Он совместим со всеми предыдущими типами модемных протоколов и позволяет динамически адаптироваться к

состоянию канала, возобновлять прерванную передачу файла с места сбоя, осуществлять групповую передачу файлов.

13. Цифровая плезиохронная иерархия представляет собой транспортную среду с широким диапазоном цифровых каналов от 64 кбит/с до 140 Мбит/с. Недостатком технологии PDH является невозможность извлечения на промежуточных узлах сети потоков данных со скоростью 64 кбит/с или 2 Мбит/с, входящих в групповой поток со скоростью 140 Мбит/с без полного демультиплексирования и удаления выравнивающих битов.

14. Этот недостаток был ликвидирован в синхронной цифровой иерархии SDH. Она позволяет гибко формировать цифровые каналы широкого диапазона скоростей: от единиц мегабит до десятков гигабит в секунду. Каналы SDH относятся к классу полупостоянных. Формирование канала происходит по инициативе оператора сети SDH. Пользователи же лишены такой возможности, в связи с чем каналы SDH обычно применяются для передачи достаточно устойчивых во времени потоков.

15. SDH обладает гибкой иерархической схемой мультиплексирования цифровых потоков разных скоростей. Это позволяет вводить в магистральный канал и выводить из него пользовательскую информацию любого поддерживаемого технологией уровня скорости, не демультиплексируя поток в целом. Основной информационной единицей технологии SDH является синхронный транспортный модуль, представляющий собой кадр, передаваемый со скоростью 155,52 Мбит/с.

16. Селективный доступ к отдельным группам данных без демультиплексирования общего транспортного потока в SDH достигается посредством байт-ориентированного мультиплексирования и непосредственной адресации начала каждого из контейнеров. В контейнере имеются указатели, определяющие текущее положение контейнера в структуре более высокого уровня. Это позволяет мультиплексору находить положение пользовательских данных "на лету", без полного демультиплексирования, как это производится в PDH.

17. Одной из самых перспективных технологий создания цифровых каналов для глобальных сетей является технология спектрального мультиплексирования WDM основанная на способности оптического волокна одновременно передавать свет различных длин волн (цветов) без взаимной интерференции. Каждая длина волны представляет отдельный оптический канал в волоконно-оптической линии связи (ВОЛС). Число каналов на входе и выходе одного волокна может достигать 32, а в отдельных случаях и более со скоростями передачи в каждом их них 2,5 Гбит/с.

18. К одной из перспективных для создания глобальных компьютерных сетей относится сеть интегрального обслуживания ISDN, представляющая собой цифровую сеть с коммутацией каналов. Основу иерархии состав-

ляет цифровой канал со скоростью передачи 64 кбит/с, называемый B-каналом.

19. Старейшим представителем цифровых сетей с коммутацией пакетов является сеть X.25. Сети X.25 наилучшим образом подходит для передачи трафика низкой интенсивности, характерного для терминалов. В сети осуществляется сборка нескольких низкоскоростных старт-стопных потоков байтов от алфавитно-цифровых терминалов в пакеты, передаваемые по сети и направляемые компьютерам для обработки.

20. Сеть X.25 состоит из узлов коммутации пакетов УКП, расположенных в различных географических точках и соединенных высокоскоростными выделенными каналами. Асинхронный старт-стопный терминал подключается к сети коммутации пакетов через устройство сборки/разборки пакетов данных. Пакет данных состоит обычно из 128 байтов, которые передаются по адресу, содержащемуся в пакете. Стандарт X.25 определяет первые три уровня эталонной модели взаимодействия открытых систем: **физический** (интерфейс X.21); **канальный** (протокол *HDLC*); **сетевой** (пакетный).

21. Усовершенствованная технология быстрой коммутации пакетов переменной длины реализована в сети Frame relay. В технологии исключены многие функции учета и контроля, используемые в сетях X.25, в связи с отсутствием их необходимости по причине высокой помехозащищенности оптоволоконных линий связи. Узлы коммутации выполняют только основные функции канального уровня, связанные с получением и дальнейшей передачей (ретрансляцией) кадров без их преобразования.

22. Высокую скорость в сети FR обеспечивает постоянный виртуальный канал, благодаря чему известен весь маршрут между конечными точками, в связи с чем устройства Frame relay избавлены от ряда постоянных процедур: фрагментации, восстановления, выбора оптимального маршрута. Большинство процедур обработки и управления передачей информации в FR осуществляется оконечным оборудованием данных с помощью протоколов более высокого уровня.

23. ATM – асинхронный режим передачи – представляет собой технологию построения и функционирования сети с пакетной коммутацией и высокоскоростным асинхронным режимом передачи. Сеть ориентирована на виртуальное (логическое) соединение и может использоваться для передачи данных, аудио- и видеоинформации. Передача информации осуществляется небольшими пакетами фиксированной длины (53 байта), называемых ячейками.

24. Для передачи ячеек по сетям ATM от источника к адресату сначала устанавливается соединение отправителя с получателем информации в процессе которого создается виртуальный канал или виртуальный путь. Со-

единение выполняется АТМ-коммутаторами, осуществляющими дополнительно функцию оптимизации маршрута между взаимодействующими абонентами.

25. Сеть АТМ делится на три уровня: физический, уровень АТМ и адаптации. На физическом уровне АТМ использует иерархию скоростей сети SDH. Кроме этого, имеются другие физические интерфейсы (T1/E1, T2/E2, FDDI). На уровне АТМ реализуются функции канального уровня. Уровень адаптации обеспечивает запрашиваемое качество услуг, предоставляет информацию уровню АТМ, необходимую для установления соединения, а также предотвращает перегрузки на узлах сети.

26. Заголовок ячейки состоит из пяти байтов и содержит идентификаторы виртуального пути и канала, команды для управления потоком, а также индикатор типа данных, биты приоритета и байт проверочной последовательности заголовка.

27. Коммутаторы АТМ соединяются между собой через узловые интерфейсы NNI, которые оснащены преимущественно интерфейсами STM-1 со скоростью передачи 155 Мбит/с, STM-4 со скоростью 622 Мбит/с или более высокого уровня.

В последние годы наиболее популярной структурой глобальной IP-сети стала многоуровневая архитектура, в которой под уровнем IP в качестве транспортных магистралей используются сети АТМ и Frame relay. Применение на двух уровнях сетей с коммутацией пакетов, использующих разные принципы работы, позволили относительно просто передавать мультимедийный трафик и широко применять методы обеспечения качества обслуживания QoS.

28. Дальнейшим шагом на пути интеграции IP-технологии с технологиями виртуальных каналов явилась технология MPLS (*MultiProtocol Label Switching*). Она занимает промежуточное положение между уровнем IP и уровнями таких технологий как АТМ, Frame Relay или Ethernet, объединяя их в единую технологию. Она объединяет технологии виртуальных каналов и коммутации пакетов. В соответствии с этой технологией в IP-пакеты на основе их приоритета добавляется специальный идентификатор – **метка**. Помеченные пакеты передаются по специально созданным LSP-путям коммутации по меткам

29. Более детально ознакомиться с особенностями построения и проектирования модемов в компьютерных сетях можно в литературе [28,34], с архитектурой плезиохронных и синхронных иерархий – в [1,7,15,16]. Вопросы построения цифровых сетей X.25, Frame Relay, ATM, ISDN достаточно полно освещены в [7,9,15,16, 30].

5.10. Контрольные вопросы

1. По какой причине для передачи данных используют аналоговые каналы тональной частоты?
2. Какие сети передачи сигналов относят к первичным и какие линии и каналы связи входят в состав первичной сети?
3. В чем состоят особенности построения глобальных компьютерных сетей и кто является типичными абонентами этих сетей?
4. Какие действия включает процедура установления соединения в коммутируемых телефонных сетях?
5. Назовите иерархию аналоговых каналов связи и их основные параметры и характеристики.
6. С какой целью в аналоговых сетях применяются модемы и почему компьютерные данные нельзя непосредственно передавать по аналоговым линиям?
7. Какая несущая частота должна быть в модема, предназначенного для передачи данных по первичному широкополосному каналу связи?
8. Что общего в стыке С2 и интерфейсе RS-232C и в чем они различаются?
9. Каким образом приемник на удаленной стороне определяет, что источник начал передачу?
10. С какой целью в модемах введен канал квитирования и каковы его параметры?
11. Каким образом удалось достичь в телефонных каналах ТЧ скорости передачи 56 кбит/с, если полоса пропускания канала не превышает 3400 Гц?
12. Зачем в модемах стали применять сверточный код с исправлением одиночных ошибок, если в нем применялся и применяется циклический код, способный обнаруживать кроме одиночных и пачки ошибок?
13. При каких условиях и за счет чего модем V.90 может обеспечить передачу данных со скоростью 56 кбит/с?
14. Каким образом DSL-модему удается осуществлять передачу данных по абонентским линиям со скоростью около одного Мбит/с, если лучший модем V.90 мог передавать по таким линиям максимум 56 кбит/с?
15. С какой целью в DSL-модемах применяется многочастотная модуляция OFDM?
16. Дайте сравнительную характеристику способов защиты от ошибок, используемых в ADSL-модеме?
17. Какие функции определяют протоколы передачи файлов, по какому признаку получатель определяет вид протокола?
18. Каковы функции протокола PPP и как реализован процесс конфигура-

ции в протоколе PPP?

19. Дайте сравнительную характеристику плезиохроннннной и синхронной цифровой иерархии.
20. Покажите рассчетным путем, какие скорости передачи должны обеспечиваться в потоках E1 и T1.
21. За счет чего возможно выделение отдельных цифровых потоков в сетях SDH без демультиплексирования общего транспортного потока?
22. Рассчитайте, с какой скоростью передаются полезные данные в кадре STM-1.
23. В чем состоит суть и преимущество спектрального мультиплексирования в волоконно-оптических линиях связи?
24. Назовите преимущества цифровой сети интегрального обслуживания ISDN по сравнению с традиционными сетями с коммутацией каналов?
25. Какие каналы связи используются для передачи сигналов в цифровых сетях интегрального обслуживания?
26. Расскажите о принципах и особенностях функционирования сети X.25.
27. Какие уровни эталонной модели взаимодействия открытых систем реализованы в сетях X.25?
28. Как организована адресация в сетях X.25?
29. За счет чего обеспечивается высокая скорость и помехоустойчивость передачи данных с ретрансляцией кадров?
30. Почему в сетях Frame Relay убрали приоритезацию трафика? Каким образом исключаются перегрузки в таких сетях?
31. Чем обусловлен сравнительно малый размер пакета (ячейки) в сети ATM?
32. Как реализована процедура установления соединения в сетях ATM?
33. Расскажите о сетевых уровнях и их функциях в технологии ATM.
34. Каково различие между виртуальным путем и виртуальным каналом?
35. В чем состоит особенность маршрутизации в сетях ATM?
36. Поясните понятия "чистая" и "наложенная" IP-сеть.
37. С какой целью производят объединение IP-технологии с технологиями виртуальных каналов?
38. С какой целью в IP-пакеты добавляются метки? Поясните функционирование MPLS-сети.

Раздел 6

БЕЗОПАСНОСТЬ КОМПЬЮТЕРНЫХ СЕТЕЙ

Почему в процессе проектирования объединенных сетей не были изначально предприняты меры по их безопасности? Чем отличается политика безопасности от политики вообще? Что представляют собой модели защиты информации в сетях? Можно ли обнаружить вторжение злоумышленника в компьютерную систему? Каким образом можно узнать пароль доступа? Почему злоумышленника интересуют сведения об используемой операционной системе? В чем состоит суть компьютерной атаки? Что такое брандмауэр? На эти и многие другие вопросы Вы найдете ответ, ознакомившись с материалом данного раздела.

6.1. Общая характеристика проблемы безопасности в компьютерных сетях

6.1.1. Уязвимости компьютерных сетей и их причины

С учетом постоянно растущего значения компьютерных сетей сегодня нельзя пренебрежительно относиться к проблеме их безопасности, так как проникновение злоумышленника в сеть подчас может оказаться губительным для самого существования организации. В большинстве случаев овладение сетью означает то же самое, что и возможность перехвата почтовых сообщений, финансовых данных, перенаправление потока информации на неавторизованные системы.

Вопрос безопасности всегда стоял перед компьютерными сетями, но сегодня как никогда растет осознание того, насколько важна безопасность компьютерных сетей в корпоративных инфраструктурах. В настоящее время для каждой корпоративной сети необходимо иметь четкую политику в области безопасности. Эта политика разрабатывается на основе анализа рисков, определения критически важных ресурсов и возможных угроз. Политикой безопасности можно назвать и простые правила использования сетевых ресурсов, и детальные описания всех соединений и их особенностей. Стандартное определение, изложенное в RFC 2196, которое считается несколько узким и ограниченным, излагает политику безопасности следующим образом: "**Политика безопасности** – это формальное изложение правил, которым должны подчиняться лица, получающие доступ к корпоративной техно-

логии и информации".

Политика безопасности включает правила пользования компьютером и действия при обнаружении нарушения системы безопасности. Так в правилах пользования компьютером оговаривается следующее: на центральном сервере должны ежедневно создаваться резервные копии данных; запрещается отключать антивирусное обеспечение; запрещается отключать брандмауэр; пользовательские пароли должны содержать 6…8 знаков и иметь как минимум один небуквенный символ и т.д. В действиях персонала при обнаружении им нарушений системы безопасности предписывается, что обо всех нарушениях следует немедленно информировать службу безопасности и следовать ее указаниям, при подозрениях о наличии вируса необходимо тотчас же сообщить об этом в службу поддержки сети.

Злоумышленники часто проверяют сети организаций на возможность проникновения в них путем методического сканирования систем на наличие уязвимых мест. При этом они зачастую используют средства автоматического зондирования, т.е. программы, которые последовательно просматривают (*сканируют*) все компьютеры, присоединенные к сети организации. Такие действия злоумышленников приводят к нарушению нормальной работы организаций и предприятий, а также наносят ущерб их репутации. В некоторых случаях организации вынуждены были временно отсоединиться от Интернета, и потратить значительные средства для решения возникших проблем с конфигурацией хостов и сети. Организации, которые не осведомлены или игнорируют проблемы сетевой безопасности, подвергают себя большому риску быть атакованными сетевыми злоумышленниками. Даже те организации, в которых безопасности уделяется значительное внимание, могут подвергаться риску из-за появления новых уязвимых мест в сетевом программном обеспечении или упорства некоторых злоумышленников.

Одна из основных причин уязвимости сетей – **игнорирование** разработчиками при проектировании сети Интернет **требований безопасности**. Это обстоятельство объясняется тем, что главным условием в ходе реализации объединенной сети было обеспечение удобства в процессе обмена информацией при проведении научных исследований. Применение сетевых технологий для коммерческих целей в гигантских масштабах открыло неограниченный доступ к сети широкому кругу пользователей. Появились люди, пытающиеся выявить уязвимые места компьютерных сетей и получить выгоду от проникновения к чужим информационным ресурсам, злонамеренно (или по легкомыслию) затруднить или нарушить функционирование сети.

Другими причинами уязвимости сетей являются следующие:

- **уязвимость сервисов TCP/IP** – ряд сервисов TCP/IP являются небезопасными и могут быть скомпрометированы технически подготовленными злоумышленниками; особенно уязвимы сервисы для улучшения управления

сетью, использующиеся в локальных компьютерных сетях;

- **легкость наблюдения за каналами и магистралями** – большинство трафика Интернета не зашифровано; электронная почта, пароли и передаваемые файлы могут быть перехвачены, после чего злоумышленники могут использовать пароли для проникновения в системы;
- **отсутствие политики безопасности** – многие сети могут быть сконфигурированы по незнанию таким образом, что будут позволять доступ к ним со стороны Интернета, многие сети допускают использование большего числа сервисов TCP/IP, чем это требуется для деятельности их организации, при этом они не пытаются ограничить доступ к информации о своих компьютерах, которая может помочь злоумышленникам проникнуть в сеть;
- **сложность конфигурирования** – средства управления доступом в хостах зачастую являются сложными в настройке и контроле за ними; неправильно сконфигурированные средства часто приводят к неавторизованному доступу.

Проникновение в компьютерную систему осуществляется в форме атак. **Атака** – это событие, при котором злоумышленник ("хакер", либо "взломщик") пытается проникнуть внутрь чужой компьютерной системы или совершить по отношению к ней какие-либо злоупотребления.

Существует большое число средств для системных администраторов, позволяющих повысить безопасность систем и обеспечить улучшенные возможности по протоколированию. Такие средства могут проверять пароли, журналы с информацией о соединениях, обнаруживать изменения системных файлов или обеспечивать другие меры безопасности, которые помогут администраторам обнаружить деятельность злоумышленников и проникновения в их системы. Для успешной реализации мер безопасности в сети администраторы должны быть хорошо осведомлены в путях возможного проникновения злоумышленников в сеть, способах перехвата и дешифрования паролей, сценариях нарушения работы сетей, а также знать способы противодействия нарушению функционирования сети.

6.1.2. Категории информационной безопасности

Информация с точки зрения информационной безопасности обладает следующими категориями:

- *конфиденциальность* – конкретная информация доступная только тому кругу лиц, для кого она предназначена; нарушение этой категории определяется как хищение либо раскрытие информации;

- *целостность* – информация существует в исходном виде, т.е. при ее хранении или передаче не было произведено несанкционированных изменений; нарушение этой категории называется фальсификацией сообщения;
- *аутентичность* – источником информации является именно то лицо, которое заявлено как ее автор; нарушение этой категории также называется фальсификацией, но уже автора сообщения;
- *апеллируемость* – гарантия того, что при необходимости можно будет доказать, что автором сообщения является именно заявленный человек, и не может являться никто другой; отличие этой категории от предыдущей в том, что при подмене автора, кто-то другой пытается заявить, что он автор сообщения, а при нарушении апеллируемости – сам автор пытается отказаться от своих слов, подписанных им однажды (категория, довольно часто применяемая в электронной коммерции).

В отношении технических характеристик информационных систем применяются следующие параметры и показатели:

- *надежность* – вероятность того, что система ведет себя в штатном и аварийных режимах так, как заложено при ее проектировании;
- *контроль доступа* – гарантия того, что различные группы лиц имеют различный доступ к информационным объектам, и эти ограничения доступа постоянно выполняются;
- *контролируемость* – возможность проведения полноценной проверки любого компонента программного комплекса в любой момент времени;
- *контроль идентификации* – вероятность того, что клиент, подключенный в данный момент к системе, является именно тем, за кого себя выдает;
- *устойчивость к умышленным сбоям* – вероятность того, что при умышленном внесении ошибок в пределах заранее оговоренных норм система будет вести себя так, как установлено техническим заданием на проектирование.

6.1.3. Абстрактные модели защиты информации

Одной из первых моделей защиты является **модель Биба** (*Biba*), опубликованная в 1977 г. Согласно ей все субъекты и объекты предварительно разделяются по нескольким уровням доступа, а затем на их взаимодействия накладываются следующие ограничения: 1) субъект не может вызывать на исполнение субъекты с более низким уровнем доступа; 2) субъект не может модифицировать объекты с более высоким уровнем доступа.

Под субъектами понимают активные объекты (пользователи, программы), а под объектами – пассивные (пароли, иные данные). К атрибутам доступа

относят всевозможные действия субъектов над объектами: чтение, изменение, дополнение (без чтения!), поиск, исполнение. Уровни безопасности представляют собой определенное дополнение к субъектам и объектам, определяющее возможность их взаимодействия. У объекта только один уровень безопасности. Уровень безопасности субъекта делится на 2 части.

- *Класс доступа* – определяет возможность доступа субъекта к определенному классу информации: "Совершенно секретно", "Секретно", "Конфеденциально" и "Для общего пользования". Субъект с высоким уровнем доступа имеет доступ ко всем последующим нижележащим уровням.
- *Категория доступа* – задают возможные области безопасности. В отличие от предыдущей части субъект может обладать одними категориями доступа из ряда наличных и не иметь доступа к иным.

Компьютерная система осуществляет переходы: текущее состояние → запрос → решение → последующее состояние. Работа системы регулируется набором свойств и правил. Модель полностью формализована математически.

Модель Гогена-Мезигера (*Goguen-Meseguer*), разработанная в 1982 году, основана на теории автоматов. Согласно ей система может при каждом действии переходить из одного разрешенного состояния только в несколько других. Субъекты и объекты в данной модели защиты разбиваются на группы – домены, и переход системы из одного состояния в другое выполняется только в соответствии с так называемой таблицей разрешений, в которой указано какие операции может выполнять субъект, скажем, из домена C над объектом из домена D. В данной модели при переходе системы из одного разрешенного состояния в другое выполняются действия, обеспечивающие общую целостность системы.

Сазерлендская (от англ. *Sutherland*) **модель защиты**, опубликованная в 1986 году, делает акцент на взаимодействии субъектов и потоков информации. Так же, как и в предыдущей модели, здесь используется машина состояний с множеством разрешенных комбинаций состояний и некоторым набором начальных позиций. В данной модели исследуется поведение множественных композиций функций перехода из одного состояния в другое.

Важную роль в теории защиты информации играет **модель защиты Кларка-Вильсона** (*Clark-Wilson*), опубликованная в 1987 году и модифицированная в 1989 г. Данная модель основана на строгом оформлении прав доступа субъектов к объектам. Но в данной модели впервые исследована защищенность третьей стороны, то есть стороны, поддерживающей всю систему безопасности. Эту роль в информационных системах обычно играет программа-супервизор. Кроме того, в модели Кларка-Вильсона идентификация субъекта производится не только перед выполнением его команды, но и повторно после выполнения. Это позволило снять проблему подмены автора в момент между его идентификацией и собственно командой. Модель Клар-

ка-Вильсона считается одной из самых совершенных в отношении поддержания целостности информационных систем.

6.2. Пути и способы вторжения нарушителей в компьютерную сеть

6.2.1. Пути проникновения нарушителей в сеть

Физическое вторжение. Если нарушитель имеет физический доступ к компьютеру (т.е. он может использовать клавиатуру или часть системы), то возможно проникновение его в систему. Методы могут быть различными: от использования специальных привилегий, которые имеет консоль, до возможности использования части системы и снятия винчестера (и чтения/записи его на другой машине).

Системное вторжение. Нарушитель уже имеет учетную запись в системе как пользователь с невысокими привилегиями. Если в системе не установлены самые последние патчи защиты, у нарушителя есть хороший шанс попытаться совершить определенную атаку для получения дополнительных административных привилегий.

Удаленное вторжение. Злоумышленник пытается проникнуть в систему через сеть с удаленной машины. Нарушитель действует без каких-либо специальных привилегий. Существует несколько видов такой хакерской деятельности. Например, нарушитель тратит гораздо больше времени и усилий, если между ним и выбранной машиной установлен межсетевой защитный экран.

Одним из распространенных способов доступа к системе является **взлом пароля.** Нарушители пытаются использовать все слабые стороны этого вида защиты. К ним относятся следующие.

Действительно слабые пароли. Большинство людей используют в качестве паролей свои имена, имена своих детей, супруга/супруги, любимого(ой) или модели машины. Есть также пользователи, которые в качестве пароля выбрали слово "пароль" или "password" или вообще никакого слова. В целом существует более 30 возможностей, которыми может воспользоваться нарушитель для подбора паролей.

Атака по словарю. Потерпев неудачу в случае вышеуказанной атаки, нарушитель может затем попытаться использовать "атаку по словарю". В этом случае злоумышленник пытается использовать программу, которая формирует в качестве пароля каждое возможное слово, приведенное в словаре. Атаки по словарю могут осуществляться либо путем неоднократных регистраций в атакуемой системе, либо путем сбора шифрованных паролей и

попыток найти им незашифрованную пару. Для этих целей нарушители, как правило, имеют копию русского и английского словарей, а также словари других иностранных языков. Все они применяют дополнительные словари, как с именами, так и со списками наиболее распространенных паролей.

Подбор пароля. Аналогично атаке по паролю нарушитель старается использовать все возможные комбинации символов. Короткий пароль, состоящий из 4-х букв в нижнем регистре, может быть взломан за несколько секунд (приблизительно полмиллиона возможных комбинаций). Длинный семизначный пароль, состоящий из символов в нижнем и верхнем регистре, а также чисел и знаков препинания (10 триллионов комбинаций) может потребовать не один месяц для взлома, в расчете на то, что устройство перебора может осуществлять миллион комбинаций в секунду.

Перехват незащищенного трафика. На традиционном Ethernet возможно разместить перехватчик (*sniffer*), чтобы перехватывать весь трафик на сегменте. В настоящее время это становится все более и более трудным, так как многие организации используют коммутируемые сети, состоящие из многих изолированных сегментов. Однако в коммутируемых сетях возможно установить сниффер на сервере, тем самым получить доступ ко всей циркулирующей в сети информации. Например, злоумышленник может не знать пароля определенного пользователя, но перехват пароля, передаваемого по протоколу Telnet позволяет ему получить доступ к удаленного узлу.

Каким же образом нарушители получают пароли? Для этого они могут использовать следующие пути.

Перехват открытого текста. Большое количество протоколов (Telnet, FTP, HTTP) выполняют передачу незашифрованных паролей при обмене по сети между клиентом и сервером. Нарушитель с помощью анализатора протоколов может "слушать" сеть в поисках таких паролей. Никаких дальнейших усилий не требуется; нарушитель может начать немедленно использовать эти пароли для регистрации в системе (сети). Примером анализатора протоколов является программа **dsniff**, которая обеспечивает возможность сбора различных паролей, передаваемых через локальную сеть. Используя программу dsniff, можно собрать пользовательские пароли служб FTP, Telnet, SMTP, HTTP, POP и ряда других.

Перехват зашифрованного текста. Большинство протоколов, однако, использует некоторое шифрование паролей. В этих случаях нарушителю потребуется провести атаку по словарю или "подбор пароля" для того, чтобы попытаться провести дешифрование. Заметим, что клиенты сети не знают о присутствии нарушителя, поскольку он является полностью пассивным и ничего не передает по сети. Взлом пароля не требует того, чтобы передавать что-нибудь в сеть, собственный компьютер нарушителя используется только для аутентификации пароля законного пользователя.

Повторное использование. В некоторых случаях нарушителям нет необходимости расшифровывать пароль. Они могут повторно передать зашифрованный пароль в процессе аутентификации.

Кража файла с паролями. Вся база данных о паролях пользователя обычно хранится в одном файле на диске. В ОС UNIX этим файлом является /etc/passwd (или некоторая разновидность этого файла), а в ОС Windows NT это SAM-файл. В любом случае, как только нарушитель получает этот файл, он может запускать программы взлома (описанные выше) для того, чтобы найти слабые пароли внутри данного файла.

Наблюдение. Одна из традиционных проблем при защите паролей заключается в том, что пароли должны быть длинными и трудными для расшифровки. Однако часто такие пароли очень трудно запомнить, поэтому пользователи их записывают. Нарушители могут часто обыскивать рабочие места пользователей для того, чтобы найти пароли, записанные чаще всего на небольших клочках бумаги. Они могут также подглядывать пароли, стоя за спиной пользователя.

В общем случае в любой процедуре вторжения злоумышленника в информационную систему можно выделить пять стадий (рисунок 6.1).

Рисунок 6.1 – Процедура реализации типового вторжения

На начальной стадии нарушитель осуществляет сбор информации об объекте атаки, чтобы на ее основе спланировать дальнейшие этапы вторжения. Этим целям может служить, например, информация о типе и версии ОС, установленной на хостах информационной системы; список пользователей, зарегистрированных в системе; сведения об используемом прикладном ПО и т. д. Стадия сбора информации может подразделяться на *внешнюю* и *внутреннюю разведку*.

При **внешней разведке** нарушители собирают как можно больше информации об атакуемой системе, ничем себя не выдавая. Они могут делать это, собирая доступную информацию, или маскируясь под обычного пользователя. На этой стадии их невозможно обнаружить. Нарушитель будет высматривать "кто есть кто", чтобы собрать как можно больше информации о потенциальной жертве. Нападающий может пройтись по DNS-таблицам (применяя программы *nslookup*, *dig* или другие утилиты, используемые для

работы с DNS), чтобы найти имена машин исследуемой сети.

На стадии **внутренней разведки** нарушитель использует более мощные способы для получения информации, но по-прежнему не делает ничего вредного. Он может пройти через все Web-страницы информационной системы и посмотреть CGI-скрипты, которые очень часто подвергаются хакерским атакам. Возможен запуск утилиты *ping* для обнаружения активных компьютеров в сети. В процессе разведки возможно сканирование UDP/TCP-портов на намеченных для атаки компьютерах для того, чтобы определить доступные сервисы. Нападающий может запустить утилиты типа *rpcinfo*, *showmount*, *snmpwalk* и т.д. для того, чтобы определить, какие службы являются доступными. В данный момент нарушитель ведет "нормальную" деятельность в сети и нет ничего, что могло бы быть классифицировано как нарушение.

На этапе вторжения нарушитель получает несанкционированный доступ к ресурсам тех хостов, на которые совершается атака. Нарушитель пересекает границу и начинает использовать возможные уязвимости на выделенных компьютерах. Он может попробовать скомпрометировать CGI скрипт, посылая команды *shell* в полях входных данных. Нарушитель может попытаться использовать эксплоиты (*exploits*) или хорошо известные уязвимости "переполнения буфера", посылая большое количество данных, либо начать проверку учетных записей с легко подбираемыми (или пустыми) паролями. **Эксплоиты** – это программы, использующие ошибки в каком-то конкретном программном обеспечении. Они применяются для получения доступа к компьютеру, главным образом с правами суперпользователя.

После этапа вторжения наступает стадия атакующего воздействия. В течение этой стадии реализуются цели, ради которых и предпринималась атака. Например, если хакер смог получить доступ к учетной записи обычного пользователя, то затем он будет пытаться совершать дальнейшие действия для получения доступа к учетной записи супервизора *root/admin*. Затем может последовать нарушение работоспособности информационной системы, кража конфиденциальной информации, удаление или модификация данных и т. д.

В следующей стадии злоумышленник стремится развить атаку, т.е. расширить круг жертв атаки, чтобы продолжить несанкционированные операции на других составляющих объекта нападения. Стадия завершения вторжения характеризуется стремлением атакующего выполнить действия, направленные на удаление следов его присутствия в информационной системе путем исправления журналов регистрации. Хакер будет пытаться использовать систему в качестве опорной площадки для проникновения в другие системы или компьютеры, поскольку большинство сетей имеют незначительное число средств для защиты от внутренних атак. Ниже рассмотрены

более подробно способы реализации этих стадий.

6.2.2. Способы сканирования ресурсов сети

Последовательное просматривание (англ. *sweeping*) ресурсов компьютерной сети производится нарушителем с целью разведки и предварительного сбора информации на первой стадии вторжения.

К простейшему способу последовательного просматривания (*сканирования*) относится процедура, обозначаемая **Ping sweeps**. В течение этого процесса сканирования утилитой *ping* просматривается диапазон IP-адресов объекта нападения с целью определения активных компьютеров. Более сложные сканеры используют другие процедуры, например **SNMP sweep**.

Для зондирования открытых **TCP-портов** в поисках сервисов, которые может использовать нарушитель, он применяет процедуру TCP-сканирования. Сеансы сканирования могут использовать обычные TCP-соединения. Существуют так называемые скрытые сеансы сканирования, применяющие наполовину открытые соединения (с тем, чтобы защитить их от регистрации в журналах) или FIN-сеансы сканирования (никогда не открывают порт, но тестируют, если что-то прослушивается). Сеансы сканирования могут быть последовательными, случайными, либо сконфигурированы по перечню портов. Процесс сканирования **UDP**-портов несколько сложнее, в связи с тем, что UDP-протокол относится к дейтаграммным протоколам, т.е. без установления виртуального соединения. Суть **UDP-сканирования** заключается в том, чтобы послать "мусорный" UDP-пакет к намеченному порту. Большинство машин будут реагировать с помощью ICMP-сообщения "*destination port unreachable*", указывающего, что на данном порту нет проверяемого сервиса. Однако многие компьютеры подавляют ICMP-сообщения, поэтому злоумышленник не в состоянии осуществлять очень быстрое UDP-сканирование.

Для идентификации используемой в сети **операционной системы** (ОС) нарушители производят посылку некорректных ICMP- или TCP-пакетов. Стандарты обычно устанавливают, каким образом компьютеры должны реагировать на легальные пакеты, поэтому машины имеют тенденцию быть единообразными в своей реакции на допустимые входные данные. Однако стандарты упускают реакцию на недопустимые входные данные. Таким образом, уникальные действия каждой ОС на недопустимые входные данные позволяют злоумышленнику понять, под чьим управлением функционирует выбранный компьютер. Этот тип деятельности имеет место на нижнем уровне (вроде скрытых сеансов TCP-сканирования), на котором анализируемые системы не регистрируют события.

Злоумышленнику известно, что существует программа **Nmap,** позволяющая администраторам сканировать отдельные хосты и целые сети, определять поддерживаемые типы сервиса и другие параметры. Nmap поддерживает множество методов сканирования: UDP, TCP connect(), TCP SYN (half open), ftp proxy (bounce attack), ICMP (ping sweep), FIN, ACK sweep, SYN sweep, IP Protocol, и др. Кроме обычного сканирования программа может определять тип операционной системы удаленного хоста, выполнять скрытое сканирование, параллельное сканирование, детектирование фильтров и ряд других действий.

О наличии межсетевого защитного экрана в системе злоумышленник может узнать по признакам его переконфигурирования. Для этого он посылает большой поток ***ping***-пакетов на хост и отмечает, что через некоторое время доступ прекратился (*ping* не проходит). По этому признаку атакующий может сделать вывод, что система обнаружения вторжений провела переконфигурацию межсетевого экрана, установив новые правила запрета ping на хост. Однако есть способы обойти эту защиту. Злоумышленник, атакуя сеть, может задавать в качестве адреса отправителя IP-адреса известных фирм (атака – *ipspoofing*). В ответ на поток ping-пакетов механизм переконфигурирования межсетевого экрана закрывает доступ на сайты этих фирм. В результате возникают многочисленные телефонные звонки пользователей "закрытых" компаний в службу поддержки сайта, и администратор вынужден отключить механизм переконфигурации (чего и добивается злоумышленник).

6.2.3. Точечные атаки и их сценарии

Зачастую гораздо проще нарушить функционирование сети или системы, чем на самом деле получить к ней доступ. Сетевые протоколы типа TCP/IP были разработаны для применения в открытом и доверенном сообществе пользователей, и его текущая версия 4 унаследовала все слабые места своих предшественников. Кроме того, многие операционные системы и сетевые устройства имеют различные изъяны в используемой реализации сетевого стека, что значительно снижает их способность противостоять компьютерным атакам.

В настоящее время наиболее распространенными являются точечные атаки типа **DoS** (*Denial of Service*) и распределенные атаки **DDoS** (*Distributed DoS*), направленные на отказ в обслуживании сетевых запросов. Атаки DoS отличаются от компьютерных атак других типов. Они не предназначены для получения доступа к атакуемой сети или для извлечения из этой сети какой-либо информации. Атака DoS делает объект нападения недоступным для обычного использования за счет превышения допустимых пределов функ-

ционирования сети, операционной системы или приложения. В случае использования некоторых серверных приложений (таких как Web- или FTP-сервер) атаки DoS могут заключаться в занятии всех соединений, доступных для этих приложений, и держать их в занятом состоянии, не допуская обслуживания обычных пользователей.

Таким образом, атака DoS нарушает или полностью блокирует обслуживание легитимных пользователей, сетей, систем или других ресурсов. DoS-атака относится к точечным (сосредоточенным), так как поступает с одного источника (точки). В случае распределенной DDoS-атаки, нападение осуществляется из множества источников, распределенных в пространстве и зачастую принадлежащим разным сетям.

Большинство атак DoS опирается не на программные ошибки или бреши в системе безопасности, а на общие слабости системной архитектуры. Некоторые атаки сводят к нулю производительность сети, переполняя ее нежелательными и ненужными пакетами или сообщая ложную информацию о текущем состоянии сетевых ресурсов.

Нарушение функционирования системы при DoS-атаках осуществляется преимущественно по таким причинам: насыщение пропускной способности сети, захват системных ресурсов, генерирование ошибочных команд, подмена маршрутов.

Насыщение пропускной способности. Наиболее коварной формой DoS-атак является насыщение пропускной способности (*bandwidth consumption*). По существу, взломщики могут заполнить всю доступную полосу пропускания определенной сети. Это можно осуществить и в локальной сети, однако чаще всего злоумышленники захватывают ресурсы удаленно. Для реализации такой атаки используется два сценария.

Сценарий 1. Взломщик может насытить сетевое подключение целевой системы, воспользовавшись более широкой полосой пропускания собственной системы. Такой сценарий вполне возможен, если злоумышленник обладает сетевым подключением типа T1/E1 (1,544 / 2,048 Мбит/с) или более быстрым, и лавинно заполняет сетевое соединение с пропускной способностью 56 или 128 кбит/с. Этот тип атак не ограничивается возможностью применения к низкоскоростным сетевым соединениям. На практике встречались ситуации, когда взломщики получали доступ к сетям с полосой пропускания более 100 Мбит/с. Для атаки на Web-узел и насыщения его канала взломщику достаточно иметь канал T1 или E1.

Сценарий 2. Взломщик усиливает атаку DoS, вовлекая в процесс насыщения целевого сетевого соединения несколько узлов. Используя другие узлы для усиления атаки DoS, злоумышленник, имея аппаратуру с невысокой скоростью передачи, может насытить пропускную способность даже 100 Мбит/с.

Захват системных ресурсов. Атака, приводящая к недостатку ресурсов (*resource starvation*), отличается от предыдущей атаки тем, что она направлена на захват системных ресурсов, таких как центральный процессор, память, диски или другие системные процессы. Зачастую взломщик обладает легитимным доступом к ограниченному количеству системных ресурсов. Однако он предпринимает попытку захватить и дополнительные ресурсы. Вследствие этого система или законные пользователи будут испытывать недостаток в совместно используемых ресурсах. Атаки такого типа обычно приводят к недоступности ресурса, и следовательно, к краху системы, переполнению файловой системы или зависанию процессов.

Ошибки программирования (*programming flaw*) заключаются в неспособности приложения, операционной системы или логической схемы обрабатывать исключительные ситуации. Обычно эти ситуации возникают при передаче уязвимому элементу несанкционированных данных. Взломщики, многократно передавая пакеты, в которых не учитываются рекомендации документов RFC, пытаются определить, способен ли сетевой стек справиться с этими исключениями или это приведет к панике ядра (*kernel panic*), либо краху всей системы. Для определенных приложений, которым требуются пользовательские входные данные, взломщики будут передавать строковые данные длиной в тысячи строк. Если программой используется буфер фиксированной длины, скажем, 128 байт, то злоумышленники попробуют сгенерировать условие переполнения буфера и вызвать крах приложения. Иногда взломщики могут также заставить процессор выполнить привилегированные команды. Печально известная атака под названием Pentium f00f основывалась на том, что пользовательский процесс, выполнив некорректную инструкцию 0xf00fc7c8, приводил к краху любой операционной системы.

Атаки на маршрутизаторы и серверы DNS. Атаки DoS на маршрутизаторы заключаются в изменении или внесении новых записей таблицы маршрутизации, что приводит к прекращению обслуживания легитимных систем или сетей. Большинство протоколов маршрутизации, такие как RIP версии 1 (*Routing Information Protocol*) и BGP (*Border Gateway Proto*col), не имеют вообще или используют слабые алгоритмы аутентификации. Это предоставляет злоумышленникам возможность изменять маршруты путем указания ложных IP-адресов. В результате таких атак трафик целевой сети маршрутизируется через сеть взломщика или в несуществующую сеть.

Атаки DoS, направленные на серверы DNS, также являются достаточно чувствительными. Большинство таких атак приводит к кэшированию на целевом сервере фиктивных адресов. Когда сервер DNS выполняет обратный поиск, взломщик может перенаправить его на требуемый узел или в некоторых случаях в "черную дыру". Существуют несколько типов подобных атак, которые приводят к тому, что большие узлы в течение продолжительного

времени оказываются недоступными.

Атака с помощью переполнения пакетами SYN. До того как атака Smurf не стала такой популярной, наиболее разрушительной считалась атака SYN Flood, использующая переполнение сервера пакетами SYN. Как уже упоминалось в разделе 4, инициализация соединения TCP представляет собой процесс, состоящий из трех шагов (рисунок 6.2). В обычной ситуации пакет SYN отсылается с определенного порта станции А на конкретный порт станции Б, который находится в состоянии ожидания запроса соединения LISTEN (*Слушает*). После приема пакета SYN система Б передает станции А пакет подтверждения синхронизации SYN/ACK. В этот момент потенциальное соединение системы Б переходит в состояние SYN_RECV – ожидания приема пакета-квитанции от станции А. Если процесс проходит нормально, станция А передает обратно пакет АСК и соединение переходит в состояние ESTABLISHED (*Создано*).

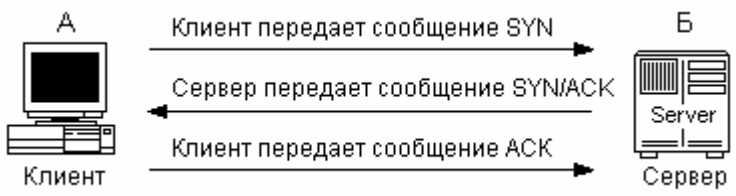


Рисунок 6.2 - Процедура согласования параметров при установлении TCP-соединения

Описанный механизм хорошо работает в большинстве случаев, однако некоторые из его изъянов позволяют взломщику сгенерировать условие DoS. Проблема заключается в том, что большинство систем заранее выделяет некоторое количество ресурсов при установке потенциального соединения, т.е. соединения, которое еще не полностью установлено. Несмотря на то, что многие системы могут поддерживать тысячи параллельных соединений с определенным портом (например, 80), достаточно нескольких десятков потенциальных запросов на соединение, чтобы израсходовать все доступные ресурсы. Именно этот механизм и применяется взломщиком для атаки с помощью переполнения пакетами SYN.

В начале атаки SYN взломщик тоже передает пакет SYN с системы А системе Б, однако при этом в качестве адреса источника указывает ложный адрес несуществующего узла. После этого система Б посылает пакет SYN/ACK по ложному адресу. Если узел по этому адресу существует, то системе Б обычно обратно отсылается пакет разъединения соединения RST, поскольку этот узел не инициировал его установку. Однако следует учесть то, что взломщик наверняка выбрал недостижимую систему. Следовательно, после того, как система Б отправила пакет SYN/ACK, она никогда не полу-

чит ответного пакета RST от системы A. Это потенциальное соединение останется в состоянии SYN_RECV и будет помещено в очередь на установку соединения. Из этой очереди потенциальное соединение может быть удалено лишь после истечения выделенного промежутка времени. Этот промежуток времени в каждой системе различен, однако он не может быть меньше 75 секунд, а в некоторых реализациях протокола IP минимальный интервал может достигать 23 минут. Поскольку очередь на установку соединения обычно имеет небольшой размер, взломщику достаточно отправить несколько пакетов SYN с интервалом 10 секунд, чтобы полностью заблокировать определенный порт.

Ситуация усугубляется вследствие того, что для ее успешной реализации достаточно небольшой полосы пропускания. Так для нарушения работоспособности промышленного Web-сервера злоумышленнику достаточно модемной линии связи 14,4 кбит/с. Кроме этого, подобная атака является скрытой, поскольку взломщики при рассылке пакетов SYN используют ложный исходный адрес. Это значительно затрудняет идентификацию источника нападения.

6.2.4. Распределенные атаки DoS путем "зомбирования"

В случае распределенной атаки DoS ее источником является несколько узлов различных компьютерных сетей (рисунок 6.3). Такой сценарий может быть реализован путем взлома функционирующих в Интернете компьютерных систем. Первый шаг любого злоумышленника, решившего прибегнуть к атаке DDoS, заключается во взломе максимального количества узлов и получении на них административных привилегий.

На этой фазе злоумышленник сканирует различные подсети и на основе анализа полученных ответов идентифицирует используемые операционные системы и версии активизированных служебных программ. Применяя программу *exploits* или другие программы, использующие существующие слабые места в целевой системе, он пытается получить привилегированные права для доступа к системе и установить в ней DDoS-агенты. Кроме того, злоумышленник может установить так называемые *root kits*, которые удаляют следы вторжения и маскируют существование DDoS-агентов.

Более прямолинейной процедурой установки DDoS-агентов является создание и рассылка троянцев, например, по электронной почте. К еще одному способу установки DDoS-агентов относится использование слабых мест при интерпретации и выполнении отдельных программ на рабочей станции.

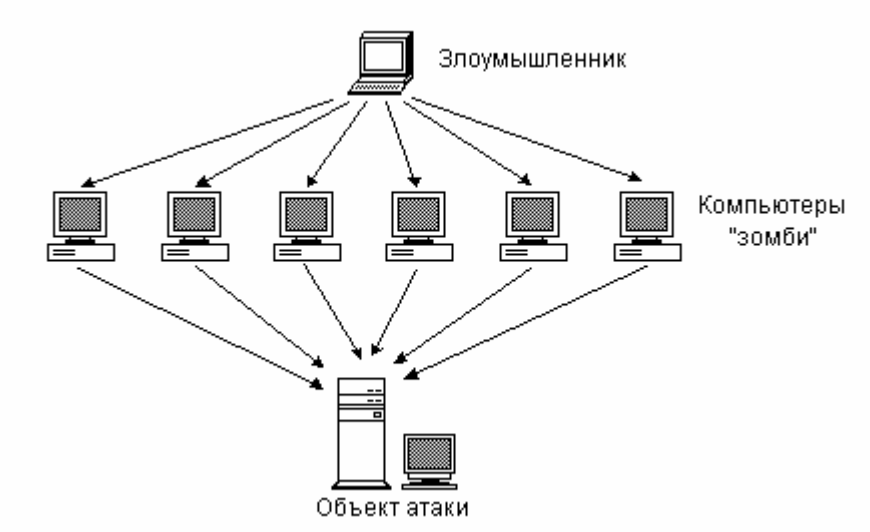


Рисунок 6.3 – DDoS-атака с использованием компьютеров "зомби"

После того как DDoS-агенты были успешно установлены, в любой момент может наступить фаза атаки на систему, являющуюся целью. Установленные и работающие в фоновом режиме DDoS-агенты могут получить команду от центрального мастера атаковать целевую систему путем насыщения ее DoS-пакетами.

К наиболее опасной разновидности атаки DDoS относится атака **Smurf**. При такой атаке возникает эффект усиления вредного воздействия за счет отправки прямых широковещательных запросов *ping* к системам, которые обязаны послать на этот запрос ответ. Атакующий посылает пакет ICMP ECHO по адресу широковещательной рассылки усиливающей сети. Адрес источника этого пакета заменяется адресом жертвы, чтобы представить дело так, будто именно целевая система инициировала запрос. Поскольку пакет ECHO послан по широковещательному адресу, все системы усиливающей сети возвращают жертве свои ответы (если только конфигурация не определяет другого поведения). Послав один пакет ICMP в сеть из 100 систем, атакующий инициирует усиление атаки DDoS в сто раз! Коэффициент усиления зависит от состава сети, поэтому атакующий ищет большую сеть, способную полностью подавить работу системы-жертвы. Воспользовавшись таким подходом, сеть с каналом доступа Т3 (45 Мбит/с) можно насытить с помощью группы каналов связи 56 кбит/с, привлекая к атакам дополнительные узлы.

Одним из первых общедоступных средств реализации атаки DDoS является пакет **TFN** (*Tribe Flood Network*), предназначенный для использования в системе UNIX. В состав пакета входит как клиентский, так и серверный компоненты. Это позволяет взломщику установить серверную часть на

удаленном взломанном узле, а затем с помощью нескольких команд, введенных с использованием клиентской части, инициировать полномасштабную распределенную атаку DoS. С помощью пакета TFN могут быть реализованы атаки с использованием ICMP-, UDP-пакетов, пакетов SYN, а также атаки Smurf. Помимо этих компонентов, в состав пакета TFN входит модуль, позволяющий получить доступ к удаленной командной оболочке, связанной с TCP-портом.

Другим распространенным пакетом, реализующим распределенную атаку DoS, является пакет **Trinoo**. В его состав входит программа удаленного управления (*клиент*), которая взаимодействует с основной программой и передает команды программе-демону (серверу). Взаимодействие между клиентом и основной программой осуществляется посредством соединения через TCP-порт 27665. При этом обычно используется пароль betaalmostdone. Связь основной программы с сервером устанавливается через UDP-порт 27444, а в обратном направлении — через UDP-порт 31335.

Еще более мощным деструктивным средством является пакет Stacheldraht, в котором комбинируется возможности Trinoo и TFN. Он реализует зашифрованный сеанс Telnet между главным и подчиненным модулем. Теперь взломщик может блокировать системы выявления вторжений и благодаря этому получать неограниченные возможности по генерации условия DoS. Как и TFN, пакет Stacheldraht предоставляет возможность инициирования ICMP-, UDP-, SYN- и Smurf-атак. Взаимодействие клиента с сервером осуществляется через комбинацию TCP- и ICMP-пакетов типа ECHO REPLY.

При взаимодействии клиента с сервером применяется алгоритм симметричного шифрования с помощью ключа. Кроме того, по умолчанию активизирован режим защиты с помощью пароля. Стоит упомянуть еще одну дополнительную возможность пакета Stacheldraht: при необходимости взломщик может обновить серверный компонент с использованием команды rcp, которая обычно используется для удаленного копирования файлов.

Усовершенствованным вариантом пакета TFN является пакет TFN2K (обозначение пакета TFN 2000). Это одно из коварных средств DDoS, принципиально отличающееся от своего предшественника и позволяющего в процессе взаимодействия использовать порты с произвольно выбранными номерами. За счет этого свойства возможен обход блокирования портов на пограничных маршрутизаторах. Как и TFN, пакет TFN2K поддерживает SYN-, UDP-, ICMP- и Smurf-атаки. Кроме того, он позволяет случайным образом переключаться между различными методами проведения атаки.

6.3. Способы борьбы с проникновением в сеть

6.3.1. Обнаружение нарушения безопасности сети на основе выявления аномалий поведения

Вывод о возможности проникновения нарушителей в сеть можно сделать по результатам наблюдения за отклонениями в поведении компьютерной сети в течение определенного промежутка времени. Такие отклонения в поведении могут быть обнаружены относительно пользователей, программ, служб и коммуникационных последовательностей.

На основании принципов функционирования сетей и опыта их эксплуатации составляются некоторые шаблоны поведения сети в целом и ее отдельных подсистем либо создаются модели поведения в обычных условиях. Модели поведения для всевозможных ситуаций на практике располагаются в так называемых "ссылочных профайлах". К аномалиям поведения сети можно отнести следующие:

- уровень ошибок идентификации и аутентификации превышает установившиеся границы;
- интенсивность использования сетевых служб выходит за обычный уровень;
- повышенная интенсивность обращения к серверу пользователей с одинаковым адресом;
- необычно большое время работы брандмауэрной системы;
- нарушение набора правил обмена;
- ненормальное поведение программного обеспечения компьютера, брандмауэра и коммуникационной аппаратуры.

В процессе поступления пакетов осуществляется анализ их заголовков, вычисляются некоторые числовые параметры, характеризующие процедуру обмена, и результаты заносятся в регистрационный журнал. Сравнивая данные регистрационного журнала с записями ссылочного профайла, можно обнаружить аномальные отклонения. Если отклонения выходят за определенные пределы, то можно сделать предварительный вывод о нарушении системы безопасности и предпринять необходимые меры.

Ссылочные профайлы должны не только содержать специфические параметры, которые характеризуют типичное поведение, но и определять границы допустимых отклонений. Так, например, вывод о возможной сетевой атаке можно сделать, если имеют место следующие ситуации:

- на протяжении установленного промежутка времени обнаружено аномальное количество попыток аутентификации со стороны различных пользователей;

- обнаружена аномально высокая активность в неурочное время (например, по выходным дням между 3 и 6 часами ночи);
- зарегистрировано необычно большое количество несанкционированных попыток установки соединения с компьютерными системами.

Поведенческие методы основываются не на моделях информационных атак, а на моделях штатного функционирования (*поведения*) компьютерной системы при отсутствии атак.

Среди поведенческих методов наиболее распространены те, что **базируются на статистических моделях**. Такие модели определяют статистические показатели, характеризующие параметры штатного поведения системы. Если с течением времени наблюдается определенное отклонение данных параметров от заданных значений, то делается вывод о наличии атаки. Как правило, в качестве таких параметров могут выступать *уровень загрузки процессора, нагрузка на каналы связи, штатное время работы пользователей системы, количество обращений к сетевым ресурсам* и т. д. Примерами подобных статистических моделей могут служить пороговая модель, модель среднего значения и среднеквадратичного отклонения или ее "расширение" – многовариационная модель.

В пороговой модели для каждого статистического параметра определены пороговые величины. Если наблюдаемый параметр превышает заданный порог, то событие, вызвавшее это превышение, считается признаком потенциальной атаки. Примером реализации такой модели является обнаружение сканирования портов или обнаружение атаки **SYN Flood**. В первом случае пороговым значением является число портов, просканированных в единицу времени. Во втором случае – число попыток установления виртуального соединения с узлом за единицу времени.

Модель среднего значения и среднеквадратичного отклонения для каждого статистического параметра на основе математического ожидания и дисперсии определяет доверительный интервал, в пределах которого должен находиться данный параметр. Если текущее значение параметра выходит за эти границы, то фиксируется существование атаки. Например, если для каждого пользователя компьютерной сети определен доверительный интервал для времени его работы в системе, то факт регистрации пользователя вне этого интервала может рассматриваться как попытка получения несанкционированного доступа к ресурсам сети.

Многовариационная модель аналогична модели среднего значения и среднеквадратичного отклонения, но позволяет одновременно учитывать корреляцию между большим количеством статистических показателей.

Поведенческие методы также реализуются при помощи **нейросетей** и **экспертных систем**. В последнем случае база правил экспертной системы описывает штатное поведение компьютерной системы. Так, при помощи

экспертной системы можно точно специфицировать взаимодействие между хостами сети, которое всегда осуществляется по определенным протоколам в соответствии с действующими стандартами. Если же в процессе обмена информацией между хостами будет выявлена неизвестная команда или нестандартное значение одного из параметров, то это может считаться признаком атаки.

6.3.2. Обнаружение вторжений на основе сигнатурного анализа

Для обнаружения атак в компьютерных сетях широко применяется метод сигнатурного анализа. Он основан на том, что большинство атак на систему известны заранее и развиваются по схожим сценариям. **Сценарий вторжения** представляет собой последовательность действий (*поведений*), описывающих атаку. При использовании сигнатурных методов каждой разновидности атаки ставится в соответствие некий шаблон – ***сигнатура***. Исходным положением сигнатурного метода является то, что сигнатуры атаки определяют характерные особенности, условия, устройства и взаимосвязь событий в сети, которые проявляются при попытках вторжения или собственно проникновения в сеть. Для реализации сигнатурного анализа система безопасности сети содержит список сигнатур вторжений. В процессе функционирования сети непрерывно наблюдается последовательность действий, выполняемых пользователем или программой, и сравнивается с известными сигнатурами. Признаком попытки нарушения безопасности может служить даже частичное соответствие последовательности событий одной из сигнатур.

В качестве сигнатуры могут применяться строка символов, семантическое выражение на специальном языке, формальная математическая модель и т. д. Например, фрагмент "cwd ~root" в FTP-сеансе однозначно определяет факт обхода механизма аутентификации на FTP-сервере и попытке перейти в корневой каталог FTP-сервера. Для регистрации сигнатур применяются специальные программные датчики, которые устанавливаются на сетевом или хостовом уровнях контролируемой сети. После выделения сигнатуры выполняется процедура ее сравнения с образцами, размещенными в специализированной базе данных известных сигнатур атак.

Одним из эффективных методов поиска сигнатуры во входном потоке данных является *метод контекстного поиска*, который заключается в обнаружении в исходной информации определенного множества символов. Так, для выявления атаки на Web-сервер, направленной на получение несанкционированного доступа к файлу паролей, проводится поиск последовательности символов "GET */etc/passwd" в заголовке HTTP-запроса. Для расширения

функциональных возможностей контекстного поиска в некоторых случаях используются специализированные языки, описывающие сигнатуру атаки. С помощью контекстного поиска эффективно выявляются атаки на основе анализа сетевого трафика, поскольку данный метод позволяет наиболее точно задать параметры сигнатуры, которую необходимо выявить в потоке исходных данных.

Существуют еще ряд разновидностей реализаций сигнатурных методов обнаружения атак: метод анализа состояний, методы, базирующиеся на экспертных системах, и методы, основанные на биологических моделях.

Суть **метода анализа состояний** состоит в формировании сигнатуры атак в виде последовательности переходов компьютерной системы из одного состояния в другое. Такие переходы зависят от появления в контролируемой системе определенного события, а последовательность этих событий определяется конкретным видом атаки. Как правило, сигнатуры атак, созданные на основе анализа состояний, описываются математическими моделями, базирующимися на теории конечных автоматов или сетей Петри.

Методы, базирующиеся на экспертных системах, позволяют описывать модели атак на естественном языке с высоким уровнем абстракции. Экспертная система, лежащая в основе таких методов, состоит из двух баз данных: *фактов* и *правил*. Факты представляют собой исходные данные о работе компьютерной системы, а правила – алгоритмы логических решений о факте атаки на основе поступившего набора фактов. Все правила экспертной системы записываются в формате "*если* <...>, *то* <...>". Результирующая база правил должна описывать характерные признаки атак, которые обязана выявлять система обнаружения вторжений.

К одним из наиболее перспективных сигнатурных методов обнаружения вторжения относятся **методы,** основанные **на биологических моделях**. Для их описания используются генетические или нейросетевые алгоритмы. Генетические изначально были предназначены для поиска оптимального решения на основе механизма естественного отбора в популяции. В компьютерных системах популяция атак (как и в биологическом мире) представляется как множество хромосом, каждая из которых моделируется битовой строкой. Популяция развивается на основе трех генетических операций: скрещивания, селекции и мутации, и ее развитие продолжается до тех пор, пока не будет достигнут заданный критерий оптимальности (он определяется в виде специальной функции). При использовании генетических алгоритмов для выявления атак в качестве хромосом популяции выступают векторы определенной длины, каждый элемент которых соответствует конкретной атаке. В результате развития такой популяции можно получить оптимальный вектор, который будет указывать, какие атаки происходят в системе в текущий момент времени.

Нейросетевой метод основан на создании сети взаимосвязанных друг с другом искусственных нейронов, каждый из которых представляет собой пороговый сумматор атак. На начальном этапе нейросеть проходит период "обучения", в течение которого она учится распознавать определенные типы атак. Для этого на ее вход подаются данные, указывающие на определенную атаку, после чего параметры нейросети настраиваются таким образом, чтобы на выходе она смогла определить тип этой атаки. Сложность использования такого метода состоит в том, что, прежде чем использующая его система обнаружения вторжений сможет выявлять большое количество атак, необходим длительный период обучения на большом количестве примеров.

Сигнатурный метод имеет ряд преимуществ, в частности такие:

- достаточно высокая точность определения факта вторжения;
- количество и тип событий, которые необходимо контролировать, ограничены данными, определенными в сигнатурах;
- реализация сигнатурного анализа является весьма эффективной, так как в нем отсутствуют вычисления с плавающей точкой над большими объемами данных, характерных для статистического анализа.

К недостаткам сигнатурного анализа можно отнести следующие:

- зависимость производительности от размера базы данных сигнатур;
- невозможность обнаружения атак, сигнатуры которых пока не определены;
- сложность обновления базы данных сигнатур в виду отсутствия общепринятого языка описания; добавление собственных сигнатур требует высокой квалификации;
- необходимость затрат значительных временных ресурсов для обновления базы данных сигнатур при обнаружении нового типа атак.

Следует отметить, что непосредственное сравнение сигнатуры вторжения с регистрируемой активностью, малоэффективно, в связи с тем, что наблюдаемые данные, относящиеся к атаке, часто имеют отклонения вследствие вариаций действий нарушителя во время атаки или изменения сценария нападения. Поэтому для автоматического принятия решения с использованием сигнатурного анализа о наличии или отсутствии атаки требуется применение систем, построенных на основе методов искусственного интеллекта.

6.3.3. Системы обнаружения вторжений

В качестве одного из базовых средств обеспечения информационной безопасности, используемого для защиты информационных ресурсов компьютерных систем любого предприятия, являются **системы обнаружения вторжений** (**СОВ**) злоумышленников, позволяющие своевременно выявлять и блокировать атаки нарушителей, желающих заполучить секреты предприятия или нарушить его работу. В англоязычной литературе такие системы обозначаются символами **IDS** (*Intrusion Detection Systems*). В основу функционирования этих систем могут быть положены различные методы, позволяющие обнаружить вторжения нарушителей (атаки).

Современные IDS предназначены для контроля работы сетевых устройств и операционных систем, выявления несанкционированных действий и автоматического реагирования на них в реальном масштабе времени. При анализе текущих информационных потоков в некоторых IDS учитываются уже произошедшие события, что позволяет идентифицировать атаки, разнесенные во времени, и тем самым прогнозировать ситуацию на будущее.

Системы IDS используются для выявления не только внешних, но и внутренних нарушителей, которых, как показывает практика, порой гораздо больше, чем внешних. В отличие от внешних нарушителей, внутренний – это авторизованный пользователь, имеющий официальный доступ к ресурсам интрасети, в том числе к тем, в которых циркулирует конфиденциальная информация. Настройка IDS для защиты от внутренних атак представляет собой очень сложную задачу, так как в этом случае требуется учет правил и профилей работы каждого из пользователей сети. Успех борьбы с внутренними атаками может быть обеспечен только при комбинировании различных систем обнаружения вторжений.

Существует много способов проникновения в компьютерную систему, детально рассмотренных в предыдущем разделе. Цель любой IDS – обнаружить атаку как можно скорее с минимальной вероятностью ошибки. При этом объекту атаки (компьютерной системе) необходимо получить ответ на следующие вопросы.

- Что происходит с системой?
- Какие компоненты подверглись нападению, и насколько опасна атака?
- Кто злоумышленник?
- Когда атака началась и откуда?
- Как и почему произошло вторжение?

В первую очередь в IDS используются различные способы определения несанкционированной активности. IDS в свою очередь пытается обнаружить атаку на систему или на сеть в целом и предупредить об этом адми-

нистратора безопасности.

Системы обнаружения вторжений классифицируют по различным признакам. Так, по способу реагирования различают пассивные и активные IDS. ***Пассивные*** просто фиксируют факт атаки, записывают данные в файл журнала регистрации и выдают предупреждения. ***Активные*** *IDS* не только определяют, но и пытаются остановить атаку, а также могут провести ответное нападение на атакующего. Наиболее распространенные типы активного реагирования – прерывание сессии и переконфигурирование межсетевого защитного экрана.

По способу выявления атаки различают системы обнаружения, основанные на **сигнатурах** (*signature-based*) либо на выявлении **аномалий** (*anomaly-based*). Первый тип основан на сравнении информации с предустановленной базой сигнатур атак. Однако системы данного типа не могут обнаруживать новые, неизвестные виды атак. Второй тип основан на контроле частоты событий или обнаружении статистических аномалий. Такая система ориентирована на выявление новых типов атак. Однако ее недостаток – необходимость постоянного обучения.

Часто системы обнаружения вторжений классифицируют по уровню сбора информации об атаке: обнаружение на сетевом уровне (*network-based*), на рабочей станции (*host-based*) и на уровне приложения (*application-based*). Системы обнаружения атак на сетевом уровне (Network IDS, **NIDS**) контролируют пакеты в сетевом окружении и пытаются обнаружить попытки злоумышленника проникнуть внутрь защищаемой системы или реализовать атаку типа "отказ в обслуживании". Примером обнаружения на сетевом уровне процесса сканирования TCP-портов является процедура контроля количества TCP-запросов на соединение (посылка пакетов SYN) со многими портами на выбранном компьютере. Система обнаружения атак на сетевом уровне (NIDS) может запускаться либо на отдельном компьютере, который контролирует свой собственный трафик, или на выделенном компьютере, который анализирует весь сетевой трафик (концентратор, маршрутизатор, зонд). Следует заметить, что "сетевые" IDS контролируют много компьютеров, тогда как индивидуальные системы обнаружения атак контролируют только тот хост, на котором они установлены.

Системы второго типа (*host-based*) предназначены для мониторинга, детектирования и реагирования на действия злоумышленников на определенном хосте. Система, располагаемая на защищаемом хосте, проверяет и выявляет направленные против него действия.

Третий тип IDS (application-based) основан на поиске проблем в определенном приложении. Существуют также гибридные IDS, представляющие собой комбинацию различных типов рассмотренных систем.

По виду анализируемых файлов различают **системы контроля цело-**

стности (*System integrity verifiers*) файлов и **мониторы регистрационных файлов** (*Log-file monitors*). Первые проверяют системные файлы с целью определения момента внесения в них изменений. Мониторы регистрационных файлов контролируют регистрационные файлы, создаваемые сетевыми сервисами и службами. Так же, как и NIDS, эти системы ищут известные сигнатуры не в сетевом трафике, а только в файлах регистрации. Обнаружение сигнатур указывает на то, что злоумышленник осуществил атаку. Типичным примером является синтаксический анализатор для log-файлов HTTP-сервера, который ищет хакеров, пытающихся использовать хорошо известные уязвимости.

В последнее время появились разработки распределенных систем обнаружения вторжений – *distributed IDS* (**dIDS**). Распределенные системы состоят из множества локальных IDS, расположенных в различных участках большой сети и связанных между собой и с центральным управляющим сервером. Такие системы усиливают защищенность корпоративной сети благодаря централизации информации об атаках, поступающей от локальных IDS. Существуют также **обманные системы** (*deception systems*), которые работают с псевдосервисами. Их цель состоит в имитации распространенных уязвимостей для того, чтобы обмануть злоумышленников.

Необходимо отметить, что выявление атак системами обнаружения вторжений должно выполняться на различных уровнях защищаемой сети. Так, на самом нижнем уровне системы обнаружения вторжений способны выявлять атаки на конкретных узлах сети: рабочих станциях, серверах и маршрутизаторах. Следующий уровень обнаружения – сетевые сегменты компьютерной системы, состоящие из нескольких хостов. Обнаружение атак также возможно и в отдельных подсетях объединенной компьютерной сети: в локальных, территориально распределенных, а также и в глобальных системах. При этом в зависимости от инфраструктуры защищаемой сети на разных уровнях могут использоваться разные методы выявления атак.

Следует подчеркнуть, что на стадии сбора злоумышленником исходных данных эффективны лишь сигнатурные методы выявления атак. Это объясняется тем, что все операции получения необходимой нарушителю информации в большинстве случаев не вызывают никакого отклонения работы компьютерной системы от штатного режима. Для этого этапа характерны такие признаки, как формирование запроса к DNS-серверу, получение сведений из базы данных управляющей информации сетевого администрирования или многократные TCP-запросы на установление соединения с различными портами. На этапе сбора исходных данных могут использоваться как сетевые, так и хостовые датчики.

На стадии вторжения обнаружить атаку можно при помощи как сигнатурных, так и поведенческих методов. Любое вторжение характеризуется

определенными признаками, которые, с одной стороны, можно представить в виде сигнатуры, а с другой – описать как некое отклонение от штатного поведения контролируемой системы. Наиболее эффективно сочетание обоих методов, при этом для получения необходимых исходных данных применимы и хостовые и сетевые датчики.

Эффективное выявление атак на этапах атакующего воздействия и развития атаки возможно только при помощи поведенческих методов, поскольку действия нарушителей зависят от целей проводимой атаки, и фиксированным множеством сигнатур атак однозначно не определяются. Учитывая тот факт, что на двух последних стадиях жизненного цикла информационной атаки самые характерные объекты – это хосты, в данном случае наиболее целесообразно применение хостовых датчиков.

Общая схема функционирования распределенной IDS показана на рисунке 6.4. Одним из наиболее важных компонентов распределенной системы обнаружения вторжений dIDS является **агент сети**. Он представляет собой небольшую по объему программу, цель которой – сообщать об атаке на центральный анализирующий сервер.

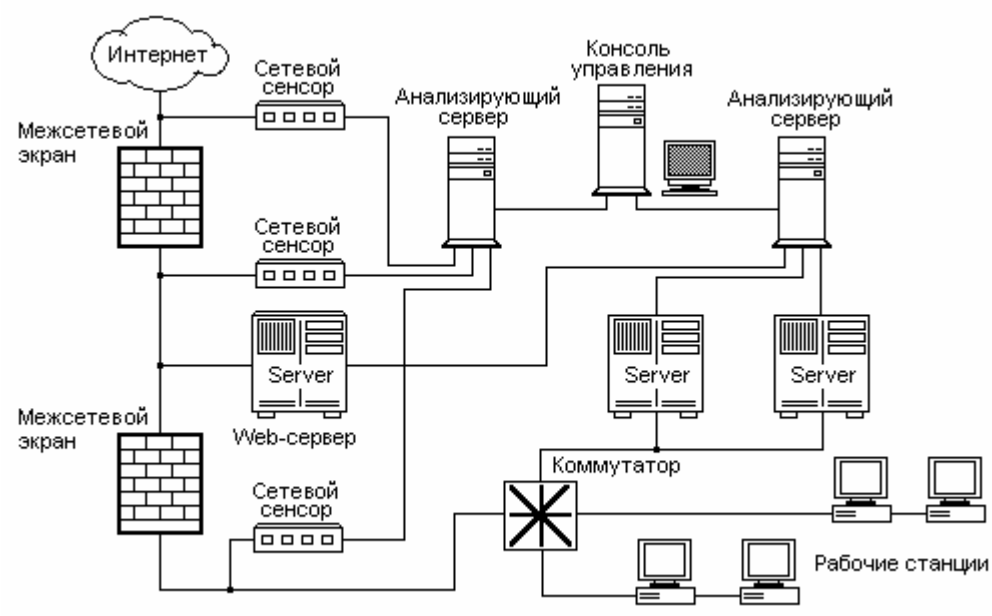


Рисунок 6.4 – Структура системы обнаружения вторжений

Сетевые сенсоры (датчики) размещаются в стратегических точках сети и контролируют проходящий трафик. Они могут анализировать заголовки и содержание каждого пакета, а также сопоставлять выбранные пакеты с образцом. Верхний сенсор анализирует трафик, поступающий из незащищен-

ной сети, средний – трафик так называемой **демилитаризованной зоны**, после первого межсетевого экрана. Нижний датчик анализирует внутренний трафик. Сетевые сенсоры реализуются, как правило, аппаратно. При фиксации факта нападения датчик инициирует ряд действий: включение сигнализации, регистрацию события, уничтожение сеанса или полный разрыв соединения. Состояние объектов в системе (компьютеров, прикладных программ, процессов и т.д.) отражается на консоли управления администратора в виде текста или пиктограмм. При этом состояние каждого компонента представляется определенным цветом: нормальному состоянию обычно соответствует зеленый, пограничному – желтый, а критическому – красный цвет. Датчики управляются с консоли управления администратора, а информация об атаках экспортируется в реляционную базу данных для последующего анализа.

Сервер сбора информации об атаке – часть системы dIDS, логически базирующаяся на центральном анализирующем сервере. Сервер определяет параметры, по которым группируется информация, полученная от агентов сети. Группировка может осуществляться по IP-адресу атакующего, порту получателя, номеру агента, дате, времени, протоколу, типу атаки и т. д.

6.3.4. Способы защиты от DoS- и DDoS-атак

Для предотвращения возможности использования злоумышленником сети или компьютера(ов) следует запретить прохождение направленных широковещательных запросов на пограничном маршрутизаторе. Кроме того, некоторые операционные системы можно настроить так, чтобы отбрасывались все широковещательные ICMP-пакеты типа ECHO.

Наилучшая защита сетевых компьютеров от использования в качестве "зомби" заключается в предотвращении их взлома на начальной стадии атаки. Для этого пользователям необходимо выполнить следующие действия:

- ограничить использование отдельных служб;
- запретить выполнение неизвестных программ;
- установить модули обновления операционной системы и приложений;
- задать только минимально необходимые разрешения на использование каталогов и файлов.

Для того чтобы защитить свои компьютеры от их использования в качестве "зомби", следует реализовать также некоторые правила фильтрации пакетов на пограничном маршрутизаторе, в частности, необходимо обеспечить фильтрацию пакетов ICMP для ограничения возможности применения атак Smurf.

Чтобы определить, подвержена ли атаке компьютерная система пакетами SYN, можно воспользоваться командой *netstat*, если она поддерживается операционной системой. Многочисленные соединения, находящиеся в состоянии SYN_RECV свидетельствуют о том, что именно в этот момент проводится атака.

Далее приводятся несколько основных способов защиты от атак с использованием пакетов SYN.

1) *Увеличение размера очереди на установку соединений*. Несмотря на то что стек протокола IP каждым производителем реализуется по-своему, вполне возможно настроить размер очереди на установку соединений таким образом, чтобы нейтрализовать воздействие атаки с использованием пакетов SYN. Это полезное, однако не самое оптимальное решение, поскольку его реализация требует дополнительных системных ресурсов, что может сказаться на общей производительности.

2) *Уменьшение периода ожидания установки соединения.* Сокращение интервала ожидания установки соединения также поможет снизить влияние атаки. Тем не менее, это тоже не самое оптимальное решение проблемы.

3) *Использование пакетов обновления программного обеспечения и защита от потенциальных атак* SYN. Большинство современных операционных систем уже имеет встроенные механизмы и средства выявления и предотвращения атак с применением пакетов SYN.

4) *Использование сетевых систем IDS*. Некоторые системы IDS уровня сети могут обнаруживать и активно противодействовать атакам SYN. Такие атаки можно обнаружить по возросшему потоку пакетов SYN, который не сопровождается потоком ответных сообщений. Система выявления вторжений может передать системе, используемой в процессе атаки, пакет RST, соответствующий начальному запросу SYN. Это будет способствовать восстановлению корректного состояния очереди на установку соединений.

Хотя каждый из подходов имеет свои достоинства и недостатки, все они способны снизить воздействие сфокусированной атаки SYN. Не следует забывать о сложности выявления злоумышленника, поскольку он использует ложный исходный адрес. Однако в решении этой задачи может помочь утилита ***dostracker,*** если у администратора сети имеется доступ к маршрутизаторам каждого сегмента пути.

Поскольку атаки SYN получили в глобальной сети широкое распространение, были разработаны и другие решения проблемы атак DoS. Например, современное ядро системы Linux версии 2.0.30 и более поздних версий имеет режим **SYN cookie**. Если этот режим включен, ядро будет выполнять выявление и регистрацию возможных атак SYN. После этого будет использоваться криптографический протокол, известный под названием SYN cookie, который позволит легитимным пользователям устанавливать соеди-

нение даже в процессе предпринятой атаки.

В других операционных системах реализован динамический механизм выделения ресурсов. Когда длина очереди на установку соединений достигает некоторого предопределенного порога, система автоматически выделяет дополнительные ресурсы, поэтому очередь никогда не будет переполнена.

6.3.5. Безопасная оболочка SSH

Безопасная оболочка **SSH** (*secure shell*), одно из самых распространенных программных средств повышения компьютерной безопасности при работе Unix-систем в сети Интернет. Под SSH понимают как собственно программу, так и задействованный в ней протокол.

Протокол SSH – это безопасный протокол, так как весь передаваемый по нему трафик шифруется. Таким образом, значительно повышается сохранность пароля пользователя и защита сайта в целом от взлома. С помощью протокола SSH можно осуществлять работу с удаленным сервером в командной строке – отлаживать, запускать программы. Кроме того, этот протокол позволяет осуществлять обычную передачу файлов.

SSH представляет собой средство организации безопасного доступа к компьютерам при работе по небезопасным каналам связи. Для организации безопасного доступа применяется процедура аутентификации с использованием асимметричного шифрования с открытым ключом. Такое решение обеспечивает более высокую безопасность, чем при использовании симметричного шифрования, хотя и порождает дополнительную вычислительную нагрузку. При последующем обмене данными применяется уже симметричное шифрование, более экономичное с точки зрения затрат процессорного времени.

SSH поддерживает режим работы со службой Telnet благодаря возможности перенаправления соответствующих данных по надежным SSH-каналам; выполняет безопасную замену многих r-команд Unix (rsh, rlogin и т.д.), с которыми традиционно связаны проблемы обеспечения безопасности.

Стандарт SSH, описывающий протокол SSH, состоит из нескольких документов, специфицирующих общую архитектуру протокола, а также протоколы трех уровней: транспортного, аутентификации и соединения. Их задача – реализовать безопасную сетевую службу в небезопасной сети.

Протокол транспортного уровня обеспечивает аутентификацию сервера, конфиденциальность и целостность данных, а протокол аутентификации – аутентификацию клиента для сервера. Таким образом, с целью повышения безопасности осуществляется не только аутентификация клиента для серве-

ра, к которому обращается клиент, но и аутентификация сервера клиентом. Протокол соединения SSH мультиплексирует безопасный (шифруемый) канал, представляя его в виде нескольких логических каналов, которые используются для различных видов служб. Клиент шлет запрос на обслуживание в первый раз, когда устанавливается безопасное соединение транспортного уровня SSH. Второй запрос направляется уже после завершения аутентификации пользователя (клиента).

Каждый работающий с SSH хост, на котором может функционировать как клиент, так и сервер, должен иметь как минимум один ключ, причем для шифрования допускаются различные криптографические алгоритмы. Несколько хостов могут иметь общий ключ. Ключ хоста-сервера используется при обмене открытыми ключами с целью проверки того, что клиент действительно общается с настоящим, а не подмененным сервером. Для этого клиент должен знать открытый ключ хоста-сервера. Это знание реализуется в рамках одной из двух моделей. В первой клиент просто имеет некий локальный файл, в котором каждому имени хоста ставится в соответствие его открытый ключ. Во второй модели вводится понятие сертификационного агента, отвечающего за проверку соответствия имени хоста его открытому ключу. При этом клиент знает только открытый ключ самого сертификационного агента. В последнем случае упрощается поддержка клиента (ему нужно знать всего один открытый ключ), но появляются высокие требования к сертификационному агенту, который должен иметь открытые ключи всех хостов, к которым обращаются клиенты.

Протоколом предусмотрена возможность отказа от проверки ключа хоста-сервера при самом первом обращении клиента к этому серверу. При этом соединение клиент-сервер будет защищено от пассивного прослушивания сети, но возникает опасность атаки типа "человек в середине" (man-in-the-middle), т. е. попытки временной подмены сервера. Если используется эта возможность, то ключ хоста-сервера будет автоматически передан клиенту и сохранен в его локальном файле.

Протоколом предусмотрено ведение переговоров между клиентом и сервером, в результате которых выбираются методы шифрования, форматы открытых ключей и т.п., предполагаемые для использования в данном сеансе связи.

6.4. Брандмауэры – межсетевые защитные экраны

6.4.1. Назначение и классификация брандмауэров

Существуют простые и надежные решения, которые могут быть использованы для улучшения безопасности сети организации. Одним из эффективных средств повышения общей безопасности сети является **система брандмауэра**, представляющая собой набор *группы программ и маршрутизаторов*, добавленных в сеть в местах ее соединения с Интернетом и *политики доступа*, определяющей правила их работы.

Основная цель брандмауэра – управление доступом ***к*** или ***из*** защищаемой сети. Он реализует политику сетевого доступа, заставляя проходить все соединения с сетью через брандмауэр, где они могут быть проанализированы и разрешены либо отвергнуты.

Брандмауэр (*межсетевой экран*) представляет собой систему, реализующую правила обмена данными между двумя или несколькими компьютерными сетями с целью защиты сети от проникновения злоумышленников. Брандмауэр устанавливается между защищаемой и незащищенной сетью (рисунок 6.5). В переводе с немецкого языка брандмауэр означает "противопожарная стена". Он обычно устанавливался между смежными зданиями для предотвращения распространения пожара. В технической литературе широко используется английское название межсетевого защитного экрана – “Firewall”.

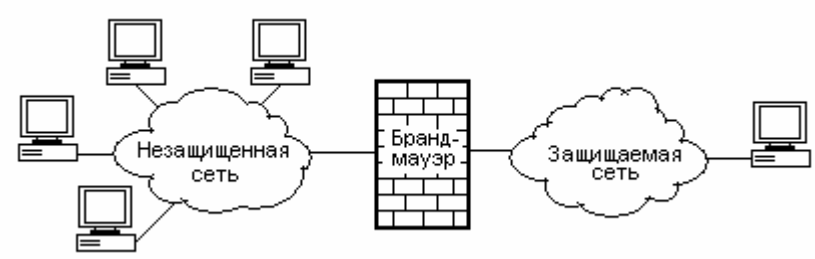


Рисунок 6.5 – Схема защиты сети посредством брандмауэра

Для реализации функций защиты весь сетевой трафик защищаемой сети должен проходить через брандмауэр, который осуществляет анализ и непрерывный контроль проходящих пакетов, а также блокирует обмен в случае нарушения установленных правил и закономерностей обмена. Брандмауэр может быть реализован на отдельных аппаратных средствах либо в виде специального программного обеспечения, устанавливаемого на каждый компьютер сети.

Брандмауэры обеспечивают несколько типов защиты:

- блокировка нежелательного трафика;
- направление входного трафика только к надежным внутренним системам;
- скрытие уязвимых систем от потенциальных атак;
- протоколирование трафика, проходящего в обоих направлениях;
- сокрытие внутренней структуры сети, типов сетевых устройств, имен систем и идентификаторов пользователей;
- обеспечение более надежной аутентификации.

Следует заметить, что ***брандмауэры*** *не обеспечивают защиту сетей автоматически*. Защита возможна лишь в случае надлежащего управления брандмауэрной системой. Только правильная настройка и эксплуатация такой системы обеспечивает техническую, персональную и организационную безопасность в защищаемой сети. К главным задачам, решаемым брандмауэрными системами в процессе их функционирования, относятся следующие:

- контроль доступа на сетевом, пользовательском и прикладном уровнях, а также уровне данных;
- управление правами доступа;
- изоляция сетевых служб;
- анализ сетевых соединений и регистрация данных;
- оповещение администрации о возникновении потенциальных или реальных опасностях вторжения.

В процессе контроля доступа на сетевом уровне принимается решение, каким хостам разрешается взаимодействие через межсетевой экран. При контроле на пользовательском уровне выполняется аутентификация пользователей, определяется, кому из них разрешается отправлять данные в незащищенную сеть, а каким принимать из нее. Контроль доступа на прикладном уровне позволяет отсеять любые переданные данные или команды, не относящиеся к данному приложению.

Управление правами доступа регламентирует виды протоколов и сетевых служб, которым разрешено взаимодействовать через межсетевой экран, а также временные рамки такого взаимодействия. Изоляция сетевых служб предназначена для избежания распространения последствий сетевых атак на смежные службы.

Регистрация данных о сетевых соединениях и событиях, имеющих отношение к безопасности необходима для последующего анализа с целью улучшения защиты. Любая деятельность внешних источников, а также пользователей защищаемой сети, выходящая за рамки установленных правил, регистрируется и может быть передана на станцию контроля и управления безопасностью для принятия соответствующих мер.

Существует несколько различных реализаций брандмауэров. К одному

из простых и достаточно экономичных типов брандмауэров относится **пакетный фильтр,** который обеспечивает минимальную безопасность за невысокую цену. Брандмауэры с фильтрацией пакетов используют маршрутизаторы с правилами фильтрации для предоставления или запрещения доступа на основе адресов отправителя и получателя, а также номера порта.

Другим видом реализации брандмауэра являются **прикладные шлюзы**, использующие программы, называемые ***прокси-серверами***, которые запускаются на брандмауэре. Прокси-сервер принимает запросы из незащищенной сети, анализирует их и передает безопасные запросы внутренним хостам, предоставляющим соответствующие услуги. Прикладные шлюзы могут также выполнять функции аутентификации пользователей и протоколирования их действий. Брандмауэры прикладного уровня настраиваются таким образом, что весь выходящий трафик представляется для внешних наблюдателей исходящим от защитного экрана, т.е. внешним сетям виден только брандмауэр.

К третьему виду реализации брандмауэров относятся **гибридные** или **сложные шлюзы**. Они объединяют в себе оба описанных вида и реализуют их последовательно. За счет последовательной реализации общий уровень безопасности повышается, в отличие от параллельной реализации, при которой общий уровень безопасности соответствует наименее безопасному из используемых видов.

Функции межсетевого экрана может выполнять маршрутизатор, сетевой компьютер или группа компьютеров. Обычно система брандмауэра создается на основе маршрутизаторов верхнего уровня, обычно на тех, которые соединяют сеть с Интернетом, хотя может быть создана и на других маршрутизаторах, для защиты только части хостов или подсетей.

6.4.2. Структура межсетевых экранов

Брандмауэр заставляет все сетевые соединения проходить через шлюз, где они могут быть проанализированы и оценены с точки зрения безопасности. Он предоставляет также другие средства, в частности, меры усиленной аутентификации вместо паролей. Кроме того, брандмауэр может ограничить доступ к тем или иным системам или доступ к Интернету от них, блокировать определенные сервисы TCP/IP, или обеспечить другие меры безопасности. Для решения таких задач брандмауэр содержит ряд взаимосвязанных программных модулей, выполняющих совместно функции защиты компьютерной сети (рисунок 6.6).

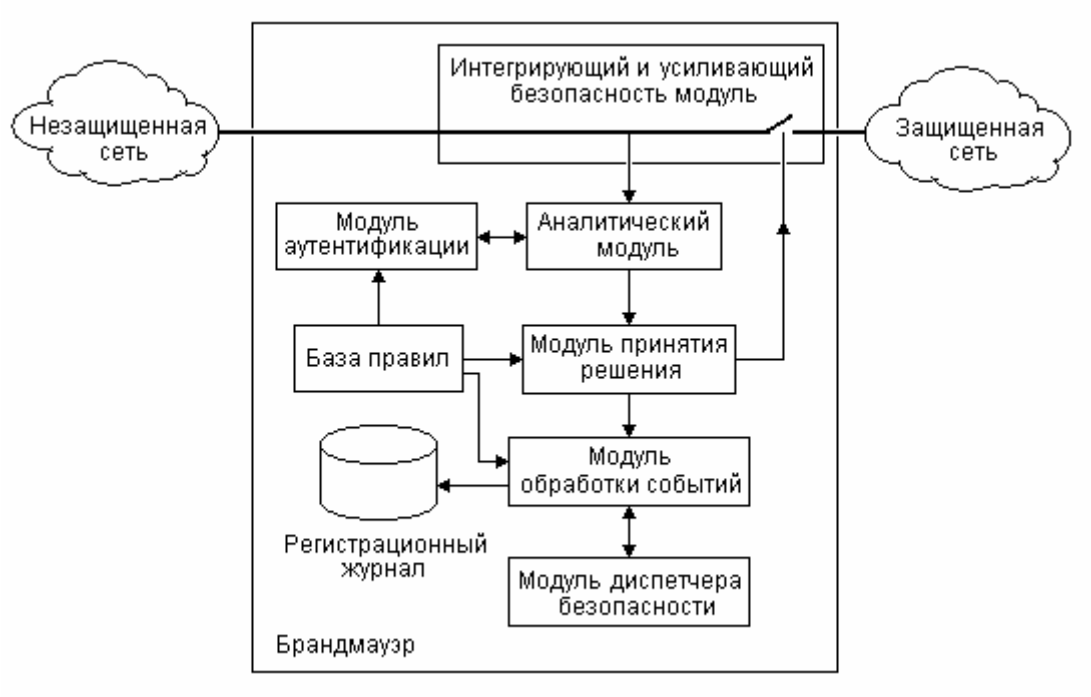


Рисунок 6.6 – Структура активных элементов брандмауэра

Все возникшие ситуации, имеющие отношение к безопасности защищаемой системы, которые были зафиксированы аналитическим модулем брандмауэра, подвергаются обработке в **модуле обработки событий**, касающихся безопасности.

В зависимости от набора правил анализа и конфигурационных настроек и установок события, имеющие отношение к безопасности, заносятся в **регистрационный журнал**, а диспетчеру безопасности выдается предупреждение. **Модуль диспетчера безопасности** отвечает за обеспечение интерфейса, посредством которого администратор сети сопровождает и обслуживает базу правил брандмауэра, отслеживает события, происходящие в системе, и анализирует информацию, содержащуюся в регистрационном журнале.

Надежность функционирования компонентов межсетевого экрана обеспечивается **модулем защиты брандмауэра**. На него возложены функции повышения активной защиты брандмауэра в целом. В его задачи входит реализация следующих механизмов:

проведение тестов целостности, позволяющих обнаруживать изменения, произошедшие в операционной системе, приложениях, модулях реализации безопасности и т.д.;

выполнение аутентификации, механизм, который разрешает вносить изменения в базу правил и читать информацию регистрационного журнала только персоналу управления безопасностью, имеющему на это право;

реализацию функциональной безопасности, при которой гарантируется безопасная работа всех компонентов брандмауэра, например, отслеживаются регистрационные журналы и жесткие диски на предмет проверки рабочего состояния, переполнения и т.д.

Кроме обеспечения безопасности, брандмауэр сам должен активно противостоять атакам, использовать программное обеспечение, в котором нет ошибок. Его программное обеспечение не должно выполнять никаких других операций кроме реализации безопасности собственно межсетевого экрана и защищаемой системы. Многие брандмауэры имеют систему проверки целостности программных кодов. При этом контрольные суммы таких кодов хранятся в защищенном месте и сравниваются при старте программы во избежание подмены программного обеспечения.

6.4.3. Пакетные фильтры

Пакетный фильтр (*packet filter*) относится к активным элементам брандмауэрной системы безопасности. Он предназначен для анализа и управления трафиком входящих и исходящих пакетов на уровне доступа к сети, сетевом и транспортном уровнях. Размещение пакетного фильтра между сетями позволяет разделить их физически. Пакетные фильтры работают как обычные мосты.

Брандмауэры с пакетными фильтрами принимают решение о том, пропускать пакет или отбросить на основе анализа IP-адресов, флагов или номера TCP-порта в заголовке этого пакета. Хотя IP-адрес и номер порта – это информация сетевого и транспортного уровней соответственно, пакетные фильтры используют также информацию прикладного уровня, так как все стандартные сервисы в TCP/IP ассоциируются с определенным номером порта.

Проверочные сведения берутся из базы правил и сверяются с результатами анализа. Для описания правил прохождения пакетов через фильтр предварительно составляются таблицы, которые затем заносятся в базу правил (таблица 6.1).

Ячейка таблицы "*Действие*" может принимать значения "пропустить" или "отбросить". В графе "*Тип пакета"* записывается тип протокола - TCP, UDP или ICMP. В ячейке "*Флаги"* указываются флаги из заголовка IP-пакета. Ячейки "*Порт источника*" и "*Порт назначения*" используются только для TCP- и UDP-пакетов.

Существует два подхода к заданию и оценке правил фильтрации: позитивный и негативный. В соответствии с позитивным подходом создания правил фильтрации указываются типы разрешенных пакетов. При этом оп-

ределяется только то, что будет разрешено. Все, что не разрешено явным образом, автоматически запрещается.

Таблица 6.1 – Таблица описания правил прохождения

Тип пакета	Адрес источника	Адрес назначения	Порт источника	Порт назначения	Флаги	Действие

Брандмауэр разрешает прохождение соединения, которое в таблице значится как "пропустить". По стратегии негативного подхода фильтрации указываются типы запрещенных пакетов. Исходным пунктом является принцип, что все разрешено. Брандмауэр запрещает прохождение только того соединения, которое в таблице доступа указано как "отбросить".

Пакетные фильтры обычно работают с IP-адресами конкретных компьютерных систем. Однако есть пакетные фильтры, которые могут контролировать соединение с пользователями. Такие фильтры называют пакетными фильтрами, ориентированными на работу с пользователями. При использовании данного типа фильтров доступу в сеть предшествуют процедуры аутентификации. В этих пакетных фильтрах имеются *профайлы пользователей*, содержащие описание профилей работы пользователей: используемые коммуникационные протоколы и приложения, время работы и компьютеры, на которых они будут работать и т.д. Пакетный фильтр активно предотвращает использование неразрешенных коммуникационных протоколов. Администратор имеет возможность задать рассылку сообщений на консоль диспетчера безопасности или запись событий в регистрационном журнале пакетного фильтра. Когда происходит событие, касающееся безопасности системы, в регистрационный журнал вводятся следующие сведения:

- дата и время события;порядковый номер события;
- тип события, имеющего отношение к безопасности;IP-адреса и номера портов;
- любая информационная запись, содержащаяся в IP-пакете.

Для дополнительной защиты в некоторых реализациях фильтров вводится режим блокировки прохождения пакетов в случае переполнения регистрационного журнала.

К положительным качествам пакетных фильтров следует отнести относительно невысокую их стоимость, гибкость в определении правил фильтрации, небольшую задержку при прохождении пакетов. Однако возможность применения пакетных фильтров имеет определенные пределы, вне которых их использование не целесообразно. Большинство реальных пакетных

фильтров имеют такие ограничения:

- данные выше транспортного уровня не анализируются;
- пакетный фильтр не обеспечивает безопасности для приложений FTP, HTTP и т.д.; так, например, в случае применения SMTP (порт 25) атаки могут производиться через утилиту Sendmail;
- возможен доступ извне к программам, работающим в защищенной сети по причине неправильной их настройки;
- типичные пакетные фильтры не помогают скрыть структуру защищаемой сети;
- регистрационные данные осуществляются только до транспортного уровня; правила фильтрации пакетов трудны в описании, необходимы глубокие знания технологий TCP и UDP;
- нарушение работоспособности брандмауэра приводит к тому, что все компьютеры за ним становятся полностью незащищенными либо недоступными.

6.4.4. Шлюзы прикладного уровня и прокси-агенты

Другой разновидностью межсетевого экрана является **шлюз прикладного уровня** (A*pplication Gateways*). Брандмауэр, построенный по принципу шлюза прикладного уровня, разделяет две сети логически и физически. Шлюз прикладного уровня имеет два сетевых интерфейса. Один соединен с незащищенной сетью (Интернет), а второй – с защищаемой. Вход в защищаемую сеть может быть выполнен только через брандмауэр, другие пути исключены. При этом осуществляется полный контроль над пакетами, циркулирующими между защищенной и незащищенной сетями. Структура шлюза прикладного уровня изображена на рисунке 6.7.

Шлюз прикладного уровня (в дальнейшем, в этом подразделе, сокращенно шлюз) получает пакеты через драйверы сетевого доступа и TCP/IP на соответствующие порты. Когда порт предназначен только для одной службы, на шлюзе должно быть установлено программное обеспечение (ПО) для передачи пакетов только к этой службе как с внешней, так и с внутренней сторон. Такое ПО, позволяющее производить передачу через шлюз пакетов, имеющих отношение только к одной службе (FTP, HTTP, Telnet и др.), называется **прокси-агентом**. На прокси-агент, работающий на шлюзе, часто возлагаются дополнительные функции, обеспечивающие безопасность службы, за которую он отвечает. В связи с тем, что прокси-агент специализируется на одной службе, диапазон возможных функций регистрации и безопасности можно значительно расширить. Он может выполнять детальный анализ по-

ступающих сообщений, так как для соответствующих служб четко определен контекст прикладных данных.

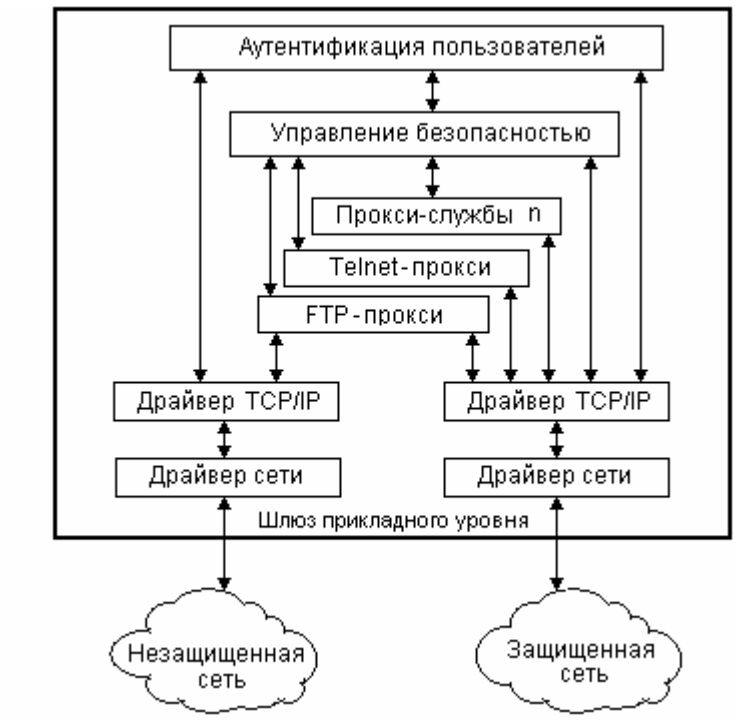


Рисунок 6.7 – Структура шлюза прикладного уровня

Число прокси-агентов, функционирующих в шлюзе прикладного уровня, определяется количеством служб, разрешаемых к использованию в защищаемой сети. В прокси-агенте может производиться дополнительное шифрование и кодировка данных.

Пользователь, желающий получить доступ к сети, сначала обязан идентифицировать себя и пройти аутентификацию. Для прохождения аутентификации пользователь предварительно должен установить соединение со шлюзом прикладного уровня. В этом случае коммуникационным партнером для пользователя будет не компьютерная система-получатель, а шлюз. Однако после завершения процедур идентификации и аутентификации шлюз становится прозрачным, транслируя пакеты получателю и попутно осуществляя их анализ. У пользователя создается впечатление, что он работает непосредственно с компьютерной системой.

Поскольку шлюзы подключены как к компьютерным системам незащищенной сети, так и к компьютерным системам защищенной сети, шлюз должен обеспечивать также и трансляцию сетевых адресов. Шлюз прикладного уровня имеет свой официальный IP-адрес в незащищенной сети и ча-

стный IP-адрес в защищенной. В процессе обмена пакетами с компьютерными системами, расположенными в незащищенной сети, шлюз использует IP-адреса незащищенной сети. При обмене пакетами с компьютерами, подключенными к защищенной сети, шлюз прикладного уровня пользуется внутренними адресами этой сети.

Прокси-агенты прикладного уровня применяются для обеспечения конкретных служб или приложений. Им известны команды определенного прикладного протокола, в связи с чем они могут анализировать их на предмет корректности применения в той или иной ситуации. Прокси-агенты прикладного уровня работают со стандартным клиентским программным обеспечением протоколов FTP или Telnet, либо с браузерами – для протокола HTTP.

В настоящее время наиболее широко используется HTTP-прокси. На него возлагаются задачи обеспечения управляемого обмена пакетами с помощью протокола HTTP и выполнения специальных функций для обеспечения безопасности этого вида сервиса. Структура шлюза прикладного уровня для службы HTTP изображена на рисунке 6.8.

В состав шлюза входят ряд блоков, осуществляющих определенные проверки. Если пакет не удовлетворяет условиям проверки, то связь хоста с сервером разрывается, а в журнал регистрации заносятся данные о параметрах соединения и причинах разъединения. В процессе установления соединения клиент сначала выходит на порт 80 шлюза прикладного уровня. Затем он выполняет процедуры идентификации и аутентификации и указывает службе HTTP желаемый адрес соединения. В случае успешной идентификации и аутентификации HTTP-прокси активизирует профайл, содержащий записи с IP-адресами отправителя и получателя, а также имя пользователя, указанное в процессе аутентификации. После этого HTTP-прокси устанавливает второй канал от шлюза прикладного уровня к порту 80 компьютерной системы получателя. Теперь пользователь может использовать службу HTTP на компьютере адресата через шлюз прикладного уровня.

Фильтр данных позволяет осуществлять доступ к одним URL и запретить к другим. Например, путем настройки параметров фильтра данных можно разрешить обращение только к серверам с доменом *.ua, исключить доступ к определенным страницам, предотвратить применение некоторых языков (Java, JavaScript или Active X) и др. Он может быть также использован для отсева известных нежелательных файлов, обнаружения вирусов, "червей" и "троянских коней". Посредством **командного фильтра** выполняется анализ и проверка разрешенных к применению на сервере протоколов (HTTP, FTP, SMTP) и команд (cd, put, get и т.д.).

Любая попытка использовать ошибочный протокол или команду приводит к разрыву соединения, результаты проверки заносятся в журнал и со-

общаются администратору.

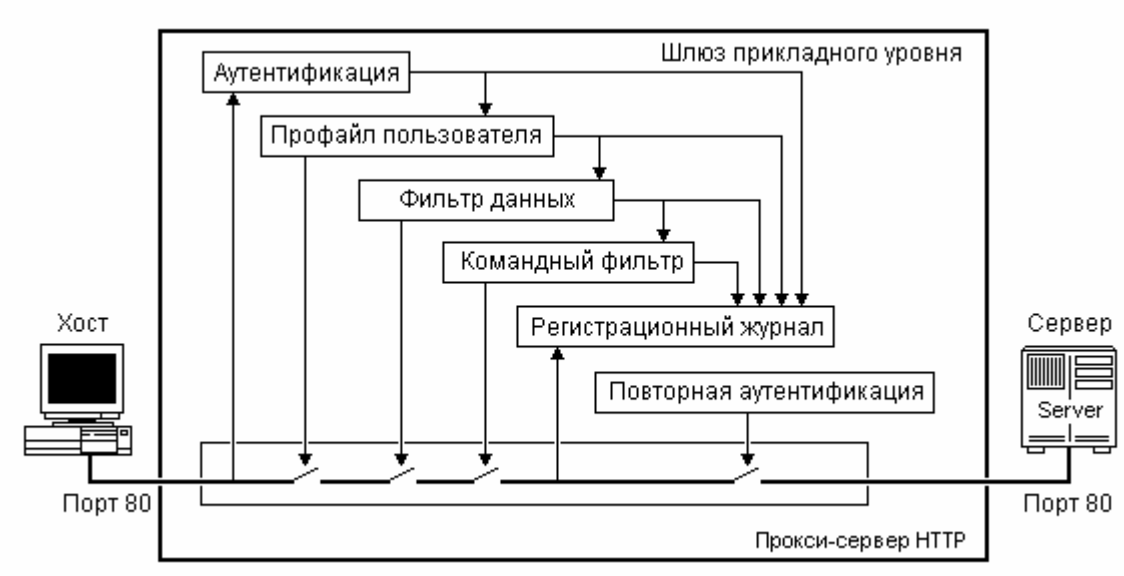


Рисунок 6.8 – Структура HTTP прокси-сервера

В процессе функционирования HTTP-прокси в его регистрационном журнале фиксируются следующие сведения:

- IP-адреса и имена компьютеров отправителя и получателя;
- имя пользователя;
- время и дата установления соединения;
- количество данных в байтах;
- имена переданных файлов или HTML-страниц;
- время и дата разрыва соединения;
- другие сведения, касающиеся безопасности сети.

В ряде случаев в шлюзах прикладного уровня после успешного выполнения первичной аутентификации запускается тайм-аут и после его истечения выполняется повторная аутентификация. При этом второй пароль может отличаться от первого. За счет такого приема повышается степень безопасности доступа к серверу службы.

К преимуществам **шлюзов прикладного уровня** следует отнести следующие:

- локальная сеть невидима из Интернета и тем самим сохраняется в секрете структура внутренней сети;
- при нарушении работоспособности брандмауэра пакеты перестают проходить через межсетевой экран, поэтому не возникает угрозы для защищаемых им компьютеров;
- защита на уровне приложений позволяет осуществлять большое коли-

чество дополнительных проверок, снижая тем самым вероятность взлома с использованием дыр в программном обеспечении;
- сведения о соединениях и прикладные данные регистрируются, документируются параметры пользователей, производящих обмен данными через шлюз;
- при аутентификации на пользовательском уровне может быть реализована система немедленного предупреждения о попытке взлома.

Недостатками этого типа межсетевых защитных экранов являются более высокая, чем для пакетных фильтров стоимость; производительность шлюзов прикладного уровня ниже, чем для пакетных фильтров.

6.4.5. Персональные брандмауэры

Центральный межсетевой защитный экран представляет собой очень эффективный механизм обеспечения безопасности сетей. Однако использование центральной брандмауэрной системы имеет определенные концептуальные ограничения. Так, в связи с тем, что брандмауэрная система защищает коммуникационные каналы, проходящие через нее, то защита окажется совершенно бесполезной в случае наличия каналов, обходящих межсетевой экран. Кроме того, центральная брандмауэр-система не защищает от внутренних атак. Она не может также обнаружить атаки на уровне данных в разрешенных соединениях. Такие атаки включают рассылку агрессивных программ в виде вложений электронной почты, загружаемого программного обеспечения из Web-узлов, Java-аплетов и др.

Эффективным решением для противодействия внутренним атакам, вредному программному обеспечению на уровне данных является использование **персональных брандмауэров**. В их задачи входит перекрытие всех возможных брешей, которые содержатся в центральном межсетевом экране и обеспечить полную защиту всех ресурсов персонального компьютера. Для мобильных станций, подключаемых к сети Интернет, персональный брандмауэр становится основным защитным экраном.

В состав персонального брандмауэра входит пользовательский интерфейс (*агент*), отображающий сведения о его состоянии. Он также контролирует атаки на компьютерное ПО и его ресурсы. В персональном брандмауэре реализованы *механизмы безопасности пользовательского режима* и *безопасности режима ядра*. **Механизм безопасности пользовательского режима** выполняет защиту высокого уровня и защищает рабочую среду способом, ориентированным на пользователя. **Механизм безопасности режима ядра** обеспечивает расширенные защитные меры низкого уровня и защища-

ет рабочую среду. Компонентом персонального брандмауэра является также **модуль удаленного доступа**, который упрощает реализацию политики безопасности данного предприятия посредством управления брандмауэром от центрального диспетчера безопасности.

Персональный брандмауэр может быть настроен для работы в режиме "off-line" или "on-line". В первом режиме настройка брандмауэра производится непосредственно на персональном компьютере, а во втором – выполняется централизованно.

6.5. Выводы по разделу

1. Вопросы безопасности компьютерных сетей в сетевых технологиях являются в настоящее время одними из важнейших, так как проникновение злоумышленника в сеть может иметь катастрофические последствия для организации, ставшей жертвой нападения.
2. Одна из причин уязвимости сетей – игнорирование разработчиками объединенной сети требований безопасности, поскольку изначально сеть предназначалась для использования в научных организациях и основывалась на принципах доверия.
3. Важную роль в обеспечении безопасности компьютерной сети играет политика безопасности, представляющая собой совокупность правил, которые должны соблюдаться всеми лицами, желающими получить доступ к корпоративной сети или технологии.
4. Проникновение в компьютерную сеть осуществляется в форме атаки, представляющей собой действия, в результате которых злоумышленник пытается получить доступ к ресурсам сети или нарушить ее функционирование.
5. Информация с точки зрения безопасности обладает такими категориями как конфиденциальность, целостность, аутентичность, аппелируемость, надежность, контроль доступа, контролируемость, устойчивость к умышленным сбоям.
6. Существует ряд моделей защиты информации: модель Биба – субъекты и объекты предварительно разбиваются на несколько уровней доступа, а на их взаимодействие накладываются определенные ограничения; модель Гогена-Мезигера, основанная на теории автоматов; переход из одного состояния в другое выполняется только в соответствии с таблицей разрешений; сазерлендская модель, в которой используется автоматная модель с множеством разрешенных комбинаций состояний, однако упор сделан на взаимодействие субъектов и потоков информации; модель Кларка-Вильсона основана на тщательном оформлении прав доступа, идентификация осуществля-

ется перед и после выполнения команды; последняя модель является самой совершенной на настоящее время.

7. Проникновение нарушителя в систему осуществляется путем физического, системного или удаленного вторжения. Одним из распространенных способов вторжения является взлом паролей.

8. Злоумышленники используют слабые стороны парольной защиты, в частности, слабые пароли, осуществляют атаки по словарю, выполняют подбор паролей, перехватывают трафик с данными о пароле, осуществляют кражу файлов с паролями.

9. Проникновению нарушителя в сеть предшествует фаза внешней и внутренней разведки, при которой он пытается получить максимум информации об атакуемом объекте. Затем последовательно наступают фазы атакующего воздействия, развития атаки и завершение вторжения.

10. Одним из распространенных способов разведки является процедура сканирования сетей с целью обнаружения активных компьютеров, открытых TCP- и UDP-портов.

11. Для идентификации используемой сетевой ОС производится посылка некорректных ICMP- или TCP-пакетов.

12. Для нарушения функционирования сети злоумышленники широко применяют сосредоточенные DoS- и распределенные DDoS-атаки, направленные на нарушение работы или полный отказ обслуживания атакуемой сети или сетевого ресурса. Это осуществляется путем насыщения пропускной способности сети, захвата системных ресурсов, генерирования ошибочных команд или подмены маршрутов.

13. Распределенные атаки выполняются путем "зомбирования" компьютеров других сетей.

14. Обнаружение вторжения злоумышленников может осуществляться на основе выявления аномалий поведения нарушителя: рост ошибок идентификации и аутентификации, повышенная интенсивность использования сетевых служб и обращений к серверу пользователей, нарушение правил обмена, ненормальное поведение программного обеспечения компьютера и сетевых компонентов.

15. Другим способом обнаружения вторжений является сигнатурный анализ. Сигнатура представляет собой шаблон разновидности компьютерной атаки (строка символов, семантическое выражение или формальная модель, последовательность переходов компьютерной системы).

16. К перспективным методам обнаружения вторжений относятся экспертные системы с использованием элементов искусственного интеллекта и методы, основанные на биологических моделях.

17. Для защиты компьютерных систем от злоумышленников разработаны системы обнаружения вторжений IDS (Intrusion Detection Systems). Они ис-

пользуются для выявления не только внешних, но и внутренних нарушителей. Различают пассивные и активные IDS. Первые только фиксируют вторжение, а вторые пытаются его остановить.

18. Эффективным средством повышения компьютерной безопасности является использование защитной оболочки SSH. Весь трафик, передаваемый по протоколу SSH, шифруется. Для организации безопасного доступа применяется процедура аутентификации с использованием асимметричного шифрования с открытым ключом.

19. Для разделения компьютеров защищаемой сети от незащищенной широко применяются межсетевые защитные экраны – брандмауэры. Брандмауэр представляет собой систему, реализующую правила обмена данными между двумя или несколькими компьютерными сетями с целью защиты сети от проникновения злоумышленников.

20. Существует несколько различных реализаций брандмауэров: пакетный фильтр, шлюз прикладного уровня и гибридные шлюзы. В прикладных и гибридных шлюзах применяются специальные программы, называемые прокси-серверами.

21. Прокси-сервер принимает запросы из незащищенной сети, анализирует их и передает безопасные запросы внутренним хостам, предоставляющим соответствующие услуги.

22. Для противодействия внутренним атакам, вредному программному обеспечению на уровне данных необходимо применять персональные брандмауэры. В их задачи входит перекрытие всех возможных брешей, которые содержатся в центральном межсетевом экране и обеспечение полной защиты всех ресурсов персонального компьютера.

23. Более детально вопросы безопасности компьютерных сетей изложены в [19, 20, 25, 29]. Особенно полезной для практического применения является книга [20], в которой приведено много фрагментов программной реализации отдельных видов атак и защиты от них.

6.6. Контрольные вопросы

1. Что входит в понятие "политика безопасности" организации?
2. Проанализируйте правила пользования компьютером и действия персонала при обнаружении нарушения системы безопасности.
3. Почему сеть Интернет обладает слабой защищенностью от проникновения злоумышленников?
4. В чем состоит суть компьютерной атаки и какие существуют виды атак?

5. Перечислите и проанализируйте категории компьютерной информационной безопасности и проведите сравнительный анализ абстрактных моделей защиты информации.
6. Как злоумышленник может заполучить пароль для доступа в компьютерную систему и с какой целью при выборе пароля следует включать небуквенные символы?
7. Каким образом можно узнать пароль, наблюдая за сетевым трафиком?
8. Какими средствами выполняется сканирование сети и что можно узнать в результате таких действий?
9. Почему злоумышленника может интересовать тип используемой операционной системы у предполагаемого объекта атаки?
10. В чем состоит суть точечной атаки типа DoS и как можно нарушить работоспособность сети?
11. Расскажите о распределенной атаке типа DDoS и поясните термин "зомбирование компьютера"?
12. Проанализируйте используемые сценарии насыщения пропускной способности сети.
13. Каким образом можно обнаружить факт нарушения безопасности сети?
14. Что означает понятие "сигнатурный анализ" и как на его основании можно обнаружить вторжение в сеть?
15. Раскройте методы, положенные в основу систем обнаружения вторжений и проанализируйте способы защиты от компьютерных атак типа DoS и DDoS.
16. Каким образом защитная оболочка SSH повышает степень безопасности компьютерной сети?
17. Какие типы защиты может выполнять брандмауэр?
18. Каким образом брандмауэр отсеивает пакеты злоумышленников?
19. Проведите сравнительную характеристику пакетного фильтра и прикладного шлюза, приведите формат таблицы описания правил прохождения пакетов через фильтр и обоснуйте необходимость введения соответствующих ячеек.
20. Поясните значение понятия "прокси-агент", перечислите его функции и укажите место его нахождения в системе безопасности.
21. Какая информация заносится в регистрационный журнал системы безопасности?
22. Какие блоки входят в состав брандмауэра и каковы их функции?
23. Какие действия необходимо выполнить при настройке пакетного фильтра и в чем состоит проблема настройки?
24. С какой целью в защищенной сети устанавливают персональные брандмауэры?

ЗАКЛЮЧЕНИЕ

В данном учебнике изложены теоретические основы построения компьютерных сетей, знание которых поможет студенту подготовиться к выполнению квалификационных выпускных работ бакалавра и магистра, а также к началу практической деятельности. Для практической работы выпускника в качестве разработчика или администратора компьютерных сетей этих знаний явно недостаточно. Каждый раздел и даже подраздел учебника – это самостоятельная область науки и техники, базирующаяся на специфическом математическом аппарате, архитектурных, системных и схемотехнических принципах, алгоритмических и программных особенностях реализации.

По этой причине начало практической деятельности специалиста состоит в глубоком изучении механизмов и особенностей функционирования конкретных компьютерных сетей и их компонентов, системы команд и особенностей программирования и настройки конкретных аппаратных средств, способов администрирования сети, тестирования ее подсистем и всей сети в целом.

Кроме этого, известно, что около половины знаний в этой бурно развивающейся области устаревает за 5 лет. История создания и развития компьютерных сетей свидетельствует, что за последние несколько десятилетий коренным образом изменились концепции построения сетей, их архитектура, способы передачи сигналов и протоколы обмена информацией, способы управления сетями и их администрирования. Достижения микроэлектроники и внедрение микропроцессорной техники и цифровой обработки сигналов, с одной стороны, привели к значительному уменьшению габаритов аппаратных средств и расширению их функциональных возможностей. С другой стороны, каждое из технических средств превратилось в специализированный одно- или многопроцессорный компьютер со своим набором команд, собственной операционной системой и языком программирования, что повышает требования к теоретическому уровню технического персонала, занимающегося обслуживанием и эксплуатацией компьютерных сетей.

Несмотря на то, что конкретные знания архитектуры сетей, системы команд и способов управления быстро меняются, базовые принципы построения компьютерных сетей на многие годы остаются незыблемыми, диалектически изменяясь по законам философии. Так, например, средства частотного уплотнения линий связи были вытеснены с появлением систем временного разделения каналов с импульсно-кодовой модуляцией. Однако с появлением в компьютерных сетях волоконно-оптических линий связи и необходимостью повышения их пропускной способности снова произошел воз-

врат к частотному разделению каналов, но на более высоком технологическом уровне (разделение каналов по длине волны). Пространственная коммутация каналов, доживающая свой век в городских АТС, в настоящее время реализована в некоторых типах коммутаторов локальных сетей.

К основополагающим принципам сетевых технологий относятся способы пакетной коммутации, передачи и синхронизации сигналов, связь полосы пропускания канала со скоростью передачи, способы борьбы с помехами и защиты от ошибок, способы маршрутизации, управления потоками и др. Знание таких принципов позволит молодому специалисту быстро освоить новые сетевые технологии и успешно управлять современными компьютерными сетями.

Авторы надеются, что знания, полученные Вами при работе с данной книгой, помогут создать прочный теоретический фундамент, который позволит Вам успешно решать насущные и перспективные сетевые задачи на протяжении Вашей трудовой деятельности.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

Основной

1. Буров Є. Комп'ютерні мережі. 2-ге оновлене і доповнене вид./ Є.Буров.- Львів: Вид-во "БаК", 2003. – 584 с.
2. Гук М. Аппаратные средства локальных сетей. Энциклопедия / М.Гук.- СПб.: Изд-во "Питер", 2005. – 576 с.
3. Комер Д.Э. Сети TCP/IP. Принципы, протоколы и структура / Д.Э.Комер. – М.: Изд-во "Вильямс", 2003.– 856 с.
4. Контроль та керування корпоративними комп'ютерними мережами: інструментальні засоби та технології: Навчальний посібник/ А.М. Гуржій, С.Ф. Коряк, В.В. Самсонов, О.Я. Скляров.- Харків: Вид-во "Компанія СМІТ", 2004.– 544 с.
5. Кулаков Ю.А. Компьютерные сети / Ю.А.Кулаков, Г.М. Луцкий. – Киев: Изд-во "Юниф", 1998.– 384 с.
6. Кулаков Ю.А. Комп'ютерні мережі. Підручник / Ю.А.Кулаков, Г.М.Луцький; За ред. Ю.С. Ковтанюка – К.: Вид-во "Юніор", 2003.– 400 с.
7. Олифер В.Г. Компьютерные сети. Принципы, технологии, протоколы: Учебник для вузов. 2-е изд. / В.Г.Олифер, Н.А.Олифер. – СПб: Изд-во "Питер", 2005. – 864 с.
8. Столлингс В. Современные компьютерные сети / В.Столлингс. – СПб.: Изд-во "Питер", 2003. – 783 с.
9. Таненбаум Э. Компьютерные сети / Э.Таненбаум.- СПб.: Изд-во "Питер", 2005. – 672 с.

Дополнительный

10. Амато В. Основы организации сетей Cisco. Том 1: Пер. с англ./ В.Амато. – М.: Изд-во "Вильямс", 2004. – 512 с.
11. Амато В. Основы организации сетей Cisco. Том 2: Пер. с англ./ В.Амато.– М.: Изд-во "Вильямс", 2004. – 464 с.
12. Валецька Т.М. Комп'ютерні мережі. Апаратні засоби. Навчальний посібник / Т.М. Валецька.- К.: Центр навчальної літератури, 2002.– 208 с.
13. Варакин Л.Е. Системы связи с шумоподобными сигналами / Л.Е.Варакин. – М.: Радио и связь, 1985.-384 с.

14. Васильев В.И. Системы связи: Учебное пособие для втузов / В.И.Васильев, А.П.Буркин, В.А.Свириденко. – М.: Высшая школа, 1987.–280 с.
15. Галкин В.А. Телекоммуникации и сети: Учебное пособие для вузов / В.А. Галкин, Ю.А. Григорьев.– М.: Изд-во МГТУ им. Н.Э. Баумана, 2003.– 608 с.
16. Гаранин М.В. Системы и сети передачи информации / М.В. Гаранин, В.И. Журавлев, С.В. Кунегин.– М.: Радио и связь, 2001. – 336 с.
17. Коммутаторы локальных сетей D-Link. – М.: 2004.– 89 с. http://www.routers.ru.
18. Кульгин М.В. Компьютерные сети. Практика построения. Для профессионалов / М.В.Кульгин. – СПб.: Изд-во "Питер", 2003.– 462 с.
19. Мамаев М. Технология защиты информации в Интернете. Специальный справочник / М.Мамаев, С.Петренко. – СПб.: Изд-во "Питер", 2002.– 848 с.
20. Медведовский И.Д. Атака из Internet / И.Д. Медведовский и др. – М.: Изд-во "СОЛОН-Р", 2002.–368 с.
21. Морозов В.К. Основы теории информационных сетей: Учебник для студентов вузов / В.К. Морозов, А.В. Долганов. – М.: Высшая школа, 1987. – 271 с.
22. Новиков Ю.В. Локальные сети. Архитектура, алгоритмы, проектирование / Ю.В. Новиков, С.В. Кондратенко.– М.: Изд-во "ЭКОМ", 2000. – 312 с.
23. Олифер Н.А. Базовые технологии локальных сетей. Центр Информационных Технологий / Н.А. Олифер, В.Г. Олифер. http://www.citforum.ru/nets/protocols2/index.shtml.
24. Паркер Т. TCP/IP. Для профессионалов / Т.Паркер, К.Сиян. – СПб.: Изд-во "Питер", 2004.– 859 с.
25. Польман Н. Архитектура брандмауэров для сетей предприятия: Пер. с англ./ Н.Польман, Т.Кразерс. – М.: Изд-во "Вильямс", 2003.– 432 с.
26. Протоколы информационно-вычислительных сетей: Справочник / С.А. Аничкин, С.А.Белов, А.В. Бернштейн и др. – М.: Радио и связь, 1990. – 504 с.
27. Семенов Ю.А. Телекоммуникационные технологии ГНЦ ИТЭФ. http://www.book.iter.ru
28. Скляр Б. Цифровая связь. Теоретические основы и практическое применение: Пер. с англ./ Б.Скляр. – М.: Изд-во "Вильямс", 2003.– 1104 с.
29. Столлингс В. Основы защиты сетей. Приложения и стандарты: Пер. с англ. /В.Столлингс.– М.: Изд-во "Вильямс", 2002. – 432 с.

30. Столлингс В. Компьютерные системы передачи данных: Пер. с англ./ В.Столлингс.– М.: Изд-во "Вильямс", 2002. – 928 с.
31. Феер К. Беспроводная цифровая связь. Методы модуляции и расширения спектра / К.Феер.– М.: Радио и связь, 2000.– 520 с.
32. Храмцов П.Б. Администрирование сети и сервисов Internet. Учебное пособие/ П.Б.Храмцов. – М.: Центр Информационных технологий, 1997.– 384 с.
33. Чернега В.С. Сжатие информации в компьютерных сетях / В.С.Чернега. – Севастополь: Изд-во "СевГТУ", 1997.– 214 с.
34. Чернега В.С. Расчет и проектирование технических средств обмена и передачи информации / В.С. Чернега, В.А. Василенко, В.Н. Бондарев . – М.: Высшая школа., 1990. – 224 с.
35. Шиллер Й. Мобильные коммуникации / Й.Шиллер. – М.: Изд-во "Вильямс", 2002.–384 с.
36. Plattner B. Datenkommunikation und elektronische Post, 2. überarbeitete und erweiterte Auflage / Plattner B. Lanz C. Lubich H. u.a. – Bonn: Addison-Wesley GmbH, 1992. – 486 S.

ПРЕДМЕТНЫЙ УКАЗАТЕЛЬ

Підручник

Чернега Віктор
Платтнер Бернард

КОМП'ЮТЕРНІ МЕРЕЖІ

Учебник

Чернега Виктор,
Платтнер Бернард

КОМПЬЮТЕРНЫЕ СЕТИ

Chernega V., Plattner B.

COMPUTER NETWORKS

Відповідальний за видання
Доценко С.В., проф., д-р фіз.-мат наук

Коректор Ю.М. Кравченко

Художник Л.Ф. Давиденко

Нормоконтролер Г.М. Персідьсков

Здано в набір 23.11.2006. Підписано до друку 28.11.2006. №1272 від XX.11.06

Формат 60 x 90 1/16. Пап. тип. №1. Офсет. друк. Ум. др. арк. 34,85
Тираж 500 прим. Зам. №

Видавництво СевНТУ, 99053, м. Севастополь-53, Студмістечко, НМЦ,
тел. 0692 23-52-10 E-mail: root@sevgtu.sebastopol.ua

Надруковано в типографії "Експрес-Друк"
99053, м. Севастополь, вул. Вакуленчука, 29