

Министерство образования и науки Республики Казахстан
Карагандинский государственный технический университет

Утверждаю
Первый проректор

_____ 2007г.

УЧЕБНО-МЕТОДИЧЕСКИЙ КОМПЛЕКС ДИСЦИПЛИНЫ ПРЕПОДАВАТЕЛЯ

по дисциплине: Вычислительные системы и сети
(код и наименование дисциплины)

для студентов специальностей 050702 - Автоматизация и управление
(шифр и наименование специальности)

Факультет электромеханический
Кафедра Автоматизации производственных процессов им.В.Ф.Бырьки

Предисловие

Учебно-методический комплекс дисциплины преподавателя разработан:
Крицким Антоном Борисовичем, преподавателем кафедры АПП

Обсужден на заседании кафедры _____ АПП
(наименование кафедры)

Протокол № _____ от " ____ " _____ 200_г

Зав. кафедрой АПП

проф., д.т.н. _____ Брейдо И.В. « ____ » _____ 200_г.
(подпись)

Одобен методическим бюро факультета _____ ЭМФ
(наименование факультета)

Протокол № _____ от « ____ » _____ 200_г.

Председатель

_____ Умбеталин Т.С. « ____ » _____ 200_г.
(подпись)

1 Рабочая учебная программа

1.1 Сведения о преподавателе и контактная информация

Крицкий Антон Борисович, преподаватель кафедры АПП КарГТУ.

Кафедра АПП им.В.Ф.Бырьки находится в главном корпусе КарГТУ, 131 аудитория, контактный телефон: 56-51-84 (кафедра), 56-53-25 (4 корпус 106 ауд.), электронный адрес преподавателя: ant55@mail.ru.

1.2 Трудоемкость дисциплины(дневное/заочное сокр/2-е высшее)

Семестр	Количество кредитов	Вид занятий					Количество часов СРС	Общее количество часов	Форма контроля
		количество контактных часов			количество часов СРС	всего часов			
		лекции	практические занятия	лабораторные занятия					
5	2	15	–	15	30	60	30	90	Экзамен

1.3 Характеристика дисциплины

Дисциплина "Вычислительные системы и сети" является одной из профилирующих для студентов специальностей 050702 "Автоматизация и управление" в соответствии с Государственным стандартом ГОСО РК и согласно учебного плана специальности входит в число обязательных.

1.4 Цель дисциплины

Целью изучения данной дисциплины является изучение современных вычислительных сетей и систем и приобретение практических навыков решения профессиональных задач в среде современных информационных технологий для предметной области "Автоматизация технологических процессов и производств".

1.5 Задачи дисциплины

Задачи дисциплины: формировать у специалиста твердые основы знаний, высокую математическую культуру и практические навыки, достаточные для успешной производственной деятельности и позволяющие ему самостоятельно осваивать новые необходимые знания и достижения в области программирования и решения инженерных задач.

В результате изучения данной дисциплины студенты должны:

иметь представление: Об архитектуре и топологии многопроцессорных вычислительных систем.

Об основах сетевых технологий, аппаратном и программном обеспечении сетей.

знать:

Назначение, область применения и способы оценки производительности многопроцессорных вычислительных систем.

Архитектуры вычислительных систем.

Математические основы, способы организации и особенности проектирования высокопроизводительных процессоров.

Общие принципы построения вычислительных сетей.

Основы передачи дискретных данных.

Базовые технологии локальных сетей.

Построение локальных сетей по стандартам физического и канального уровней.

уметь:

Грамотно производить комплектацию и агрегатирование вычислительных систем и сетевого оборудования, осуществлять проектирование топологии локальной сети.

1.6 Пререквизиты

Для изучения данной дисциплины необходимо усвоение следующих дисциплин (с указанием разделов (тем)):

Дисциплина	Наименование разделов (тем)
1.Микропроцессорные средства и системы	Логические устройства, компараторы, аналоговые и цифровые преобразователи
2.Информатика	Устройство персонального компьютера
3.Технологические процессы и производственные линии	Автоматизированные и автоматические производственные линии

1.7 Постреквизиты

Знания, полученные при изучении дисциплины «Вычислительные системы и сети», используются при освоении следующих дисциплин: «Компьютерные системы технологического контроля и управления».

1.8 Содержание дисциплины

1.8.1 Содержание дисциплины по видам занятий и их трудоемкость

Наименование раздела, (темы)	Трудоемкость по видам занятий, ч.				
	лекции	практические	лабораторные	СРСП	СРС
Лекция 1. Назначение, область применения и способы оценки производительности многопроцессорных вычислительных систем . Архитектура вычислительных систем	1	–	1	2	2
Лекция 2. Принципы построения коммуникационных сред. Математические основы, способы организации и особенности проектирования высокопроизводительных процессоров	1	–	1	2	2
Лекция 3. Коммутаторы для многопроцессорных вычислительных систем. Требования к компонентам МВС	1	–	1	2	2
Лекция 4. Кластеры и массивно-параллельные системы различных производителей	1	–	1	2	2
Лекция 5. От централизованных систем - к вычислительным сетям. Основные проблемы построения сетей	2	–	2	2	2
Лекция 6. Модель взаимодействия открытых систем и проблемы стандартизации	2	–	2	4	4
Лекция 7. Локальные и глобальные сети. Требования, предъявляемые к современным вычислительным сетям	2	–	2	4	4
Лекция 8. Линии связи	2	–	2	4	4
Лекция 9. Методы передачи дискретных данных на физическом уровне	2	–	2	4	4
Лекция 10. Методы передачи данных канального уровня. Методы коммутации.	1		1	4	4
ИТОГО:	15	–	15	30	30

1.8.2 Тематика курсовых работ (проектов):

Не предусмотрен

1.9 Список основной литературы

1. В.В.Воеводин, Вл.В.Воеводин - "Параллельные вычисления", издательство "БХВ ", 2002.
2. В. Корнеев В. , "Современные микропроцессоры", издательство "Нолидж", 2003.
3. Э. Таненбаум, "Архитектура компьютера", СПб, издательство «Питер», 2002.
4. В.Г.Олифер, Н.А.Олифер Компьютерные сети, издательство "Питер", 2000.
5. К. Хамахер, З. Вранешич, С. Заки, Организация ЭВМ. СПб, издательство «Питер», 2003 г.

1.10 Список дополнительной литературы

1. С.Немнюгин, О.Стефик, "Параллельное программирование для многопроцессорных вычислительных систем" СПб.: "БХВ-Петербург", 2002
2. Ю.И.Нечаев. Искусственный интеллект: концепции и приложения. СПб: Изд.центр СПбГМТУ, 2002 г.
3. СуперЭВМ. Аппаратная и программная реализация / Под ред. С.Фернбаха : Пер. с англ. - М.: Радио и связь, 1991. - 320 с.
4. ТИИЭР. Тематический выпуск "Техника супер-ЭВМ", 1989. - N 12
5. Каган Б.М. Электронные вычислительные машины и системы : Учебн.пособие для вузов. - М.: Энергоатомиздат, 1985. - 552 с.

1.11 Критерии оценки знаний студентов

Экзаменационная оценка по дисциплине определяется как сумма максимальных показателей успеваемости по рубежным контролям (до 50%) и итоговой аттестации (экзамену) (до 50%) и составляет значение до 100% в соответствии с таблицей.

Оценка по буквенной системе	Баллы	%-ное содержание	Оценка по традиционной системе
A	4,0	95-100	Отлично
A-	3,67	90-94	
B+	3,33	85-89	Хорошо
B	3,0	80-84	
B-	2,67	75-89	

C+	2,33	70-74	Удовлетворительно
C	2,0	65-69	
C-	1,67	60-64	
D+	1,33	55-59	
D	1,0	50-54	
F	0	0-49	Неудовлетворительно

Рубежный контроль проводится на 5-й, 10-й и 15-й неделях обучения и складывается исходя из следующих видов контроля:

Вид контроля	%ое содержание	Академический период обучения, неделя															Итого, %	
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15		
Посещаемость	15	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	15	
Лаб. работы	20		3		3		3		3			3					5	20
СРСП	10			2			2			2			2		2		10	
Модуль	15					5					5						5	15
Экзамен	40																40	
Всего	100	1	4	3	4	6	6	1	4	3	6	5	2	1	3	11	60	

Продолжение приложения В

1.12 Политика и процедуры

При изучении дисциплины «Вычислительные системы и сети» прошу соблюдать следующие правила:

1. Не опаздывать на занятия.
2. Не пропускать занятия без уважительной причины, в случае болезни прошу предоставлять справку, в других случаях – объяснительную записку.
3. Быть подготовленным: на лекционных занятиях по материалам предыдущих лекций, на СРСП – по поставленным вопросам, на лабораторных занятиях – знать цели и ход выполнения текущей работы.
4. Аккуратно вести конспект лекций, оформлять лабораторные работы в соответствии с правилами и требованиями КарГТУ.
5. Активно участвовать в учебном процессе.
6. Быть терпимыми, открытыми, откровенными и доброжелательными к сокурсникам и преподавателям.

1.13 Учебно-методическая обеспеченность дисциплины

Ф.И.О автора	Наименование учебно-методической литературы	Издательство, год издания	Количество экземпляров	
			в библиотеке	на кафедре
Основная литература				
В.В.Воеводин, Вл.В.Воеводин	Параллельные вычисления	"БХВ", 2002	3	–
В. Корнеев	Современные микропроцессоры	"Нолидж", 2003	2	–
Э. Таненбаум	Архитектура компьютера	«Питер», 2002.	3	–
В.Г.Олифер, Н.А.Олифер	Компьютерные сети	«Питер», 2000	2	–
К. Хамахер, З. Вранешич, С. Заки	Организация ЭВМ	«Питер», 2003	2	–

Дополнительная литература				
С.Немнюгин, О.Стесик	Параллельное программирование для многопроцессорных вычислительных систем	БХВ-Петербург, 2002	1	—
Ю.И.Нечаев	Искусственный интеллект: концепции и приложения	СПб: Изд.центр СПбГМТУ, 2002	1	—
Под ред. С.Фернбаха	СуперЭВМ. Аппаратная и программная реализация	Радио и связь, 1991	1	—
—	. ТИИЭР. Тематический выпуск "Техника супер-ЭВМ" 1989. - N 12	—	1	—
Каган Б.М.	Электронные вычислительные машины и системы	Энергоатомиздат, 1985	2	—

2 График выполнения и сдачи заданий по дисциплине

Вид контроля	Цель и содержание задания	Рекомендуемая литература	Продолжительность выполнения, ч.	Форма контроля	Срок сдачи
Лабораторная работа № 1	Изучение особенностей функционирования локальной вычислительной сети со случайным методом доступа к моноканалу	Крицкий А.Б. Лабораторные работы по курсу “Вычислительные системы и сети”	8	Отчет по лабораторной работе	4 недели обучения
Лабораторная работа № 2	Изучение особенностей функционирования локальной вычислительной сети с маркерным методом доступа к моноканалу	Крицкий А.Б. Лабораторные работы по курсу “Вычислительные системы и сети”	8	Отчет по лабораторной работе	8 недели обучения
Лабораторная работа № 3	Проектирование систем логического управления технологическими процессами	Крицкий А.Б. Лабораторные работы по курсу “Вычислительные системы и сети”	8	Отчет по лабораторной работе	12 недели обучения
Лабораторная работа № 4	Исследование возможностей ЛВС и кластера.	Крицкий А.Б. Лабораторные работы по курсу “Вычислительные системы и сети”	6	Отчет по лабораторной работе	15 недели обучения

3 Конспект лекций

Лекция 1

Тема: Назначение, область применения и способы оценки производительности многопроцессорных вычислительных систем. Архитектура вычислительных систем.

План лекции

1. Введение.
2. Области применения
3. Производительность МВС
4. Классификация архитектур по параллельной обработке данных

Раздел 1 : Назначение, область применения и способы оценки производительности многопроцессорных вычислительных систем.

1.1 Введение.

В настоящее время сфера применения многопроцессорных вычислительных систем (МВС) непрерывно расширяется, охватывая все новые области в самых различных отраслях науки, бизнеса и производства. Стремительное развитие кластерных систем создает условия для использования многопроцессорной вычислительной техники в реальном секторе экономики.

1.2 Области применения

Наряду с расширением области применения, по мере совершенствования МВС происходит усложнение и увеличение количества задач в областях, традиционно использующих высокопроизводительную вычислительную технику. В настоящее время выделен круг фундаментальных и прикладных проблем, объединенный понятием "Grand challenges", эффективное решение которых возможно только с использованием сверхмощной вычислительных ресурсов. Этот круг включает следующие задачи:

- Предсказания погоды, климата и глобальных изменений в атмосфере
- Науки о материалах
- Построение полупроводниковых приборов
- Сверхпроводимость
- Структурная биология
- Разработка фармацевтических препаратов
- Генетика
- Квантовая хромодинамика
- Астрономия
- Транспортные задачи
- Гидро- и газодинамика
- Управляемый термоядерный синтез
- Эффективность систем сгорания топлива
- Геоинформационные системы
- Разведка недр

- Наука о мировом океане
- Распознавание и синтез речи
- Распознавание изображений

1.3 Производительность МВС

Главной отличительной особенностью многопроцессорной вычислительной системы является ее производительность, т.е. количество операций, производимых системой за единицу времени. Различают пиковую и реальную производительность. Под пиковой понимают величину, равную произведению пиковой производительности одного процессора на число таких процессоров в данной машине. При этом предполагается, что все устройства компьютера работают в максимально производительном режиме. Пиковая производительность компьютера вычисляется однозначно, и эта характеристика является базовой, по которой производят сравнение высокопроизводительных вычислительных систем. Чем больше пиковая производительность, тем теоретически быстрее пользователь сможет решить свою задачу. Пиковая производительность есть величина теоретическая и, вообще говоря, недостижимая при запуске конкретного приложения. Реальная же производительность, достигаемая на данном приложении, зависит от взаимодействия программной модели, в которой реализовано приложение, с архитектурными особенностями машины, на которой приложение запускается.

Существуют два способа оценки пиковой производительности компьютера. Один из них опирается на число команд, выполняемых компьютером в единицу времени. Единицей измерения, как правило, является MIPS (Million Instructions Per Second). Производительность, выраженная в MIPS, говорит о скорости выполнения компьютером своих же инструкций. Но, во-первых, заранее не ясно, в какое количество инструкций отобразится конкретная программа, а, во-вторых, каждая программа обладает своей спецификой, и число команд от программы к программе может меняться очень сильно. В связи с этим данная характеристика дает лишь самое общее представление о производительности компьютера.

Другой способ измерения производительности заключается в определении числа вещественных операций, выполняемых компьютером в единицу времени. Единицей измерения является Flops (Floating point operations per second) – число операций с плавающей точкой, производимых компьютером за одну секунду. Такой способ является более приемлемым для пользователя, поскольку последний знает вычислительную сложность своей программы и, пользуясь этой характеристикой, может получить нижнюю оценку времени ее выполнения.

2.1. Классификация архитектур по параллельной обработке данных

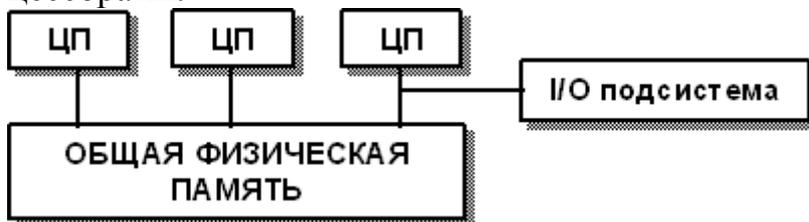
В 1966 году **М.Флинном** (Flynn) был предложен чрезвычайно удобный подход к классификации архитектур вычислительных систем. В основу было положено понятие потока, под которым понимается последовательность эле-

ментов, команд или данных, обрабатываемая процессором. Соответствующая система классификации основана на рассмотрении числа потоков инструкций и потоков данных и описывает четыре архитектурных класса:

SISD	=	Single	Instruction	Single	Data
MISD	=	Multiple	Instruction	Single	Data
SIMD	=	Single	Instruction	Multiple	Data
MIMD	=	Multiple	Instruction	Multiple	Data

2.2. SMP архитектура ▲

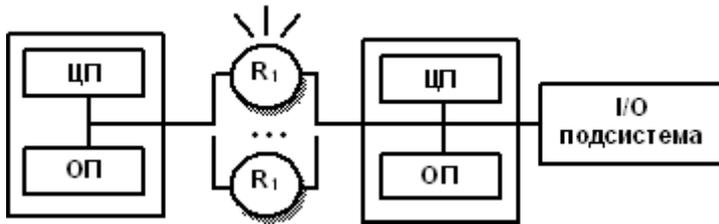
SMP архитектура (symmetric multiprocessing) - симметричная многопроцессорная архитектура. Главной особенностью систем с архитектурой SMP является наличие общей физической памяти, разделяемой всеми процессорами.



Схематический вид SMP-архитектуры.

2.3. MPP архитектура ▲

MPP архитектура (massive parallel processing) - массивно-параллельная архитектура. Главная особенность такой архитектуры состоит в том, что память физически разделена. В этом случае система строится из отдельных модулей, содержащих процессор, локальный банк оперативной памяти (ОП), два коммуникационных процессора (рутера) или сетевой адаптер, иногда - жесткие диски и/или другие устройства ввода/вывода. Один рутер используется для передачи команд, другой - для передачи данных. По сути, такие модули представляют собой полнофункциональные компьютеры (см. рис.). Доступ к банку ОП из данного модуля имеют только процессоры (ЦП) из этого же модуля. Модули соединяются специальными коммуникационными каналами. Пользователь может определить логический номер процессора, к которому он подключен, и организовать обмен сообщениями с другими процессорами. Используются два варианта работы операционной системы (ОС) на машинах MPP архитектуры. В первом полноценная операционная система (ОС) работает только на управляющей машине (front-end), на каждом отдельном модуле работает сильно урезанный вариант ОС, обеспечивающий работу только расположенной в нем ветви параллельного приложения. Во втором варианте на каждом модуле работает полноценная UNIX-подобная ОС, устанавливаемая отдельно на каждом модуле.



Схематический вид архитектуры с отдельной памятью

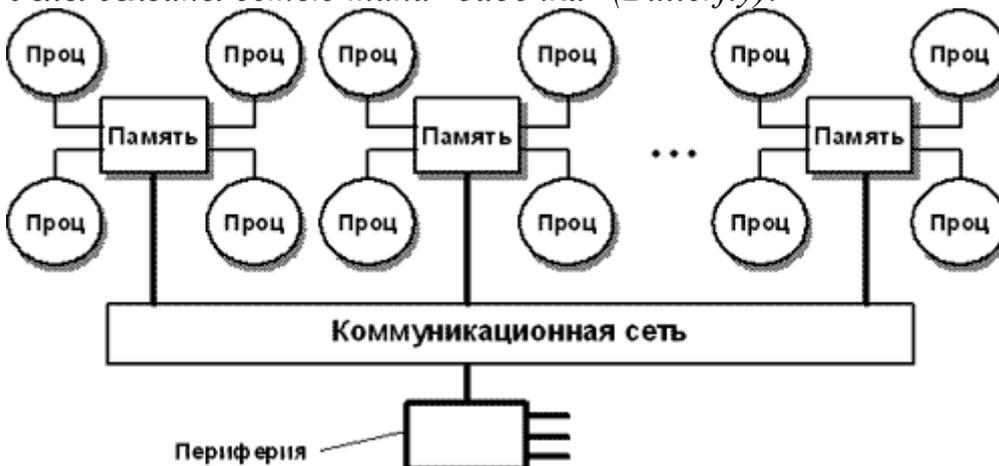
2.4. Гибридная архитектура (NUMA) Организация когерентности многоуровневой иерархической памяти. ▲

Гибридная архитектура NUMA (nonuniform memory access). Главная особенность такой архитектуры - неоднородный доступ к памяти.

Гибридная архитектура воплощает в себе удобства систем с общей памятью и относительную дешевизну систем с отдельной памятью. Суть этой архитектуры - в особой организации памяти, а именно: память является физически распределенной по различным частям системы, но логически разделяемой, так что пользователь видит единое адресное пространство. Система состоит из однородных базовых модулей (плат), состоящих из небольшого числа процессоров и блока памяти. Модули объединены с помощью высокоскоростного коммутатора. Поддерживается единое адресное пространство, аппаратно поддерживается доступ к удаленной памяти, т.е. к памяти других модулей. При этом доступ к локальной памяти осуществляется в несколько раз быстрее, чем к удаленной. По существу архитектура NUMA является MPP (массивно-параллельная архитектура) архитектурой, где в качестве отдельных вычислительных элементов берутся SMP (симметричная многопроцессорная архитектура) узлы.

Структурная схема компьютера с гибридной сетью: четыре процессора связываются между собой при помощи кроссбара в рамках одного SMP узла.

Узлы связаны сетью типа "бабочка" (Butterfly):



2.5. PVP архитектура ▲

PVP (Parallel Vector Process) - параллельная архитектура с векторными процессорами.

Основным признаком PVP-систем является наличие специальных векторно-

конвейерных процессоров, в которых предусмотрены команды однотипной обработки векторов независимых данных, эффективно выполняющиеся на конвейерных функциональных устройствах. Как правило, несколько таких процессоров (1-16) работают одновременно с общей памятью (аналогично SMP) в рамках многопроцессорных конфигураций. Несколько таких узлов могут быть объединены с помощью коммутатора (аналогично MPP). Поскольку передача данных в векторном формате осуществляется намного быстрее, чем в скалярном (максимальная скорость может составлять 64 Гб/с, что на 2 порядка быстрее, чем в скалярных машинах), то проблема взаимодействия между потоками данных при распараллеливании становится несущественной. И то, что плохо распараллеливается на скалярных машинах, хорошо распараллеливается на векторных. Таким образом, системы PVP архитектуры могут являться машинами общего назначения (general purpose systems). Однако, поскольку векторные процессоры весьма дороги, эти машины не будут являться общедоступными.

2.6. Кластерная архитектура ▲

Кластер представляет собой два или больше компьютеров (часто называемых узлами), объединяемых при помощи сетевых технологий на базе шинной архитектуры или коммутатора и предстающих перед пользователями в качестве единого информационно-вычислительного ресурса. В качестве узлов кластера могут быть выбраны серверы, рабочие станции и даже обычные персональные компьютеры. Преимущество кластеризации для повышения работоспособности становится очевидным в случае сбоя какого-либо узла: при этом другой узел кластера может взять на себя нагрузку неисправного узла, и пользователи не заметят прерывания в доступе. Возможности масштабируемости кластеров позволяют многократно увеличивать производительность приложений для большего числа пользователей. технологий (**Fast/Gigabit Ethernet**, **Myrinet**) на базе шинной архитектуры или коммутатора. Такие суперкомпьютерные системы являются самыми дешевыми, поскольку собираются на базе стандартных комплектующих элементов ("off the shelf"), процессоров, коммутаторов, дисков и внешних устройств.

Литература:

1. В.В.Воеводин, Вл.В.Воеводин - "Параллельные вычисления", издательство "БХВ ", 2002.
2. В. Корнеев В. , "Современные микропроцессоры", издательство "Нолидж", 2003.
3. Э. Таненбаум, "Архитектура компьютера", СПб, издательство «Питер», 2002.
4. В.Г.Олифер, Н.А.Олифер Компьютерные сети, издательство "Питер", 2000.

5. К. Хамахер, З. Вранешич, С. Заки, Организация ЭВМ. СПб, издательство «Питер», 2003 г.

Контрольные задания для СРС

- 1 Привести пример 10 самых производительных МВС.*
- 2 Найти сведения о выполнении научных или производственных задач, решаемых при помощи МВС.*
- 3 Привести пример компьютера с SMP архитектурой.*
- 4 Привести пример компьютера с MPP архитектурой.*
- 5 Привести пример компьютера с Гибридной архитектурой.*

Лекция 2

Тема: Принципы построения коммуникационных сред. Математические основы, способы организации и особенности проектирования высокопроизводительных процессоров

План лекции

1. Введение.
2. Коммуникационные среды
3. Разновидности процессоров.

Раздел 1: Принципы построения коммуникационных сред.

1.1 Введение

В самом общем смысле архитектуру компьютера можно определить как способ соединения компьютеров между собой, с памятью и с внешними устройствами. Реализация этого соединения может идти различными путями. Конкретная реализация соединений такого рода называется коммуникационной средой компьютера. Одна из самых простых реализаций – это использование общей шины, к которой подключаются как процессоры, так и память. Сама шина состоит из определенного числа линий связи, необходимых для передачи адресов, данных и управляющих сигналов между процессором и памятью. Этот способ реализован в SMP системах. Основным недостатком таких систем, как было указано выше, является плохая масштабируемость. Увеличение, даже незначительное числа устройств на шине вызывает заметные задержки при обмене с памятью и катастрофическое падение производительности системы в целом. Необходимы другие подходы для построения коммуникационной среды, и одним из них является разделение памяти на независимые модули и обеспечение возможности доступа разных процессоров к различным модулям одновременно посредством использования различного рода коммутаторов (см. раздел 5 настоящего учебника).

1.2 Примеры построения коммуникационных сред на основе масштабируемого когерентного интерфейса SCI

SCI (Scalable Coherent Interface) принят как стандарт в 1992 (ANSI/IEEE Std 1596-1992). Предназначен для достижения высоких скоростей передачи с малым временем задержки, при этом обеспечивая масштабируемую архитектуру, позволяющую строить системы, состоящие из множества блоков. Представляет собой комбинацию шины и локальной сети, обеспечивает реализацию когерентности кэш-памяти, размещаемой в узле SCI, посредством механизма распределенных директорий, который улучшает производительность, скрывая затраты на доступ к удаленным данным в модели с распределенной разделяемой памятью. Производительность передачи данных обычно находится в пределах от 200 Мбайт/с до 1000 Мбайт/с на расстояниях десятков метров с использованием электрических кабелей и километров с использованием оптоволокон. SCI уменьшает время межузловых коммуникаций по

сравнению с традиционными схемами передачи данных в сетях путем устранения обращений к программным уровням – операционной системе и библиотекам времени выполнения; коммуникации представляются как часть простой операции загрузки данных процессором (командами load или store). Обычно обращение к данным, физически расположенным в памяти другого вычислительного узла и не находящимся в кэше, приводит к формированию запроса на удаленный узел для получения необходимых данных, которые в течение нескольких микросекунд доставляются в локальный кэш, и выполнение программы продолжается. Старый подход требовал формирования пакетов на программном уровне с последующей передачей их аппаратному обеспечению. Точно также происходил и прием, в результате чего задержки были в сотни раз больше, чем у SCI. Однако, для совместимости SCI имеет возможность переносить пакеты других протоколов. Другое преимущество SCI – использование простых протоколов типа RISC, которые обеспечивают большую пропускную способность. Узлы с адаптерами SCI могут использоваться для соединения коммутаторы или же соединяться в кольцо. Обычно каждый узел оказывается включенным в два кольца (рис. 3.1).

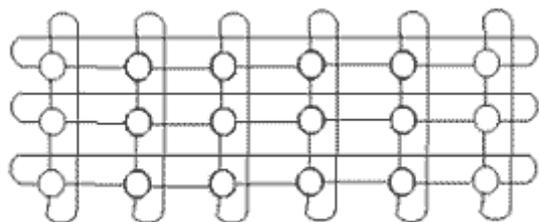


Рис.3.1. Матрица узлов кластера на основе сети SCI

1.3 Коммуникационная среда MYRINET

Сетевую технологию **Myrinet** представляет компания **Myricom**, которая впервые предложила свою коммуникационную технологию в 1994 году, а на сегодня имеет уже более 1000 инсталляций по всему миру. Технология Myrinet основана на использовании многопортовых коммутаторов при ограниченных несколькими метрами длинах связей узлов с портами коммутатора. Узлы в Myrinet соединяются друг с другом через коммутатор (до 16 портов). Максимальная длина линий связи варьируется в зависимости от конкретной реализации.

Как коммутируемая сеть, аналогичная по структуре сегментам Ethernet, соединенным с помощью коммутаторов, Myrinet может одновременно передавать несколько пакетов, каждый из которых идет со скоростью, близкой к 2 Гбит/с. В отличие от некоммутированных Ethernet и FDDI сетей, которые разделяют общую среду передачи, совокупная пропускная способность сети Myrinet возрастает с увеличением количества машин. На сегодняшний день Myrinet чаще всего используют как локальную сеть (LAN) сравнительно небольшого физического размера, связывая вместе компьютеры внутри комнаты или здания. Из-за своей высокой скорости, малого времени задержки, прямой коммутации и умеренной стоимости, Myrinet особенно популярен для объединения компьютеров в кластеры. Myrinet также используется как

системная сеть (System Area Network, SAN), которая может объединять компьютеры в кластер внутри стойки с той же производительностью, но с более низкой стоимостью, чем Myrinet LAN. Пакеты Myrinet могут иметь любую длину. Таким образом, они могут включать в себя другие типы пакетов, включая IP пакеты. Соединение вычислительных узлов с адаптерами Myrinet в сеть происходит с помощью коммутаторов, которые имеют сейчас 4, 8, 12 или 16 портов. В коммутаторах используется передача пакетов путем установления соединения на время передачи, для маршрутизации сообщений используется алгоритм прокладки пути (wormhole, «чреваточина»). Коммутаторы, как и сетевые адаптеры, построены на специализированных микропроцессорах LANai фирмы Myricom.

1.4 Коммуникационная среда Raceway ▲

Коммуникационная среда **Raceway** обеспечивает пропускную способность на уровне 1 Гбайт/с; среда передачи создается с помощью коммутатора фирмы Cypress и соответствующих сетевых адаптеров. Коммутатор имеет 6 портов, пропускная способность каждого порта составляет 160 Мбайт/с. Порт состоит из 32 сигнальных линий данных и 5 управляющих линий. При начале транзакции среда Raceway предварительно устанавливает соединение, задержка в коммутаторе при установлении соединения составляет примерно 125 нс. Структуры вычислительных систем, создаваемых при помощи Raceway, аналогичны тем, которые применяются в случае использования сети Myrinet или коммутаторов и адаптеров SCI. Разница заключается в количестве портов коммутаторов, форматах передаваемых пакетов и в протоколах.

Коммуникационная среда Raceway принята в качестве стандарта (ANSI/VINA 5-1994).

1.5 Коммуникационные среды на базе транспьютероподобных процессоров ▲

Транспьютер – это слово, производное от слов транзистор и компьютер.

В настоящее время такие системы на используются и информация о них представляет чисто исторический интерес.

Транспьютер - это микроэлектронный прибор, объединяющий на одном кристалле микропроцессор, быструю память, интерфейс внешней памяти и каналы ввода-вывода (линки), предназначенные для подключения аналогичных приборов. Прибор спроектирован таким образом, чтобы максимально облегчить построение параллельных вычислительных систем. При соединении транспьютерных элементов между собой требуется минимальное число дополнительных интегральных схем. Связь между транспьютерами осуществляется путем непосредственного соединения линка одного прибора с линком другого. Это позволяет создавать сети с различными топологиями с большим числом элементов.

Транспьютер представляет собой микропроцессор, в состав которого входят:

1. ЦПУ с сокращенным набором команд (RISC).
2. 64-разрядный сопроцессор (FPU) плавающей арифметики с высокой пиковой производительностью, работающий параллельно с ЦПУ.
3. Внутрикристалльное ОЗУ
4. 32-разрядная шина памяти
5. Четыре последовательных двунаправленных линии связи (Link), обеспечивающих взаимодействие транспьютера с внешним миром, работающих параллельно.
6. Таймер.
7. Системные управляющие сигналы Инициализация, Анализ, Ошибка, управляющие загрузкой и анализом состояния транспьютера, сигнализирующие об ошибках.
8. Интерфейс внешних событий (Event), обеспечивающий асинхронную связь внутреннего процесса и внешнего события.

Раздел 2: Математические основы, способы организации и особенности проектирования высокопроизводительных процессоров

2.1. Ассоциативные процессоры ▲

Существующие в настоящее время алгоритмы прикладных задач, системное программное обеспечение и аппаратные средства преимущественно ориентированы на традиционную адресную обработку данных. Данные должны быть представлены в виде ограниченного количества форматов (например, массивы, списки, записи), должна быть явно создана структура связей между элементами данных посредством указателей на адреса элементов памяти, при обработке этих данных должна быть выполнена совокупность операций, обеспечивающих доступ к данным по указателям. Такой подход обуславливает громоздкость операционных систем и систем программирования, а также служит препятствием к созданию вычислительных средств с архитектурой, ориентированной на более эффективное использование параллелизма обработки данных.

Ассоциативный способ обработки данных позволяет преодолеть многие ограничения, присущие адресному доступу к памяти, за счет задания некоторого критерия отбора и проведение требуемых преобразований, только над теми данными, которые удовлетворяют этому критерию. Критерием отбора может быть совпадение с любым элементом данных, достаточным для выделения искомым данным из всех данных. Поиск данных может происходить по фрагменту, имеющему большую или меньшую корреляцию с заданным элементом данных.

2.2. Конвейерные процессоры. ▲

Процессоры современных компьютеров используют особенную технологию - конвейеры, которые позволяют обрабатывать более одной команды одновременно.

Обработка команды может быть разделена на несколько основных этапов, назовем их микрокомандами. Выделим основные пять микрокоманд:

- Выборка команды
- Расшифровка команды
- Выборка необходимых операндов
- Выполнение команды
- Сохранение результатов

Все этапы команды задействуются только один раз и всегда в одном и том же порядке: одна за другой. Это, в частности означает, что если первая микрокоманда выполнила свою работу и передала результаты второй, то для выполнения текущей команды она больше не понадобится, и, следовательно, может приступить к выполнению следующей команды. Выделим каждую команду в отдельную часть устройства и расположим их в порядке выполнения. В первый момент времени выполняется первая микрокоманда. Она завершает свою работу и начинает выполняться вторая микрокоманда, в то время как первая готова для выполнения следующей инструкции. Первая инструкция может считаться выполненной, когда завершать работу все пять микрокоманд.

2.3. Матричные процессоры ▲

Наиболее распространенными из систем, класса: **один поток команд - множество - потоков данных (SIMD)**, являются **матричные системы**, которые лучше всего приспособлены для решения задач, характеризующихся параллелизмом независимых объектов или данных. Организация систем подобного типа на первый взгляд достаточно проста. Они имеют общее управляющее устройство, генерирующее поток команд и большое число процессорных элементов, работающих параллельно и обрабатывающих каждая свой поток данных. Таким образом, производительность системы оказывается равной сумме производительностей всех процессорных элементов. Однако на практике, чтобы обеспечить достаточную эффективность системы при решении широкого круга задач необходимо организовать связи между процессорными элементами с тем, чтобы наиболее полно загрузить их работой. Именно характер связей между процессорными элементами и определяет разные свойства системы.

2.4. Клеточные и ДНК процессоры ▲

В настоящее время в поисках реальной альтернативы полупроводниковым технологиям создания новых вычислительных систем ученые обращают все большее внимание на биотехнологии, или **биокомпьютинг**, который представляет собой гибрид информационных, молекулярных технологий, также биохимии. Биокомпьютинг позволяет решать сложные вычислительные задачи, пользуясь методами, принятыми в биохимии и молекулярной биологии, организуя вычисления при помощи живых тканей, клеток, вирусов и биомолекул. Наибольшее распространение получил подход, где в качестве основ-

ного элемента (процессора) используются молекулы дезоксирибонуклеиновой кислоты. Центральное место в этом подходе занимает так называемый ДНК - процессор. Кроме ДНК в качестве био-процессора могут быть использованы также белковые молекулы и биологические мембраны.

ДНК-процессоры

Так же, как и любой другой процессор, ДНК процессор характеризуется структурой и набором команд. В нашем случае структура процессора - это структура молекулы ДНК. А набор команд - это перечень биохимических операций с молекулами. Принцип устройства компьютерной ДНК-памяти основан на последовательном соединении четырех нуклеотидов (основных кирпичиков ДНК-цепи). Три нуклеотида, соединяясь в любой последовательности, образуют элементарную ячейку памяти - кодон, которые затем формируют цепь ДНК. Основная трудность в разработке ДНК-компьютеров связана с проведением избирательных однокодонных реакций (взаимодействий) внутри цепи ДНК. Однако прогресс есть уже и в этом направлении. Уже есть экспериментальное оборудование, позволяющее работать с одним из 1020 кодонов или молекул ДНК. Другой проблемой является самосборка ДНК, приводящая к потере информации. Ее преодолевают введением в клетку специальных ингибиторов - веществ, предотвращающих химическую реакцию самосшивки.

Клеточные компьютеры

Клеточные компьютеры представляют собой самоорганизующиеся колонии различных "умных" микроорганизмов, в геном которых удалось включить некую логическую схему, которая могла бы активизироваться в присутствии определенного вещества. Для этой цели идеально подошли бы бактерии, стакан с которыми и представлял бы собой компьютер. Такие компьютеры очень дешевы в производстве. Им не нужна столь стерильная атмосфера, как при производстве полупроводников.

Главным свойством компьютера такого рода является то, что каждая их клетка представляет собой миниатюрную химическую лабораторию. Если биоорганизм запрограммирован, то он просто производит нужные вещества. Достаточно вырастить одну клетку, обладающую заданными качествами, и можно легко и быстро вырастить тысячи клеток с такой же программой.

2.5. Коммуникационные процессоры ▲

Коммуникационные процессоры - это микрочипы, являющие собой нечто среднее между жесткими специализированными интегральными микросхемами и гибкими процессорами общего назначения. Коммуникационные процессоры программируются, как и привычные нам ПК-процессоры, но построены с учетом сетевых задач, **оптимизированы для сетевой работы**, и на их основе производители - как процессоров, так и оборудования - пишут программное обеспечение для специфических приложений. Коммуникационный процессор имеет собственную память и оснащен высокоскоростными внешними каналами для соединения с другими процессорными узлами. Его присутствие позволяет в значительной мере освободить вычислительный

процессор от нагрузки, связанной с передачей сообщений между процессорными узлами. Скоростной коммуникационный процессор с RISC-ядром позволяет управлять обменом данными по нескольким независимым каналам, поддерживать практически все распространенные протоколы обмена, гибко и эффективно распределять и обрабатывать последовательные потоки данных с временным разделением каналов.

2.6. Процессоры баз данных

Процессорами (машинами) баз данных в настоящее время принято называть программно- аппаратные комплексы, предназначенные для выполнения всех или некоторых функций систем управления базами данных (*СУБД*). Если в свое время системы управления базами данных предназначались в основном для хранения текстовой и числовой информации, то теперь они рассчитаны на самые различные форматы данных, в том числе графические, звуковые и видео. Процессоры баз данных выполняют функции управления и распространения, обеспечивают дистанционный доступ к информации через шлюзы, а также репликацию обновленных данных с помощью различных механизмов тиражирования.

2.7. Поточковые процессоры

Поточковыми называют процессора, в основе работы которых лежит принцип обработки многих данных с помощью одной команды. Согласно классификации Флинна (см. раздел **2.1**) они принадлежат к **SIMD** (single instruction stream / multiple data stream) архитектуре. Технология SIMD позволяет выполнять одно и то же действие, например вычитание и сложение, над несколькими наборами чисел одновременно. **SIMD-операции** для чисел двойной точности с плавающей запятой ускоряют работу ресурсоемких приложений для создания **контента, трехмерного рендеринга, финансовых расчетов и научных задач**. Кроме того, усовершенствованы возможности 64-разрядной технологии MMX (целочисленных SIMD-команд); эта технология распространена на 128-разрядные числа, что позволяет ускорить обработку видео, речи, шифрование, обработку изображений и фотографий. Поточковый процессор повышает общую производительность, что особенно важно при работе с 3D-графическими объектами.

2.8. Нейронные процессоры ▲

Одно из наиболее перспективных направлений разработки принципиально новых архитектур вычислительных систем тесно связано с созданием компьютеров нового поколения на основе принципов обработки информации, заложенных в **искусственных нейронных сетях** (НС). Первые практические работы по искусственным нейросетям и нейрокомпьютерам начались еще в 40-50-е годы. Под нейронной сетью обычно понимают совокупность элементарных преобразователей информации, называемых «нейронами», которые определены образом соединены друг с другом каналами обмена информации «синаптическими связями».

2.9. Процессоры с многозначной (нечеткой) логикой ▲

Идея построения процессоров с нечеткой логикой (**fuzzy logic**) основывается на нечеткой математике. Математическая теория нечетких множеств,

предложенная проф. Л.А. Заде, являясь предметом интенсивных исследований, открывает все большие возможности перед системными аналитиками. Основанные на этой теории различные компьютерные системы, в свою очередь, существенно расширяют область применения нечеткой логики.

Литература:

1. В.В.Воеводин, Вл.В.Воеводин - "Параллельные вычисления", издательство "БХВ ", 2002.
2. В. Корнеев В. , "Современные микропроцессоры", издательство "Нолидж", 2003.
3. Э. Таненбаум, "Архитектура компьютера", СПб, издательство «Питер», 2002.
4. В.Г.Олифер, Н.А.Олифер Компьютерные сети, издательство "Питер", 2000.
5. К. Хамахер, З. Вранешич, С. Заки, Организация ЭВМ. СПб, издательство «Питер», 2003 г.

Контрольные задания для СРС

1. Привести примеры наиболее распространенных процессоров с конвейерной архитектурой.
2. Оценить применение Ассоциативных процессоров при работе базами данных.
3. Применение Матричных процессоров для научных расчетов в настоящее время.
4. Современное состояние дел в области разработок Клеточных и ДНК процессоров.
5. Современный рынок Коммуникационных процессоров.

Лекция 3

Тема: Коммутаторы для многопроцессорных вычислительных систем. Требования к компонентам МВС.

План лекции

1. Коммутаторы для МВС
2. Требования к компонентам МВС
3. Надежность и отказоустойчивость МВС

Раздел 1 : Коммутаторы для многопроцессорных вычислительных систем.

Коммуникационные среды вычислительных систем (ВС) состоят из адаптеров вычислительных модулей (ВМ) и коммутаторов, обеспечивающих соединения между ними.

Используются как простые коммутаторы, так и составные, komponуемые из набора простых. Простые коммутаторы могут соединять лишь малое число ВМ в силу физических ограничений, однако обеспечивают при этом минимальную задержку при установлении соединения. Составные коммутаторы, обычно строящиеся из простых в виде многокаскадных схем с помощью линий «точка-точка», преодолевают ограничение на малое количество соединений, однако увеличивают и задержки.

1.1. Простые коммутаторы

Типы простых коммутаторов:

- с временным разделением
- с пространственным разделением

Достоинства: простота управления и высокое быстродействие. Недостатки: малое количество входов и выходов.

Примеры использования:

- простые коммутаторы с временным разделением используются в системах SMP Power Challenge от SGI,
- простые коммутаторы с пространственным разделением (Gigaplane) используются в семействе Sun Ultra Enterprise.

1.2. Составные коммутаторы

Простые коммутаторы имеют ограничения на число входов и выходов, а также могут требовать большого количества оборудования при увеличении этого числа (в случае пространственных коммутаторов). Поэтому для построения коммутаторов с большим количеством входов и выходов используют совокупность простых коммутаторов, объединенных с помощью линий «точка-точка».

Составные коммутаторы имеют задержку, пропорциональную количеству простых коммутаторов, через которые проходит сигнал от входа до выхода, т.е. числу каскадов. Однако объем оборудования составного коммутатора меньше, чем простого с тем же количеством входов и выходов.

Чаще всего составные коммутаторы строятся из прямоугольных коммутаторов 2×2 с двумя входами и выходами. Они имеют два состояния: прямое пропускание входов на соответствующие выходы и перекрестное пропускание. Коммутатор 2×2 состоит из собственно блока коммутации данных и блока управления. Блок управления в зависимости от поступающих на него управляющих сигналов определяет, какой тип соединения следует осуществить в блоке коммутации: прямой или перекрестный. При этом если оба входа хотят соединиться с одним выходом, то коммутатор разрешает конфликт и связывает с данным выходом только один вход, а запрос на соединение со стороны второго блокируется или отвергается.

1.3. Распределенные составные коммутаторы

В распределенных вычислительных системах ресурсы разделяются между задачами, каждая из которых выполняется на своем подмножестве процессоров. В связи с этим возникает понятие близости процессоров, которая является важной для активно взаимодействующих процессоров. Обычно близость процессоров выражается в различной каскадности соединений, различных расстояниях между ними.

Один из вариантов создания составных коммутаторов заключается в объединении прямоугольных коммутаторов $(v+1 \times v+1)$, $v > 1$, таким образом, что один вход и один выход каждого составляющего коммутатора служат входом и выходом составного коммутатора. К каждому внутреннему коммутатору подсоединяются процессор и память, образуя вычислительный модуль с v каналами для соединения с другими вычислительными модулями. Свободные v выходов и v входов каждого вычислительного модуля соединяются линиями “точка-точка” с входами и выходами других коммутаторов, образуя граф межмодульных связей.

Наиболее эффективным графом межмодульных связей с точки зрения организации обмена данными между вычислительными модулями является полный граф. В этом случае между каждой парой вычислительных модулей существует прямое соединение. При этом возможны одновременные соединения между произвольными вычислительными модулями.

Однако обычно создать полный граф межмодульных связей невозможно по различным причинам. Обмен данными приходится производить через цепочки транзитных модулей. Из-за этого увеличиваются задержки, и ограничивается возможность установления одновременных соединений. Таким образом, эффективный граф межмодульных связей должен минимизировать время межмодульных обменов и максимизировать количество одновременно активизированных соединений. Кроме того, на выбор графа межмодульных связей влияет учет отказов и восстановлений вычислительных модулей и линий связи.

Раздел 2: Требования к компонентам МВС

Данный заголовок нужно понимать более в широком смысле, чем просто требования к техническим характеристикам компонент вычислительной системы: процессору, дисковым массивам, памяти, коммутаторам и т.п. аппаратным средствам. Гораздо более важное значение имеют требования, предъявляемые к вычислительной системе, которую собираются построить для реализации конкретных целей — решения задач определённого круга (научных, экономических, информационных систем и т.п.), модель программирования. Отсюда вытекает выбор архитектуры МВС.

Разработчикам необходимо, прежде всего, проанализировать следующие связанные между собой вопросы:

- Отношение стоимость/производительность
- Надежность и отказоустойчивость
- Масштабируемость
- Совместимость программного обеспечения

Требования к надёжности и отказоустойчивости системы обсуждаются в другой главе.

2.1. Отношение стоимость/производительность

Добиться дополнительного повышения производительности в МВС тяжелее, чем произвести масштабирование внутри узла. Основным барьером является трудность организации эффективных межузловых связей. Коммуникации, которые происходят между узлами, должны быть устойчивы к большим задержкам программно поддерживаемой когерентности. Приложения с большим количеством взаимодействующих процессов работают лучше на основе SMP-узлов, в которых коммуникационные связи более быстрые. В кластерах, как и в MPP системах, масштабирование приложений более эффективно при уменьшении объема коммуникаций между процессами, работающими в разных узлах. Это обычно достигается путем разбиения данных. Именно такой подход используется в наиболее известном приложении на основе кластеров OPS (Oracle Parallel Server).

2.2. Масштабируемость

Масштабируемость представляет собой возможность наращивания числа и мощности процессоров, объемов оперативной и внешней памяти и других ресурсов вычислительной системы. Масштабируемость должна обеспечиваться архитектурой и конструкцией компьютера, а также соответствующими средствами программного обеспечения.

Так, например, возможность масштабирования кластера ограничена значением отношения скорости процессора к скорости связи, которое не должно быть слишком большим (реально это отношение для больших систем не может быть более 3-4, в противном случае не удастся даже реализовать режим единого образа операционной системы). С другой стороны, последние 10 лет истории развития процессоров и коммутаторов показывают, что разрыв по скорости между ними все увеличивается. Добавление каждого нового процессора в действительно масштабируемой системе должно давать прогнозируемое увеличение производительности и пропускной способности при приемлемых затратах. Одной из основных задач при построении масштабируемых систем является минимизация стоимости расширения компьютера и упрощение планирования. В идеале добавление процессоров к системе должно приводить к линейному росту ее производительности. Однако это не всегда так. Потери производительности могут возникать, например, при недостаточной пропускной способности шин из-за возрастания трафика между процессорами и основной памятью, а также между памятью и устройствами ввода/вывода. В действительности реальное увеличение производительности трудно оценить заранее, поскольку оно в значительной степени зависит от динамики поведения прикладных задач.

2.3. Совместимость и мобильность программного обеспечения

Концепция программной совместимости впервые в широких масштабах была применена разработчиками системы IBM/360. Основная задача при проектировании всего ряда моделей этой системы заключалась в создании такой архитектуры, которая была бы одинаковой с точки зрения пользователя для всех моделей системы независимо от цены и производительности каждой из них. Огромные преимущества такого подхода, позволяющего сохранять существующий задел программного обеспечения при переходе на новые (как правило, более производительные) модели, были быстро оценены как производителями компьютеров, так и пользователями и, начиная с этого времени, практически все фирмы-поставщики компьютерного оборудования взяли на вооружение эти принципы, поставляя серии совместимых компьютеров. Следует заметить, однако, что со временем даже самая передовая архитектура неизбежно устаревает и возникает потребность внесения радикальных изменений архитектуру и способы организации вычислительных систем.

Раздел 3: Надежность и отказоустойчивость МВС

- Одной из основных проблем построения вычислительных систем остаётся **задача обеспечения их продолжительного функционирования.**

Важнейшей характеристикой вычислительных систем является

•

Литература:

1. В.В.Воеводин, Вл.В.Воеводин - "Параллельные вычисления", издательство "БХВ ", 2002.
2. В. Корнеев В. , "Современные микропроцессоры", издательство "Нолидж", 2003.
3. Э. Таненбаум, "Архитектура компьютера", СПб, издательство «Питер», 2002.
4. В.Г.Олифер, Н.А.Олифер Компьютерные сети, издательство «Питер», 2000.
5. К. Хамахер, З. Вранешич, С. Заки, Организация ЭВМ. СПб, издательство «Питер», 2003 г.

Контрольные задания для СРС

1. *Ведущие фирмы производители коммутаторов для современных МВС и их продукция.*
2. *Провести оценку стоимости/производительности персонального компьютера и мощного сервера от известного производителя.*
3. *Оценить способность к масштабируемости персонального компьютера и ноутбука.*
4. *Оценить совместимость системного и прикладного программного обеспечения для персонального компьютера по отношению к предыдущим и последующим архитектурам.*
5. *Средства повышения надежности и отказоустойчивости современных МВС.*

Лекция 4

Тема: Кластеры и массивно-параллельные системы различных производителей.

План лекции

1. Введение
2. Примеры кластерных решений
3. SMP системы
4. Суперкомпьютеры

Развитие сетевых технологий привело к появлению недорогих, но эффективных коммуникационных решений. Это и предопределило появление кластерных вычислительных систем, фактически являющихся одним из направлений развития компьютеров с массовым параллелизмом. Классические суперкомпьютеры, использующие специализированные процессоры таких фирм как, например, Cray, NEC (векторно-параллельные или массивно-параллельные), обычно недешевы, поэтому и стоимость подобных систем не сравнима со стоимостью систем, находящихся в массовом производстве.

Вычислительные системы, создаваемые из массово выпускаемых компонентов, стали притягательной альтернативой традиционным суперкомпьютерным системам. При выполнении многих прикладных задач такие ВС, даже с небольшим или средним (до 128–256) числом вычислительных модулей, показывают производительность, не уступающую или даже превосходящую производительность традиционных суперкомпьютеров как с распределенной, так и с разделяемой памятью. Наряду с этим, эти ВС обладают рядом преимуществ, среди которых: более низкая стоимость, короткий цикл разработки и возможность оперативно использовать наиболее эффективные вычислительные и коммуникационные компоненты из имеющихся на рынке во время создания системы. Поэтому неудивительно, что ведущие фирмы разработчики высокопроизводительной техники приступили к созданию кластерных систем.

1.1. Примеры кластерных решений IBM

В начале 2000 года IBM создала Linux-кластер из установленных в стойке серверов IBMxSeries, интегрировав их с соответствующими сетями, системами управления (аппаратное и программное обеспечение) и необходимыми услугами. После выпуска в 2001 году кластера 1300, IBM представила недавно кластер 1350 на процессорах Intel Xeon.

Стандартным вычислительным узлом для кластера 1350 является **IBMxSeries 335**. Это позволяет одному или двум процессорам Intel Pentium 4 (Xeon) с быстрой динамической памятью и диском размещаться в стандартном корпусе размером «1U». Символ 1U обозначает 1,75 дюймов высоты в стандартном 19-и дюймовом корпусе. X335 имеет встроенный сервисный процессор и два слота для соединения с другими компонентами системами.

1.2. Примеры кластерных решений HP

- Слияние HP и Compaq обеспечило HP прочное положение лидера по продаже Linux-систем, соответствующих лучшим промышленным стандартам на базе архитектур IA-32 и IA-64. Данная технология дополнена мощной поддержкой разработок ядра Linux на базе семейства Itanium, а также разработок с открытым кодом в целом.

Поддержка ОС Linux со стороны HP охватывает всю линейку серверов HP, основанных на архитектуре Intel (IA-32 и IA-64), включая все серверы промышленного стандарта **HP ProLiant**, сверхплотную блейд-архитектуру, рабочие станции HP, настольные компьютеры Evo, отдельные портативные компьютеры, серверы ProLiant для применения в качестве межсетевых экранов и даже портативные устройства iPAQ. HP также продолжает поддерживать технологию ОС Linux для архитектуры **AlphaServer**, разработанную компанией Compaq. ОС Linux работает на Alpha системах, начиная с 1994 года, став первым примером 64-разрядной системы с поддержкой Linux. Это открыло путь для современных разработок ОС Linux на базе семейства Itanium. HP поддерживает на своих серверах дистрибутивы Red Hat и SuSE, планируя осуществлять поддержку дистрибутивов операционной системы UnitedLinux после ее выпуска. HP предлагает заказчикам возможность

8.3. Примеры кластерных решений SGI

Седьмого января 2003 года компания SGI представила новое семейство 64-х разрядных Linux-серверов и суперкластеров, названных **SGI Altix 3000**. Система SGI Altix 3000 использует процессоры Intel Itanium 2 и основана на архитектуре глобальной разделяемой памяти SGI Numaflex, которая является реализацией архитектуры неоднородного доступа к памяти (NUMA). NUMAflex появилась в 1996 году и с тех пор использовалась в известной серии серверов и суперкомпьютеров SGI Origin, основанных на процессорах MIPS и 64-разрядной операционной системе IRIX. Дизайн NUMAflex позволяет помещать процессор, память, систему ввода/вывода, соединительные провода, графическую подсистему в модульные компоненты, иначе называемые блоками или кирпичиками. Эти кирпичики могут комбинироваться и конфигурироваться с большой гибкостью, чтобы удовлетворять потребности клиента в ресурсах и рабочей нагрузке. Используя этот дизайн третьего поколения, компания SGI смогла создать систему SGI Altix 3000 на основе традиционных блоков ввода/вывода (IX- и PX-блоки), хранения данных (D-блоки) и соединительных компонентов (маршрутизирующие блоки/R-блоки). Основным отличием этой новой системы является процессорный блок (C-блок), который содержит процессоры Itanium 2.

1.4 SMP Power Challenge фирмы Silicon Graphics

Компания Silicon Graphics (SGI) была создана в 1981 году. Основным направлением работы компании в течение многих лет было создание высокопроизводительных графических рабочих станций. В настоящее время ее интересы распространяются на рынок высокопроизводительных вычислений как для технических, так и для коммерческих приложений. В частности она концентрирует свои усилия на разработке и внедрении современных технологий визуализации вычислений, трехмерной графики, обработки звука и мультимедиа.

1.5. Семейство SUN Ultra Enterprise фирмы SUN

Sun Ultra Enterprise — это серия мощных, масштабируемых, удобных в управлении и надежных серверов. Эту серию можно разделить на следующие группы:

Серверы рабочих групп

- Sun Enterprise 10s
- Sun Enterprise 250

- Sun Enterprise 220R
- Sun Enterprise 450
- Sun Enterprise 420R

Серверы отдела предприятия

- Sun Enterprise 3500
- Sun Enterprise 4500

Серверы масштаба предприятия

- Sun Enterprise 5500
- Sun Enterprise 6500
- Sun Enterprise 10000

1.6. Семейство массово-параллельных машин ВС МВС-100 и МВС-1000

Массово-параллельные масштабируемые системы МВС предназначены для решения прикладных задач, требующих большого объема вычислений и обработки данных. Суперкомпьютерная установка системы МВС представляет собой мультипроцессорный массив, объединенный с внешней дисковой памятью и устройствами ввода-вывода информации под общим управлением персонального компьютера или рабочей станции.

Основа программного обеспечения МВС:

- языки FORTRAN и C (C++), дополнительные средства описания параллельных процессов;
- программные средства PVM и MPI (общепринятые для систем параллельной обработки);
- средства реализации многопользовательских режимов и удаленного доступа.

Основные области фактического применения суперкомпьютеров МВС-100/1000:

1. Решение задач расчета аэродинамики летательных аппаратов, в том числе явления интерференции при групповом движении.
2. Расчет трехмерных нестационарных течений вязко сжимаемого газа.
3. Расчеты течений с локальными тепловыми неоднородностями в потоке.
4. Разработка квантовой статистики моделей поведения вещества при экс-

тремальных условиях. Расчеты баз данных по уравнениям соблюдения в широкой области температуры и плотности.

5. Расчеты структурообразования биологических макромолекул.

6. Моделирование динамики молекулярных и биомолекулярных систем.

7. Решение задач линейных дифференциальных игр. Динамические задачи конфликтов управления.

8. Решение задач механики деформируемых твердых тел, в том числе с учетом процессов разрушения.

В 1999 году 5 ноября был официально открыт Межведомственный суперкомпьютерный центр (МСЦ). В его создании принимали участие Миннауки РФ, РАН, Минобразования России и Российский фонд фундаментальных исследований. К тому моменту в МСЦ в качестве основного вычислителя была установлена 16-процессорная система фирмы Хьюлетт-Паккард V2250 производительностью 15 млрд. опер/с. Параллельно с этой системой в МСЦ работал также 96-процессорный вариант отечественной системы МВС-1000. Суммарная производительность всех систем центра достигала 230 млрд. опер/с ($2,3 \times 10^{11}$).

1.7. ВС с распределённой памятью фирм Sequent и DATA GENERAL Sequent

В 1999 году крупнейшие компьютерные корпорации **IBM** и **Sequent** объявили о своем слиянии. Sequent фактически стала подразделением IBM. После этого IBM остановила значительное обновление линии серверов **NUMA Q**, выпускавшихся Sequent.

Sequent являлся поставщиком масштабируемых NUMA-серверов (серии NUMA-Q 1000 и NUMA-Q 2000), включающих до 64 процессоров Intel и предназначенных в основном для коммерческих систем онлайн-обработки транзакций и поддержки СУБД.

Кроме того, NUMA-технологии, разработанные специалистами Sequent, используются и в серверах от IBM. IBM утверждает, что NUMA станет определяющей технологией для UNIX- и NT-серверов уже в начале 21 века.

Sequent была, по-видимому, второй после IBM компанией, осуществившей поставки UNIX-кластеров баз данных в середине 1993 года. Она предлагала решения, соответствующие среднему и высокому уровню готовности своих систем. Первоначально Sequent Hi-Av Systems обеспечивали дублирование систем, которые разделяли общие диски. Пользователи могли выбирать ручной или автоматический режим переключения на резерв в случае отказа. Hi-Av Systems обеспечивает также горячее резервирование IP адресов и позволяет кластеру, в состав которого входят до четырех узлов, иметь единственный сетевой адрес.

Компания Sequent одной из первых освоила технологию Fast-Wide SCSI, что позволило ей добиться значительного увеличения производительности систем при обработке транзакций. Компания поддерживает дисковые подсистемы RAID уровней 1, 3 и 5. Кроме того она предлагает в качестве

разделяемого ресурса ленточные накопители SCSI. Модель SE90 поддерживает кластеры, в состав которых могут входить два, три или четыре узла, представляющих собой многопроцессорные системы Symmetry 2000 или Symmetry 5000 в любой комбинации. Это достаточно мощные системы. Например, Sequent Symmetry 5000 Series 790 может иметь от 2 до 30 процессоров Pentium 66 МГц, оперативную память емкостью до 2 Гбайт и дисковую память емкостью до 840 Гбайт.

При работе с Oracle Parallel Server все узлы кластера работают с единственной копией базы данных, расположенной на общих разделяемых дисках.

8.8. Современные кластеры DIGITAL

Как и предыдущие фирмы (Sequent и Data General) фирма **Digital (DEC)** прекратила своё самостоятельное существование, сначала став отделением фирмы Compac, а затем вместе с ней вошла в состав Hewlett-Packard. Компания DEC известна тем, что она являлась разработчиком серверов **AlphaServer**, на основе которых строились Alpha-кластеры. Перейдя в другую компанию, коллектив Digital под маркой HP продолжает работу по обновлению этой линии серверов.

Архитектура систем на основе процессора Alpha была разработана в 1988-1991 году с перспективой на 20-25 лет. Выпускаемое в настоящее время четвёртое поколение процессоров содержит 4-канальное суперскалярное ядро, 80 регистров для целочисленных операций, 72 регистра с плавающей запятой, причём в процессе обработки одновременно может находиться до 80 инструкций. Для архитектуры процессора Alpha характерно внеочередное исполнение команд, логика предсказания ветвлений, полностью интегрированная на кристалле кэш-память первого уровня, многоканальные устройства доступа к оперативной памяти. В семейство Alpha-сервер входит четыре серии серверов: AlphaServer DS, AlphaServer ES, AlphaServer GS и AlphaServer SC. Если в моделях AlphaServer DS используется не более 64-разрядных процессоров, в серверах AlphaServer SC их может быть несколько сотен (до 512 и более). Максимальная тактовая частота процессоров в настоящее время составляет 1,25 ГГц. Все компьютеры семейства построены по коммутируемой технологии, что позволяет избежать недлстатков, присущих системной шине.

8.9. Современные суперкомпьютеры — Cray T3E-1200.

Системы **Cray T3E** — это масштабируемые параллельные системы, которые используют DECchip 21164 (DEC Alpha EV5) RISK-процессоры с пиковой производительностью 600 Мфлоп и 21164А для машин Cray T3E-900 и Cray T3E-1200. Каждый процессорный элемент (ПЭ) Cray T3E имеет свою собственную DRAM-память объёмом от 64 Мбайт до 2 Гбайт. В отличие от системы CRAY T3D, в которой исполняемая задача запрашивает фиксированное количество процессоров на все время выполнения, в CRAY T3E неиспользуемые процессоры могут использоваться другими задачами. Модели T3E, T3E-900, T3E-1200, T3E-1350.

•

Система T3E-1200

Серия Cray T3E-1200 в два раза превышает производительность систем Cray T3E при уменьшенной вдвое стоимости за Мфлоп. Конфигурации в воздушно-жидкостном охлаждении имеют от 6-и процессоров, а в жидкостном — от 32 процессоров. Каждый процессор имеет производительность в 1,2 Тфлоп, для всей системы пиковая производительность меняется от 7,2 Гфлоп до 2,5 Тфлоп. Масштабируется до тысяч процессоров. Серия выпущена в 1997 году.

Система предназначена для наиболее важных научных и технических задач в аэрокосмической, автомобильной, финансовой, химико-фармацевтической, нефтяной и т.д. промышленности, также в широких областях прикладных исследований, включая химию, гидродинамику, предсказание погоды и сейсмические процессы.

Лекция 5

Тема: От централизованных систем - к вычислительным сетям. Основные проблемы построения сетей

План лекции

1. Эволюция вычислительных систем
2. Проблемы физической передачи данных по линиям связи
3. Проблемы объединения нескольких компьютеров
4. Структуризация как средство построения больших сетей

Раздел 1: От централизованных систем - к вычислительным сетям

1.1. Эволюция вычислительных систем

Концепция вычислительных сетей является логическим результатом эволюции компьютерной технологии. Первые компьютеры 50-х годов - большие, громоздкие и дорогие - предназначались для очень небольшого числа избранных пользователей. Часто эти монстры занимали целые здания. Такие компьютеры не были предназначены для интерактивной работы пользователя, а использовались в режиме пакетной обработки.

Системы пакетной обработки

Системы пакетной обработки, как правило, строились на базе мэйн-фрейма - мощного и надежного компьютера универсального назначения. Пользователи подготавливали перфокарты, содержащие данные и команды программ, и передавали их в вычислительный центр. Операторы вводили эти карты в компьютер, а распечатанные результаты пользователи получали обычно только на следующий день (рис. 1.1), Таким образом, одна неверно набитая карта означала как минимум суточную задержку.

Многотерминальные системы - прообраз сети

По мере удешевления процессоров в начале 60-х годов появились новые способы организации вычислительного процесса, которые позволили учесть интересы пользователей. Начали развиваться интерактивные многотерминальные системы разделения времени (рис. 1.2). В таких системах компьютер отдавался в распоряжение сразу нескольким пользователям. Каждый пользователь получал в свое распоряжение терминал, с помощью которого он мог вести диалог с компьютером. Причем время реакции вычислительной системы было достаточно мало для того, чтобы пользователю была не слишком заметна параллельная работа с компьютером и других пользователей. Разделяя таким образом компьютер, пользователи получили возможность за сравнительно небольшую плату пользоваться преимуществами компьютеризации.

Появление глобальных сетей

Тем не менее потребность в соединении компьютеров, находящихся на большом расстоянии друг от друга, к этому времени вполне назрела. Началось все с решения более простой задачи - доступа к компьютеру с терминалов, удаленных от него на многие сотни, а то и тысячи километров. Термина-

лы соединялись с компьютерами через телефонные сети с помощью модемов. Такие сети позволяли многочисленным пользователям получать удаленный доступ к разделяемым ресурсам нескольких мощных компьютеров класса суперЭВМ. Затем появились системы, в которых наряду с удаленными соединениями типа терминал-компьютер были реализованы и удаленные связи типа компьютер-компьютер. Компьютеры получили возможность обмениваться данными в автоматическом режиме, что, собственно и является базовым механизмом любой вычислительной сети. Используя этот механизм, в первых сетях были реализованы службы обмена файлами, синхронизации баз данных, электронной почты и другие, ставшие теперь традиционными сетевые службы.

Первые локальные сети

В начале 70-х годов произошел технологический прорыв в области производства компьютерных компонентов - появились большие интегральные схемы. Их сравнительно невысокая стоимость и высокие функциональные возможности привели к созданию мини-компьютеров, которые стали реальными конкурентами мэйнфреймов. Закон Гроша перестал соответствовать действительности, так как десяток мини-компьютеров выполнял некоторые задачи (как правило, хорошо распараллеливаемые) быстрее одного мэйнфрейма, а стоимость такой мини-компьютерной системы была меньше.

Создание стандартных технологий локальных сетей

В середине 80-х годов положение дел в локальных сетях стало кардинально меняться. Утвердились стандартные технологии объединения компьютеров в сеть - Ethernet, Arcnet, Token Ring. Мощным стимулом для их развития послужили персональные компьютеры. Эти массовые продукты явились идеальными элементами для построения сетей - с одной стороны, они были достаточно мощными для работы сетевого программного обеспечения, а с другой - явно нуждались в объединении своей вычислительной мощности для решения сложных задач, а также разделения дорогих периферийных устройств и дисковых массивов. Поэтому персональные компьютеры стали преобладать в локальных сетях, причем не только в качестве клиентских компьютеров, но и в качестве центров хранения и обработки данных, то есть сетевых серверов, потеснив с этих привычных ролей мини-компьютеры и мэйнфреймы.

Современные тенденции

Сегодня вычислительные сети продолжают развиваться, причем достаточно быстро. Разрыв между локальными и глобальными сетями постоянно сокращается во многом из-за появления высокоскоростных территориальных каналов связи, не уступающих по качеству кабельным системам локальных сетей. В глобальных сетях появляются службы доступа к ресурсам, такие же удобные и прозрачные, как и службы локальных сетей. Подобные примеры в большом количестве демонстрирует самая популярная глобальная сеть - Internet.

Изменяются и локальные сети. Вместо соединяющего компьютеры пассивного кабеля в них в большом количестве появилось разнообразное

коммуникационное оборудование - коммутаторы, маршрутизаторы, шлюзы. Благодаря такому оборудованию появилась возможность построения больших корпоративных сетей, насчитывающих тысячи компьютеров и имеющих сложную структуру. Возродился интерес к крупным компьютерам - в основном из-за того, что после спада эйфории по поводу легкости работы с персональными компьютерами выяснилось, что системы, состоящие из сотен серверов, обслуживать сложнее, чем несколько больших компьютеров. Поэтому на новом витке эволюционной спирали мэйнфреймы стали возвращаться в корпоративные вычислительные системы, но уже как полноправные сетевые узлы, поддерживающие Ethernet или Token Ring, а также стек протоколов TCP/IP, ставший благодаря Internet сетевым стандартом де-факто.

1.2. Вычислительные сети - частный случай распределенных систем

Компьютерные сети относятся к распределенным (или децентрализованным) вычислительным системам. Поскольку основным признаком распределенной вычислительной системы является наличие нескольких центров обработки данных, то наряду с компьютерными сетями к распределенным системам относят также мультипроцессорные компьютеры и многомашинные вычислительные комплексы.

Мультипроцессорные компьютеры

В мультипроцессорных компьютерах имеется несколько процессоров, каждый из которых может относительно независимо от остальных выполнять свою программу. В мультипроцессоре существует общая для всех процессоров операционная система, которая оперативно распределяет вычислительную нагрузку между процессорами. Взаимодействие между отдельными процессорами организуется наиболее простым способом - через общую оперативную память

Многомашинные системы

Многомашинная система - это вычислительный комплекс, включающий в себя несколько компьютеров (каждый из которых работает под управлением собственной операционной системы), а также программные и аппаратные средства связи компьютеров, которые обеспечивают работу всех компьютеров комплекса как единого целого.

Вычислительные сети

В *вычислительных сетях* программные и аппаратные связи являются еще более слабыми, а автономность обрабатывающих блоков проявляется в наибольшей степени - основными элементами сети являются стандартные компьютеры, не имеющие ни общих блоков памяти, ни общих периферийных устройств. Связь между компьютерами осуществляется с помощью специальных периферийных устройств - сетевых адаптеров, соединенных относительно протяженными каналами связи. Каждый компьютер работает под управлением собственной операционной системы, а какая-либо "общая" операционная система, распределяющая работу между компьютерами сети, от-

существует. Взаимодействие между компьютерами сети происходит за счет передачи сообщений через сетевые адаптеры и каналы связи. С помощью этих сообщений один компьютер обычно запрашивает доступ к локальным ресурсам другого компьютера. Такими ресурсами могут быть как данные, хранящиеся на диске, так и разнообразные периферийные устройства - принтеры, модемы, факс-аппараты и т.д. Разделение локальных ресурсов каждого компьютера между всеми пользователями сети - основная цель создания вычислительной сети.

Каким же образом сказывается на пользователе тот факт, что его компьютер подключен к сети? Прежде всего, он может пользоваться не только файлами, дисками, принтерами и другими ресурсами своего компьютера, но и аналогичными ресурсами других компьютеров, подключенных к той же сети. Правда, для этого недостаточно снабдить компьютеры сетевыми адаптерами и соединить их кабельной системой. Необходимы еще некоторые добавления к операционным системам этих компьютеров. На тех компьютерах, ресурсы которых должны быть доступны всем пользователям сети, необходимо добавить модули, которые постоянно будут находиться в режиме ожидания запросов, поступающих по сети от других компьютеров. Обычно такие модули называются программными *серверами (server)*, так как их главная задача - обслуживать (*serve*) запросы на доступ к ресурсам своего компьютера. На компьютерах, пользователи которых хотят получать доступ к ресурсам других компьютеров, также нужно добавить к операционной системе некоторые специальные программные модули, которые должны выработать запросы на доступ к удаленным ресурсам и передавать их по сети на нужный компьютер. Такие модули обычно называют программными *клиентами (client)*. Собственно же сетевые адаптеры и каналы связи решают в сети достаточно простую задачу - они передают сообщения с запросами и ответами от одного компьютера к другому, а основную работу по организации совместного использования ресурсов выполняют клиентские и серверные части операционных систем.

Распределенные программы

Сетевые службы всегда представляют собой распределенные программы. *Распределенная программа* - это программа, которая состоит из нескольких взаимодействующих частей, причем каждая часть, как правило, выполняется на отдельном компьютере сети.

До сих пор речь шла о системных распределенных программах. Однако в сети могут выполняться и распределенные пользовательские программы - приложения. Распределенные приложения в полной мере используют потенциальные возможности распределенной обработки, предоставляемые вычислительной сетью, и поэтому часто называются *сетевыми приложениями*. Следует подчеркнуть, что не всякое приложение, выполняемое в сети, является сетевым.

Существует большое количество популярных приложений, которые не являются распределенными и целиком выполняются на одном компьютере сети. Тем не менее и такие приложения могут использовать преимущества

сети за счет встроенных в операционную систему сетевых служб. Значительная часть истории локальных сетей связана как раз с использованием таких нераспределенных приложений. Рассмотрим, например, как происходила работа пользователя с известной в свое время СУБД dBase. Обычно файлы базы данных, с которыми работали все пользователи сети, располагались на файловом сервере. Сама же СУБД хранилась на каждом клиентском компьютере в виде единого программного модуля.

1.3. Что дает предприятию использование сетей

Этот вопрос можно уточнить следующим образом: в каких случаях развертывание на предприятии вычислительных сетей предпочтительнее использования автономных компьютеров или многомашинных систем? Какие новые возможности появляются на предприятии с появлением там вычислительной сети? И наконец, всегда ли предприятию нужна сеть?

Если не вдаваться в частности, то конечной целью использования вычислительных сетей на предприятии является повышение эффективности его работы, которое может выражаться, например, в увеличении прибыли предприятия. Действительно, если благодаря компьютеризации снизились затраты на производство уже существующего продукта, сократились сроки разработки новой модели или ускорилось обслуживание заказов потребителей - это означает, что данному предприятию действительно нужна была сеть.

В каждом конкретном случае для создания сети были свои резоны, но верно и общее утверждение: что-то в этих сетях все-таки есть. Использование вычислительных сетей дает предприятию следующие возможности:

- разделение дорогостоящих ресурсов;
- совершенствование коммуникаций;
- улучшение доступа к информации;
- быстрое и качественное принятие решений;
- свобода в территориальном размещении компьютеров.

Раздел 2: Основные проблемы построения сетей

2.1. Проблемы физической передачи данных по линиям связи

Даже при рассмотрении простейшей сети, состоящей всего из двух машин, можно увидеть многие проблемы, присущие любой вычислительной сети, в том числе проблемы, связанные с физической передачей сигналов по линиям связи, без решения которой невозможен любой вид связи.

В вычислительной технике для представления данных используется двоичный код. Внутри компьютера единицам и нулям данных соответствуют дискретные электрические сигналы. Представление данных в виде электрических или оптических сигналов называется *кодированием*. Существуют различные способы кодирования двоичных цифр 1 и 0, например, потенциа-

ный способ, при котором единице соответствует один уровень напряжения, а нулю - другой, или импульсный способ, когда для представления цифр используются импульсы различной или одной полярности.

Аналогичные подходы могут быть использованы для кодирования данных и при передаче их между двумя компьютерами по линиям связи. Однако эти линии связи отличаются по своим электрическим характеристикам от тех, которые существуют внутри компьютера. Главное отличие внешних линий связи от внутренних состоит в их гораздо большей протяженности, а также в том, что они проходят вне экранированного корпуса по пространствам, зачастую подверженным воздействию сильных электромагнитных помех. Все это приводит к значительно большим искажениям прямоугольных импульсов (например, "заваливанию" фронтов), чем внутри компьютера. Поэтому для надежного распознавания импульсов на приемном конце линии связи при передаче данных внутри и вне компьютера не всегда можно использовать одни и те же скорости и способы кодирования.

В вычислительных сетях применяют как потенциальное, так и импульсное кодирование дискретных данных, а также специфический способ представления данных, который никогда не используется внутри компьютера, - *модуляцию* (рис. 2.1). При модуляции дискретная информация представляется синусоидальным сигналом той частоты, которую хорошо передает имеющаяся линия связи.

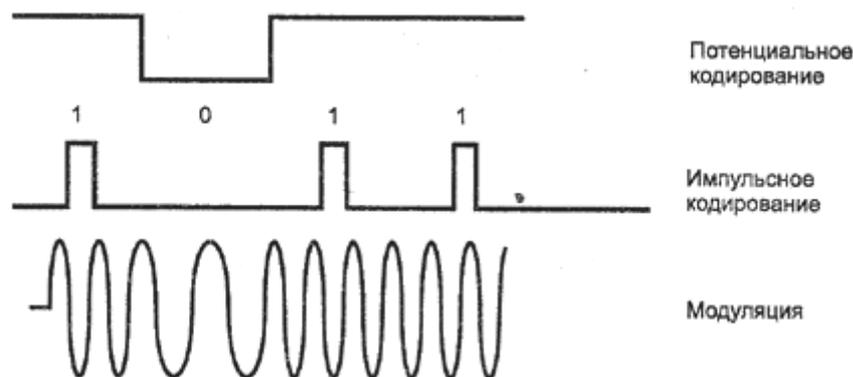


Рис. 2.1. Примеры представления дискретной информации

Потенциальное или импульсное кодирование применяется на каналах высокого качества, а модуляция на основе синусоидальных сигналов предпочтительнее в том случае, когда канал вносит сильные искажения в передаваемые сигналы. Обычно модуляция используется в глобальных сетях при передаче данных через аналоговые телефонные каналы связи, которые были разработаны для передачи голоса в аналоговой форме и поэтому плохо подходят для непосредственной передачи импульсов.

2.2. Проблемы объединения нескольких компьютеров

До сих пор мы рассматривали вырожденную сеть, состоящую всего из двух машин. При объединении в сеть большего числа компьютеров возникает целый комплекс новых проблем.

Топология физических связей

В первую очередь необходимо выбрать способ организации физических связей, то есть *топологию*. Под топологией вычислительной сети понимается конфигурация графа, вершинам которого соответствуют компьютеры сети (иногда и другое оборудование, например концентраторы), а ребрам - физические связи между ними. Компьютеры, подключенные к сети, часто называют *станциями* или *узлами сети*.

Заметим, что конфигурация *физических связей* определяется электрическими соединениями компьютеров между собой и может отличаться от конфигурации *логических связей* между узлами сети. Логические связи представляют собой маршруты передачи данных между узлами сети и образуются путем соответствующей настройки коммуникационного оборудования.

Выбор топологии электрических связей существенно влияет на многие характеристики сети. Например, наличие резервных связей повышает надежность сети и делает возможным балансирование загрузки отдельных каналов. Простота присоединения новых узлов, свойственная некоторым топологиям, делает сеть легко расширяемой. Экономические соображения часто приводят к выбору топологий, для которых характерна минимальная суммарная длина линий связи.

Организация совместного использования линий связи

Только в сети с полносвязной топологией для соединения каждой пары компьютеров имеется отдельная линия связи. Во всех остальных случаях неизбежно возникает вопрос о том, как организовать совместное использование линий связи несколькими компьютерами сети. Как и всегда при разделении ресурсов, главной целью здесь является удешевление сети.

В вычислительных сетях используют как индивидуальные линии связи между компьютерами, так и *разделяемые (shared)*, когда одна линия связи попеременно используется несколькими компьютерами. В случае применения разделяемых линий связи возникает комплекс проблем, связанных с их совместным использованием, который включает как чисто электрические проблемы обеспечения нужного качества сигналов при подключении к одному и тому же проводу нескольких приемников и передатчиков, так и логические проблемы разделения во времени доступа к этим линиям.

Адресация компьютеров

Еще одной новой проблемой, которую нужно учитывать при объединении трех и более компьютеров, является проблема их адресации. К адресу узла сети и схеме его назначения можно предъявить несколько требований.

- Адрес должен уникально идентифицировать компьютер в сети любого масштаба.
- Схема назначения адресов должна сводить к минимуму ручной труд администратора и вероятность дублирования адресов.
- Адрес должен иметь иерархическую структуру, удобную для построения больших сетей. Эту проблему хорошо иллюстрируют международные почтовые адреса, которые позволяют почтовой службе, органи-

- Адрес должен быть удобен для пользователей сети, а это значит, что он должен иметь символьное представление например, Server3 или www.dsco.com.
- Адрес должен иметь по возможности компактное представление, чтобы не перегружать память коммуникационной аппаратуры - сетевых адаптеров, маршрутизаторов и т.п.
- *Аппаратные (hardware) адреса.* Эти адреса предназначены для сети небольшого или среднего размера, поэтому они не имеют иерархической структуры. Типичным представителем адреса такого типа является адрес сетевого адаптера локальной сети. Такой адрес обычно используется только аппаратурой, поэтому его стараются сделать по возможности компактным и записывают в виде двоичного или шестнадцатеричного значения, например 0081005e24a8. При задании аппаратных адресов обычно не требуется выполнение ручной работы, так как они либо встраиваются в аппаратуру компанией-изготовителем, либо генерируются автоматически при каждом новом запуске оборудования, причем уникальность адреса в пределах сети обеспечивает оборудование. Помимо отсутствия иерархии, использование аппаратных адресов связано еще с одним недостатком - при замене аппаратуры, например, сетевого адаптера, изменяется и адрес компьютера. Более того, при установке нескольких сетевых адаптеров у компьютера появляется несколько адресов, что не очень удобно для пользователей сети.
- *Символьные адреса или имена.* Эти адреса предназначены для запоминания людьми и поэтому обычно несут смысловую нагрузку. Символьные адреса легко использовать как в небольших, так и крупных сетях. Для работы в больших сетях символическое имя может иметь сложную иерархическую структуру, например ftp-arch1.ucl.ac.uk. Этот адрес говорит о том, что данный компьютер поддерживает ftp-архив в сети одного из колледжей Лондонского университета (University College London - ucl) и эта сеть относится к академической ветви (ac) Internet Великобритании (United Kingdom - uk). При работе в пределах сети Лондонского университета такое длинное символическое имя явно избыточно и вместо него удобно пользоваться кратким символическим именем, на роль которого хорошо подходит самая младшая составляющая полного имени, то есть имя ftp-arch1.
- *Числовые составные адреса.* Символьные имена удобны для людей, но из-за переменного формата и потенциально большой длины их передача по сети не очень экономична. Поэтому во многих случаях для рабо-

2.3. Ethernet - пример стандартного решения сетевых проблем

Рассмотрим, каким образом описанные выше общие подходы к решению наиболее важных проблем построения сетей воплощены в наиболее популярной сетевой технологии - *Ethernet*.

Сетевая технология - это согласованный набор стандартных протоколов и реализующих их программно-аппаратных средств (например, сетевых адаптеров, драйверов, кабелей и разъемов), достаточный для построения вычислительной сети. Эпитет "достаточный" подчеркивает то обстоятельство, что этот набор представляет собой минимальный набор средств, с помощью которых можно построить работоспособную сеть. Возможно, эту сеть можно улучшить, например, за счет выделения в ней подсетей, что сразу потребует кроме протоколов стандарта Ethernet применения протокола IP, а также специальных коммуникационных устройств - маршрутизаторов. Улучшенная сеть будет, скорее всего, более надежной и быстродействующей, но за счет надстроек над средствами технологии Ethernet, которая составила базис сети. Термин "сетевая технология" чаще всего используется в описанном выше узком смысле, но иногда применяется и его расширенное толкование как любого набора средств и правил для построения сети, например, "технология сквозной маршрутизации", "технология создания защищенного канала", "технология IP-сетей".

Протоколы, на основе которых строится сеть определенной технологии (в узком смысле), специально разрабатывались для совместной работы, поэтому от разработчика сети не требуется дополнительных усилий по организации их взаимодействия. Иногда сетевые технологии называют *базовыми технологиями*, имея в виду то, что на их основе строится базис любой сети. Примерами базовых сетевых технологий могут служить наряду с Ethernet такие известные технологии локальных сетей как, Token Ring и FDDI, или же технологии территориальных сетей X.25 и frame relay. Для получения работоспособной сети в этом случае достаточно приобрести программные и аппаратные средства, относящиеся к одной базовой технологии - сетевые адапте-

ры с драйверами, концентраторы, коммутаторы, кабельную систему и т.п., - и соединить их в соответствии с требованиями стандарта на данную технологию.

Стандарт Ethernet был принят в 1980 году. Число сетей, построенных на основе этой технологии, к настоящему моменту оценивается в 5 миллионов, а количество компьютеров, работающих в таких сетях, - в 50 миллионов.

2.4. Структуризация как средство построения больших сетей

В сетях с небольшим (10-30) количеством компьютеров чаще всего используется одна из типовых топологий - общая шина, кольцо, звезда или полносвязная сеть. Все перечисленные топологии обладают свойством однородности, то есть все компьютеры в такой сети имеют одинаковые права в отношении доступа к другим компьютерам (за исключением центрального компьютера при соединении звезда). Такая однородность структуры делает простой процедуру наращивания числа компьютеров, облегчает обслуживание и эксплуатацию сети.

Однако при построении больших сетей однородная структура связей превращается из преимущества в недостаток. В таких сетях использование типовых структур порождает различные ограничения, важнейшими из которых являются:

- ограничения на длину связи между узлами;
- ограничения на количество узлов в сети;
- ограничения на интенсивность трафика, порождаемого узлами сети.

Физическая структуризация сети

Простейшее из коммуникационных устройств - *повторитель (repeater)* - используется для физического соединения различных сегментов кабеля локальной сети с целью увеличения общей длины сети. Повторитель передает сигналы, приходящие из одного сегмента сети, в другие ее сегменты (рис. 2.6). Повторитель позволяет преодолеть ограничения на длину линий связи за счет улучшения качества передаваемого сигнала - восстановления его мощности и амплитуды, улучшения фронтов и т.п.

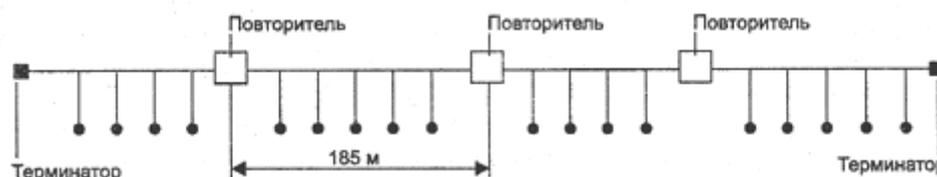


Рис. 2.6. Повторитель позволяет увеличить длину сети Ethernet

Повторитель, который имеет несколько портов и соединяет несколько физических сегментов, часто называют *концентратором (concentrator)* или *хабом (hub)*. Эти названия (hub - основа, центр деятельности) отражают тот факт, что в данном устройстве сосредоточиваются все связи между сегментами сети.

Концентраторы характерны практически для всех базовых технологий локальных сетей - Ethernet, ArcNet, Token Ring, FDDI, Fast Ethernet, Gigabit Ethernet, 100VG-AnyLAN.

Логическая структуризация сети

Физическая структуризация сети полезна во многих отношениях, однако в ряде случаев, обычно относящихся к сетям большого и среднего размера, невозможно обойтись без логической структуризации сети. Наиболее важной проблемой, не решаемой путем физической структуризации, остается проблема перераспределения передаваемого трафика между различными физическими сегментами сети.

В большой сети естественным образом возникает неоднородность информационных потоков: сеть состоит из множества подсетей рабочих групп, отделов, филиалов предприятия и других административных образований. Очень часто наиболее интенсивный обмен данными наблюдается между компьютерами, принадлежащими к одной подсети, и только небольшая часть обращений происходит к ресурсам компьютеров, находящихся вне локальных рабочих групп. Для повышения эффективности работы сети неоднородность информационных потоков необходимо учитывать.

Сеть с типовой топологией (шина, кольцо, звезда), в которой все физические сегменты рассматриваются в качестве одной разделяемой среды, оказывается неадекватной структуре информационных потоков в большой сети. Например, в сети с общей шиной взаимодействие любой пары компьютеров занимает ее на все время обмена, поэтому при увеличении числа компьютеров в сети шина становится узким местом. Компьютеры одного отдела вынуждены ждать, когда окончит обмен пара компьютеров другого отдела, и это при том, что необходимость в связи между компьютерами двух разных отделов возникает гораздо реже и требует совсем небольшой пропускной способности.

Этот случай иллюстрирует рис. 2.8, а. Здесь показана сеть, построенная с использованием концентраторов. Пусть компьютер А, находящийся в одной подсети с компьютером В, посылает ему данные. Несмотря на разветвленную физическую структуру сети, концентраторы распространяют любой кадр по всем ее сегментам. Поэтому кадр, посылаемый компьютером А компьютеру В, хотя и не нужен компьютерам отделов 2 и 3, в соответствии с логикой работы концентраторов поступает на эти сегменты тоже. И до тех пор, пока компьютер В не получит адресованный ему кадр, ни один из компьютеров этой сети не сможет передавать данные.

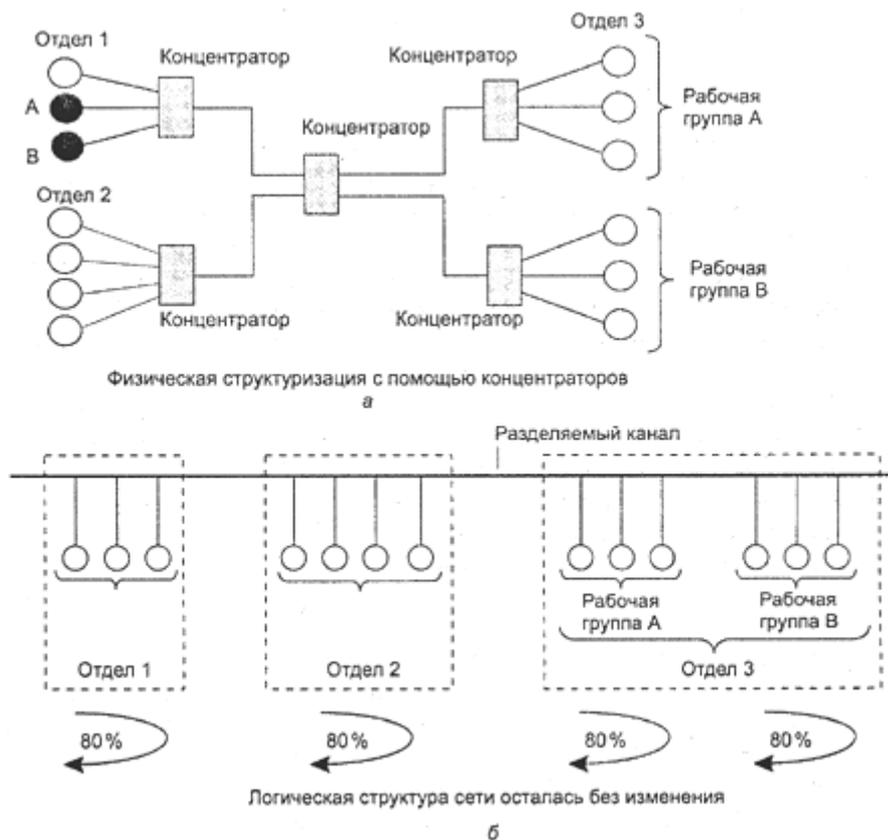


Рис. 2.8. Противоречие между логической структурой сети и структурой информационных потоков

Такая ситуация возникает из-за того, что логическая структура данной сети осталась однородной - она никак не учитывает увеличение интенсивности трафика внутри отдела и предоставляет всем парам компьютеров равные возможности по обмену информацией (рис. 2.8, б).

Решение проблемы состоит в отказе от идеи единой однородной разделяемой среды. Например, в рассмотренном выше примере желательно было бы сделать так, чтобы кадры, которые передают компьютеры отдела 1, выходили бы за пределы этой части сети в том и только в том случае, если эти кадры направлены какому-либо компьютеру из других отделов. С другой стороны, в сеть каждого из отделов должны попадать те и только те кадры, которые адресованы узлам этой сети. При такой организации работы сети ее производительность существенно повысится, так как компьютеры одного отдела не будут простаивать в то время, когда обмениваются данными компьютеры других отделов.

Нетрудно заметить, что в предложенном решении мы отказались от идеи общей разделяемой среды в пределах всей сети, хотя и оставили ее в пределах каждого отдела. Пропускная способность линий связи между отделами не должна совпадать с пропускной способностью среды внутри отделов. Если трафик между отделами составляет только 20 % трафика внутри отдела (как уже отмечалось, эта величина может быть другой), то и пропускная способность линий связи и коммуникационного оборудования, соединяющего отделы, может быть значительно ниже внутреннего трафика сети отдела.

ВНИМАНИЕ Распространение трафика, предназначенного для компьютеров некоторого сегмента сети, только в пределах этого сегмента, называется локализацией трафика. Логическая структуризация сети - это процесс разбиения сети на сегменты с локализованным трафиком.

Для логической структуризации сети используются такие коммуникационные устройства, как мосты, коммутаторы, маршрутизаторы и шлюзы.

Мост (bridge) делит разделяемую среду передачи сети на части (часто называемые логическими сегментами), передавая информацию из одного сегмента в другой только в том случае, если такая передача действительно необходима, то есть если адрес компьютера назначения принадлежит другой подсети. Тем самым мост изолирует трафик одной подсети от трафика другой, повышая общую производительность передачи данных в сети. Локализация трафика не только экономит пропускную способность, но и уменьшает возможность несанкционированного доступа к данным, так как кадры не выходят за пределы своего сегмента и их сложнее перехватить злоумышленнику.

На рис. 2.9 показана сеть, которая была получена из сети с центральным концентратором (см. рис. 2.8) путем его замены на мост. Сети 1-го и 2-го отделов состоят из отдельных логических сегментов, а сеть отдела 3 - из двух логических сегментов. Каждый логический сегмент построен на базе концентратора и имеет простейшую физическую структуру, образованную отрезками кабеля, связывающими компьютеры с портами концентратора.

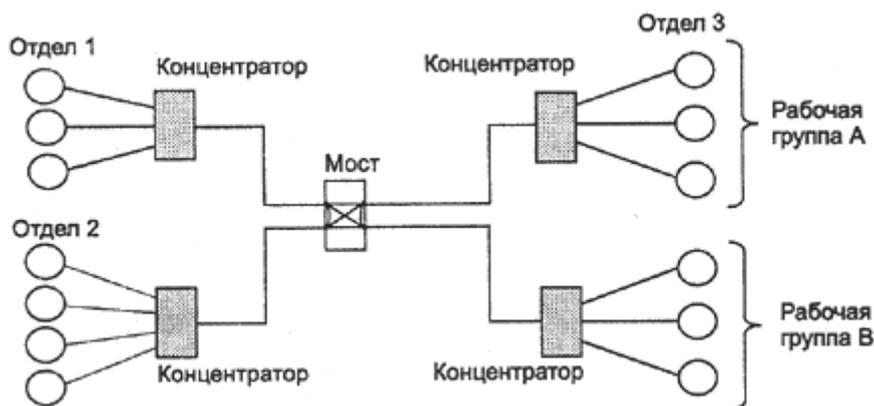


Рис. 2.9. Логическая структуризация сети с помощью моста

Мосты используют для локализации трафика аппаратные адреса компьютеров. Это затрудняет распознавание принадлежности того или иного компьютера к определенному логическому сегменту - сам адрес не содержит никакой информации по этому поводу. Поэтому мост достаточно упрощенно представляет деление сети на сегменты - он запоминает, через какой порт на него поступил кадр данных от каждого компьютера сети, и в дальнейшем передает кадры, предназначенные для этого компьютера, на этот порт. Точной топологии связей между логическими сегментами мост не знает. Из-за этого применение мостов приводит к значительным ограничениям на конфигурацию связей сети - сегменты должны быть соединены таким образом, чтобы в сети не образовывались замкнутые контуры.

Коммутатор (switch, switching hub) по принципу обработки кадров ничем не отличается от моста. Основное его отличие от моста состоит в том, что он является своего рода коммуникационным мультипроцессором, так как каждый его порт оснащен специализированным процессором, который обрабатывает кадры по алгоритму моста независимо от процессоров других портов. За счет этого общая производительность коммутатора обычно намного выше производительности традиционного моста, имеющего один процессорный блок. Можно сказать, что коммутаторы - это мосты нового поколения, которые обрабатывают кадры в параллельном режиме.

Ограничения, связанные с применением мостов и коммутаторов - по топологии связей, а также ряд других, - привели к тому, что в ряду коммуникационных устройств появился еще один тип оборудования - *маршрутизатор (router)*. Маршрутизаторы более надежно и более эффективно, чем мосты, изолируют трафик отдельных частей сети друг от друга. Маршрутизаторы образуют логические сегменты посредством явной адресации, поскольку используют не плоские аппаратные, а составные числовые адреса. В этих адресах имеется поле номера сети, так что все компьютеры, у которых значение этого поля одинаково, принадлежат к одному сегменту, называемому в данном случае подсетью (subnet).

Кроме локализации трафика маршрутизаторы выполняют еще много других полезных функций. Так, маршрутизаторы могут работать в сети с замкнутыми контурами, при этом они осуществляют выбор наиболее рационального маршрута из нескольких возможных. Сеть, представленная на рис. 2.10, отличается от своей предшественницы (см. рис. 2.9) тем, что между подсетями отделов 1 и 2 проложена дополнительная связь, которая может использоваться как для повышения производительности сети, так и для повышения ее надежности.

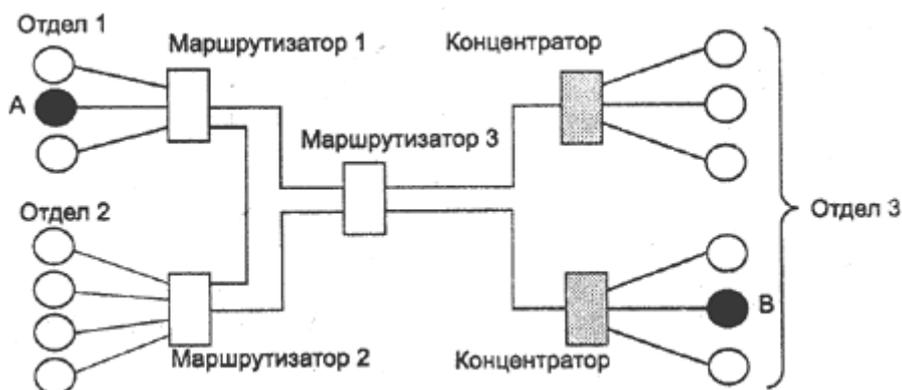


Рис. 2.10. Логическая структуризация сети с помощью маршрутизаторов

2.5. Сетевые службы

Для конечного пользователя сеть - это не компьютеры, кабели и концентраторы и даже не информационные потоки, для него сеть - это, прежде всего, тот набор сетевых служб, с помощью которых он получает возможность просмотреть список имеющихся в сети компьютеров, прочитать уда-

ленный файл, распечатать документ на "чужом" принтере или послать почтовое сообщение. Именно совокупность предоставляемых возможностей - насколько широк их выбор, насколько они удобны, надежны и безопасны - определяет для пользователя облик той или иной сети.

Литература:

1. В.В.Воеводин, Вл.В.Воеводин - "Параллельные вычисления", издательство "БХВ ", 2002.
2. В. Корнеев В. , "Современные микропроцессоры", издательство "Нолидж", 2003.
3. Э. Таненбаум, "Архитектура компьютера", СПб, издательство «Питер», 2002.
4. В.Г.Олифер, Н.А.Олифер Компьютерные сети, издательство "Питер", 2000.
5. К. Хамахер, З. Вранешич, С. Заки, Организация ЭВМ. СПб, издательство «Питер», 2003 г.

Контрольные задания для СРС

1. *Привести пример системы пакетной обработки и многотерминальной системы.*
2. *Использование мэйнфреймов в настоящее время. Характеристики мэйнфреймов.*
3. *Сетевая технология ARCNet.*
4. *Беспроводные сетевые технологии.*
5. *Корпоративные сети современных предприятий.*

Лекция 6

Тема: Модель взаимодействия открытых систем и проблемы стандартизации

План лекции

1. Многоуровневый подход. Протокол. Интерфейс. Стек протоколов
2. Модель OSI
3. Уровни модели OSI

Универсальный тезис о пользе стандартизации, справедливый для всех отраслей, в компьютерных сетях приобретает особое значение. Суть сети - это соединение разного оборудования, а значит, проблема совместимости является одной из наиболее острых. Без принятия всеми производителями общепринятых правил построения оборудования прогресс в деле "строительства" сетей был бы невозможен. Поэтому все развитие компьютерной отрасли в конечном счете отражено в стандартах - любая новая технология только тогда приобретает "законный" статус, когда ее содержание закрепляется в соответствующем стандарте.

В компьютерных сетях идеологической основой стандартизации является многоуровневый подход к разработке средств сетевого взаимодействия. Именно на основе этого подхода была разработана стандартная семиуровневая модель взаимодействия открытых систем, ставшая своего рода универсальным языком сетевых специалистов.

1.1 Многоуровневый подход. Протокол. Интерфейс. Стек протоколов

Организация взаимодействия между устройствами в сети является сложной задачей. Как известно, для решения сложных задач используется универсальный прием - декомпозиция, то есть разбиение одной сложной задачи на несколько более простых задач-модулей. Процедура декомпозиции включает в себя четкое определение функций каждого модуля, решающего отдельную задачу, и интерфейсов между ними. В результате достигается логическое упрощение задачи, а кроме того, появляется возможность модификации отдельных модулей без изменения остальной части системы.

Многоуровневое представление средств сетевого взаимодействия имеет свою специфику, связанную с тем, что в процессе обмена сообщениями участвуют две машины, то есть в данном случае необходимо организовать согласованную работу двух "иерархий". При передаче сообщений оба участника сетевого обмена должны принять множество соглашений. Например, они должны согласовать уровни и форму электрических сигналов, способ определения длины сообщений, договориться о методах контроля достоверности и т.п. Другими словами, соглашения должны быть приняты для всех уровней, начиная от самого низкого - уровня передачи битов - до самого высокого, реализующего сервис для пользователей сети.

На рис. 3.1 показана модель взаимодействия двух узлов. С каждой стороны средства взаимодействия представлены четырьмя уровнями. Процедура взаимодействия этих двух узлов может быть описана в виде набора правил

взаимодействия каждой пары соответствующих уровней обеих участвующих сторон. нормализованные правила, определяющие последовательность и формат сообщений, которыми обмениваются сетевые компоненты, лежащие на одном уровне, но в разных узлах, называются *протоколом*.

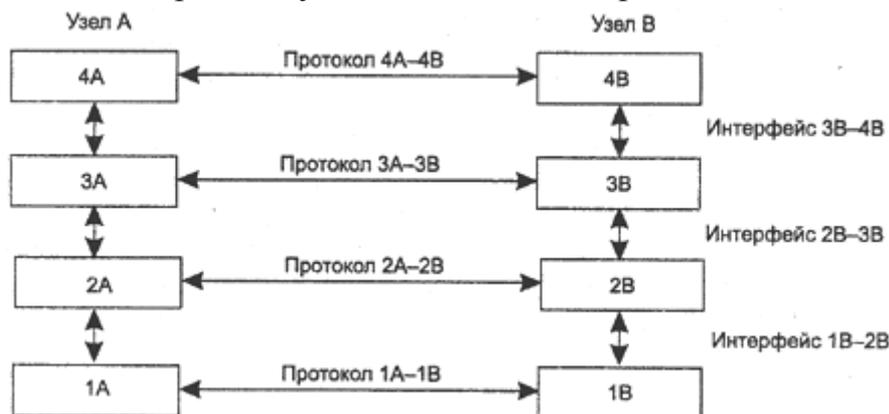


Рис. 3.1. Взаимодействие двух узлов

Модули, реализующие протоколы соседних уровней и находящиеся в одном узле, также взаимодействуют друг с другом в соответствии с четко определенными правилами и с помощью стандартизованных форматов сообщений. Эти правила принято называть *интерфейсом*. Интерфейс определяет набор сервисов, предоставляемый данным уровнем соседнему уровню. В сущности, протокол и интерфейс выражают одно и то же понятие, но традиционно в сетях за ними закрепили разные области действия: протоколы определяют правила взаимодействия модулей одного уровня в разных узлах, а интерфейсы - модулей соседних уровней одном узле.

Средства каждого уровня должны обрабатывать, во-первых, свой собственный протокол, а во-вторых, интерфейсы с соседними уровнями. Иерархически организованный набор протоколов, достаточный для организации взаимодействия узлов в сети, называется *стеком коммуникационных протоколов*.

2.2. Модель OSI

Из того, что протокол является соглашением, принятым двумя взаимодействующими объектами, в данном случае двумя работающими в сети компьютерами, совсем не следует, что он обязательно является стандартным. Но на практике при реализации сетей стремятся использовать стандартные протоколы. Это могут быть фирменные, национальные или международные стандарты.

В начале 80-х годов ряд международных организаций по стандартизации - ISO, ITU-T и некоторые другие - разработали модель, которая сыграла значительную роль в развитии сетей. Эта модель называется моделью взаимодействия открытых систем (Open System Interconnection, OSI) или моделью OSI. Модель OSI определяет различные уровни взаимодействия систем, дает им стандартные имена и указывает, какие функции должен выполнять каждый уровень. Модель OSI была разработана на основании большого опыта, полу-

ченного при создании компьютерных сетей, в основном глобальных, в 70-е годы. Полное описание этой модели занимает более 1000 страниц текста. В модели OSI (рис. 3.2) средства взаимодействия делятся на семь уровней: прикладной, представительный, сеансовый, транспортный, сетевой, канальный и физический. Каждый уровень имеет дело с одним определенным аспектом взаимодействия сетевых устройств.

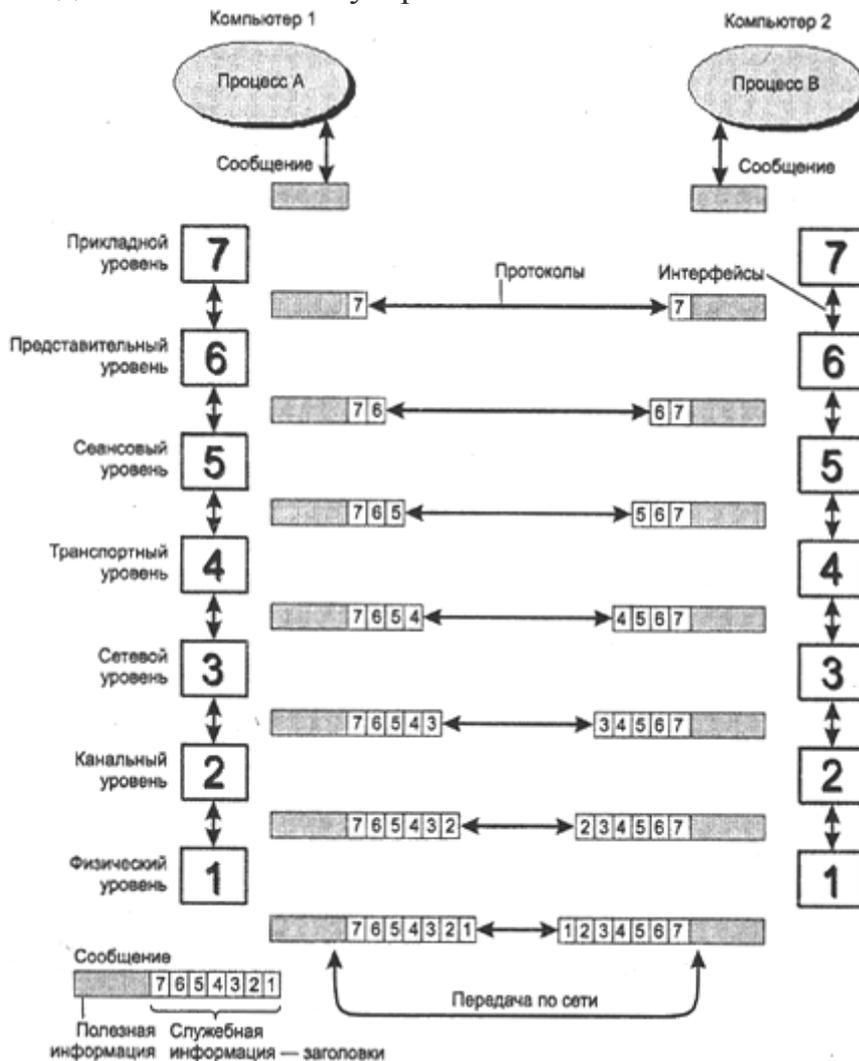


Рис. 3.2. Модель взаимодействия открытых систем ISO/OSI

Модель OSI описывает только системные средства взаимодействия, реализуемые операционной системой, системными утилитами, системными аппаратными средствами. Модель не включает средства взаимодействия приложений конечных пользователей. Свои собственные протоколы взаимодействия приложения реализуют, обращаясь к системным средствам. Поэтому необходимо различать уровень взаимодействия приложений и прикладной уровень.

Следует также иметь в виду, что приложение может взять на себя функции некоторых верхних уровней модели OSI. Например, некоторые СУБД имеют встроенные средства удаленного доступа к файлам. В этом случае приложение, выполняя доступ к удаленным ресурсам, не использует системную файловую службу; оно обходит верхние уровни модели OSI и обращается напрямую к системным средствам, ответственным за транспортировку.

ку сообщений по сети, которые располагаются на нижних уровнях модели OSI.

В модели OSI различаются два основных типа протоколов. В протоколах *с установлением соединения (connection-oriented)* перед обменом данными отправитель и получатель должны сначала установить соединение и, возможно, выбрать некоторые параметры протокола, которые они будут использовать при обмене данными. После завершения диалога они должны разорвать это соединение. Телефон - это пример взаимодействия, основанного на установлении соединения.

Вторая группа протоколов - *протоколы без предварительного установления соединения (connectionless)*. Такие протоколы называются также дейтаграммными протоколами. Отправитель просто передает сообщение, когда оно готово. Опускание письма в почтовый ящик - это пример связи без предварительного установления соединения. При взаимодействии компьютеров используются протоколы обоих типов.

1.3 Уровни модели OSI

Физический уровень

Физический уровень (Physical layer) имеет дело с передачей битов по физическим каналам связи, таким, например, как коаксиальный кабель, витая пара, оптоволоконный кабель или цифровой территориальный канал. К этому уровню имеют отношение характеристики физических сред передачи данных, такие как полоса пропускания, помехозащищенность, волновое сопротивление и другие. На этом же уровне определяются характеристики электрических сигналов, передающих дискретную информацию, например, крутизна фронтов импульсов, уровни напряжения или тока передаваемого сигнала, тип кодирования, скорость передачи сигналов. Кроме этого, здесь стандартизируются типы разъемов и назначение каждого контакта.

Функции физического уровня реализуются во всех устройствах, подключенных к сети. Со стороны компьютера функции физического уровня выполняются сетевым адаптером или последовательным портом.

Канальный уровень

На физическом уровне просто пересылаются биты. При этом не учитывается, что в некоторых сетях, в которых линии связи используются (разделяются) попеременно несколькими парами взаимодействующих компьютеров, физическая среда передачи может быть занята. Поэтому одной из задач канального уровня (Data Link layer) является проверка доступности среды передачи. Другой задачей канального Уровня является реализация механизмов обнаружения и коррекции ошибок. Для этого на канальном уровне биты группируются в наборы, называемые *кадрами (frames)*. Канальный уровень обеспечивает корректность передачи каждого кадра, помещая специальную последовательность бит в начало и конец каждого кадра, для его выделения, а также вычисляет контрольную сумму, обрабатывая все байты кадра опре-

деленным способом и добавляя контрольную сумму к кадру. Когда кадр приходит по сети, получатель снова вычисляет контрольную сумму полученных данных и сравнивает результат с контрольной суммой из кадра. Если они совпадают, кадр считается правильным и принимается. Если же контрольные суммы не совпадают, то фиксируется ошибка. Канальный уровень может не только обнаруживать ошибки, но и исправлять их за счет повторной передачи поврежденных кадров. Необходимо отметить, что функция исправления ошибок не является обязательной для канального уровня, поэтому в некоторых протоколах этого уровня она отсутствует, например, в Ethernet и frame relay.

Сетевой уровень

Сетевой уровень (Network layer) служит для образования единой транспортной системы, объединяющей несколько сетей, причем эти сети могут использовать совершенно различные принципы передачи сообщений между конечными узлами и обладать произвольной структурой связей. Функции сетевого уровня достаточно разнообразны. Начнем их рассмотрение на примере объединения локальных сетей.

Протоколы канального уровня локальных сетей обеспечивают доставку данных между любыми узлами только в сети с соответствующей типовой топологией, например топологией иерархической звезды. Это очень жесткое ограничение, которое не позволяет строить сети с развитой структурой, например, сети, объединяющие несколько сетей предприятия в единую сеть, или высоконадежные сети, в которых существуют избыточные связи между узлами. Можно было бы усложнять протоколы канального уровня для поддержания петлевидных избыточных связей, но принцип разделения обязанностей между уровнями приводит к другому решению. Чтобы с одной, стороны сохранить простоту процедур передачи данных для типовых топологий, а с другой допустить использование произвольных топологий, вводится дополнительный сетевой уровень.

На сетевом уровне сам термин *сеть* наделяют специфическим значением. В данном случае под сетью понимается совокупность компьютеров, соединенных между собой в соответствии с одной из стандартных типовых топологий и использующих для передачи данных один из протоколов канального уровня, определенный для этой топологии.

Транспортный уровень

На пути от отправителя к получателю пакеты могут быть искажены или потеряны. Хотя некоторые приложения имеют собственные средства обработки ошибок, существуют и такие, которые предпочитают сразу иметь дело с надежным соединением. Транспортный уровень (Transport layer) обеспечивает приложениям или верхним уровням стека - прикладному и сеансовому - передачу данных с той степенью надежности, которая им требуется. Модель OSI определяет пять классов сервиса, предоставляемых транспортным уровнем. Эти виды сервиса отличаются качеством предоставляемых услуг: срочностью, возможностью восстановления прерванной связи, наличием средств мультиплексирования нескольких соединений между различными приклад-

ными протоколами через общий транспортный протокол, а главное - способностью к обнаружению и исправлению ошибок передачи, таких как искажение, потеря и дублирование пакетов.

Сеансовый уровень

Сеансовый уровень (Session layer) обеспечивает управление диалогом: фиксирует, какая из сторон является активной в настоящий момент, предоставляет средства синхронизации. Последние позволяют вставлять контрольные точки в длинные передачи, чтобы в случае отказа можно было вернуться назад к последней контрольной точке, а не начинать все с начала. На практике немногие приложения используют сеансовый уровень, и он редко реализуется в виде отдельных протоколов, хотя функции этого уровня часто объединяют с функциями прикладного уровня и реализуют в одном протоколе.

Представительный уровень

Представительный уровень (Presentation layer) имеет дело с формой представления передаваемой по сети информации, не меняя при этом ее содержания. За счет уровня представления информация, передаваемая прикладным уровнем одной системы, всегда понятна прикладному уровню другой системы. С помощью средств данного уровня протоколы прикладных уровней могут преодолеть синтаксические различия в представлении данных или же различия в кодах символов, например кодов ASCII и EBCDIC. На этом уровне может выполняться шифрование и дешифрование данных, благодаря которому секретность обмена данными обеспечивается сразу для всех прикладных служб. Примером такого протокола является протокол Secure Socket Layer (SSL), который обеспечивает секретный обмен сообщениями для протоколов прикладного уровня стека TCP/IP.

Прикладной уровень

Прикладной уровень (Application layer) - это в действительности просто набор разнообразных протоколов, с помощью которых пользователи сети получают доступ к разделяемым ресурсам, таким как файлы, принтеры или гипертекстовые Web-страницы, а также организуют свою совместную работу, например, с помощью протокола электронной почты. Единица данных, которой оперирует прикладной уровень, обычно называется *сообщением* (message).

Существует очень большое разнообразие служб прикладного уровня. Приведем в качестве примера хотя бы несколько наиболее распространенных реализаций файловых служб: NCP в операционной системе Novell NetWare, SMB в Microsoft Windows NT, NFS, FTP и TFTP, входящие в стек TCP/IP.

Сетезависимые и сетезависимые уровни

Функции всех уровней модели OSI могут быть отнесены к одной из двух групп: либо к функциям, зависящим от конкретной технической реализации сети, либо к функциям, ориентированным на работу с приложениями.

Три нижних уровня - физический, канальный и сетевой - являются сетезависимыми, то есть протоколы этих уровней тесно связаны с технической реализацией сети и используемым коммуникационным оборудованием. На-

пример, переход на оборудование FDDI означает полную смену протоколов физического и канального уровней во всех узлах сети.

Три верхних уровня - прикладной, представительный и сеансовый - ориентированы на приложения и мало зависят от технических особенностей построения сети. На протоколы этих уровней не влияют какие бы то ни было изменения в топологии сети, замена оборудования или переход на другую сетевую технологию. Так, переход от Ethernet на высокоскоростную технологию 100VG-AnyLAN не потребует никаких изменений в программных средствах, реализующих функции прикладного, представительного и сеансового уровней.

Транспортный уровень является промежуточным, он скрывает все детали функционирования нижних уровней от верхних. Это позволяет разрабатывать приложения, не зависящие от технических средств непосредственной транспортировки сообщений.

1.4. Стандартные стеки коммуникационных протоколов

Важнейшим направлением стандартизации в области вычислительных сетей является стандартизация коммуникационных протоколов. В настоящее время в сетях используется большое количество стеков коммуникационных протоколов. Наиболее популярными являются стеки: TCP/IP, IPX/SPX, NetBIOS/SMB, DECnet, SNA и OSI. Все эти стеки, кроме SNA на нижних уровнях - физическом и канальном, - используют одни и те же хорошо стандартизованные протоколы Ethernet, Token Ring, FDDI и некоторые другие, которые позволяют использовать во всех сетях одну и ту же аппаратуру. Зато на верхних уровнях все стеки работают по своим собственным протоколам. Эти протоколы часто не соответствуют рекомендуемой модели OSI разбиению на уровни. В частности, функции сеансового и представительного уровня, как правило, объединены с прикладным уровнем. Такое несоответствие связано с тем, что модель OSI появилась как результат обобщения уже существующих и реально используемых стеков, а не наоборот.

Стек TCP/IP

Стек TCP/IP был разработан по инициативе Министерства обороны США более 20 лет назад для связи экспериментальной сети ARPAnet с другими сетями как набор общих протоколов для разнородной вычислительной среды. Большой вклад в развитие стека TCP/IP, который получил свое название по популярным протоколам IP и TCP, внес университет Беркли, реализовав протоколы стека в своей версии ОС UNIX. Популярность этой операционной системы привела к широкому распространению протоколов TCP, IP и других протоколов стека. Сегодня этот стек используется для связи компьютеров всемирной информационной сети Internet, а также в огромном числе корпоративных сетей.

Стек IPX/SPX

Этот стек является оригинальным стеком протоколов фирмы Novell, разработанным для сетевой операционной системы NetWare еще в начале 80-х годов. Протоколы сетевого и сеансового уровней Internetwork Packet

Exchange (IPX) и Sequenced Packet Exchange (SPX), которые дали название стеку, являются прямой адаптацией протоколов XNS фирмы Херох, распространенных в гораздо меньшей степени, чем стек IPX/SPX. Популярность стека IPX/SPX непосредственно связана с операционной системой Novell NetWare, которая еще сохраняет мировое лидерство по числу установленных систем, хотя в последнее время ее популярность несколько снизилась и по темпам роста она отстает от Microsoft Windows NT.

Стек NetBIOS/SMB

Этот стек широко используется в продуктах компаний IBM и Microsoft. На физическом и канальном уровнях этого стека используются все наиболее распространенные протоколы Ethernet, Token Ring, FDDI и другие. На верхних уровнях Работают протоколы NetBEUI и SMB.

Протокол NetBIOS (Network Basic Input/Output System) появился в 1984 году как сетевое расширение стандартных функций базовой системы ввода/вывода (BIOS) IBM PC для сетевой программы PC Network фирмы IBM. В дальнейшем этот протокол был заменен так называемым протоколом расширенного пользовательского интерфейса NetBEUI - NetBIOS Extended User Interface. Для обеспечения совместимости приложений в качестве интерфейса к протоколу NetBEUI был сохранен интерфейс NetBIOS. Протокол NetBEUI разрабатывался как эффективный протокол, потребляющий немного ресурсов и предназначенный для сетей, насчитывающих не более 200 рабочих станций. Этот протокол содержит много полезных сетевых функций, которые можно отнести к сетевому, транспортному и сеансовому уровням модели OSI, однако с его помощью невозможна маршрутизация пакетов. Это ограничивает применение протокола NetBEUI локальными сетями, не разделенными на подсети, и делает невозможным его использование в составных сетях. Некоторые ограничения NetBEUI снимаются реализацией этого протокола NBF (NetBEUI Frame), которая включена в операционную систему Microsoft Windows NT.

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

Стеки протоколов SNA фирмы IBM, DECnet корпорации Digital Equipment и AppleTalk/AFP фирмы Apple применяются в основном в операционных системах и сетевом оборудовании этих фирм.

Литература:

1. В.В.Воеводин, Вл.В.Воеводин - "Параллельные вычисления", издательство "БХВ", 2002.
2. В. Корнеев В. , "Современные микропроцессоры", издательство "Нолидж", 2003.
3. Э. Таненбаум, "Архитектура компьютера", СПб, издательство «Питер», 2002.

4. В.Г.Олифер, Н.А.Олифер Компьютерные сети, издательство “Питер”, 2000.

5. К. Хамахер, З. Вранешич, С. Заки, Организация ЭВМ. СПб, издательство «Питер», 2003 г.

Контрольные задания для СРС

1. *Дать понятие “открытой системы”.*
2. *Дать понятие модульности и стандартизации.*
3. *Источники стандартов отрасли.*
4. *Стек DECnet.*
5. *Стек OSI.*

Лекция 7

Тема: Локальные и глобальные сети. Требования, предъявляемые к современным вычислительным сетям

План лекции

1. Введение
2. Локальные и глобальные сети
3. Требования, предъявляемые к современным вычислительным сетям

1.1 Локальные и глобальные сети

Для классификации компьютерных сетей используются различные признаки, но чаще всего сети делят на типы по территориальному признаку, то есть по величине территории, которую покрывает сеть.

Классифицируя сети по территориальному признаку, различают *локальные (Local Area Networks - LAN)*, *глобальные (Wide Area Networks - WAN)* и *городские (Metropolitan Area Networks - MAN) сети*.

- LAN - сосредоточены на территории не более 1-2 км; построены с использованием дорогих высококачественных линий связи, которые позволяют, применяя простые методы передачи данных, достигать высоких скоростей обмена данными порядка 100 Мбит/с. Предоставляемые услуги отличаются широким разнообразием и обычно предусматривают реализацию в режиме on-line.
- WAN - объединяют компьютеры, рассредоточенные на расстоянии сотен и тысяч километров. Часто используются уже существующие не очень качественные линии связи. Более низкие, чем в локальных сетях, скорости передачи данных (десятки килобит в секунду) ограничивают набор предоставляемых услуг передачей файлов, преимущественно не в оперативном, а в фоновом режиме, с использованием электронной почты. Для устойчивой передачи дискретных данных применяются более сложные методы и оборудование, чем в локальных сетях.
- MAN - занимают промежуточное положение между локальными и глобальными сетями. При достаточно больших расстояниях между узлами (десятки километров) они обладают качественными линиями связи и высокими скоростями обмена, иногда даже более высокими, чем в классических локальных сетях. Как и в случае локальных сетей, при построении MAN уже существующие линии связи не используются, а прокладываются заново.

1.2 Требования, предъявляемые к современным вычислительным сетям

Главным требованием, предъявляемым к сетям, является выполнение сетью ее основной функции - обеспечение пользователям потенциальной возможности доступа к разделяемым ресурсам всех компьютеров, объединенных в сеть. Все остальные требования - производительность, надежность, совместимость, управляемость, защищенность, расширяемость и масштабируемость - связаны с качеством выполнения этой основной задачи.

Производительность

Потенциально высокая производительность - это одно из основных свойств распределенных систем, к которым относятся компьютерные сети. Это свойство обеспечивается возможностью распараллеливания работ между несколькими компьютерами сети. К сожалению, эту возможность не всегда удается реализовать.

Существует несколько основных характеристик производительности сети:

- время реакции;
- пропускная способность;
- задержка передачи и вариация задержки передачи.

Время реакции сети является интегральной характеристикой производительности сети с точки зрения пользователя. Именно эту характеристику имеет в виду пользователь, когда говорит: "Сегодня сеть работает медленно".

В общем случае время реакции определяется как интервал времени между возникновением запроса пользователя к какой-либо сетевой службе и получением ответа на этот запрос.

Пропускная способность отражает объем данных, переданных сетью или ее частью в единицу времени. Пропускная способность уже не является пользовательской характеристикой, так как она говорит о скорости выполнения внутренних операций сети - передачи пакетов данных между узлами сети через различные коммуникационные устройства. Зато она непосредственно характеризует качество выполнения основной функции сети - транспортировки сообщений - и поэтому чаще используется при анализе производительности сети, чем время реакции.

Пропускная способность измеряется либо в битах в секунду, либо в пакетах в секунду. Пропускная способность может быть мгновенной, максимальной и средней.

Средняя пропускная способность вычисляется путем деления общего объема переданных данных на время их передачи, причем выбирается достаточно длительный промежуток времени - час, день или неделя.

Мгновенная пропускная способность отличается от средней тем, что для усреднения выбирается очень маленький промежуток времени - например, 10 мс или 1с.

Максимальная пропускная способность - это наибольшая мгновенная пропускная способность, зафиксированная в течение периода наблюдения.

Пропускную способность можно измерять между любыми двумя узлами или точками сети, например между клиентским компьютером и сервером, между входным и выходным портами маршрутизатора. Для анализа и настройки сети очень полезно знать данные о пропускной способности отдельных элементов сети.

Задержка передачи определяется как задержка между моментом поступления пакета на вход какого-либо сетевого устройства или части сети и моментом появления его на выходе этого устройства. Этот параметр произво-

дительности по смыслу близок ко времени реакции сети, но отличается тем, что всегда характеризует только сетевые этапы обработки данных, без задержек обработки компьютерами сети. Обычно качество сети характеризуют величинами *максимальной задержки передачи* и *вариацией задержки*.

Надежность и безопасность

Одной из первоначальных целей создания распределенных систем, к которым относятся и вычислительные сети, являлось достижение большей надежности по сравнению с отдельными вычислительными машинами.

Важно различать несколько аспектов надежности. Для технических устройств используются такие показатели надежности, как среднее время наработки на отказ, вероятность отказа, интенсивность отказов. Однако эти показатели пригодны для оценки надежности простых элементов и устройств, которые могут находиться только в двух состояниях - работоспособном или неработоспособном. Сложные системы, состоящие из многих элементов, кроме состояний работоспособности и неработоспособности, могут иметь и другие промежуточные состояния, которые эти характеристики не учитывают. В связи с этим для оценки надежности сложных систем применяется другой набор характеристик.

Готовность или коэффициент готовности (availability) означает долю времени, в течение которого система может быть использована. Готовность может быть улучшена путем введения избыточности в структуру системы: ключевые элементы системы должны существовать в нескольких экземплярах, чтобы при отказе одного из них функционирование системы обеспечивали другие.

Чтобы систему можно было отнести к высоконадежным, она должна как минимум обладать высокой готовностью, но этого недостаточно. Необходимо обеспечить *сохранность данных* и защиту их от искажений. Кроме этого, должна поддерживаться согласованность (непротиворечивость) данных, например, если для повышения надежности на нескольких файловых серверах хранятся несколько копий данных, то нужно постоянно обеспечивать их идентичность.

Так как сеть работает на основе механизма передачи пакетов между конечными узлами, то одной из характерных характеристик надежности является *вероятность доставки пакета* узлу назначения без искажений. Наряду с этой характеристикой могут использоваться и другие показатели: вероятность потери пакета (по любой из причин - из-за переполнения буфера маршрутизатора, из-за несовпадения контрольной суммы, из-за отсутствия работоспособного пути к узлу назначения и т.д.), вероятность искажения отдельного бита передаваемых данных, отношение потерянных пакетов к доставленным.

Другим аспектом общей надежности является *безопасность (security)*, то есть способность системы защитить данные от несанкционированного доступа. В распределенной системе это сделать гораздо сложнее, чем в централизованной. В сетях сообщения передаются по линиям связи, часто проходящим через общедоступные помещения, в которых могут быть установ-

лены средства прослушивания линий. Другим уязвимым местом могут быть оставленные без присмотра персональные компьютеры. Кроме того, всегда имеется потенциальная угроза взлома защиты сети от неавторизованных пользователей, если сеть имеет выходы в глобальные сети общего пользования.

Еще одной характеристикой надежности является *отказоустойчивость (fault wrance)*. В сетях под отказоустойчивостью понимается способность системы скрыть от пользователя отказ отдельных ее элементов. Например, если копии таблицы базы данных хранятся одновременно на нескольких файловых серверах, то пользователи могут просто не заметить отказ одного из них. В отказоустойчивой системе отказ одного из ее элементов приводит к некоторому снижению качества ее работы (деградации), а не к полному останову. Так, при отказе одного из файловых серверов в предыдущем примере увеличивается только время доступа к базе данных из-за уменьшения степени распараллеливания запросов, но в целом система будет продолжать выполнять свои функции.

Расширяемость и масштабируемость

Термины расширяемость и масштабируемость иногда используют как синонимы но это неверно - каждый из них имеет четко определенное самостоятельное значение.

Расширяемость (extensibility) означает возможность сравнительно легкого добавления отдельных элементов сети (пользователей, компьютеров, приложений, служб), наращивания длины сегментов сети и замены существующей аппаратуры более мощной. При этом принципиально важно, что легкость расширения системы иногда может обеспечиваться в некоторых весьма ограниченных пределах. Например, локальная сеть Ethernet, построенная на основе одного сегмента толстого коаксиального кабеля, обладает хорошей расширяемостью, в том смысле, что позволяет легко подключать новые станции. Однако такая сеть имеет ограничение на число станций - их число не должно превышать 30-40. Хотя сеть допускает физическое подключение к сегменту и большего числа станций (до 100), но при этом чаще всего резко снижается производительность сети. Наличие такого ограничения и является признаком плохой масштабируемости системы при хорошей расширяемости.

Масштабируемость (scalability) означает, что сеть позволяет наращивать количество узлов и протяженность связей в очень широких пределах, при этом производительность сети не ухудшается. Для обеспечения масштабируемости сети приходится применять дополнительное коммуникационное оборудование и специальным образом структурировать сеть. Например, хорошей масштабируемостью обладает многосегментная сеть, построенная с использованием коммутаторов и маршрутизаторов и имеющая иерархическую структуру связей. Такая сеть может включать несколько тысяч компьютеров и при этом обеспечивать каждому пользователю сети нужное качество обслуживания.

Прозрачность

Прозрачность (transparency) сети достигается в том случае, когда сеть представляется пользователям не как множество отдельных компьютеров, связанных между собой сложной системой кабелей, а как единая традиционная вычислительная машина с системой разделения времени. Известный лозунг компании Sun Microsystems: "Сеть - это компьютер" - говорит именно о такой прозрачной сети.

Поддержка разных видов трафика

Компьютерные сети изначально предназначены для совместного доступа пользователя к ресурсам компьютеров: файлам, принтерам и т.п. Трафик, создаваемый этими традиционными службами компьютерных сетей, имеет свои особенности и существенно отличается от трафика сообщений в телефонных сетях или, например, в сетях кабельного телевидения. Однако 90-е годы стали годами проникновения в компьютерные сети трафика мультимедийных данных, представляющих в цифровой форме речь и видеоизображение. Компьютерные сети стали использоваться для организации видеоконференций, обучения и развлечения на основе видеофильмов и т.п. Естественно, что для динамической передачи мультимедийного трафика требуются иные алгоритмы и протоколы и, соответственно, другое оборудование. Хотя доля мультимедийного трафика пока невелика, он уже начал свое проникновение как в глобальные, так и локальные сети, и этот процесс, очевидно, будет продолжаться с возрастающей скоростью.

Управляемость

Управляемость сети подразумевает возможность централизованно контролировать состояние основных элементов сети, выявлять и разрешать проблемы, возникающие при работе сети, выполнять анализ производительности и планировать развитие сети. В идеале средства управления сетями представляют собой систему, осуществляющую наблюдение, контроль и управление каждым элементом сети - от простейших до самых сложных устройств, при этом такая система рассматривает сеть как единое целое, а не как разрозненный набор отдельных устройств.

Хорошая система управления наблюдает за сетью и, обнаружив проблему, активизирует определенное действие, исправляет ситуацию и уведомляет администратора о том, что произошло и какие шаги предприняты. Одновременно с этим система управления должна накапливать данные, на основании которых можно планировать развитие сети. Наконец, система управления должна быть независима от производителя и обладать удобным интерфейсом, позволяющим выполнять все действия с одной консоли.

Совместимость

Совместимость или *интегрируемость* означает, что сеть способна включать в себя самое разнообразное программное и аппаратное обеспечение, то есть в ней могут сосуществовать различные операционные системы, поддерживающие разные стеки коммуникационных протоколов, и работать аппаратные средства и приложения от разных производителей. Сеть, состоя-

щая из разнотипных элементов, называется неоднородной или гетерогенной, а если гетерогенная сеть работает без проблем, то она является интегрированной. Основным путем построения интегрированных сетей - использование модулей, выполненных в соответствии с открытыми стандартами и спецификациями.

Литература:

1. В.В.Воеводин, Вл.В.Воеводин - "Параллельные вычисления", издательство "БХВ ", 2002.
2. В. Корнеев В. , "Современные микропроцессоры", издательство "Нолидж", 2003.
3. Э. Таненбаум, "Архитектура компьютера", СПб, издательство «Питер», 2002.
4. В.Г.Олифер, Н.А.Олифер Компьютерные сети, издательство "Питер", 2000.
5. К. Хамахер, З. Вранешич, С. Заки, Организация ЭВМ. СПб, издательство «Питер», 2003 г.

Контрольные задания для СРС

1. *Особенности локальных, глобальных и городских сетей.*
2. *Отличия локальных сетей от глобальных.*
3. *Тенденция к сближению локальных и глобальных сетей.*
4. *Сети отделов.*
5. *Корпоративные сети.*

Лекция 8

Тема: Линии связи

План лекции

1. Типы линий связи
2. Аппаратура линий связи
3. Характеристики линий связи
4. Стандарты кабелей

Любая сетевая технология должна обеспечить надежную и быструю передачу дискретных данных по линиям связи. И хотя между технологиями имеются большие различия, они базируются на общих принципах передачи дискретных данных, которые рассматриваются в этой главе. Эти принципы находят свое воплощение в методах представления двоичных единиц и нулей с помощью импульсных или синусоидальных сигналов в линиях связи различной физической природы, методах обнаружения и коррекции ошибок.

1.1. Типы линий связи

Линия связи (рис. 5.1) состоит в общем случае из физической среды, по которой передаются электрические информационные сигналы, аппаратуры передачи данных и промежуточной аппаратуры. Синонимом термина *линия связи (line)* является термин *канал связи (cannel)*.



Рис. 5.1. Состав линии связи

Физическая среда передачи данных (medium) может представлять собой кабель, то есть набор проводов, изоляционных и защитных оболочек и соединительных разъемов, а также земную атмосферу или космическое пространство, через которые распространяются электромагнитные волны.

В зависимости от среды передачи данных линии связи разделяются на следующие:

- проводные (воздушные);
- кабельные (медные и волоконно-оптические);
- радиоканалы наземной и спутниковой связи.

Проводные (воздушные) линии связи представляют собой провода без каких-либо изолирующих или экранирующих оплеток, проложенные между столбами и висящие в воздухе. По таким линиям связи традиционно передаются телефонные или телеграфные сигналы, но при отсутствии других возможностей эти линии используются и для передачи компьютерных данных. Скоростные качества и помехозащищенность этих линий оставляют желать много лучшего. Сегодня проводные линии связи быстро вытесняются кабельными.

Кабельные линии представляют собой достаточно сложную конструкцию. Кабель состоит из проводников, заключенных в несколько слоев изоляции: электрической, электромагнитной, механической, а также, возможно, климатической. Кроме того, кабель может быть оснащен разъемами, позволяющими быстро выполнять присоединение к нему различного оборудования. В компьютерных сетях применяются три основных типа кабеля: кабели на основе скрученных пар медных проводов, коаксиальные кабели с медной жилой, а также волоконно-оптические кабели,

Радиоканалы наземной и спутниковой связи образуются с помощью передатчика и приемника радиоволн. Существует большое количество различных типов радиоканалов отличающихся как используемым частотным диапазоном, так и дальностью канала. Диапазоны коротких, средних и длинных волн (КВ, СВ и ДВ), называемые также диапазонами амплитудной модуляции (Amplitude Modulation, АМ) по типу используемого в них метода модуляции сигнала, обеспечивают дальнюю связь, но при невысокой скорости передачи данных. Более скоростными являются каналы, работающие на диапазонах ультракоротких волн (УКВ), для которых характерна частотная модуляция (Frequency Modulation, FM), а также диапазонах сверхвысоких частот (СВЧ или microwaves). В диапазоне СВЧ (свыше 4 ГГц) сигналы уже не отражаются ионосферой Земли и для устойчивой связи требуется наличие прямой видимости между передатчиком и приемником. Поэтому такие частоты используют либо спутниковые каналы, либо радиорелейные каналы, где это условие выполняется.

1.2. Аппаратура линий связи

Аппаратура передачи данных (АПД или DCE - Data Circuit terminating Equipment) непосредственно связывает компьютеры или локальные сети пользователя с линией связи и является, таким образом, пограничным оборудованием. Традиционно аппаратуру передачи данных включают в состав линии связи. Примерами DCE являются модемы, терминальные адаптеры сетей ISDN, оптические модемы, устройства подключения к цифровым каналам. Обычно DCE работает на физическом уровне, отвечая за передачу и прием сигнала нужной формы и мощности в физическую среду.

Аппаратура пользователя линии связи, вырабатывающая данные для передачи по линии связи и подключаемая непосредственно к аппаратуре передачи данных, обобщенно носит название *оконечное оборудование данных (ООД или DTE - Data Terminal Equipment)*. Примером DTE могут служить компьютеры или маршрутизаторы локальных сетей. Эту аппаратуру не включают в состав линии связи.

Разделение оборудования на классы DCE и DTE в локальных сетях является достаточно условным. Например, адаптер локальной сети можно считать как принадлежностью компьютера, то есть DTE, так и составной частью канала связи, то есть DCE.

Промежуточная аппаратура обычно используется на линиях связи большой протяженности. Промежуточная аппаратура решает две основные задачи:

- улучшение качества сигнала;
- создание постоянного составного канала связи между двумя абонентами сети.

1.3. Характеристики линий связи

Типы характеристик и способы их определения

К основным характеристикам линий связи относятся:

- амплитудно-частотная характеристика;
- полоса пропускания;
- затухание;
- помехоустойчивость;
- перекрестные наводки на ближнем конце линии;
- пропускная способность;
- достоверность передачи данных;
- удельная стоимость.

1.4. Стандарты кабелей

Кабель - это достаточно сложное изделие, состоящее из проводников, слоев экрана и изоляции. В некоторых случаях в состав кабеля входят разъемы, с помощью которых кабели присоединяются к оборудованию. Кроме этого, для обеспечения быстрой перекоммутации кабелей и оборудования используются различные электромеханические устройства, называемые кроссовыми секциями, кроссовыми коробками или шкафами.

- *Затухание (Attenuation)*. Затухание измеряется в децибелах на метр для определенной частоты или диапазона частот сигнала.
- *Перекрестные наводки на ближнем конце (Near End Cross Talk, NEXT)*. Измеряются в децибелах для определенной частоты сигнала.
- *Импеданс (волновое сопротивление)* - это полное (активное и реактивное) сопротивление в электрической цепи. Импеданс измеряется в Омах и является относительно постоянной величиной для кабельных систем (например, для коаксиальных кабелей, используемых в стандартах Ethernet, импеданс кабеля должен составлять 50 Ом). Для неэкранированной витой пары наиболее часто используемые значения импе-

- *Активное сопротивление* - это сопротивление постоянному току в электрической цепи. В отличие от импеданса активное сопротивление не зависит от частоты и возрастает с увеличением длины кабеля.
- *Емкость* - это свойство металлических проводников накапливать энергию. Два электрических проводника в кабеле, разделенные диэлектриком, представляют собой конденсатор, способный накапливать заряд. Емкость является нежелательной величиной, поэтому следует стремиться к тому, чтобы она была как можно меньше (иногда применяют термин "паразитная емкость"). Высокое значение емкости в кабеле приводит к искажению сигнала и ограничивает полосу пропускания линии.
- *Уровень внешнего электромагнитного излучения или электрический шум*. Электрический шум - это нежелательное переменное напряжение в проводнике. Электрический шум бывает двух типов: фоновый и импульсный. Электрический шум можно также разделить на низко-, средне- и высокочастотный. Источниками фонового электрического шума в диапазоне до 150 кГц являются линии электропередачи, телефоны и лампы дневного света; в диапазоне от 150 кГц до 20 МГц - компьютеры, принтеры, ксероксы; в диапазоне от 20 МГц до 1 ГГц - телевизионные и радиопередатчики, микроволновые печи. Основными источниками импульсного электрического шума являются моторы, переключатели и сварочные агрегаты. Электрический шум измеряется в милливольтках.
- *Диаметр или площадь сечения проводника*. Для медных проводников достаточно употребительной является американская система AWG (American Wire Gauge), которая вводит некоторые условные типы проводников, например 22 AWG, 24 AWG, 26 AWG. Чем больше номер типа проводника, тем меньше его диаметр. В вычислительных сетях наиболее употребительными являются типы проводников, приведенные выше в качестве примеров. В европейских и международных стандартах диаметр проводника указывается в миллиметрах.

Кабели на основе экранированной витой пары

Экранированная витая пара STP хорошо защищает передаваемые сигналы от внешних помех, а также меньше излучает электромагнитных колебаний вовне, что защищает, в свою очередь, пользователей сетей от вредного для здоровья излучения. Наличие заземляемого экрана удорожает кабель и усложняет его прокладку, так как требует выполнения качественного заземления. Экранированный кабель применяется только для передачи данных, а голос по нему не передают.

Основным стандартом, определяющим параметры экранированной витой пары, является фирменный стандарт IBM. В этом стандарте кабели делятся не на категории, а на типы; Type 1, Type 2,..., Type 9.

Коаксиальные кабели

Существует большое количество типов коаксиальных кабелей, используемых в сетях различного типа - телефонных, телевизионных и компьютерных. Ниже приводятся основные типы и характеристики этих кабелей.

- RG-8 и RG-11 - "толстый" коаксиальный кабель, разработанный для сетей Ethernet 10Base-5. Имеет волновое сопротивление 50 Ом и внешний диаметр 0,5 дюйма (около 12 мм). Этот кабель имеет достаточно толстый внутренний проводник диаметром 2,17 мм, который обеспечивает хорошие механические и электрические характеристики (затухание на частоте 10 МГц - не хуже 18 дБ/км). Зато этот кабель сложно монтировать - он плохо гнется.
- RG-58/U, RG-58 A/U и RG-58 C/U - разновидности "тонкого" коаксиального кабеля для сетей Ethernet 10Base-2. Кабель RG-58/U имеет сплошной внутренний проводник, а кабель
- RG-58 A/U - многожильный. Кабель RG-58 C/U проходит "военную приемку". Все эти разновидности кабеля имеют волновое сопротивление 50 Ом, но обладают худшими механическими и электрическими характеристиками по сравнению с "толстым" коаксиальным кабелем. Тонкий внутренний проводник 0,89 мм не так прочен, зато обладает гораздо большей гибкостью, удобной при монтаже. Затухание в этом типе кабеля выше, чем в "толстом" коаксиальном кабеле, что приводит к необходимости уменьшать длину кабеля для получения одинакового затухания в сегменте. Для соединения кабелей с оборудованием используется разъем типа BNC.
- RG-59 - телевизионный кабель с волновым сопротивлением 75 Ом. Широко применяется в кабельном телевидении.
- RG-62 - кабель с волновым сопротивлением 93 Ома, использовался в сетях ArcNet, оборудование которых сегодня практически не выпускается.

Волоконно-оптические кабели

Волоконно-оптические кабели состоят из центрального проводника света (сердцевины) - стеклянного волокна, окруженного другим слоем стекла - оболочкой, обладающей меньшим показателем преломления, чем сердцевина. Распространяясь по сердцевине, лучи света не выходят за ее пределы, отражаясь от покрывающего слоя оболочки. В зависимости от распределения показателя преломления и от величины диаметра сердечника различают:

- многомодовое волокно со ступенчатым изменением показателя преломления (рис. 5.6, а);
- многомодовое волокно с плавным изменением показателя преломления (рис. 5.6, б);
- одномодовое волокно (рис. 5.6, в).

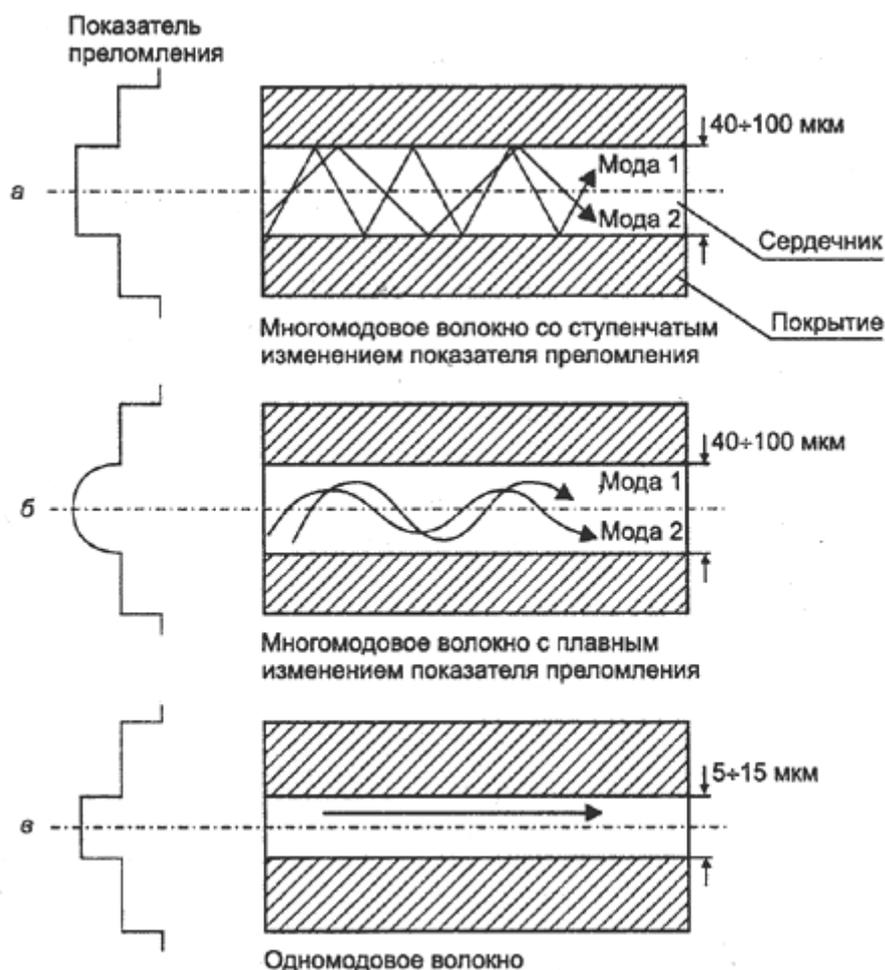


Рис. 5.6. Типы оптического кабеля

Понятие "мода" описывает режим распространения световых лучей во внутреннем сердечнике кабеля. В *одномодовом кабеле (Single Mode Fiber, SMF)* используется центральный проводник очень малого диаметра, соизмеримого с длиной волны света - от 5 до 10 мкм. При этом практически все лучи света распространяются вдоль оптической оси световода, не отражаясь от внешнего проводника. Полоса пропускания одномодового кабеля очень широкая - до сотен гигагерц на километр. Изготовление тонких качественных волокон для одномодового кабеля представляет сложный технологический процесс, что делает одномодовый кабель достаточно дорогим. Кроме того, в волокно такого маленького диаметра достаточно сложно направить пучок света, не потеряв при этом значительную часть его энергии.

В многомодовых кабелях (*Multi Mode Fiber, MMF*) используются более широкие внутренние сердечники, которые легче изготовить технологически. В стандартах определены два наиболее употребительных многомодовых кабеля: 62,5/125 мкм и 50/125 мкм, где 62,5 мкм или 50 мкм - это диаметр центрального проводника, а 125 мкм - диаметр внешнего проводника.

В многомодовых кабелях во внутреннем проводнике одновременно существует несколько световых лучей, отражающихся от внешнего проводника под разными углами. Угол отражения луча называется модой луча. В многомодовых кабелях с плавным изменением коэффициента преломления режим распространения каждой моды имеет более сложный характер.

Многомодовые кабели имеют более узкую полосу пропускания - от 500 до 800 МГц/км. Сужение полосы происходит из-за потерь световой энергии при отражениях, а также из-за интерференции лучей разных мод. В качестве источников излучения света в волоконно-оптических кабелях применяются:

- светодиоды;
- полупроводниковые лазеры.

Литература:

1. В.В.Воеводин, Вл.В.Воеводин - "Параллельные вычисления", издательство "БХВ ", 2002.
2. В. Корнеев В. , "Современные микропроцессоры", издательство "Нолидж", 2003.
3. Э. Таненбаум, "Архитектура компьютера", СПб, издательство «Питер», 2002.
4. В.Г.Олифер, Н.А.Олифер Компьютерные сети, издательство "Питер", 2000.
5. К. Хамахер, З. Вранешич, С. Заки, Организация ЭВМ. СПб, издательство «Питер», 2003 г.

Контрольные задания для СРС

1. *Технология DSL.*
2. *Цифровые каналы T1 и E1.*
3. *Передача голоса по каналам Интернет.*
4. *Методы преобразования и передачи изображения.*
5. *Методы сжатия информации.*

Лекция 9

Тема: Методы передачи дискретных данных на физическом уровне

План лекции

1. Аналоговая модуляция
2. Цифровое кодирование
3. Логическое кодирование
4. Дискретная модуляция аналоговых сигналов

При передаче дискретных данных по каналам связи применяются два основных типа физического кодирования - на основе синусоидального несущего сигнала и на основе последовательности прямоугольных импульсов. Первый способ часто называется также *модуляцией* или *аналоговой модуляцией*, подчеркивая тот факт, что кодирование осуществляется за счет изменения параметров аналогового сигнала. Второй способ обычно называют *цифровым кодированием*.

1.1 Аналоговая модуляция

Аналоговая модуляция применяется для передачи дискретных данных по каналам с узкой полосой частот, типичным представителем которых является *канал тональной частоты*, предоставляемый в распоряжение пользователям общественных телефонных сетей. Типичная амплитудно-частотная характеристика канала тональной частоты представлена на рис. 6.1. Этот канал передает частоты в диапазоне от 300 до 3400 Гц, таким образом, его полоса пропускания равна 3100 Гц. Хотя человеческий голос имеет гораздо более широкий спектр - примерно от 100 Гц до 10 кГц, - для приемлемого качества передачи речи диапазон в 3100 Гц является хорошим решением. Строгое ограничение полосы пропускания тонального канала связано с использованием аппаратуры уплотнения и коммутации каналов в телефонных сетях.

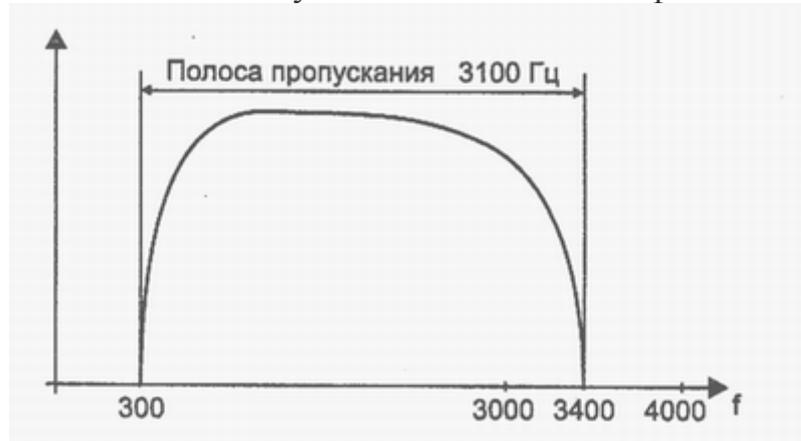


Рис. 6.1. Амплитудно-частотная характеристика канала тональной частоты. Устройство, которое выполняет функции модуляции несущей синусоиды на передающей стороне и демодуляции на приемной стороне, носит название *модем* (модулятор-демодулятор).

Методы аналоговой модуляции

Аналоговая модуляция является таким способом физического кодирования, при котором информация кодируется изменением амплитуды, частоты или фазы синусоидального сигнала несущей частоты. Основные способы аналоговой модуляции показаны на рис. 6.2. На диаграмме (рис. 6.2, а) показана последовательность бит исходной информации, представленная потенциалами высокого уровня для логической единицы и потенциалом нулевого уровня для логического нуля. Такой способ кодирования называется потенциальным кодом, который часто используется при передаче данных между блоками компьютера.

При *амплитудной модуляции* (рис. 6.2, б) для логической единицы выбирается один уровень амплитуды синусоиды несущей частоты, а для логического нуля - другой. Этот способ редко используется в чистом виде на практике из-за низкой помехоустойчивости, но часто применяется в сочетании с другим видом модуляции - фазовой модуляцией.

При *частотной модуляции* (рис. 6.2, в) значения 0 и 1 исходных данных передаются синусоидами с различной частотой - f_0 и f_1 . Этот способ модуляции не требует сложных схем в модемах и обычно применяется в низкоскоростных модемах, работающих на скоростях 300 или 1200 бит/с.

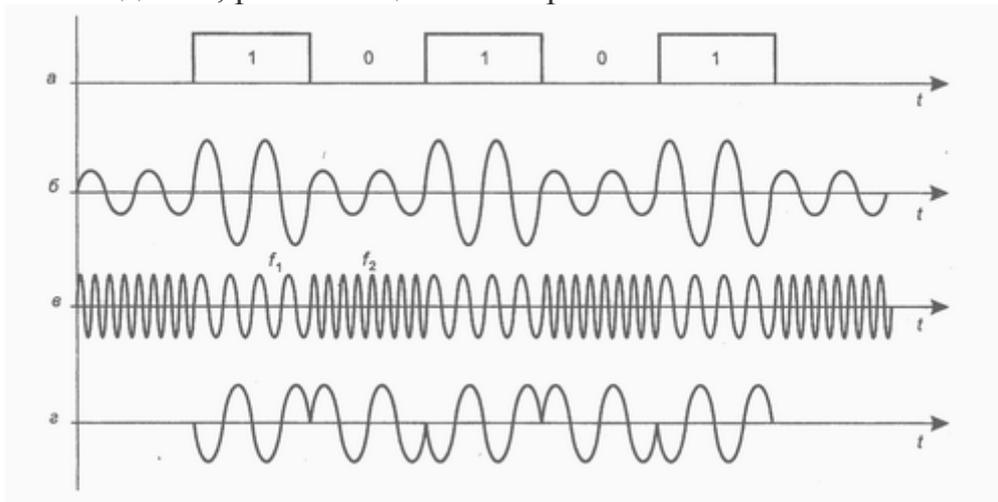


Рис. 6.2. Различные типы модуляции

При *фазовой модуляции* (рис. 6.2, г) значениям данных 0 и 1 соответствуют сигналы одинаковой частоты, но с различной фазой, например 0 и 180 градусов или 0,90,180 и 270 градусов.

В скоростных модемах часто используются комбинированные методы модуляции, как правило, амплитудная в сочетании с фазовой.

Спектр модулированного сигнала

Спектр результирующего модулированного сигнала зависит от типа модуляции и скорости модуляции, то есть желаемой скорости передачи бит исходной информации.

Рассмотрим сначала спектр сигнала при потенциальном кодировании. Пусть логическая единица кодируется положительным потенциалом, а логический ноль - отрицательным потенциалом такой же величины. Для упрощения вычислений предположим, что передается информация, состоящая из

бесконечной последовательности чередующихся единиц и нулей, как это и показано на рис. 6.2, а. Заметим, что в данном случае величины бод и бит в секунду совпадают.

1.2 Цифровое кодирование

При цифровом кодировании дискретной информации применяют потенциальные и импульсные коды.

В потенциальных кодах для представления логических единиц и нулей используется только значение потенциала сигнала, а его перепады, формирующие законченные импульсы, во внимание не принимаются. Импульсные коды позволяют представить двоичные данные либо импульсами определенной полярности, либо частью импульса - перепадом потенциала определенного направления.

Требования к методам цифрового кодирования

При использовании прямоугольных импульсов для передачи дискретной информации необходимо выбрать такой способ кодирования, который одновременно достигал бы нескольких целей:

- имел при одной и той же битовой скорости наименьшую ширину спектра результирующего сигнала;
- обеспечивал синхронизацию между передатчиком и приемником;
- обладал способностью распознавать ошибки;
- обладал низкой стоимостью реализации.

1.3 Логическое кодирование

Логическое кодирование используется для улучшения потенциальных кодов типа AMI, NRZI или 2Q1B. Логическое кодирование должно заменять длинные последовательности бит, приводящие к постоянному потенциалу, вкраплениями единиц. Как уже отмечалось выше, для логического кодирования характерны два метода - избыточные коды и скремблирование.

1.4. Дискретная модуляция аналоговых сигналов

Одной из основных тенденций развития сетевых технологий является передача в одной сети как дискретных, так и аналоговых по своей природе данных. Источниками дискретных данных являются компьютеры и другие вычислительные устройства, а источниками аналоговых данных являются такие устройства, как телефоны, видеокамеры, звуко- и видеовоспроизводящая аппаратура. На ранних этапах решения этой проблемы в территориальных сетях все типы данных передавались в аналоговой форме, при этом дискретные по своему характеру компьютерные данные преобразовывались в аналоговую форму с помощью модемов.

Однако по мере развития техники съема и передачи аналоговых данных выяснилось, что передача их в аналоговой форме не позволяет улучшить качество принятых на другом конце линии данных, если они существенно исказились при передаче. Сам аналоговый сигнал не дает никаких указаний ни о

том, что произошло искажение, ни о том, как его исправить, поскольку форма сигнала может быть любой, в том числе и такой, которую зафиксировал приемник. Улучшение же качества линий, особенно территориальных, требует огромных усилий и капиталовложений. Поэтому на смену аналоговой технике записи и передачи звука и изображения пришла цифровая техника. Эта техника использует так называемую дискретную модуляцию исходных непрерывных во времени аналоговых процессов.

1.5 Асинхронная и синхронная передачи

При обмене данными на физическом уровне единицей информации является бит, поэтому средства физического уровня всегда поддерживают побитовую синхронизацию между приемником и передатчиком.

Канальный уровень оперирует кадрами данных и обеспечивает синхронизацию между приемником и передатчиком на уровне кадров. В обязанности приемника входит распознавание начала первого байта кадра, распознавание границ-полей кадра и распознавание признака окончания кадра.

Обычно достаточно обеспечить синхронизацию на указанных двух уровнях - битовом и кадровом, - чтобы передатчик и приемник смогли обеспечить устойчивый обмен информацией. Однако при плохом качестве линии связи (обычно это относится к телефонным коммутируемым каналам) для удешевления аппаратуры и повышения надежности передачи данных вводят дополнительные средства синхронизации на уровне байт.

Такой режим работы называется *асинхронным* или *старт-стопным*. Другой причиной использования такого режима работы является наличие устройств, которые генерируют байты данных в случайные моменты времени. Так работает клавиатура дисплея или другого терминального устройства, с которого человек вводит данные для обработки их компьютером.

Литература:

1. В.В.Воеводин, Вл.В.Воеводин - "Параллельные вычисления", издательство "БХВ", 2002.
2. В. Корнеев В., "Современные микропроцессоры", издательство "Нолидж", 2003.
3. Э. Таненбаум, "Архитектура компьютера", СПб, издательство «Питер», 2002.
4. В.Г.Олифер, Н.А.Олифер Компьютерные сети, издательство «Питер», 2000.
5. К. Хамахер, З. Вранешич, С. Заки, Организация ЭВМ. СПб, издательство «Питер», 2003 г.

Контрольные задания для СРС

1. Стандарт MPEG-4.
2. Стандарт MPEG-7
3. Обнаружение ошибок.
4. Коррекция ошибок.
5. Статистическая теория каналов связи.

Лекция 10

Тема: Методы передачи данных канального уровня. Методы коммутации

План лекции

1. Асинхронные протоколы
2. Синхронные символично-ориентированные и бит-ориентированные протоколы
3. Передача с установлением соединения и без установления соединения
4. Обнаружение и коррекция ошибок

1.1. Методы передачи данных канального уровня

Канальный уровень обеспечивает передачу пакетов данных, поступающих от протоколов верхних уровней, узлу назначения, адрес которого также указывает протокол верхнего уровня. Протоколы канального уровня оформляют переданные им пакеты в кадры собственного формата, помещая указанный адрес назначения в одного из полей такого кадра, а также сопровождая кадр контрольной суммой. Протокол канального уровня имеет локальный смысл, он предназначен для доставки кадров данных, как правило, в пределах сетей с простой топологией связей и однотипной или близкой технологией, например в односегментных сетях Ethernet или же в многосегментных сетях Ethernet и Token Ring иерархической топологии, разделенных только мостами и коммутаторами.

Другой областью действия протоколов канального уровня являются связи типа "точка-точка" глобальных сетей, когда протокол канального уровня ответственен за доставку кадра непосредственному соседу. Адрес в этом случае не имеет принципиального значения, а на первый план выходит способность протокола восстанавливать искаженные и утерянные кадры, так как плохое качество территориальных каналов, особенно коммутируемых телефонных, часто требует выполнения подобных действий.

Если же перечисленные выше условия не соблюдаются, например связи между сегментами Ethernet имеют петлевидную структуру, либо объединяемые сети используют различные способы адресации, как это имеет место в сетях Ethernet и X.25, то протокол канального уровня не может в одиночку справиться с задачей передачи кадра между узлами и требует помощи протокола сетевого уровня.

Наиболее существенными характеристиками метода передачи, а значит, и протокола, работающего на канальном уровне, являются следующие:

- асинхронный/синхронный;
- символично-ориентированный/бит-ориентированный;
- с предварительным установлением соединения/дейтаграммный;
- с обнаружением искаженных данных/без обнаружения;
- с обнаружением потерянных данных/без обнаружения;

- с восстановлением искаженных и потерянных данных/без восстановления;
- с поддержкой динамической компрессии данных/без поддержки.

Многие из этих свойств характерны не только для протоколов канального уровня, но и для протоколов более высоких уровней.

Асинхронные протоколы

Асинхронные протоколы представляют собой наиболее старый способ связи. Эти протоколы оперируют не с кадрами, а с отдельными символами, которые представлены байтами со старт-стоповыми символами. Асинхронные протоколы ведут свое происхождение от тех времен, когда два человека связывались с помощью телетайпов по каналу "точка-точка". С развитием техники асинхронные протоколы стали применяться для связи телетайпов, разного рода клавиатур и дисплеев с вычислительными машинами. Единицей передаваемых данных был не кадр данных, а отдельный символ. Некоторые символы имели управляющий характер, например символ <CR> предписывал телетайпу или дисплею выполнить возврат каретки на начало строки. В этих протоколах существуют управляющие последовательности, обычно начинающиеся с символа <ESC>. Эти последовательности вызывали на управляемом устройстве достаточно сложные действия - например, загрузку нового шрифта на принтер.

В асинхронных протоколах применяются стандартные наборы символов, чаще всего ASCII или EBCDIC. Так как первые 32 или 27 кодов в этих наборах являются специальными кодами, которые не отображаются на дисплее или принтере, то они использовались асинхронными протоколами для управления режимом обмена данными. В самих пользовательских данных, которые представляли собой буквы, цифры, а также такие знаки, как @, %, \$ и т. п., специальные символы никогда не встречались, так что проблемы их отделения от пользовательских данных не существовало.

Постепенно асинхронные протоколы усложнились и стали наряду с отдельными символами использовать целые блоки данных, то есть кадры. Например, популярный протокол XMODEM передает файлы между двумя компьютерами по асинхронному модему. Начало приема очередного блока файла инициируется символьной командой - принимающая сторона постоянно передает символ ASCII NAK. Передающая сторона, приняв NAK, отправляет очередной блок файла, состоящий из 128 байт данных, заголовка и концевики. Заголовок состоит из специального символа SOH (Start Of Header) и номера блока. Концевик содержит контрольную сумму блока данных. Приемная сторона, получив новый блок, проверяла его номер и контрольную сумму. В случае совпадения этих параметров с ожидаемыми приемник отправлял символ ACK, а в противном случае - символ NAK, после чего передатчик должен был повторить передачу данного блока. В конце передачи файла передавался символ EOH.

Как видно из описания протокола XMODEM, часть управляющих операций выполнялась в асинхронных протоколах посылкой в асинхронном режиме отдельных символов, в то же время часть данных пересылалась блоками, что более характерно для синхронных протоколов.

Синхронные символьно-ориентированные и бит-ориентированные протоколы

В синхронных протоколах между пересылаемыми символами (байтами) нет стартовых и стоповых сигналов, поэтому отдельные символы в этих протоколах пересылать нельзя. Все обмены данными осуществляются кадрами, которые имеют в общем случае заголовок, поле данных и концевик (рис. 7.1). Все биты кадра передаются непрерывным синхронным потоком, что значительно ускоряет передачу данных.



Рис. 7.1. Кадры синхронных протоколов

Так как байты в этих протоколах не отделяются друг от друга служебными сигналами, то одной из первых задач приемника является распознавание границы байт. Затем приемник должен найти начало и конец кадра, а также определить границы каждого поля кадра - адреса назначения, адреса источника, других служебных полей заголовка, поля данных и контрольной суммы, если она имеется.

Передача с установлением соединения и без установления соединения

При передаче кадров данных на канальном уровне используются как дейтаграммные процедуры, работающие без установления соединения (*connectionless*), так и процедуры с предварительным установлением логического соединения (*connection-oriented*).

При дейтаграммной передаче кадр посылается в сеть "без предупреждения", и никакой ответственности за его утерю протокол не несет (рис. 7.3, а). Предполагается, что сеть всегда готова принять кадр от конечного узла. Дейтаграммный метод работает быстро, так как никаких предварительных действий перед отправкой данных не выполняется. Однако при таком методе трудно организовать в рамках протокола отслеживание факта доставки кадра узлу назначения. Этот метод не гарантирует доставку пакета.

Передача с установлением соединения более надежна, но требует больше времени для передачи данных и вычислительных затрат от конечных узлов.

В этом случае узлу-получателю отправляется служебный кадр специального формата с предложением установить соединение (рис. 7.3, б). Если узел-получатель согласен с этим, то он посылает в ответ другой служебный кадр, подтверждающий установление соединения и предлагающий для данного логического соединения некоторые параметры, например идентификатор соединения, максимальное значение поля данных кадров, которые будут использоваться в рамках данного соединения, и т. п. Узел-инициатор соединения может завершить процесс установления соединения отправкой третье-

го служебного кадра, в котором сообщит, что предложенные параметры ему подходят. На этом логическое соединение считается установленным, и в его рамках можно передавать информационные кадры с пользовательскими данными. После передачи некоторого законченного набора данных, например определенного файла, узел инициирует разрыв данного логического соединения, посылая соответствующий служебный кадр.



Рис. 7.3. Протоколы без установления соединения (а) и с установлением соединения (б)

Заметим, что, в отличие от протоколов дейтаграммного типа, которые поддерживают только один тип кадра - информационный, протоколы, работающие по процедуре с установлением соединения, должны поддерживать несколько типов кадров - служебные, для установления (и разрыва) соединения, и информационные, переносящие собственно пользовательские данные.

Логическое соединение обеспечивает передачу данных как в одном направлении - от инициатора соединения, так и в обоих направлениях.

Обнаружение и коррекция ошибок

Канальный уровень должен обнаруживать ошибки передачи данных, связанные с искажением бит в принятом кадре данных или с потерей кадра, и по возможности их корректировать.

Большая часть протоколов канального уровня выполняет только первую задачу - обнаружение ошибок, считая, что корректировать ошибки, то есть повторно передавать данные, содержавшие искаженную информацию, должны протоколы верхних уровней. Так работают такие популярные протоколы локальных сетей, как Ethernet, Token Ring, FDDI и другие. Однако существуют протоколы канального уровня, например LLC2 или LAP-B, которые самостоятельно решают задачу восстановления искаженных или потерянных кадров.

Очевидно, что протоколы должны работать наиболее эффективно в типичных условиях работы сети. Поэтому для сетей, в которых искажения и потери кадров являются очень редкими событиями, разрабатываются протоколы типа Ethernet, в которых не предусматриваются процедуры устранения ошибок. Действительно, наличие процедур восстановления данных потребовало бы от конечных узлов дополнительных вычислительных затрат, которые в условиях надежной работы сети являлись бы избыточными.

Напротив, если в сети искажения и потери случаются часто, то желательно уже на канальном уровне использовать протокол с коррекцией ошибок, а не оставлять эту работу протоколам верхних уровней. Протоколы верхних уровней, например транспортного или прикладного, работая с большими тайм-аутами, восстановят потерянные данные с большой задержкой. В глобальных сетях первых поколений, например сетях X.25, которые работали через ненадежные каналы связи, протоколы канального уровня всегда выполняли процедуры восстановления потерянных и искаженных кадров.

Поэтому нельзя считать, что один протокол лучше другого потому, что он восстанавливает ошибочные кадры, а другой протокол - нет. Каждый протокол должен работать в тех условиях, для которых он разработан.

Компрессия данных

Компрессия (сжатие) данных применяется для сокращения времени их передачи. Так как на компрессию данных передающая сторона тратит дополнительное время, с которому нужно еще прибавить аналогичные затраты времени на декомпрессию этих данных принимающей стороной, то выгоды от сокращения времени на передачу сжатых данных обычно бывают заметны только для низкоскоростных каналов. Этот порог скорости для современной аппаратуры составляет около 64 Кбит/с. Многие программные и аппаратные средства сети способны выполнять *динамическую компрессию* данных в отличие от статической, когда данные предварительно компрессируются (например, с помощью популярных архиваторов типа WinZip), а уже затем отсылаются в сеть.

На практике может использоваться ряд алгоритмов компрессии, каждый из которых применим к определенному типу данных. Некоторые модемы (называемые интеллектуальными) предлагают *адаптивную компрессию*, при которой в зависимости от передаваемых данных выбирается определенный алгоритм компрессии. Рассмотрим некоторые из общих алгоритмов компрессии данных.

1.2. Методы коммутации

Любые сети связи поддерживают некоторый способ коммутации своих абонентов между собой. Этими абонентами могут быть удаленные компьютеры, локальные сети, факс-аппараты или просто собеседники, общающиеся с помощью телефонных аппаратов. Практически невозможно предоставить каждой паре взаимодействующих абонентов свою собственную некоммутируемую физическую линию связи, которой они могли бы монопольно "владеть" в течение длительного времени. Поэтому в любой сети всегда применяется

какой-либо способ коммутации абонентов, который обеспечивает доступность имеющихся физических каналов одновременно для нескольких сеансов связи между абонентами сети. На рис. 7.5 показана типичная структура сети с коммутацией абонентов.

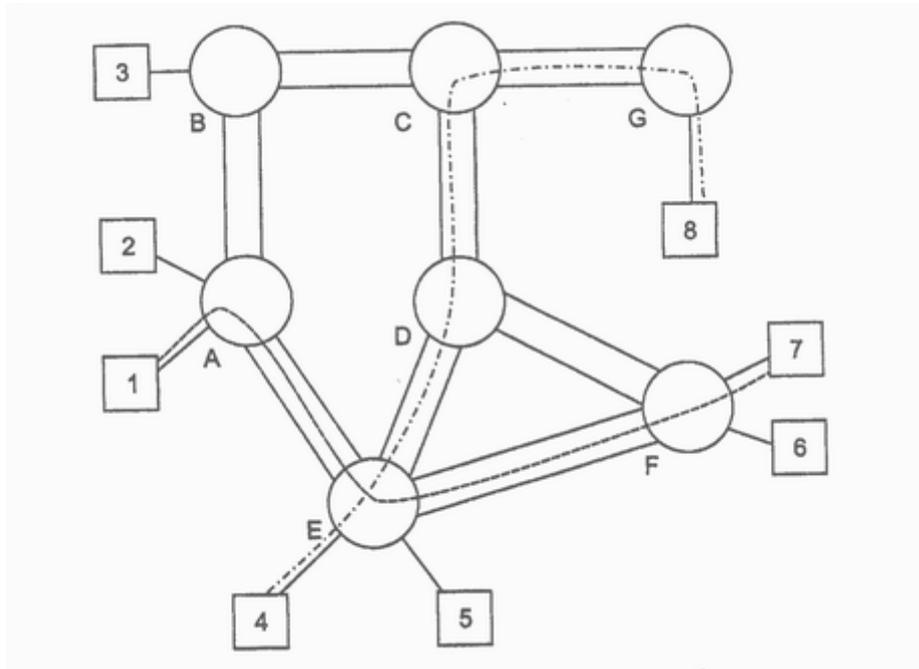


Рис. 7.5. Общая структура сети с коммутацией абонентов

Абоненты соединяются с коммутаторами индивидуальными линиями связи, каждая из которых используется в любой момент времени только одним, закрепленным за этой линией абонентом. Между коммутаторами линии связи разделяются несколькими абонентами, то есть используются совместно.

Существуют три принципиально различные схемы коммутации абонентов в сетях: *коммутация каналов (circuit switching)*, *коммутация пакетов (packet switching)* и *коммутация сообщений (message switching)*. Внешне все эти схемы соответствуют приведенной на рис. 7.5 структуре сети, однако возможности и свойства их различны. Сети с коммутацией каналов имеют более богатую историю, они ведут свое происхождение от первых телефонных сетей. Сети с коммутацией пакетов сравнительно молоды, они появились в конце 60-х годов как результат экспериментов с первыми глобальными компьютерными сетями. Сети с коммутацией сообщений послужили прототипом современных сетей с коммутацией пакетов и сегодня они в чистом виде практически не существуют.

Коммутация каналов

Коммутация каналов подразумевает образование непрерывного составного физического канала из последовательно соединенных отдельных канальных участков для прямой передачи данных между узлами. Отдельные каналы соединяются между собой специальной аппаратурой - коммутаторами, которые могут устанавливать связи между любыми конечными узлами сети. В сети с коммутацией каналов перед передачей данных всегда необхо-

димо выполнить процедуру установления соединения, в процессе которой и создается составной канал.

Коммутация пакетов

Принципы коммутации пакетов

Коммутация пакетов - это техника коммутации абонентов, которая была специально разработана для эффективной передачи компьютерного трафика. Эксперименты по созданию первых компьютерных сетей на основе техники коммутации каналов показали, что этот вид коммутации не позволяет достичь высокой общей пропускной способности сети. Суть проблемы заключается в пульсирующем характере трафика, который генерируют типичные сетевые приложения. Например, при обращении к удаленному файловому серверу пользователь сначала просматривает содержимое каталога этого сервера, что порождает передачу небольшого объема данных. Затем он открывает требуемый файл в текстовом редакторе, и эта операция может создать достаточно интенсивный обмен данными, особенно если файл содержит объемные графические включения. После отображения нескольких страниц файла пользователь некоторое время работает с ними локально, что вообще не требует передачи данных по сети, а затем возвращает модифицированные копии страниц на сервер - и это снова порождает интенсивную передачу данных по сети.

Литература:

1. В.В.Воеводин, Вл.В.Воеводин - "Параллельные вычисления", издательство "БХВ ", 2002.
2. В. Корнеев В. , "Современные микропроцессоры", издательство "Нолидж", 2003.
3. Э. Таненбаум, "Архитектура компьютера", СПб, издательство «Питер», 2002.
4. В.Г.Олифер, Н.А.Олифер Компьютерные сети, издательство "Питер", 2000.
5. К. Хамахер, З. Вранешич, С. Заки, Организация ЭВМ. СПб, издательство «Питер», 2003 г.

Контрольные задания для СРС

1. *Протокол SLIP .*
2. *Протокол PPP.*
3. *Протокол G.703.*
4. *Видеоконференции по каналам Интернет.*
5. *Технология ATM .*

4 Методические указания для выполнения практических (семинарских) занятий

Практические занятия не предусмотрены

5 Методические указания для выполнения лабораторных работ

Методические указания к лабораторной работе №1

Тема: Изучение особенностей функционирования локальной вычислительной сети со случайным методом доступа к моноканалу

Цель работы - В данной работе будут исследованы особенности функционирования ЛВС типа ETHERNET и и проведена оценка ее эффективности.

Содержание работы

*Локальные вычислительные сети.
Множественный доступ с контролем несущей
и обнаружением коллизий CSMA/CD*

Ethernet - это самый распространенный на сегодняшний день стандарт локальных сетей. Общее количество сетей, работающих по протоколу Ethernet в настоящее время, оценивается в 5 миллионов, а количество компьютеров с установленными сетевыми адаптерами Ethernet - в 50 миллионов.

Когда говорят Ethernet, то под этим обычно понимают любой из вариантов этой технологии. В более узком смысле Ethernet - это сетевой стандарт, основанный на экспериментальной сети Ethernet Network, которую фирма Xerox разработала и реализовала в 1975 году.

Институт инженеров по электротехнике и электронике (*IEEE — Institute of Electrical and Electronic Engineers*) определил и опубликовал в качестве стандарта наборы физических характеристик, которым должны удовлетворять ЛВС с обнаружением коллизий. Эти документы называются стандартами IEEE 802.3 (Ethernet). Однако необходимо иметь в виду, что реальный Ethernet и IEEE 802.3 немного отличаются в определении пакета. Некоторые ЛВС не отвечают ни стандарту IEEE 802.3. Например, система ARCNet, выпускаемая фирмами Datapoint Corporation, Standard Microsystems и Thomas-Conrad, а также StarLan фирмы AT&T, VistaLan фирмы Allen-Bradley, LANtastic фирмы Artisoft, Omninet фирмы Corvus, PC Net фирмы IBM, ProNet фирмы Proteon.

В ЛВС, работающей по схеме обнаружения коллизий (полное ее название *CSMA/CD — Множественный доступ с проверкой несущей и обнаружением коллизий*), адаптеры непрерывно находятся в состоянии прослушивания сети. При необходимости передачи данных узел должен дожидаться освобож-

дения ЛВС, и только после этого он может приступить к передаче. Однако в этом случае передача сообщений может начаться одновременно двумя или более узлами ЛВС. Это и называется *коллизией*. Узлы должны будут повторить свои сообщения, причем повтор сообщений в таких случаях производится адаптером самостоятельно без вмешательства прикладной программы. Время, затрачиваемое на обнаружение и обработку таких событий, обычно не превышает микросекунды.

В ЛВС Ethernet узлы передают сообщения со скоростью 10 Мбит/с. Все узлы принимают каждое сообщение, но только тот из них, кому адресовано данное сообщение, посылает в ЛВС подтверждение о его приеме. На рис. 1 показан пример ЛВС Ethernet.



Рис. 1. Сеть Ethernet

Канал связи в ЛВС может быть однополосным и широкополосным. Различия этих вариантов находятся на аппаратном уровне.

Для широкополосной ЛВС характерно применение аналоговой техники. Здесь используется модем для ввода сигналов несущей в передающую среду. Сигналы несущей затем модифицируются (модулируются) цифровым сигналом. Вследствие аналогового характера сети в широкополосных системах часто используется мультиплексирование с частотным разделением, что обеспечивает передачу нескольких несущих или работу нескольких подканалов в одной и той же передающей среде. Аналоговые сигналы несущей работают в высокочастотном радиодиапазоне. Если ЛВС не работает с высокими частотами, она не считается широкополосной.

В однополосной сети используется цифровая техника. Усилитель-формирователь линии вводит в канал перепады напряжения. Канал далее работает как транспортный механизм, с помощью которого происходит распространение цифровых импульсов напряжения. Множественный доступ к среде может быть обеспечен мультиплексированием с временным квантованием или с помощью специальных протоколов. Мультиплексор представляет собой устройство, основная функция которого – обеспечить совместное использование одной линии связи, несколькими устройствами (терминалами) или портами.

Основными поставщиками оборудования для сетей Ethernet являются фирмы DEC и 3Com, но его предлагают также AST Research, Data General, Excelan, Gateway Communications, Micom-Interlan, Proteon, RAD Data Communications, Thomas-Conrad, Ungermann-Bass, Western Digital, Zenith.

Впервые экспериментальная ЛВС Ethernet, послужившая основой для стандарта, была создана в Исследовательском Центре Пало-Альто (*PARC — Palo Alto Research Center*) фирмы Херох в 1975 году. Обычно Ethernet работает с производительностью 10 Мбит/с с 50-Омным коаксиальным кабелем. Современная версия Ethernet 2.0 была принята в 1982 году.

Первая версия стандарта IEEE 802.3, аналогичного Ethernet, была опубликована в 1985 году. Различия между двумя стандартами находятся в области архитектуры ЛВС и форматов пакетов. В 1995 году был принят стандарт Fast Ethernet, который во многом не является самостоятельным стандартом, о чем говорит и тот факт, что его описание просто является дополнительным разделом к основному стандарту 802.3- разделом 802.3u. Аналогично, принятый в 1998 году стандарт Gigabit Ethernet описан в разделе 802.3z основного документа.

Для передачи двоичной информации по кабелю для всех вариантов физического уровня технологии Ethernet, обеспечивающих пропускную способность 10 Мбит/с, используется манчестерский код.

Все виды стандартов Ethernet (в том числе Fast Ethernet и Gigabit Ethernet) используют один и тот же метод разделения среды передачи данных - метод *CSMA/CD*.

CSMA/CD и стандарт IEEE 802.3

Как уже было сказано, наиболее известным механизмом управления локальной сетью шинной конфигурации является метод множественного доступа с контролем несущей и обнаружением коллизий (*CSMA/CD*). Метод *CSMA/CD* является неприоритетным методом управления с контролем несущей (и коллизиями). При разработке системы Ethernet были частично использованы принципы работы системы ALOHA (применяется в спутниковых коммуникациях). Спецификация *CSMA/CD* Ethernet базируется на концепции уровневых протоколов. На рис. 2 показаны уровни, которые имеются в *CSMA/CD*.

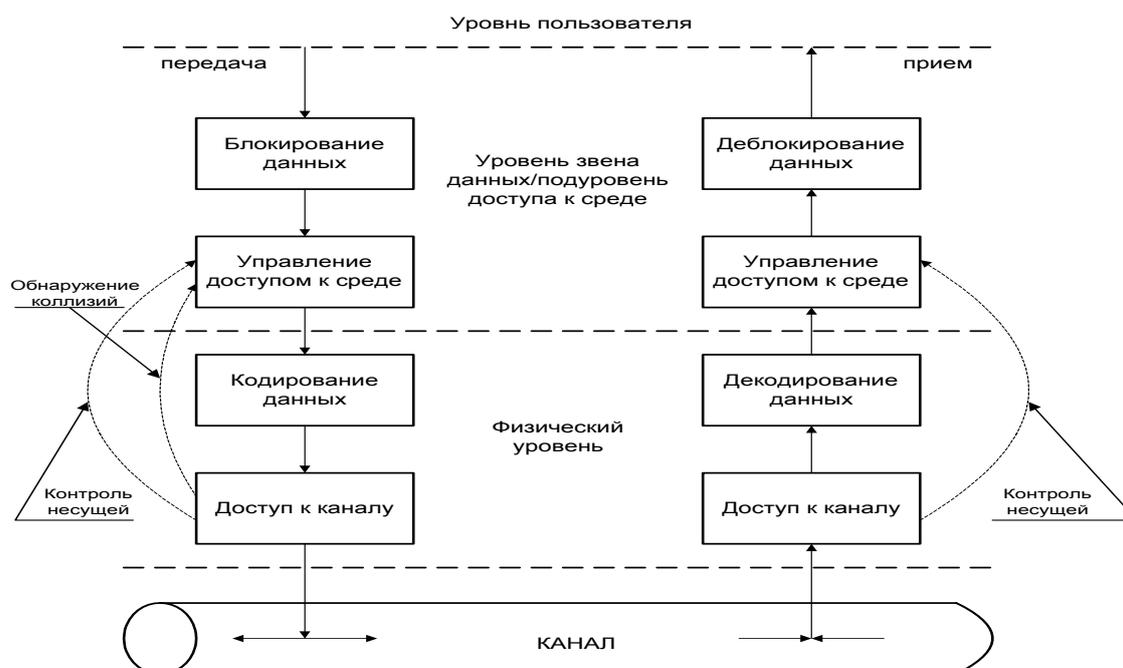


Рис. 2. Уровни ISO в ЛВС м методом доступа CSMA/CD

Уровень пользователя обслуживается двумя уровнями CSMA/CD, а именно уровнем звена данных (канальным уровнем) и физическим уровнем. Каждый из двух самых нижних уровней состоит из двух различных объектов. Уровень звена данных фактически содержит алгоритмы, управляющие работой сети CSMA/CD. Они не зависят от среды, следовательно, сеть может быть широкополосной или однополосной; это безразлично с точки зрения управления звеном данных. Стандарт IEEE 802 включает как широкополосный, так и однополосный варианты.

Уровень звена данных состоит из логического объекта, осуществляющего блокирование/деблокирование данных и логического объекта административного управления доступом к среде передачи/приема. (В спецификации Ethernet логический объект управления доступом к среде называется логическим объектом управления звеном). Основные функции этих объектов таковы:

Блокирование/деблокирование данных:

- формирует кадр CSMA/CD (кадр MAC); включает адрес отправителя и адрес назначения (получателя); вычисляет поле обнаружения ошибок в передающем узле и использует это поле для вычисления в принимающем узле признака (индикатора) обнаружения ошибок.

Управление доступом к среде:

- передает кадр в физический уровень и принимает кадр из физического уровня;
- помещает кадр в буфер;
- обеспечивает устранение коллизий (передающая сторона);

- обеспечивает обработку коллизий (передающая сторона).

Физический уровень зависит от типа передающей среды. Он отвечает за такие услуги, как ввод электрических сигналов в канал, обеспечение синхронизации работы канала, кодирование и декодирование данных. Подобно уровню звена данных, физический уровень состоит из двух основных логических объектов: логического объекта кодирования/декодирования и доступа к каналу приема/передачи (хотя в документах стандарта IEEE 802.3 эти два логических объекта объединены). Основными функциями этих объектов являются:

Кодирование/декодирование данных:

- формирует сигналы для синхронизации станций, подключенных к каналу (такой сигнал синхронизации называется преамбулой);
- кодирует поток двоичных данных в самосинхронизирующийся код в передающем узле и декодирует манчестерский код обратно в двоичный код в принимающем узле.

Манчестерский код (рис. 3) обеспечивает изменение состояния сигнала при представлении каждого бита. Следовательно, это хороший синхронизирующий код. Однако он требует удвоенной скорости для передачи заданного количества битов.

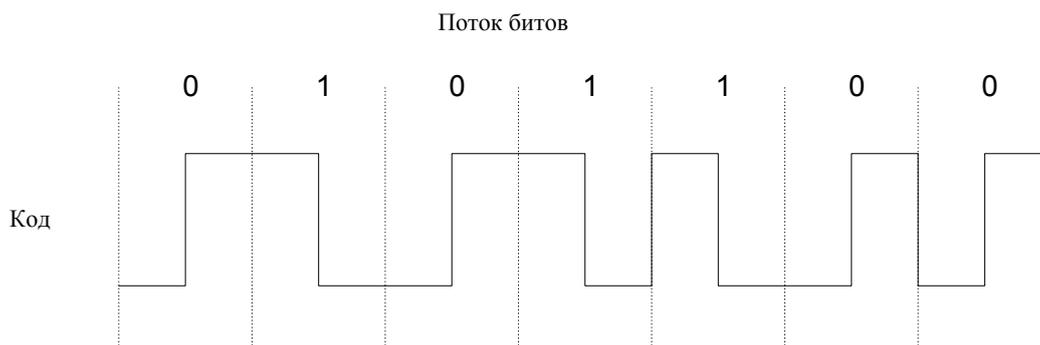


Рис. 3. Манчестерский код

Манчестерский код обычно используется в технике записи на магнитную ленту, оптоволоконных каналах, коаксиальных линиях и локальных сетях.

Доступ к каналу:

- вводит физический сигнал в канал на передающей стороне и получает сигнал на принимающей стороне интерфейса;
- контролирует несущую в канале как на передающей, так и на принимающей стороне (что означает, что канал занят);
- обнаруживает коллизию в канале на передающей стороне (указывая, что произошло наложение двух сигналов).

В сети CSMA/CD каждая станция имеет как передающую, так и принимающую сторону для обеспечения входящего/исходящего потока данных.

Передающая сторона вызывается, когда пользователь желает передать данные другому ООД в сети; принимающая сторона вызывается, когда данные передаются в кабель станциям, подключенным к сети.

Логический объект передающей стороны, выполняющий функции блокирования данных, принимает пользовательские данные и формирует кадр MAC. Он добавляет к данным поле контрольной последовательности кадра и передает кадр управлению доступом к среде, которое помещает и хранит кадр в буфере до тех пор, пока не освободится канал. Канал считается свободным, когда произведен сброс сигнала контроля несущей, индицируемого логическому объекту доступа к каналу (на передающей стороне) в физическом уровне. После небольшой задержки управление доступом к среде передает кадр физическому уровню.

На физическом уровне передающей стороны при выполнении функции кодирования данных производится передача сигнала синхронизации (преамбулы). Кроме того, производится манчестерское кодирование потока двоичных данных. Сигнал синхронизации передается затем передающей стороне доступа к каналу, которая вводит сигнал в канал.

Кадр CSMA/CD (MAC) передается всем станциям, подключенным к каналу. Сигнал распространяется от узла-источника к другим узлам в обоих направлениях. Принимающая станция контролирует преамбулу, синхронизируется с сигналом и устанавливает сигнал контроля несущей. Далее на принимающей стороне доступа к каналу сигнал поступает на декодирование. Объект декодирования данных переводит манчестерский код обратно в двоичную последовательность данных и передает кадр управлению доступом к среде.

Подобно своему партнеру на передающей стороне, управление доступом к среде хранит кадр в буфере до тех пор, пока не будет произведен сброс сигнала контроля несущей, индицируемого логическому объекту доступа к каналу на принимающей стороне. Сброс сигнала контроля несущей означает, что приняты все биты. Управление доступом к среде может теперь передать данные для их деблокирования. При деблокировании данных производится контроль ошибок, которые могли возникнуть в процессе передачи. Если ошибок не было, производится проверка адресного поля, чтобы определить правильность адресации кадра при передаче данному узлу. Если адрес правильный, логический объект деблокирования данных передает кадр уровню пользователя вместе с адресом назначения (DA), адресом отправителя (SA) и, конечно, блоком данных LLC.

Различия между сетями Ethernet и IEEE 802.3 незначительны. Ethernet обеспечивает связь, соответствующую уровням 1 и 2 эталонной модели OSI, в то время как IEEE 802.3 обеспечивает физический уровень 1 и только часть канального уровня 2, определяя доступ к каналу, но не логический протокол его контроля. Оба протокола (Ethernet и IEEE 802.3) реализованы аппаратно. Выполняются они типично - либо в виде карты, либо на самой системной плате.

Краткие характеристики Ethernet версии 2 и IEEE 802.3 показаны в табл. 1.

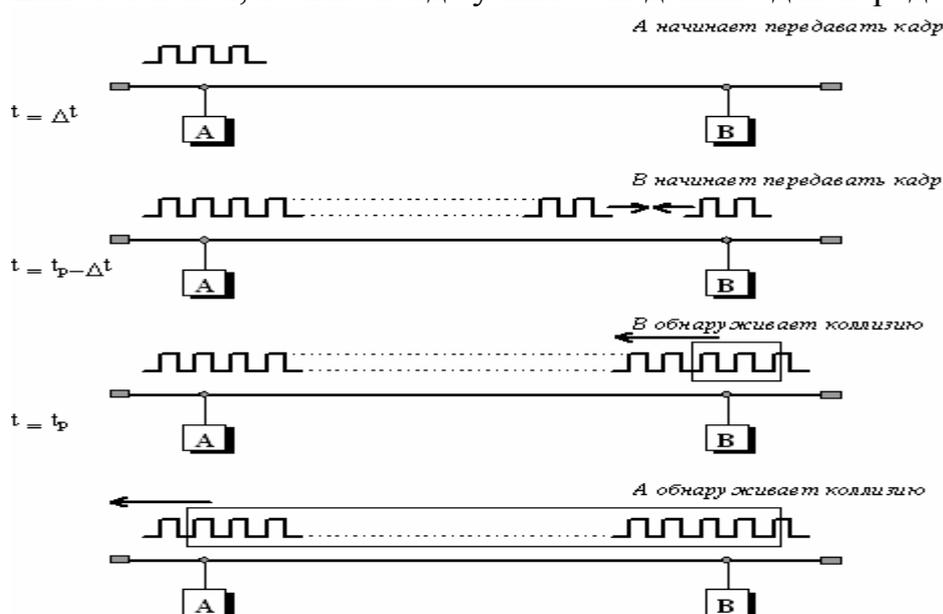
Таблица 1. Краткие характеристики Ethernet

Характеристика	Ethernet	IEEE 802.3	IEEE 802.3	IEEE 802.3	IEEE 802.3
Протокол	Ethernet	10Base5	10Base2	1Base5	10BaseT
Скорость передачи (Мбит/с)	10	10	10	1	10
Метод передачи сигнала	Цифровой	Цифровой	Цифровой	Цифровой	Цифровой
Максимальная длина сегмента (м)	500	500	185	250	100
Кабель	50-Ом толстый коакс. каб.	50-Ом толстый коакс. каб.	50-Ом толстый коакс. каб.	Витая пара	Витая пара
Топология	Шина	Шина	Шина	Звезда	Звезда

Больше всего протокол Ethernet похож на 10Base5 стандарта IEEE 802.3. Оба эти протокола используют конфигурацию типа "шина" с соединительным кабелем между конечными станциями и непосредственно сетевым оборудованием. В Ethernet такой кабель называется *кабелем приемопередатчика*. Он связан с приемопередатчиком, который соединен с физическим сетевым оборудованием (сегментом). Конфигурация IEEE 802.3 в общем такая же, но соединительный кабель называется *интерфейсом блока доступа* (AUI - attachment unit interface), а передатчик - *блоком доступа к среде* (MAU - medium attachment unit). В обоих случаях соединительный кабель подключается к интерфейсной плате (или устройству) внутри конечной станции.

Коллизии

Так как сеть CSMA/CD является равноранговой сетью, станции запрашивают канал, только когда у них есть данные для передачи.



t_p – время распространения сигнала, Δt – время захвата канала.

Рис. 4. Схема образования коллизий

Соперничество за канал может возникнуть тогда, когда сигналы вводятся в кабель от разных станций примерно одновременно. Когда это происходит, возникает наложение и искажение сигналов. Их правильный прием станциями невозможен. Центральным аспектом коллизий является окно коллизий. Этим термином описывается интервал времени, необходимый для распространения сигнала по каналу и обнаружения его любой станцией сети. Например, предположим, что в сети имеется кабель длиной 0,6 миль. Если станции расположены в самом дальнем конце кабеля, расстояние до самой отдаленной станции составляет около 0,6 мили. Передача сигнала на это расстояние потребует 4,2 мкс. Когда станция A готова передавать данные, она «прослушивает» кабель, чтобы определить, имеется ли сигнал в цепи. Если станция B ранее передала кадр в канал, но он еще не достиг станции A, то станция A ошибочно будет считать, что канал свободен, и начнет передачу своего пакета. В данной ситуации произойдет коллизия двух сигналов (рис. 4).

В наихудших для однополосной сети условиях время, необходимое для обнаружения столкновения (и захвата канала), в два раза больше задержки распространения, так как сигнал, образовавшийся в результате коллизии, должен распространиться обратно к передающим станциям. Задержка распространения и обнаружение коллизий в случае широкополосной сети составляют еще большее время, поскольку в такой сети используются два кабеля для передачи и приема сигналов. В наихудшем случае время обнаружения коллизии в четыре раза больше задержки распространения.

Коллизия является нежелательным явлением, так как приводит к ошибкам в работе сети. Более того, при передаче длинных кадров коллизия поглощает больше канального времени, чем при использовании коротких кадров. CSMA/CD учитывает эту проблему на уровне управления доступа к среде путем прекращения передачи кадра сразу же после обнаружения коллизии.

Другую точку зрения на коллизии представляет длительность слота (кванта)—время, необходимое для распространения кадра по всему каналу вместе с задержкой получения канала. Канал Ethernet 10 Мбит/с (однополосный) имеет задержку распространения, составляющую 450 битовых длительностей ($45 \text{ мкс} \times 10\,000\,000 = 450$). Ethernet требует, чтобы длительность слота была больше суммы времени распространения (450 бит) и максимального времени подавления (передачи сигналов подавления или затора) (48 бит).

Если сигнал распространился во все части канала без коллизий, говорят, что станция, которая передала этот сигнал, получила или захватила канал. Если это произошло, коллизии устранены, поскольку все станции обнаружили сигнал и уступили ему. Однако, если произошла коллизия, компонент доступа к каналу на передающей стороне замечает наложение сигналов в канале (в виде аномального изменения напряжения) и устанавливает для управления доступом к среде (передающей стороны) специальный сигнал обнаружения коллизии.

Для обработки коллизии управление доступом к среде выполняет две функции. Во-первых, усиливается эффект коллизии путем передачи специальной последовательности битов, называемой затором (jam). Цель затора состоит в том, чтобы сделать коллизию настолько продолжительной, чтобы ее смогли заметить все другие передающие станции, которые вовлечены в коллизию. В локальной сети CSMA/CD требуется, чтобы затор состоял, по меньшей мере, из 32 бит, но не более 48 бит. Это гарантирует, что продолжительность коллизии будет достаточно большой, чтобы ее обнаружили все передающие станции в сети. Ограничение сверху длины последовательности необходимо для того, чтобы станции ошибочно не приняли ее за действительный кадр. Любой кадр, содержащий менее 64 байт (октетов), считается фрагментом испорченного коллизией сообщения и игнорируется любой принимающей станцией сети.

Компонент управления доступом к среде (передающая сторона) выполняет после этого еще одну функцию: после посылки затора он прекращает передачу и планирует передачу на более позднее время, определяемое на основе случайного выбора интервала ожидания. Прерывание передачи кадра уменьшает отрицательный эффект коллизий при передаче длинных кадров.

В принимающей станции или станциях биты, которые образовались в результате коллизии, декодируются физическим уровнем. Фрагменты кадров, вовлеченных в коллизию, опознаются уровнем управления доступом к среде (на принимающей стороне) как недействительные кадры. Он замечает, что коллизионный фрагмент меньше, чем самый короткий действительный кадр, и игнорирует подобный фрагмент. Следовательно, затор используется с той

целью, чтобы гарантировать, что все *передающие* станции заметят коллизию, а передача фрагментарного кадра гарантирует, что любая *принимающая* станция проигнорирует эту передачу.

Как в спецификации Ethernet, так и в стандарте IEEE 802.3 обычно используется «1-настойчивый» метод для управления коллизиями и соперничеством за канал. Однако этот «1-настойчивый» алгоритм применим к целому кратному длительности слота (512 бит), а планирование повторной передачи производится управляемым вероятностным процессом, называемым *усеченным двоичным экспоненциальным возвратом* (truncated binary exponential back-off).

По окончании последовательности затора уровень доступа к среде ожидает, прежде чем сделать попытку повторной передачи максимум 16 раз. Каждый раз среднее значение случайного числа r удваивается:

«Длительность ожидания является целым кратным длительности слота... Число этих слотовых интервалов, в течение которых имеет место ожидание перед n -й попыткой повторной передачи, выбирается как равномерно распределенная случайная величина r в диапазоне $0 < r < s^k$ где $k = \min(n, 10)$ ». \min – функция, определяющая наименьшую из двух величин: n или 10.

Однако существуют методы, позволяющие избежать появления коллизий в сети. Один из подходов состоит в том, чтобы использовать в сети дополнительное устройство, называемое *таймером* или *арбитром*. Это устройство определяет, когда станция может вести передачу без опасности коллизий. Временные параметры определяются каждой станцией; главная станция для управления использованием канала не предусмотрена.

Каждый порт имеет предварительно установленный временной порог. После того как этот временной порог пройден, порт на основании некоторого временного параметра определяет, когда можно вести передачу. (Это напоминает концепцию «захвата» маркера.) Значения времени могут устанавливаться на приоритетной основе, причем у порта с наивысшим приоритетом переполнение таймера наступает раньше всего. Если этот порт не намерен вести передачу, канал будет находиться в состоянии покоя. Станция со следующим по величине приоритетом обнаруживает, что канал свободен. Ее таймер показывает, что лимит времени, когда может вестись передача, не исчерпан, поэтому она может захватить канал.

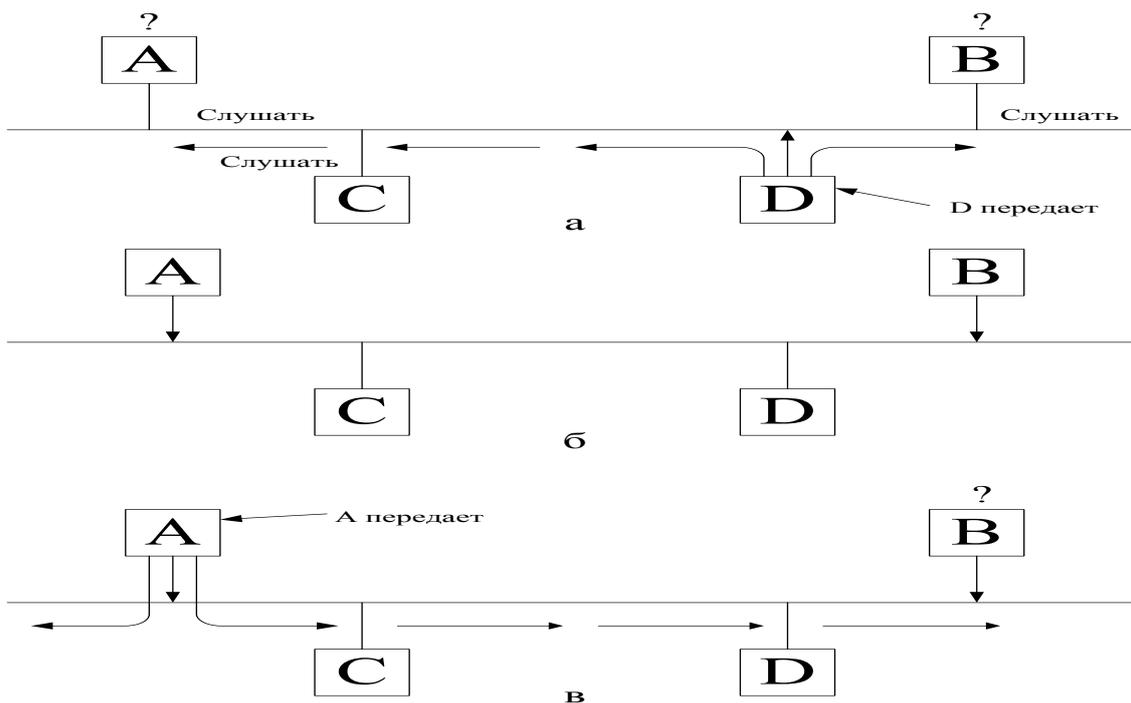
Станции с высоким приоритетом в случае, если они не ведут передачу, переводят канал в состояние покоя, что позволяет станциям с более низким приоритетом использовать его. В традиционных слотовых сетях время покоя представляет собой не что иное, как упущенные возможности для передачи данных. Однако сеть без коллизий использует арбитра, чтобы дать возможность станции со следующим по величине приоритетом в канале захватить время покоя, если у нее есть данные, которые необходимо передать. Этот подход значительно уменьшает время покоя канала.

Метод CSMA/CD наиболее эффективен в условиях относительно низкой общей загрузки канала (менее 30%). Однако в случаях, когда локальная сеть более равномерно загружена, больше подходят другие типы сетей. Например,

в условиях большей загрузки канала более высокую производительность, как правило, можно обеспечить, используя такой тип локальной сети, как маркерное кольцо.

Методы захвата канала

В сети с контролем несущей все станции считаются равноправными, поэтому все они имеют равное право на использование канала. (Однако можно ввести систему приоритетов на основе различных времен выдержки для различных устройств.) Прежде чем начать передачу, требуется, чтобы станции «прослушали» канал и определили, является ли канал активным (т. е. ведет ли какая-либо другая станция передачу данных по каналу). Если канал находится в состоянии покоя, любая станция, имеющая данные для передачи, может послать свой кадр в канал. Если канал занят, станции должны ждать завершения передачи сигнала.



а – станции А и В ожидают, контролируя несущую;
 б – канал свободен, станции А и В пытаются захватить его;
 в – станция А передает, а станция В ожидает конца передачи.

Рис. 5. Система доступа с контролем несущей

Иллюстрацией сети с контролем несущей (с коллизией) является рис. 5. Станции А, В, С и D подключены к шине или каналу (с горизонтальной топологией) с помощью шинных интерфейсных устройств ВІU. Предположим, что станции А и В должны передать данные; однако в это время канал использует станция D, поэтому устройства ВІU станций А и В «слушают» и ждут окончания передачи кадра из станции D. Как только линия переходит в состояние покоя (рис. 5б), станции А и В пытаются захватить канал.

В сетях с контролем несущей предусмотрено несколько методов захвата канала (табл. 2). Одним из методов является метод «ненастойчивого» контроля несущей, обеспечивающего всем станциям возможность начинать передачу немедленно после того, как обнаруживается, что канал свободен (без арбитража перед передачей). В случае если канал занят, станции выжидают случайный период времени перед тем, как снова проверить состояние канала.

Другим методом, который используется в системах с квантованием времени, является метод «р-настойчивого» контроля несущей; он предусматривает для каждой станции некоторый алгоритм ожидания (*p* означает вероятность). Например, станции А и В не начинают немедленно передачу после того, как контроль обнаружил, что канал перешел в состояние покоя; в этом случае каждая станция вызывает программу генерации случайного числа—времени ожидания (обычно несколько микросекунд). Если станция обнаруживает, что канал занят, она выжидает некоторый период времени (слот) и делает новую попытку. Она произведет передачу в освободившийся канал с вероятностью *p* или вероятностью $1-p$ отложит передачу до следующего слота.

Однако имеется еще один метод – «1-настойчивого» контроля несущей, предусматривающий, что станция начинает передачу сразу же после того, как обнаруживает, что канал находится в состоянии покоя. Когда возникает коллизия, перед тем как снова произвести контроль канала, станции выжидают в течение случайного периода времени. Этот метод называется «1-настойчивым» потому, что станция производит передачу с вероятностью 1, когда обнаруживает, что канал свободен.

Таблица 2. Методы доступа к каналу

Условие	Ненастойчивый	р-настойчивый	1-настойчивый
Канал свободен	Передать немедленно	Передать с вероятностью <i>p</i> , отложить с вероятностью $1-p$	Передать немедленно
Канал занят	Случайное время ожидания и контроль	Передать с вероятностью <i>p</i> , отложить с вероятностью $1-p$	Непрерывно контролировать несущую

Коллизия	Повторно передать после случайного времени ожидания	Повторно передать после случайного времени ожидания	Повторно передать после случайного времени ожидания

Метод «р-настойчивого» контроля разработан с двойкой целью: во-первых, уменьшить время пребывания канала в состоянии покоя, что обеспечивается методом «1-настойчивого» контроля несущей, и, во-вторых, уменьшить вероятность коллизий, на что направлен метод «ненастойчивого контроля». Однако величина p должна быть выбрана достаточно небольшой, чтобы обеспечить приемлемые эксплуатационные характеристики. Это может показаться удивительным, но многие поставщики и рабочие группы по стандартизации оказывают предпочтение методу «1-настойчивого» контроля.

Использование пакетов в Ethernet и IEEE 802.3

На рис. 6 показано расположение и определение полей пакета в оригинальной системе Ethernet.

Описание полей:

- *Преамбула.* Это поле длиной 8 байт используется для синхронизации пакета. Преамбула всегда содержит код 10101010 в первых семи байтах и код 10101011 в последнем байте.

- *Назначение.* Это поле длиной 6 байт содержит адрес узла ЛВС, которому предназначено сообщение. Старший (самый левый) бит в первом байте имеет специальное назначение. Если он равен 0, то адрес назначения является физическим адресом и уникален в ЛВС. В соответствии со схемой присвоения имен, принятой фирмой Хегох, первые три байта задают адрес группы, а следующие три байта задают локальный адрес в группе. Если же этот бит равен 1, то пакет является широковещательным, и тогда остальные байты в этом поле могут адресовать пакет какой-нибудь конкретной группе рабочей станции или всем рабочим станциям в ЛВС (если все биты равны 1).

Преамбула	Назначение	Источник	Тип	Данные	CRC-сумма
7	6	6	2	46 - 1500	4

Длина каждого поля, в байтах

Рис. 6. Пакет Ethernet.

- *Источник.* Это поле также имеет длину 6 байт и идентифицирует узел, отправивший пакет. Старший бит первого байта в этом поле всегда равен 0.

- *Тип.* Это поле содержит два байта и идентифицирует тип протокола более высокого уровня, используемого для его передачи или приема. Оно было введено фирмой Xerox для внутреннего употребления и никак не интерпретируется в Ethernet. Это поле позволяет множеству протоколов высокого уровня (называемых *уровнями клиента*) разделять ЛВС без того, чтобы вникать в содержимое пакетов друг друга.

- *Данные.* Это поле может иметь длину от 46 до 1500 байт и содержит данные, составляющие сообщение.

- *CRC-сумма.* Последнее поле длиной 4 байта содержит *остаток избыточной циклической суммы (cyclic redundancy checksum)*, вычисленный с помощью полиномов типа CRC-32. Узел, получивший сообщение, должен также произвести вычисления типа CRC-32 и сравнить полученный результат с содержимым этого поля для выявления ошибок передачи.

Если не учитывать преамбулу, то можно видеть, что сообщение Ethernet имеет длину от 64 до 1518 байт, и минимальная длина данных сообщения равна 46 байтам.

На рис. 7 приведен формат пакета по стандарту IEEE 802.3, который содержит следующие поля:

- *Преамбула.* Это поле содержит 7 одинаковых байтов 10101010, предназначенных для синхронизации.

- *Признак начала пакета.* Это поле содержит один байт 10101011. Как видим, признак начала пакета вместе с преамбулой в точности соответствуют полю преамбулы в Ethernet.

- *Назначение.* Содержит 2 или 6 байт в зависимости от типа установленной ЛВС IEEE 802.3. Он указывает для какой рабочей станции предназначен данный пакет. Заметим, что в конкретной ЛВС длина адресного поля должна быть или 2 или 6 байтов. Самый популярный тип ЛВС из IEEE 802.3, называемый 10BASE5, имеет длину адресного поля 6 байт. Первый бит адреса назначения, называемый I/G, определяет, кому предназначен пакет — группе или отдельному узлу. Значение этого бита равно 1, если пакет предназначен группе (широковещательное сообщение), и равно 0, если он предназначен отдельному узлу. В случае двухбайтового адресного поля остальные 15 бит определяют адрес узла, которому предназначено сообщение. Если же длина адресного поля равна 6 байтам, то тогда второй бит, следующий за битом I/G, называется битом признака универсального или локального адреса U/L. Значение этого бита равно 0, если адрес является глобальным, и 1, если локальным. Остальные 46 бит определяют адрес узла.

- *Источник.* Длина этого поля равна 2 или 6 байтам и содержит адрес отправителя. Бит I/G (первый) всегда равен 0.

- *Длина.* Это двухбайтное поле содержит информацию о длине данных в пакете.

- *Данные.* Это поле может иметь длину от 0 до 1500 байт. Если длина этого поля меньше 46 байт, то следующее поле (набивка) используется для доведения длины пакета до нужного уровня.
- *Набивка.* В это поле вставляются пустые символы для доведения длины пакета до минимально допустимой величины. При достаточно большой длине поля данных поле набивки может отсутствовать.
- *CRC-сумма.* Поле длиной 4 байта содержит *остаток избыточной циклической* суммы, вычисленный посредством полиномов типа СКС-32, такое же, как в Ethernet.

Преамбула	Признак начала пакета	Назначение	Источник	Длина	Данные	Набивка	CRC-сумма
7	1	2 или 6	2 или 6	2	0 - 1500	?	4

Длина каждого поля, в байтах

Рис. 7. Пакет стандарта IEEE 802.3.

Длина пакета как для Ethernet, так и для IEEE 802.3 (предполагаем тип 10BASE5), находится в диапазоне от 64 до 1518 байт, не считая преамбулы и признака начала пакета. Однако в стандарте IEEE 802.3 имеется возможность передачи прикладной программой данных длиной менее 46 байт, благодаря тому, что на уровне MAC пакет будет автоматически дополнен до нужного размера пустыми символами. В противоположность этому в стандарте Ethernet такие ситуации будут рассматриваться как ошибочные.

Спецификации физической среды Ethernet

Исторически первые сети технологии Ethernet были созданы на коаксиальном кабеле диаметром 0,5 дюйма. В дальнейшем были определены и другие спецификации физического уровня для стандарта Ethernet, позволяющие использовать различные среды передачи данных. Метод доступа CSMA/CD и все временные параметры остаются одними и теми же для любой спецификации физической среды технологии Ethernet 10 Мбит/с.

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

- 10 Base-5 — коаксиальный кабель диаметром 0,5 дюйма, называемый «толстым» коаксиалом. Имеет волновое сопротивление 50 Ом. Максимальная длина сегмента — 500 метров (без повторителей).

- 10 Base-2 — коаксиальный кабель диаметром 0,25 дюйма, называемый «тонким» коаксиалом. Имеет волновое сопротивление 50 Ом. Максимальная длина сегмента — 185 метров (без повторителей).

- в 10 Base-T — кабель на основе неэкранированной витой пары (Unshielded Twisted Pair, UTP). Образуется звездообразную топологию на основе

концентратора. Расстояние между концентратором и конечным узлом — не более 100 м.

- 10 Base-F — волоконно-оптический кабель. Топология аналогична топологии стандарта 10 Base-T. Имеется несколько вариантов этой спецификации — FOIRL (расстояние до 1000 м), 10 Base-FL (расстояние до 2000 м), 10 Base-FB (расстояние до 2000 м).

Число 10 в указанных выше названиях обозначает битовую скорость передачи данных этих стандартов — 10 Мбит/с, а слово Base — метод передачи на одной базовой частоте 10 МГц (в отличие от методов, использующих несколько несущих частот, которые называются Broadband — широкополосными). Последний символ в названии стандарта физического уровня обозначает тип кабеля.

Стандарт 10 Base-T

Стандарт принят в 1991 году, как дополнение к существующему набору стандартов Ethernet, и имеет обозначение 802.31.

Сети 10Base-T используют в качестве среды две *неэкранированные витые пары* (Unshielded Twisted Pair, UTP). Многопарный кабель на основе неэкранированной витой пары категории 3 (категория определяет полосу пропускания кабеля, величину перекрестных наводок NEXT и некоторые другие параметры его качества) телефонные компании уже достаточно давно использовали для подключения телефонных аппаратов внутри зданий. Этот кабель носит также название Voice Grade, говорящее о том, что он предназначен для передачи голоса.

Идея приспособить этот популярный вид кабеля для построения локальных сетей оказалась очень плодотворной, так как многие здания уже были оснащены нужной кабельной системой. Оставалось разработать способ подключения сетевых адаптеров и прочего коммуникационного оборудования к витой паре таким образом, чтобы изменения в сетевых адаптерах и программном обеспечении сетевых операционных систем были бы минимальными по сравнению с сетями Ethernet на коаксиале. Это удалось, поэтому переход на витую пару требует только замены трансивера сетевого адаптера или порта маршрутизатора, а метод доступа и все протоколы канального уровня остались теми же, что и в сетях Ethernet на коаксиале.

Конечные узлы соединяются по топологии «точка-точка» со специальным устройством — многопортовым повторителем с помощью двух витых пар. Одна витая пара требуется для передачи данных от станции к повторителю (выход Tx сетевого адаптера), а другая — для передачи данных от повторителя к станции (вход Rx сетевого адаптера). На рис. 11. показан пример трехпортового повторителя. Повторитель принимает сигналы от одного из конечных узлов и синхронно передает их на все свои остальные порты, кроме того, с которого поступили сигналы.

Многопортовые повторители в данном случае обычно называются концентраторами (англоязычные термины — hub или concentrator). Концентратор осуществляет функции повторителя сигналов на всех отрезках витых пар, подключенных к его портам, так что образуется единая среда передачи данных — логический моноканал (логическая общая шина). Повторитель обнаруживает коллизию в сегменте в случае одновременной передачи сигналов по нескольким своим Rx-входам и посылает jam-последовательность на все свои Tx-выходы. Стандарт определяет битовую скорость передачи данных 10 Мбит/с и максимальное расстояние отрезка витой пары между двумя непосредственно связанными узлами (станциями и концентраторами) не более 100 м при наличии витой пары качества не ниже категории 3. Это расстояние определяется полосой пропускания витой пары. На длине 100 м она позволяет передавать данные со скоростью 10 бит/с при использовании манчестерского кода.

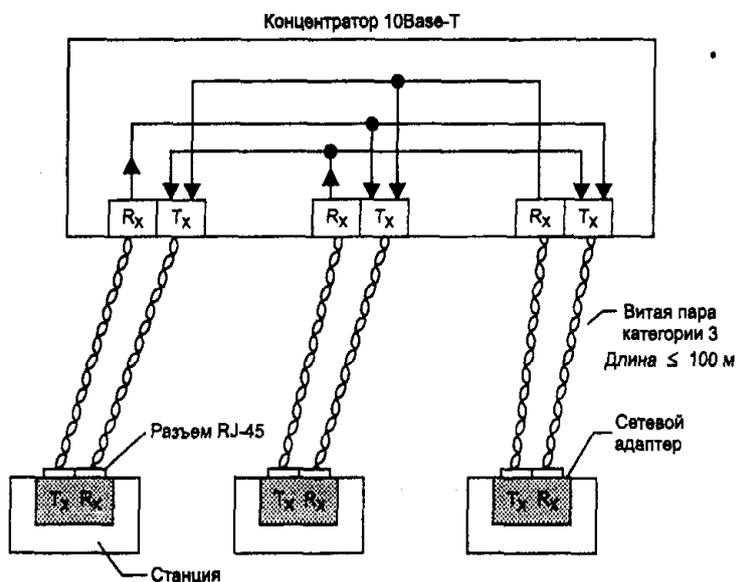


Рис. 11. Сеть стандарта 10Base-T: T, - передатчик; R, - приемник

Концентраторы 10 Base-T можно соединять друг с другом с помощью тех же портов, которые предназначены для подключения конечных узлов. При этом нужно позаботиться о том, чтобы передатчик и приемник одного порта были соединены соответственно с приемником и передатчиком другого порта.

Для обеспечения синхронизации станций при реализации процедур доступа CSMA/CD и надежного распознавания станциями коллизий в стандарте определено максимально число концентраторов между любыми двумя станциями сети, а именно 4. Это правило носит название «правила 4-х хабов» и оно заменяет «правило 5-4-3», применяемое к коаксиальным сетям. При создании сети 10Base-T с большим числом станций концентраторы можно соединять друг с другом иерархическим способом, образуя древовидную структуру (рис. 12.).

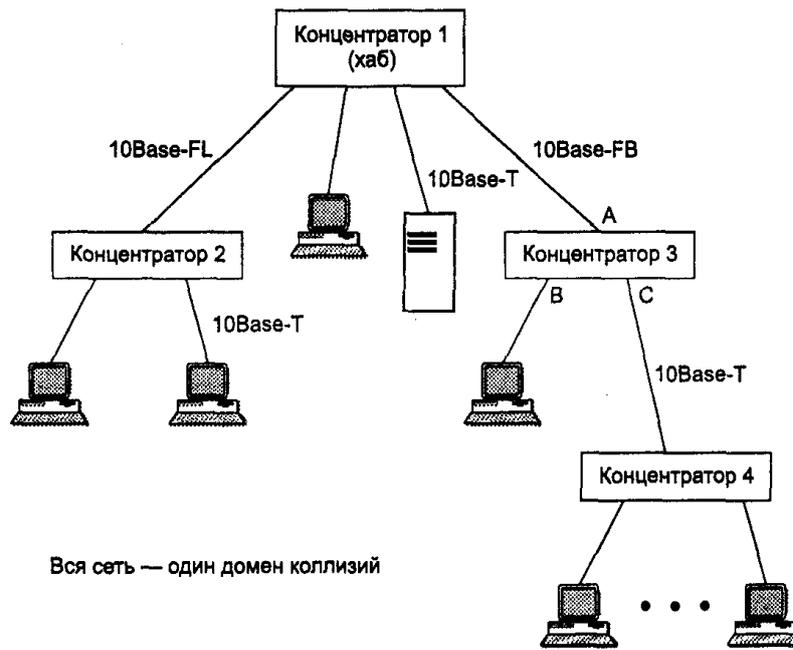


Рис. 12. Иерархическое соединение концентраторов Ethernet

ВНИМАНИЕ ! Петлевидное соединение концентраторов в стандарте 10Base-T запрещено, так как оно приводит к некорректной работе сети. Это требование означает, что в сети 10Base-T не разрешается создавать параллельные каналы связи между критически важными концентраторами для резервирования связей на случай отказа порта, концентратора или кабеля. Резервирование связей возможно только за счет перевода одной из параллельных связей в неактивное (заблокированное) состояние

Общее количество станций в сети 10Base-T не должно превышать общего предела в 1024, и для данного типа физического уровня это количество действительно можно достичь. Для этого достаточно создать двухуровневую иерархию концентраторов, расположив на нижнем уровне достаточное количество концентраторов с общим количеством портов 1024 (рис. 13). Конечные узлы нужно подключить к портам концентраторов нижнего уровня. Правило 4-х хабов при этом выполняется — между любыми конечными узлами будет ровно 3 концентратора.

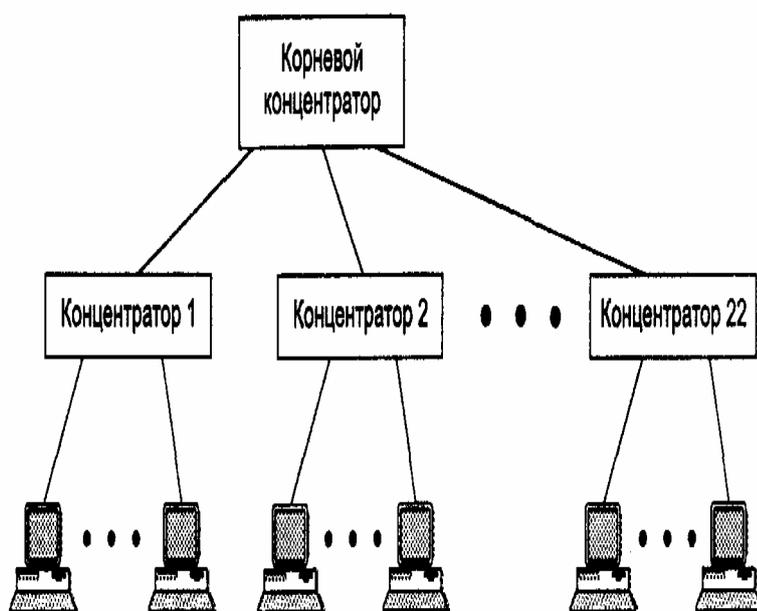


Рис. 13. Схема с максимальным количеством станций

Максимальная длина сети в 2500 м здесь понимается как максимальное расстояние между любыми двумя конечными узлами сети (часто применяется также термин «максимальный диаметр сети»). Очевидно, что если между любыми двумя узлами сети не должно быть больше 4-х повторителей, то максимальный диаметр сети 10Base-T составляет $5 \times 100 = 500$ м.

Сети, построенные на основе стандарта 10Base-T, обладают по сравнению с коаксиальными вариантами Ethernet многими преимуществами. Эти преимущества связаны с разделением общего физического кабеля на отдельные кабельные отрезки, подключенные к центральному коммуникационному устройству. И хотя логически эти отрезки по-прежнему образуют общую разделяемую среду, их физическое разделение позволяет контролировать их состояние и отключать в случае обрыва, короткого замыкания или неисправности сетевого адаптера на индивидуальной основе. Это обстоятельство существенно облегчает эксплуатацию больших сетей Ethernet, так как концентратор обычно автоматически выполняет такие функции, уведомляя при этом администратора сети о возникшей проблеме.

В стандарте 10Base-T определена процедура тестирования физической работоспособности двух отрезков витой пары, соединяющих трансивер конечного узла и порт повторителя. Эта процедура называется *тестом связности (link test)*, и она основана на передаче каждые 16 мс специальных импульсов J и K манчестерского кода между передатчиком и приемником каждой витой пары. Если тест не проходит, то порт блокируется и отключает проблемный узел от сети. Так как коды J и K являются запрещенными при передаче кадров, то тестовые последовательности не влияют на работу алгоритма доступа к среде.

Появление между конечными узлами активного устройства, которое может контролировать работу узлов и изолировать от сети некорректно рабо-

тающие, является главным преимуществом технологии 10Base-T по сравнению со сложными в эксплуатации коаксиальными сетями. Благодаря концентраторам сеть Ethernet приобрела некоторые черты отказоустойчивой системы.

Оптоволоконный Ethernet

В качестве среды передачи данных 10 мегабитный Ethernet использует оптическое волокно. Оптоволоконные стандарты в качестве основного типа кабеля рекомендуют достаточно дешевое многомодовое оптическое волокно, обладающее полосой пропускания 500-800 МГц при длине кабеля 1 км. Допустимо и более дорогое од-номодовое оптическое волокно с полосой пропускания в несколько гигагерц, но при этом нужно применять специальный тип трансивера.

Функционально сеть Ethernet на оптическом кабеле состоит из тех же элементов, что и сеть стандарта 10 Base-T — сетевых адаптеров, многопортового повторителя и отрезков кабеля, соединяющих адаптер с портом повторителя. Как и в случае витой пары, для соединения адаптера с повторителем используются два оптоволоконка — одно соединяет выход Tx адаптера со входом Rx повторителя, а другое — вход Rx адаптера с выходом Tx повторителя.

Стандарт FOIRL (Fiber Optic Inter-Repeater Link) представляет собой первый стандарт комитета 802.3 для использования оптоволоконна в сетях Ethernet. Он гарантирует длину оптоволоконной связи между повторителями до 1 км при общей длине сети не более 2500 м. Максимальное число повторителей между любыми узлами сети — 4. Максимального диаметра в 2500 м здесь достичь можно, хотя максимальные отрезки кабеля между всеми 4 повторителями, а также между повторителями и конечными узлами недопустимы — иначе получится сеть длиной 5000 м.

Стандарт 10Base-FL представляет собой незначительное улучшение стандарта FOIRL. Увеличена мощность передатчиков, поэтому максимальное расстояние между узлом и концентратором увеличилось до 2000 м. Максимальное число повторителей между узлами осталось равным 4, а максимальная длина сети — 2500 м.

Стандарт 10Base-FB предназначен только для соединения повторителей. Конечные узлы не могут использовать этот стандарт для присоединения к портам концентратора. Между узлами сети можно установить до 5 повторителей 10Base-FB при максимальной длине одного сегмента 2000 м и максимальной длине сети 2740 м.

Повторители, соединенные по стандарту 10Base-FB, при отсутствии кадров для передачи постоянно обмениваются специальными последовательностями сигналов, отличающимися от сигналов кадров данных, для поддержания синхронизации. Поэтому они вносят меньшие задержки при передаче данных из одного сегмента в другой, и это является главной причиной, по ко-

торой количество повторителей удалось увеличить до 5. В качестве специальных сигналов используются манчестерские коды J и K в следующей последовательности: J-J-K-K-J-J-... Эта последовательность порождает импульсы частоты 2,5 МГц, которые и поддерживают синхронизацию приемника одного концентратора с передатчиком другого. Поэтому стандарт 10Base-FB имеет также название *синхронный Ethernet*.

Как и в стандарте 10Base-T, оптоволоконные стандарты Ethernet решают соединять концентраторы только в древовидные иерархические структуры. Любые петли между портами концентраторов не допускаются.

Домен коллизий

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

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

Приведенная на рис. 12 сеть представляет собой один домен коллизий. Если, например, столкновение кадров произошло в концентраторе 4, то в соответствии с логикой работы концентраторов 10Base-T сигнал коллизии распространится по всем портам всех концентраторов.

Если же вместо концентратора 3 поставить в сеть мост, то его порт С, связанный с концентратором 4, воспримет сигнал коллизии, но не передаст его на свои остальные порты, так как это не входит в его обязанности. Мост просто отработает ситуацию коллизии средствами порта С, который подключен к общей среде, где эта коллизия возникла. Если коллизия возникла из-за того, что мост пытался передать через порт С кадр в концентратор 4, то, зафиксировав сигнал коллизии, порт С приостановит передачу кадра и попытается передать его повторно через случайный интервал времени. Если порт С принимал в момент возникновения коллизии кадр, то он просто отбросит полученное начало кадра и будет ожидать, когда узел, передававший кадр через концентратор 4, не сделает повторную попытку передачи. После успешного принятия данного кадра в свой буфер мост передаст его на другой порт в соответствии с таблицей продвижения, например на порт А. Все события, связанные с обработкой коллизий портом С, для остальных сегментов сети, которые подключены к другим портам моста, останутся просто неизвестными.

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

Общие характеристики стандартов Ethernet 10 Мбит/с

В табл. 3. и 4. сведены основные ограничения и характеристики стандартов Ethernet.

Таблица 3. Общие ограничения для всех стандартов Ethernet

Номинальная пропускная способность	10 Мбит/с
Максимальное число станций в сети	1024
Максимальное расстояние между узлами в сети	2500 м (в 10Base-FB 2750 м)
Максимальное число коаксиальных сегментов в сети	5

Таблица 4. Параметры спецификаций физического уровня для стандарта Ethernet

	10Base-5	10Base-2	10Base-T	10Base-F
Кабель	Толстый коаксиальный кабель RG-8 или RG-11	Тонкий коаксиальный кабель RG-58	Неэкранированная витая пара категорий 3,4, 5	Многомодовый волоконно-оптический кабель
Максимальная длина сегмента, м	500	185	100	2000
Максимальное расстояние между узлами сети (при использовании повторителей), м				
Максимальное число станций	100	30	1024	1024

ций в сег- менте				
Максималь- ное				
число повто- рителей ме- жду любыми	4	4	4	4 (5 для 10 Base-FB)
станциями сети				

Методика расчета конфигурации сети Ethernet

Соблюдение многочисленных ограничений, установленных для различных стандартов физического уровня сетей Ethernet, гарантирует корректную работу сети (естественно, при исправном состоянии всех элементов физического уровня).

Наиболее часто приходится проверять ограничения, связанные с длиной отдельного сегмента кабеля, а также количеством повторителей и общей длиной сети. Правила «5-4-3» для коаксиальных сетей и «4-х хабов» для сетей на основе витой пары и оптоволокну не только дают гарантии работоспособности сети, но и оставляют большой «запас прочности» сети. Например, если посчитать время двойного оборота в сети, состоящей из 4-х повторителей 10Base-5 и 5-ти сегментов максимальной длины 500 м, то окажется, что оно составляет 537 битовых интервала. А так как время передачи кадра минимальной длины, состоящего вместе с преамбулой 72 байт, равно 575 битовым интервалам, то видно, что разработчики стандарта Ethernet оставили 38 битовых интервала в качестве запаса для надежности. Тем не менее комитет 802.3 говорит, что и 4 дополнительных битовых интервала создают достаточный запас надежности.

Комитет IEEE 802.3 приводит исходные данные о задержках, вносимых повторителями и различными средами передачи данных, для тех специалистов, которые хотят самостоятельно рассчитывать максимальное количество повторителей и максимальную общую длину сети, не довольствуясь теми значениями, которые приведены в правилах «5-4-3» и «4-х хабов». Особенно такие расчеты полезны для сетей, состоящих из смешанных кабельных систем, например коаксиала и оптоволокну, на которые правила о количестве повторителей не рассчитаны. При этом максимальная длина каждого отдельного физического сегмента должна строго соответствовать стандарту, то есть 500 м для «толстого» коаксиала, 100 м для витой пары и т. д.

Чтобы сеть Ethernet, состоящая из сегментов различной физической природы, работала корректно, необходимо выполнение четырех основных условий:

о количество станций в сети не более 1024;

- максимальная длина каждого физического сегмента не более величины, определенной в соответствующем стандарте физического уровня;
- время двойного оборота сигнала (Path Delay Value, PDV) между двумя самыми удаленными друг от друга станциями сети не более 575 битовых интервала;
- сокращение межкадрового интервала IPG (Path Variability Value, PW) при прохождении последовательности кадров через все повторители должно быть не больше, чем 49 битовых интервала. Так как при отправке кадров конечные узлы обеспечивают начальное межкадровое расстояние в 96 битовых интервала, то после прохождения повторителя оно должно быть не меньше, чем $96 - 49 = 47$ битовых интервала.

Соблюдение этих требований обеспечивает корректность работы сети даже в случаях, когда нарушаются простые правила конфигурирования, определяющие максимальное количество повторителей и общую длину сети в 2500 м.

Выводы

- * Ethernet — это самая распространенная на сегодняшний день технология локальных сетей. В широком смысле Ethernet — это целое семейство технологий, включающее различные фирменные и стандартные варианты, из которых наиболее известны фирменный вариант Ethernet DIX, 10-мегабитные варианты стандарта IEEE 802.3, а также новые высокоскоростные технологии Fast Ethernet и Gigabit Ethernet. Почти все виды технологий Ethernet используют один и тот же метод разделения среды передачи данных — метод случайного доступа CSMA/CD, который определяет облик технологии в целом.
- * В узком смысле Ethernet — это 10-мегабитная технология, описанная в стандарте IEEE 802.3.
- * Важным явлением в сетях Ethernet является коллизия — ситуация, когда две станции одновременно пытаются передать кадр данных по общей среде. Наличие коллизий — это неотъемлемое свойство сетей Ethernet, являющееся следствием принятого случайного метода доступа. Возможность четкого распознавания коллизий обусловлена правильным выбором параметров сети, в частности соблюдением соотношения между минимальной длиной кадра и максимально возможным диаметром сети.
- * На характеристики производительности сети большое значение оказывает коэффициент использования сети, который отражает ее загруженность. При значениях этого коэффициента свыше 50 % полезная пропускная способность сети резко падает: из-за роста интенсивности коллизий, а также увеличения времени ожидания доступа к среде.
- * Максимально возможная пропускная способность сегмента Ethernet в кадрах в секунду достигается при передаче кадров минимальной длины и составляет 14 880 кадр/с. При этом полезная пропускная способность сети составляет всего 5,48 Мбит/с, что лишь ненамного превышает половину номинальной пропускной способности — 10 Мбит/с.

- * Максимально возможная полезная пропускная способность сети Ethernet составляет 9,75 Мбит/с, что соответствует использованию кадров максимальной длины в 1518 байт, которые передаются по сети со скоростью 513 кадр/с.
- * При отсутствии коллизий и ожидания доступа коэффициент использования сети зависит от размера поля данных кадра и имеет максимальное значение 0,96.
- * Технология Ethernet поддерживает 4 разных типа кадров, которые имеют общий формат адресов узлов. Существуют формальные признаки, по которым сетевые адаптеры автоматически распознают тип кадра.
- * В зависимости от типа физической среды стандарт IEEE 802.3 определяет различные спецификации: 10Base-5, 10Base-2, 10Base-T, FOIRL, 10Base-FL, 10Base-FB. Для каждой спецификации определяются тип кабеля, максимальные длины непрерывных отрезков кабеля, а также правила использования повторителей для увеличения диаметра сети: правило «5-4-3» для коаксиальных вариантов сетей, и правило «4-х хабов» для витой пары и оптоволокна.
- * Для «смешанной» сети, состоящей из физических сегментов различного типа, полезно проводить расчет общей длины сети и допустимого количества повторителей. Комитет IEEE 802.3 приводит исходные данные для таких расчетов, в которых указываются задержки, вносимые повторителями различных спецификаций физической среды, сетевыми адаптерами и сегментами кабеля.

В данной работе будет исследованы особенности функционирования ЛВС типа ETHERNET и проведена оценка ее эффективности, для чего необходимо определить следующие величины:

$T_{дост}$ - среднее время доставки сообщения между абонентами.

$P_{кон}$ - вероятность возникновения конфликта.

ρ - загрузка моноканала.

$\rho_{эфф}$ - загрузка моноканала непосредственно данными.

Эти значения определяются с помощью программы аналитического моделирования ЛВС со случайным методом доступа. Прежде чем начать работу с этой программой необходимо ответить преподавателю на несколько контрольных вопросов.

Порядок выполнения работы

Изучение реально работающей сети представляет собой трудновыполнимую задачу требующую достаточно сложный комплекс измерительной аппаратуры, поэтому исследования в данной лабораторной работы будут проводиться при помощи специальной моделирующей программы.

Моделирующая программа позволяет получить следующие основные характеристики работы сети со случайным доступом:

Общая интенсивность абонентов - суммарная интенсивность выдачи в сеть пакетов.

Загрузка моноканала данными - загрузка сети данными.

Загрузка служебной информацией - загрузка сети служебными полями кадров.

Загрузка моноканала конфликтами - загрузка моноканала пакетами, попавшими в конфликт.

Среднее время доставки пакета - среднее время, затрачиваемое на передачу пакета от одного абонента другому.

Вероятность конфликта - вероятность возникновения конфликтной ситуации.

При этом исходными являются данные:

Число абонентов - число абонентов подключенных к моноканалу.

Интенсивность абонента - интенсивность абонента.

Время восстановления - время восстановления сети после конфликта.

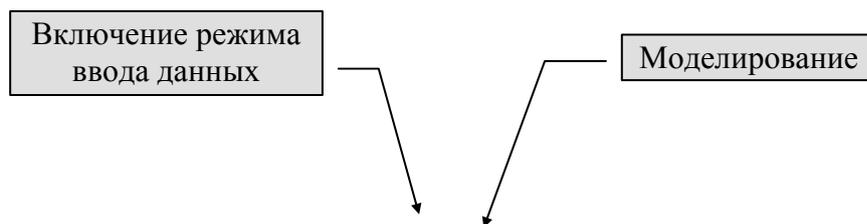
Это время определяется технической реализацией ЛВС.

Пропускная способность сети - скорость передачи данных определяемая техническими возможностями адаптеров и моноканала.

Информационная часть пакета - размер области данных в передаваемых кадрах.

Для моделирования работы ЛВС необходимо установить программу в режим ввода данных, нажав соответствующую кнопку (рис. 15). Наберите нужные исходные данные и нажмите кнопку «Моделирование», после этого программа просчитает необходимые данные.

При появлении сообщения «*Неверные исходные данные*» проверьте корректность введенных вами значений. Как правило, такие значения высвечиваются красным цветом. Сообщение «*Сеть перегружена*» указывает на то, что при введенных вами данных сеть не может нормально функционировать.



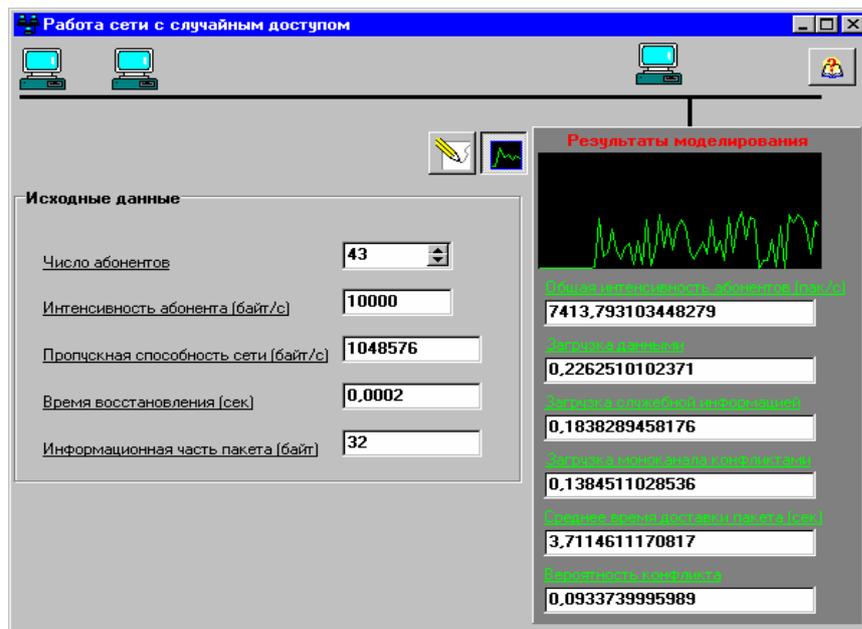


Рис. 15. Среда моделирования.

4. Задание к выполнению работы

Приступить к индивидуальной работе можно только после ответа преподавателю на контрольные вопросы.

В ходе этой работы вам необходимо получить значения характеристик и объяснить некоторые особенности функционирования сети со случайным методом доступа. Эти особенности хорошо иллюстрируются зависимостями эффективности работы сети от исходных данных. Вам необходимо получить по 10-15 значений исследуемого параметра и занести их в таблицу для дальнейшего построения графиков зависимостей. При этом значения величины влияющий на исследуемый параметр необходимо брать таким образом, чтобы он изменялся от своего минимально возможного значения до максимума. После этого необходимо объяснить полученные результаты. Используя значения указанные в вашем варианте необходимо исследовать следующие зависимости:

1. Зависимость эффективной загрузки моноканала $\rho_{эфф}$ от общей загрузки ρ . Общая загрузка определяется как сумма загрузок данными, служебной информацией и конфликтами.
2. Зависимость загрузки моноканала конфликтами от общей загрузки моноканала.
3. Зависимость загрузки моноканала от интенсивности абонентов.
4. Зависимость загрузки моноканала конфликтами от общей загрузки моноканала.
5. Зависимость загрузки моноканала от количества абонентов.
6. Зависимость загрузки моноканала от его пропускной способности.

7. Зависимость времени доставки сообщения от загрузки моноканала.
8. Зависимость времени доставки сообщения от вероятности конфликта.
9. Зависимость времени доставки от размера пакета информации.
10. Зависимость времени доставки от суммарной интенсивности абонентов.
11. Зависимость времени доставки от пропускной способности моноканала.
12. Зависимость вероятности возникновения конфликта от числа абонентов.
13. Зависимость вероятности возникновения конфликта от интенсивности абонентов.
14. Зависимость вероятности возникновения конфликта от времени восстановления.
15. Зависимость вероятности возникновения конфликта от общей загрузки.
16. Зависимость вероятности возникновения конфликта от длины информационной части пакета.

Варианты заданий

№ вар.	Исследуем. зависимости	Число абонент.	Информационная часть пакета [байт]	Пропускн. способн. сети [Мбит/с]	Время восстановления [мкс]
1	1, 7, 14, 4, 5	10	16	10	50
2	11, 6, 3, 2, 7	15	32	10	50
3	2, 8, 5, 14, 1	20	64	10	50
4	5, 15, 6, 13, 2	25	192	10	50
5	12, 14, 1, 5, 3	10	320	10	50
6	6, 8, 4, 14, 5	15	448	10	25
7	8, 3, 14, 5, 13	20	576	10	25
8	16, 8, 15, 14, 1	25	704	10	25
9	3, 16, 9, 1, 10	20	168	10	25
10	1, 13, 11, 13, 16	18	32	10	25
11	9, 4, 11, 10, 16	10	64	15	45
12	11, 1, 4, 7, 5	16	192	15	45
13	2, 13, 8, 15, 9	20	320	15	45
14	10, 16, 11, 6, 1	15	448	15	45
15	12, 16, 13, 3, 4	25	576	15	45
16	3, 15, 13, 10, 14	30	704	15	20
17	12, 5, 2, 11, 7	25	1024	15	20
18	15, 3, 5, 9, 16	15	16	15	20
19	3, 13, 5, 8, 10	32	32	15	20
20	8, 5, 3, 10, 12	35	64	15	20
21	12, 5, 4, 1, 2	20	192	5	100

№ вар.	Исследуем. зависимости	Число абонент.	Информационная часть пакета [байт]	Пропускн. способн. сети [Мбит/с]	Время восстановления [мкс]
22	13, 6, 7, 10, 117	15	320	5	100
23	12, 8, 6, 3, 14	10	448	5	100
24	3, 2, 12, 11, 4	30	576	5	100
25	11, 4, 10, 5, 6	33	704	5	100
26	14, 2, 6, 11, 12	35	1024	5	150
27	15, 6, 8, 5, 13	15	16	5	150
28	12, 9, 1, 7, 9	10	32	5	150
29	10, 15, 11, 2, 8	20	40	5	150
30	7, 2, 1, 9, 11	25	64	5	150

Содержание отчёта

Результаты исследования необходимо оформить в виде таблиц и графиков. Допускается для построения графиков использовать табличный процессор EXCEL. Пример оформления приведен в табл 7.

Таблица 7. Пример оформления результатов исследования.

Номер опыта	1	2	3	4	5									12
Загрузка моноканала														
Время доставки (с)														

Контрольные вопросы

1. Дайте описание метода доступа к моноканалу CSMA/CD.
2. Охарактеризуйте особенности случайных методов доступа.
3. Структура кадра ETHERNET. Назначение его полей.
4. Что такое коллизия. Причины ее возникновения.
5. Опишите применяемый в ETHERNET алгоритм отсрочки доступа к моноканалу в случае возникновения коллизии.
6. Охарактеризуйте особенности известных методов захвата канала.
7. Назовите основные характеристики ЛВС со случайным доступом.
8. Назовите преимущества и недостатки рассмотренного метода доступа.

Литература

1. Климанов А.П. Методы разработки аналитических моделей для анализа локальных вычислительных сетей, используемых в управлении технологическими процессами. М.: МЭИ, 1995. 115с.

2. Нанс Б. Программирование в локальных сетях. Пермь:1992.-756с.
3. Шварц М. Сети связи: протоколы, моделирование и анализ. Ч 2. М.: Наука., 1992.- 272 с.
4. Олифер В.Г., Олифер Н.А. Компьютерные сети. Принципы, технологии, протоколы - СПб: Издательство «Питер», 1999. - 672 с.

Контрольные задания для СРС

1. *Поясните разницу между расширяемостью и масштабируемостью на примере технологии Ethernet.*
2. *Что такое домен коллизий?*
3. *В чем состоят функции преамбулы и начального ограничителя кадра в стандарте Ethernet?*
4. *Какие сетевые средства осуществляют jabber control?*
5. *Чем объясняется, что минимальный размер кадра в стандарте 10Base-5 был выбран равным 64 байт?*

Методические указания к лабораторной работе №2

Тема: Изучение особенностей функционирования локальной вычислительной сети с маркерным методом доступа к моноканалу

Цель работы - В данной работе будут исследованы особенности функционирования ЛВС типа Token Ring и проведена оценка ее эффективности.

Содержание работы

Основные сведения

Как правило, в локальных вычислительных сетях все абоненты связываются друг с другом по одному моноканалу. При передаче информации абонент отправитель сначала разбивает ее на отдельные информационные пакеты (кадры). Посылка и прием таких кадров сетевыми адаптерами является основой работы всех локальных сетей. При отправке кадров в общий моноканал возникает необходимость разделения доступа адаптеров к нему.

В силу своих высоких эксплуатационных характеристик наибольшее распространение в практике разработки ЛВС нашли два следующих метода доступа:

- управляемый селективный метод доступа с «передачей маркера» ;
- свободный метод доступа с проверкой несущей и обнаружением конфликтов (CSMA/CD).

Метод доступа CSMA/CD подробно рассмотрен в лабораторной работе №2.

В данной лабораторной работе будут рассмотрены особенности работы ЛВС с методом доступа «передача маркера» . Иногда этот метод доступа называют *маркерным* .

Маркерный метод доступа имеет несколько модификаций и основные две из них связаны с особенностями его использования в двух топологиях ЛВС:

- физическая «шина» (IEEE-802.4);
- физическое «кольцо» (IEEE-802.5).

Обе модификации маркерного метода доступа включены в международный стандарт.

Особенности функционирования ЛВС с маркерным методом доступа по «физическому кольцу»

Канальные станции, подключенные к моноканалу, связаны кольцевой сетью, по которой между станциями передается сигнал (маркер), разрешающий доступ к моноканалу.

Если канальная станция находится в состоянии готовности и к ней приходит маркер, то она направляет в моноканал пакет информации. Пакет поступает в следующую станцию, которая ретранслирует его к очередной станции и т.д. Каждая канальная станция сравнивает свой адрес с адресом получателя.

Если адреса совпадают, то станция принимает пакет в свою память и одновременно транслирует его дальше. Факт приема пакета канальной станцией отмечается установкой кодовой комбинации приема, выделенной в пакете.

Передающая станция находится в состоянии передачи до возвращения в нее отправленного пакета.

Принимая ранее переданный пакет, станция сравнивает его с хранимым в памяти текстом, анализирует значение битов приема и, установив правильность передачи и факт приема пакета, посылает маркер следующей канальной станции, которая при получении маркера имеет право на передачу своего пакета. Если данная станция не имеет данных для передачи, то она передает маркер следующей станции ЛВС.

Таким образом, маркер последовательно передается между станциями кольца, поочередно предоставляя им право на передачу данных.

Определим некоторые исходные параметры, характеризующие особенности функционирования маркерного метода доступа.

Среднее время передачи маркера (управления) - τ (Тау - представлено в исходных данных).

$$\tau = \tau_{ki} + \tau_{\text{маркера}} .$$

$$\tau_{ki} = L_{ki} / C .$$

L_{ki} - среднее расстояние между двумя соседними абонентами «к» и «i».

C - скорость распространения сигнала в физической среде моноканала.

$$\tau_{\text{маркера}} = b_{\text{мар}} / V .$$

$b_{\text{мар}}$ - длина маркера в битах.

V - скорость передачи информации в среде моноканала [бит/сек].

T_0 - среднее время, затрачиваемое на передачу пакета в кольце (с момента времени начала передачи пакета «i-й» канальной станции до момента времени генерации «i-й» канальной станцией маркера для передачи управления следующей канальной станцией).

$$T_0 = b/V + t_{\text{зад}} ; \text{ где}$$

b - средняя длина пакета (без маркера) в битах;

$t_{\text{зад}}$ - среднее время задержки при передаче пакета в кольце.

$$t_{\text{зад}} = n * L / C$$

n - число канальных станций;

L - среднее расстояние между двумя соседними канальными станциями;

C - скорость распространения сигнала в физической среде кольца.

Выше была представлена идея реализации маркерного метода доступа, однако конкретная реализация имеет свои специфические особенности.

Часть 1. Token Ring. Основные понятия. Алгоритм работы. Управление
Сеть Token Ring первоначально была разработана компанией IBM в 1970 гг. Она по-прежнему является основной технологией IBM для локальных сетей (LAN), уступая по популярности среди технологий LAN только Ethernet/IEEE 802.3. Спецификация IEEE (IEEE – институт инженеров по электротехнике и радиоэлектронике) 802.5 почти идентична и полностью совместима с сетью Token Ring IBM. Спецификация IEEE 802.5 была фактически создана по образцу Token Ring IBM, и она продолжает отслеживать ее разработку. Термин "Token Ring" обычно применяется как при ссылке на сеть Token Ring IBM, так и на сеть IEEE 802.5.

1.1 Как работает сеть Token Ring

Станции (ПК) получают доступ к передающей среде посредством маркера, передаваемого по кольцу. Маркер представляет собой трехбайтовый кадр, который циркулирует по кольцу, оставаясь в пассивном состоянии, пока какой-либо станции не потребуется выполнить передачу данных по кольцу. Станция, намеревающаяся передать данные должна захватить маркер (рис. 1.1).



Рис. 1.1. Работа сети Token Ring

Перед тем как станция передаст маркер дальше, она присоединяет к кадру маркера информационный кадр. После этого маркер становится либо кадром управления доступом, к передающей среде (medium access control — MAC), либо кадром, управления логическим каналом передачи данных (logical link control — LLC). Эта информация включает также адрес (или адреса) получателя пересылаемой информации,

После того как станция-адресат (или несколько таких станций) примет предназначенные ей данные, маркер возвращается к исходной станции (источнику), которая изымает его из кольца.

1.2. Вниз по потоку и вверх против потока

Двумя логическими направлениями перемещения по кольцу являются направления вверх против потока (upstream) и вниз по потоку (downstream). Направление перемещения данных по кольцу — всегда вниз по потоку.

Маркер всегда передается на следующую станцию в направлении вниз по потоку. Станция, которая передает маркер, называется ближайшим активным соседом, расположенным вверх против потока (nearest active upstream neighbor — NAUN) — по отношению к следующей активной станции вниз по потоку, которая принимает маркер (рис. 1.2).



Рис. 1.2 NAUN

Концепция NAUN очень важна в среде Token Ring, поскольку является непосредственной точкой отсчета для управления связью и адресации станций в кольце. Одна из наиболее важных задач NAUN заключается в функции уведомления соседа. Процесс уведомления соседа (Neighbor Notification process) позволяет кольцевой станции (ring station — RS) узнать адрес своего NAUN.

Схемы адресации

В архитектуре Token Ring каждая станция в кольце имеет уникальный адрес. Существует три различных типа адресов: индивидуальный, групповой и функциональный.

- *Индивидуальная адресация.* Каждая станция имеет в кольце уникальный адрес.
- *Групповая адресация.* Адресуется одна или несколько станций. Обращение к станциям, определенным как часть общего группового адреса, может осуществляться посредством ширококвещательной передачи, направленной по их групповому адресу, Широковещательная передача — это метод, позволяющий станции в кольце взаимодействовать с одной или несколькими станциями по общему адресу.
- *Функциональная адресация.* Станциям необходимо взаимодействовать с определенной станцией, которая предоставляет общие функции. Одни функциональные адреса заранее определены архитектурой Token Ring, а другие назначаются пользователем в соответствии с его потребностями. Предопределенные функциональные адреса указаны в таблице.

Сервер отчетов о конфигурациях	C00000000010
Монитор ошибок кольца	C00000000008
Сервер параметров кольца	C00000000002
Активный монитор	C00000000001
Мост	C00000000100
Менеджер LAN	C00000002000

1.3. Варианты топологий Token Ring

Существует 2 основные топологии Token Ring – 4 Мбит/с и 16 Мбит/с.

Основное различие между 4 Мбит/с и 16 Мбит/с Token Ring заключается в полосе пропускания, однако есть и другие различия. Например, 16 Мбит/с Token Ring поддерживает кадры большего размера, чем Мбит/с Token Ring. Кадры 4 Мбит/с Token Ring составляют примерно 4500 байт, а кадры 16 Мбит/с – 18000 байт.

Существенным отличием 16 Мбит/с сети является введение в архитектуру Token Ring понятия раннего освобождения маркера (early token release - ETR). ETR позволяет перемещать по кольцу два маркера одновременно. При скорости 4 Мбит/с в каждый данный момент времени по кольцу перемещается только один маркер (см. рис. 1.3).



Для 16 Мбит/с сетей Token Ring механизм раннего освобождения маркера (ETR) позволяет перемещать по кольцу два маркера одновременно

Рис. 1.3 Топология Token Ring 16 Мбит/с с механизмом раннего освобождения маркера

ETR допускает наличия двух маркеров в кольце одновременно благодаря тому, что при скорости 16 Мбит/с кадры данных затрачивают меньше времени по кольцу и в самом кольце обеспечивается более широкий свободный участок полосы пропускания. С целью заполнения свободного участка полосы пропускания архитектура Token Ring требует передачи нуль-символов. Конечно, этот свободный участок полосы пропускания расходуется непроизводительно, но в случае комбинации 16 Мбит/с с ETR передающая станция освобождает маркер сразу же после отправки кадра данных, что позволяет повысить пропускную способность сети до 80%. В 4 Мбит/с сети маркер удерживается до тех пор, пока «старый» кадр не вернется на передающую станцию.

В настоящее время 16 Мбит/с сети чаще всего используются в качестве магистралей в составе крупных сетей с топологией Token Ring.

Еще одна топология, построенная на основе Token Ring – FDDI – это распределенный интерфейс передачи данных по волоконно-оптическому кабелю, обеспечивающий скорость передачи данных 100 Мбит/с. Более подробно эта топология будет рассмотрена в Части 3.

1.4. Типы кадров

Архитектура Token Ring характеризуется тремя различными типами кадров:

- *Кадр маркера.* Трехбайтовый кадр, циркулирующий по кольцу в качестве управляющего сигнала.
- *Кадр данных.* Этот кадр является носителем либо MAC-информации, либо LLC-информации. MAC-информация используется для управления потоком данных в кольце, LLC-информация представляет собой данные пользователя, которые необходимо передать по кольцу.
- *Кадр последовательности аварийного завершения.* Используется для очистки кольца в том случае, когда с тем или иным кадром возникает какая-то проблема. В этой ситуации станция, которая в данный момент управляет маркером (исходная станция), передает кадр последовательности аварийного завершения.

Кадр маркера

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

- *Поле начального ограничителя* появляется в начале маркера, а также в начале любого кадра, проходящего по сети. Поле состоит из уникальной серии электрических импульсов, которые отличаются от тех импульсов, которыми кодируются единицы и нули в байтах данных. Поэтому начальный ограничитель нельзя спутать ни с какой битовой последовательностью.
- *Поле контроля доступа.* Разделяется на четыре элемента данных: **PPP-T-M-RRR**, где PPP - биты приоритета, T - бит маркера, M - бит монитора, RRR - резервные биты.

Каждый кадр или маркер имеет приоритет, устанавливаемый битами приоритета (значение от 0 до 7, 7 - наивысший приоритет). Станция может воспользоваться маркером, если только она получила маркер с приоритетом, меньшим или равным, чем ее собственный. Сетевой адаптер станции, если ему не удалось захватить маркер, помещает свой приоритет в резервные биты маркера, но только в том случае, если записанный в резервных битах приоритет ниже его собственного. Эта станция будет иметь преимущественный доступ при последующем поступлении к ней маркера.

Схема использования приоритетного метода захвата маркера показана на рисунке 1.4.

Сначала монитор помещает в поле текущего приоритета P максимальное значение приоритета, а поле резервного приоритета R обнуляется (маркер 7110). Маркер проходит по кольцу, в котором станции имеют текущие приоритеты 3, 6 и 4. Так как эти значения меньше, чем 7, то захватить маркер станции не могут, но они записывают свое значение приоритета в поле резервного приоритета, если их приоритет выше его текущего значения. В результате маркер возвращается к монитору со значением резервного приори-

тета R = 6. Монитор переписывает это значение в поле P, а значение резервного приоритета обнуляет, и снова отправляет маркер по кольцу. При этом обороте его захватывает станция с приоритетом 6 - наивысшим приоритетом в кольце в данный момент времени.

Бит маркера имеет значение 0 для маркера и 1 для кадра.

Бит монитора устанавливается в 1 активным монитором и в 0 любой другой станцией, передающей маркер или кадр. Если активный монитор видит маркер или кадр, содержащий бит монитора в 1, то активный монитор знает, что этот кадр или маркер уже однажды обошел кольцо и не был обработан станциями. Если это кадр, то он удаляется из кольца. Если это маркер, то активный монитор переписывает приоритет из резервных битов полученного маркера в поле приоритета. Поэтому при следующем проходе маркера по кольцу его захватит станция, имеющая наивысший приоритет.

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

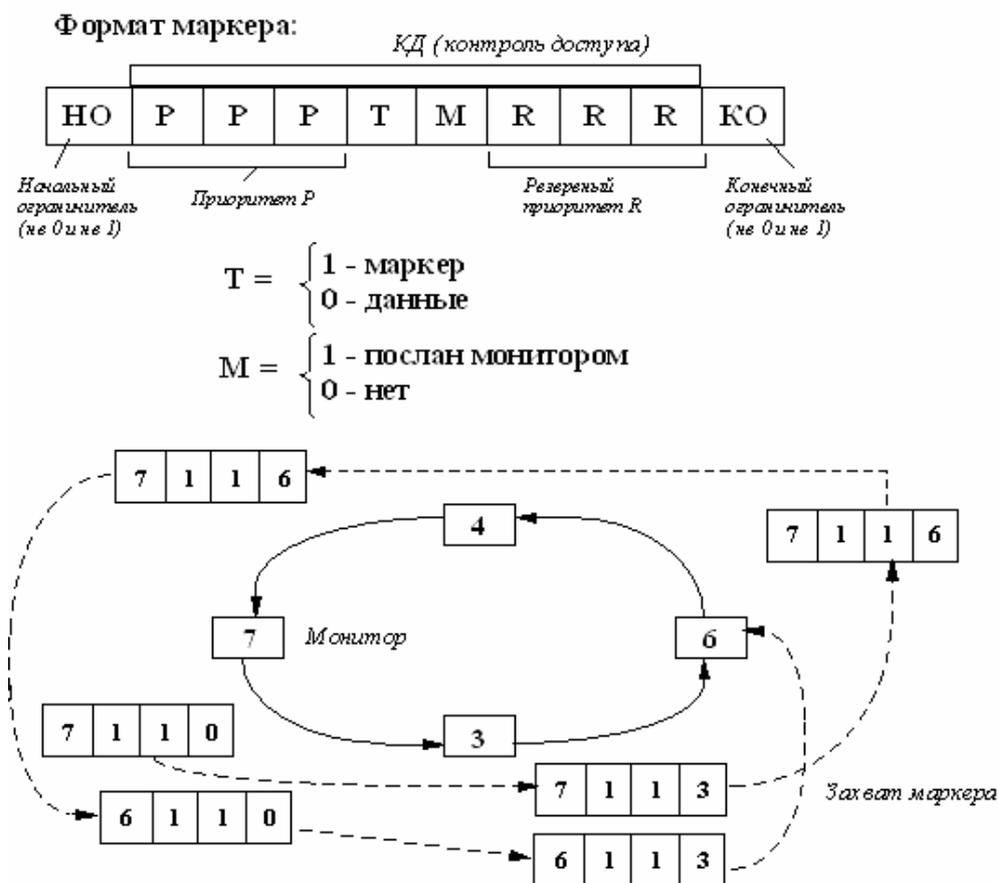


Рис. 1.4 Захват маркера

Кадр данных

Кадр данных состоит из нескольких групп полей:

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

Кадр данных может переносить данные либо для управления кольцом (данные MAC-уровня), либо пользовательские данные (LLC-уровня).

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

- Начальный ограничитель, такой же, как и для маркера;
- Управление доступом, также совпадает для кадров и для маркеров;
- Контроль кадра - это однобайтовое поле, содержащее два подполя - тип кадра и идентификатор управления MAC: 2 бита типа кадра имеют значения 00 для кадров MAC и 01 для кадров LLC. Биты идентификатора управления MAC определяют тип кадра управления кольцом.

Адрес получателя (либо 2, либо 6 байтов). Первый бит определяет групповой или индивидуальный адрес как для 2-х байтовых, так и для 6-ти байтовых адресов. Второй бит в 6-ти байтовых адресах говорит, назначен адрес локально или глобально.

Адрес отправителя имеет тот же размер и формат, что и адрес получателя.

Поле данных кадра может содержать данные одного из управляющих кадров MAC или запись пользовательских данных, предназначенных для (или получаемых от) протокола более высокого уровня, такого как IPX или NetBIOS. Это поле не имеет определенной максимальной длины.

Последовательность контроля кадра - используется для обнаружения ошибок, состоит из четырех байтов остатка циклически избыточной контрольной суммы (вычисляемой по алгоритму CRC-32, осуществляющему циклическое суммирование по модулю 32).

Последовательность конца кадра состоит из двух полей: конечный ограничитель и статус кадра.

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

Статус кадра имеет длину 1 байт и содержит 4 резервных бита и два подполя: бит распознавания адреса и бит копирования кадра.

Прерывающая последовательность

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

1.5. Управляющая роль кольца

Управление физическим уровнем Token Ring осуществляется посредством ряда функций, присущих архитектуре Token Ring. Каждая станция в кольце имеет сетевую плату, которая содержит агента. Этот агент взаимодействует с определенными управляющими станциями Token Ring в кольце посредством передачи кадров MAC. Роль управляющей станций заранее определена архитектурой Token Ring.

Роль этих станции очень важна для поддержания целостности связи в кольце. Управляющие станции обеспечивают функции управления локальным кольцом и функции сервера, связанные с управлением кольцом. Функции, связанные с управлением локальным кольцом, служат для синхронизации связи в локальном кольце. Функции сервера по управлению кольцом взаимодействуют со средой LAN Manager/IBM Systems Networking Architecture (SNA). Для обеспечения взаимодействия управляющих станций и серверов при управлении кольцом используется протокол IBM Network Management (IBMNM).

В процессе управления кольцом управляющая станция может выполнять следующие функции:

- пассивного монитора (Standby monitor — SM)
- активного монитора (Active monitor — AM)
- сервера отчетов о конфигурациях (Configuration report server — CRS)
- сервера параметров кольца (Ring parameter server - RPS)
- монитора ошибок кольца (Ring error monitor - REM)
- сервера моста LAN (LAN bridge server - LBS)
- механизма выдачи отчетов LAN (LAN reporting mechanism - LRM)

Пассивный монитор

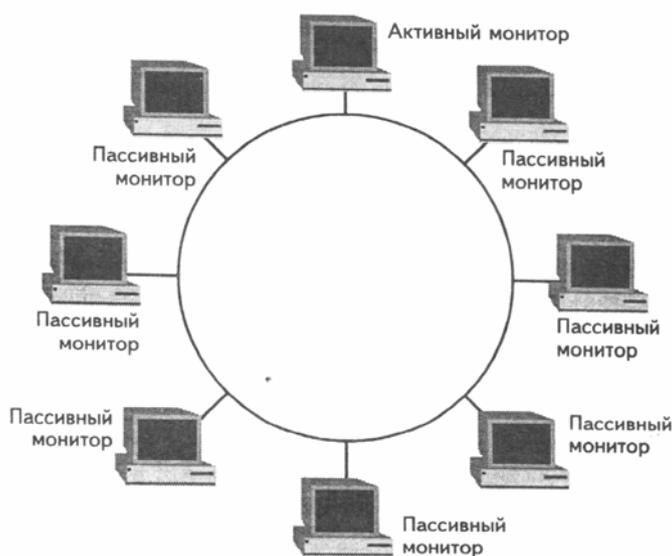
Роль пассивных мониторов (SM) не ограничена управляющими функциями. Скорее все они являются кольцевыми станциями (RS) общего назначения. Пассивные мониторы действительно время от времени обеспечивают функции локального управления, поскольку они отвечают за обнаружение сбоев, происходящих в активном мониторе. Если SM не обнаруживают в кольце MAC-кадра активного монитора, то они вступают в состязание за роль активного монитора (AM).

Важно отметить, что агент на сетевой плате каждой RS используется в MAC-взаимодействии с такими основными управляющими станциями, как AM, CRS, RPS, REM, LBS и LRM.

SM могут также запрашивать некоторые важные параметры кольца от одной из управляющих станций.

Активный монитор

Активный монитор является главным менеджером связи в кольце. Он отвечает за поддержание передачи данных и управляющую информацию, циркулирующую между всеми станциями кольца. Архитектура Token Ring основана на поддержке AM целостности кольца за счет стабилизационных "реперных точек" (рис. 1.5).



Одна станция в кольце работает как активный монитор (AM); другие станции в кольце выполняют функции пассивных мониторов

Рис. 1.5 Активный монитор в кольце

Сервер отчетов о конфигурациях

Сервер отчетов о конфигурациях (CRS) - это функция управления кольцом, выполняемая одной из RS, в среде с несколькими кольцами, где контроль работы колец должен осуществляться из центрального пункта (LAN Manager). Каждое кольцо в среде с несколькими кольцами содержит один CRS. CRS может также выполнять функции обычной RS.

Основная задача CRS — сбор важной статистической информации из кольца и передача этой информации на консоль LAN Manager. Эта информация включает данные по отдельным RS, например, об изменении NAUN и передаче MAC-кадров New Monitor.

Сервер параметров кольца

Сервер параметров кольца (RPS) выполняет функцию локального управления и играет роль сервера кольца. В каждом кольце обычно имеется одна RS, действующая в качестве RPS. Эта роль также очень важна в среде с несколь-

кими кольцами, поскольку RPS передает определенную информацию локального кольца на консоль LAN Manager для Token Ring. RPS может также выполнять функции обычной RS.

Монитор ошибок кольца

При поиске неисправностей в сетях самую существенную информацию предоставляет монитор ошибок кольца REM. Единственное его назначение – сбор информации об ошибках кольца. REM может выполнять локальное управление и быть сервером кольца. Обычно REM не применяется в качестве рядовой RS; он предназначен исключительно для сбора статистики об ошибках.

Функцию REM выполняет и большая часть анализаторов протокола Token Ring при поиске неисправностей в кольце,

Сервер моста LAN

Сервер моста LAN (LBS) играет роль сервера кольца и, следовательно, выполняет мониторинг статистических данных, маршрутизируемых между двумя или несколькими кольцами, соединенными мостом. Этот сервер может также выполнять функции обычной RS.

1.6. Подключение станции к сети Token Ring

Каждая станция при подключении к сети Token Ring выполняет процедуру вставки в кольцо (Ring Insertion). Этот процесс обычно инициируется пользователем соответствующей станции — либо включением питания станции, либо программным методом.

Станция считается активным SM только по завершении процесса вставки в кольцо, включающего пять фаз:

- Фаза 0: Тест "Проверка абонентского носителя" - проверка частоты битовых ошибок в шлейфе между сетевой платой и портом устройства многостанционного доступа. В результате осуществляется физическое подключение станции к кольцу.
- Фаза 1: Станция, только что подключившаяся к кольцу в течение заданного интервала времени проверяет, имеется ли в кольце AM, после чего ожидает приема одного из трех MAC-кадров: "активный монитор присутствует", "резервный монитор присутствует" или "очистка кольца". Если станция обнаружит один из этих трех MAC-кадров до момента истечения заданного интервала времени, она делает вывод о наличии AM в кольце и переходит к фазе 2. Если станция не обнаружит ни одного из этих трех MAC-кадров, она запускает процесс объявления маркера.
- Фаза 2: Основная цель этой фазы — удостовериться в том, что ни одна из станций в кольце не имеет того же адреса, что и у станции, пытающейся подключиться к кольцу. Станция выдает в кольцо MAC-кадр

- Фаза 3: В ходе этой фазы станция впервые участвует в процессе уведомления соседа. Станция узнает адрес своего NAUN и уведомляет его о своем собственном адресе. Если в это время произойдет какое-либо прерывание (например, выдача сигнального кадра - beacon), станция исключает себя из кольца и перезапускает процесс Ring Insertion.
- Фаза 4: В ходе этой фазы станция выдает в кольцо MAC-кадр Request Initialization, который адресуется RPS. Этот кадр сообщает RPS о том, что к кольцу подключилась какая-то станция, готовая принять любые специальные рабочие параметры RPS, предназначенные для данного кольца. Затем RPS передаст MAC-кадр Initialize Ring Station, устанавливающему правильные значения номера кольца и таймера отчетов о нерегулярных ошибках для новой станции. Если у RPS имеются какие-либо специальные параметры, он также передает их на новую станцию.

По завершении перечисленных пяти фаз новая станция считается физически и логически подключенной.

1.7. Наиболее интересные механизмы работы Token Ring

Объявление маркера

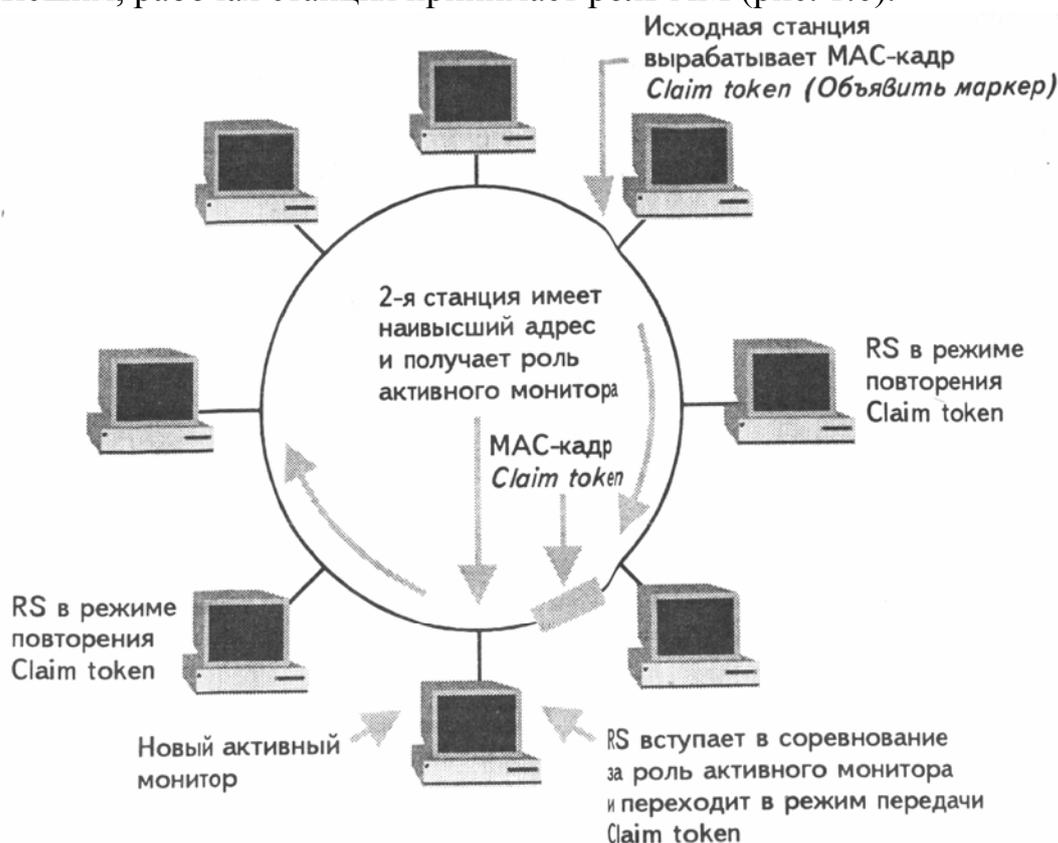
Только одна RS в кольце обозначается как AM. Объявление маркера – это процесс, в ходе которого SM вступают в состязание за право выполнять роль AM. Объявление маркера происходит в кольце в следующих случаях:

- Новая станция подключается к кольцу, но не обнаруживает в этом кольце AM. Это объясняется тем, что, возможно, новая станция не приняла в кольце MAC-кадра Active Monitor Present в ходе процесса Ring Insertion.
- AM не может обнаружить в кольце каких-либо кадров, а время ожидания Token Ring T(Receive_Notification) уже истекло.
- Какой-либо SM определяет отсутствие AM или не может обнаружить в кольце каких-либо кадров, а его таймеры протокола T(Good-Token) или T(Receive_Notification) уже исчерпаны.

Не все станции участвуют в процессе Token Claiming, поскольку по умолчанию это участие не предполагается. В ходе этого процесса станции работают в одном из двух основных режимов: Claim Token Transmit или Claim Token Repeat.

Каждая RS в режиме Claim Token Repeat сравнивает свой адрес с адресом в принятом MAC-кадре Claim Token. Если ее адрес выше, она также становится участником процесса Token Claiming и вырабатывает в свою очередь

собственный MAC-кадр Claim Token. Этот процесс продвигается по кольцу до тех пор, пока какой-либо из участников трижды не примет свой собственный MAC-кадр Claim Token, Определив, что именно ее адрес является наивысшим, рабочая станция принимает роль АМ (рис. 1.6).



Состязание RS за роль активного монитора

Рис. 1.6 Состязание кольцевых станций за роль активного монитора
Приоритетный доступ

В поле управления доступом первые три бита — это биты приоритета, определяющие текущий приоритет определенного маркера или кадра. Последние три бита поля управления доступом зарезервированы для запроса определенной рабочей станцией приоритета при необходимости доступа к кольцу. Биты приоритета и резервные биты могут принимать значения от 111 (высший приоритет) до 000 (низкий приоритет). Все RS могут запрашивать и поднимать уровень своего приоритета в кольце.

RS руководствуется полями управления доступом при выяснении приоритета данного кадра или маркера и при резервировании кадра или маркера для какого-либо определенного приоритета.

Если этот бит принят на АМ в состоянии 1, АМ полагает, что соответствующий кадр или маркер обошел кольцо, по меньшей мере, один раз, но выдавшая его станция так и не приняла этот маркер обратно. АМ сбрасывает указанный бит в 0 и инициирует процесс Ring Purge для очистки кольца.

Этот механизм уравнивает шансы всех станций на доступ к кольцу. Принцип равенства шансов подкрепляется еще и тем, что станция, подни-

мающая уровень своего приоритета в кольце, должна вернуться к исходному уровню приоритета.

Уведомление соседа (Neighbor Notification) - это последовательный логический процесс, в ходе которого каждая RS узнает адрес своего NAUN.

В структуре каждого кадра MAC/LLC имеется поле статуса кадра (frame Status — FS) — однобайтовое поле, которое отражает текущий статус соответствующего кадра. Поле FS включает два бита: адрес распознан (А-бит) и кадр скопирован (С-бит). Выполнение процесса Neighbor Notification преследует следующие цели:

1. Выяснить в ходе проверки А-битов, распознан ли данный кадр последней станцией по адресу источника (SA), включенному в него.
2. Определить в ходе проверки С-битов, успешно ли скопирован кадр станцией.

Очистка кольца

У каждой станции есть состояние, называемое режимом нормального повторения. В этом состоянии, RS опрашивает все принимаемые ею маркеры и может соответствующим образом копировать и повторять их.

Процесс очистки кольца представляет собой попытку установки кольца в режим нормального повторения. АМ инициирует процесс очистки кольца в следующих случаях:

- При обнаружении в кольце ошибки (например, потеря маркера или кадров, сбой в активной синхронизации между станциями кольца или неправильное выполнение какого-либо процесса Token Ring)
- Если в поле управления доступом маркера или кадра М-бит установлен в 1.
- Если кольцо требуется перевести в режим нормального повторения.
- Когда АМ обнаруживает, что интервал таймера T(Any-Token) истек.

Сгенерировав кадр, АМ ожидает приема этого кадра обратно. Если кадр принят, АМ полагает, что кольцо стабилизировано, и сбрасывает все таймеры протоколов Token Ring, после чего инициирует процесс Neighbor Notification для перезапуска и перевода кольца обратно в режим нормального повторения.

Порядок выполнения работы

Интегрированная программная среда моделирования функционирования ЛВС

Для проведения лабораторной работы необходимо познакомиться с интегрированной программной средой моделирования.

Интегрированная программная среда моделирования построена на основе аналитических мультипликативных моделей сетей массового обслуживания

и дает возможность проводить анализ вероятностно-временных характеристик следующих локальных вычислительных сетей :

- а. Одноканальной ЛВС шинной топологии со свободным (случайным) методом доступа,
- б. Одноканальной кольцевой ЛВС с маркерным методом доступа,
- в. Двухканальная ЛВС кольцевой топологии с маркерным методом доступа.

Интегрированная программная среда моделирования работы ЛВС обеспечивает:

- комфортный интерфейс работы пользователя (иерархическая система меню на основе использования многооконной технологии);
- возможность выбора объекта моделирования;
- возможность ввода исходных данных с клавиатуры и из файла;
- обнаружение ошибок ввода исходных данных;
- динамически изменяемый формат вводимых данных;
- получение вероятностно-временных характеристик исследуемой ЛВС при заданных исходных данных.

Среда моделирования при ее настройке на одноканальную кольцевую ЛВС с маркерным методом доступа обеспечивает получение следующих вероятностно-временных характеристик:

- вероятность наличия маркера у данного сетевого адаптера;
- вероятность нахождения пакета в моноканале;
- коэффициент вариации времени обработки пакета (с момента его генерации абонентом-отправителем до момента его получения абонентом-получателем);
- среднее время обработки пакета;
- среднее время ожидания пакетом обслуживания (предоставление маркера в моноканале);
- дисперсия времени ожидания пакетом обслуживания;
- коэффициент вариации времени ожидания пакетом обслуживания;
- среднее время пребывания пакета в зоне моноканала;
- дисперсия времени пребывания пакета в зоне моноканала;
- коэффициент вариации времени пребывания в зоне моноканала;
- среднее время ожидания маркера + среднее время передачи пакета в моноканале.

1.1. Запуск интегрированной программной среды моделирования ЛВС

В среде «NC» найти директорию «SR-NET» и активизировать в ней файл «unig.exe».

В результате активизации на мониторе появится панель среды и информационное сообщение об особенностях среды.

После этого Вы должны левой клавишей «мышки» активизировать работу в моделирующей среде.

1.2. Особенности работы с интегрированной программной средой моделирования ЛВС

После успешного запуска интегрированной программной среды моделирования на экране монитора формируются три части:

- Полоса меню
- Панель экрана
- Строка статуса.

Полоса меню состоит из семи пунктов:

- File (Файл)
- Edit (Редактор)
- Run (Пуск)
- Options (Настройка среды)
- Search (Поиск)
- Windows (Окна)
- About (О программе)

File	Edit	Options	Run	Search	Windows	About
------	------	---------	-----	--------	---------	-------

Пункт меню File

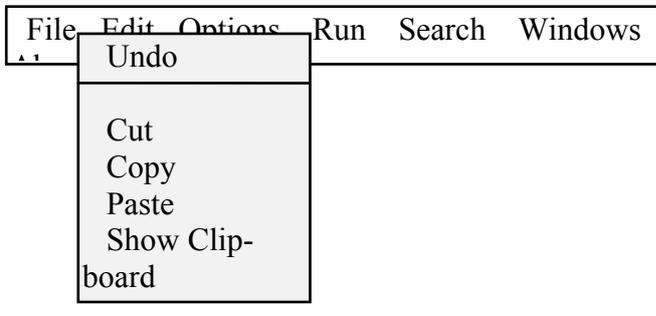
предназначен для работы с файлами.

File	Edit	Options	Run	Search	Windows
New					
Open					
F3					
Save					
F2					
Save As					
Save All					

- New - создание нового файла для редактирования.
- Open - загрузка файла для редактирования.
- Save - сохранение файла из текущего окна редактирования.
- Save As - сохранение файла из текущего окна редактирования под уникальным именем.
- Save All - сохранение всех файлов в среде.
- Change Dir - смена текущей директории.
- DOS Shell - выход ОС для выполнения какой-либо команды, программа `unir.exe` остается в памяти для возврата в программу из среды необходимо набрать `EXIT`.
- 8. Exit - выход из программы.

Пункт меню Edit

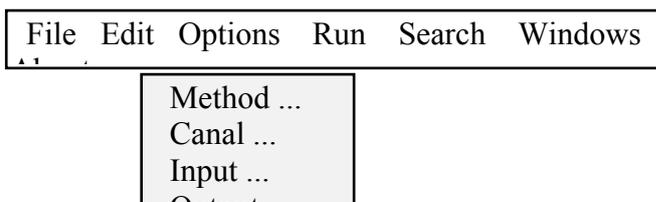
предназначен для удобства редактирования текстового файла (ов).



- Undo - отменить все изменения , сделанные в файле с момента редактирования.
- Cut - уничтожить текст в куче.
- Copy - скопировать текст в кучу.
- Paste - вставить текст из кучи.
- Show Clipboard - показать на экране текущую кучу.
- Clear - уничтожить текст в текущем редактируемом окне.

Пункт меню Options

предназначен для настройки среды.



Пункт подменю Method

предназначен для выбора метода исполнения .



Пункт подменю Canal

предназначен для выбора количества моноканалов при выборе маркерного метода доступа.

Количество каналов
(*) Один канал

Пункт подменю Input

предназначен для выбора устройства ввода : клавиатура или файл.

Ввод данных
(*) С клавиатуры

Пункт подменю Output

предназначен для задания формата выводимых данных.

А. Среда настроена на модель одноканальной ЛВС с произвольным методом доступа.

Формат выводимых данных	
Выводимые массивы данных :	Количество значений :
[x] Вер-ть нахождения пакета в к-	(*) Все значения () Предельные и

Б. Среда настроена на модель одноканальной ЛВС с маркерным методом доступа.

Формат выводимых данных

Выводимые массивы данных :

Количество значений :

Вер-ть наличия маркера у
СА
 Вер-ть нахождения пакета
в к-ле
 К вариации t обработки па-
кета
 среднее t обработки пакета
 среднее t ожидания обслу-
живания
 диспенсия t ожилаия об-

(*) Все значения
 () Предельные и

В. Среда настроена на модель двухканальной ЛВС с маркерным мето-
дом доступа.

Формат выводимых данных

Выводимые массивы данных :

Количество значений :

Вер-ть наличия маркера у
СА
 Вер-ть нахождения пакета
в к-ле
 К вариации t обработки па-
кета
 среднее t обработки пакета
 среднее t ожидания обслу-

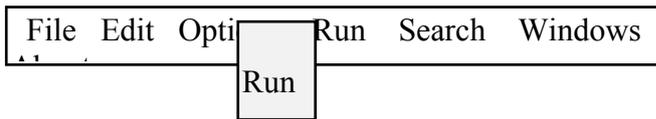
(*) Все значения
 () Предельные и

Пункт меню Run

рассчитывает математические характеристики для заданной пользователем модели ЛВС Options/Method...). По умолчанию - модель одноканальной ЛВС с произвольным методом доступа.

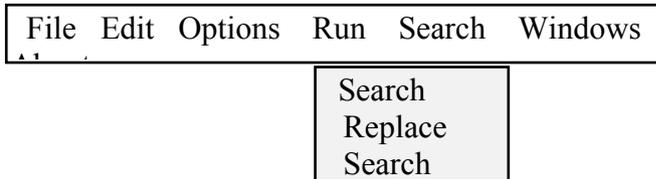
По выбору пользователя (Options/Input...) исходные данные вводятся с клавиатуры или из файла. По умолчанию - с клавиатуры. Введенные с клавиатуры данные записываются в файл, который впоследствии может быть отредактирован пользователем.

Результаты выводятся в формате, выбранном пользователем (Options/Output...). По умолчанию происходит полный вывод результатов. Сформированные конечные данные записываются в файл, который загружается в редактор.



Пункт меню Search

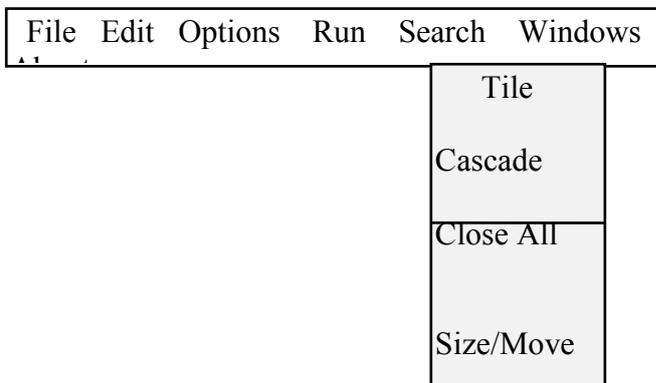
предназначен для удобства редактирования файла.



- Search - поиск слов по тексту.
- Replace - замена текста в тексте.
- Search Again - повтор поиска.

Пункт меню Windows

Предназначен для работы с окнами.



- Size/Move - перемещение и масштабирование окна.
- Zoom - масштабирование активного окна до границ панели экрана.
- Next - активизация последующего открытого окна.
- Previous - активизация предпоследнего открытого окна.
- Close - закрытие текущего окна.
- Calculator - калькулятор.
- Tile - располагает все открытые окна в порядке мозаики.
- Cascade - располагает все открытые окна в порядке каскада.
- Close All - закрытие всех открытых окон.

1.3. Особенности работы с интегрированной программной средой моделирования при выполнении данной лабораторной работы .

Изучение реально работающей сети представляет собой трудновыполнимую задачу, требующую достаточно сложный комплекс измерительной аппаратуры, поэтому исследования в данной лабораторной работе будут производиться при помощи специальной моделирующей программы.

Запуск интегрированной среды моделирования ЛВС

В среде «NC» найти директорию «SR-NET» и активизировать в ней файл «unir.exe».

В результате активизации на мониторе появится панель среды и информационное сообщение об особенностях среды.

После этого Вы должны левой клавишей «мышки» активизировать работу в моделирующей среде.

Настройка среды

Настройка среды реализуется активизацией пункта главного меню «Option». Затем активизируется пункт подменю «Method», в котором среда с помощью «мышки» настраивается в соответствии с заданием данной лабораторной работы на маркерный метод.

Выбор числа каналов определяется активизацией в пункте главного меню «Option» пункта подменю «Canal» и для выполняемой лабораторной работы - выбор «Один канал».

Затем аналогично производится настройка на форму ввода исходной информации (пункт подменю «Input»: «С клавиатуры» или «Из файла»). И на формате выводимых данных задается один из двух вариантов выводимых значений :

- перечень всех предусмотренных значений;
- перечень лишь предельных и средних значений.

Настройка на форму ввода а вывода завершается фиксацией настройки .

После завершения настройки модели переходим к режиму «Выполнение моделирования (Пункт «Run» главного меню среды моделирования).

Выполнение моделирования

В зависимости от настройки подпункта «Option» - «Input» - ввод исходной информации:

- с клавиатуры;
- из файла,

программа моделирования будет развиваться по разным сценариям.

Рассмотрим вначале первый сценарий - «ввод исходной информации с клавиатуры».

Активизируйте пункт «Run» главного меню. Затем программа последовательно запрашивает на ввод ряд значений для заданного варианта настройки, которые необходимо ввести в соответствии с Вашим заданием на выполнение лабораторной работы. В том случае, если значение вводимых исходных данных выходит за диапазон допустимых, программа просит повторить ввод корректных значений.

После завершения ввода исходных данных программа запрашивает «Имя файла для сохранения данных». - в этом случае необходимо сокращенно записать свою фамилию с расширением «.DAT», например, IVAN1.DAT .

Затем программа запрашивает «Имя файла для вывода результата моделирования». - и пользователь записывает, например, IVAN2.DAT.

После чего на экране появятся результаты моделирования, которые для нашего примера будут размещены в файле под именем «ivan2.dat» .

Таким образом, в памяти ПЭВМ вы имеете два файла:

- файл исходных данных для моделирования - «IVAN1.DAT»;
- файл результатов моделирования - «IVAN2.DAT».

Эти файлы можно вывести на принтер или скопировать на дискету для дальнейшей обработки при оформлении отчета по лабораторной работе.

Рассмотрим второй сценарий - «ввод исходных данных из файла».

Этот сценарий используется в двух случаях:

- хотите повторить моделирование ;
- хотите провести моделирование при скорректированных исходных данных (здесь в файле исходных данных можно поменять значения параметров и / или формат вывода результатов).

Итак, для выполнения второго сценария нам необходимо скорректировать значения исходных данных моделирования представленных в файле IVAN1.DAT.

Пока мы находимся в среде моделирования и поэтому сделаем следующие действия:

- активизируем пункт главного меню «Option», затем активизируем подпункт «Input»;
- в меню «выбор способа ввода данных» активизируем режим «ввод данных из файла».

Таким образом, теперь среда моделирования настроена на работу с файлом исходных данных.

- активизируем пункт меню «File» - «Open» - откроется экран с перечнем имен файлов с расширением *.DAT.

- активизируйте мышью имя файла исходной информации.

В нашем примере это будет файл с именем «IVAN1.DAT» и откройте этот файл.

На экране появится содержание рассматриваемого файла.

Теперь вы можете изменить значение параметров исходных данных в соответствии с заданием на моделирование и перейти в режим выполнения и моделирования «Run».

- активизируйте пункт «Run» главного меню.

- появится экран, в рабочем поле которого необходимо набрать имя файла для вывода результатов моделирования, например «IVAN3.DAT».

Теперь на экране появится содержание результатов моделирования, которые представлены в файле под именем «IVAN3.DAT».

В памяти ПЭВМ будете иметь исходные данные моделирования в файле «IVAN1.DAT» (они скорректированы по отношению к варианту набора данных с клавиатуры) и результаты моделирования будут представлены в файле «IVAN3.DAT».

Эти данные можно вывести на печать и / или скопировать на дискету для дальнейшей обработки при оформлении отчета по лабораторной работе.

Используя рассматриваемые выше сценарии, можно реализовать все заданные варианты моделирования ЛВС.

Для выхода из среды моделирования необходимо активизировать «мышью» пункт главного меню «File» и в окне этого меню выбрать «мышью» раздел «Exit». При этом вы вернетесь в среду «Norton Commander».

Задание на выполнение лабораторной работы

Приступать к индивидуальной работе можно только после ответа преподавателю на контрольные вопросы.

Таблица вариантов заданий*

№ вар.	Число Абонентов n	$\Sigma \lambda_i$	Tau1	T01	Точность результат.	Допуст. время обработ.	Значение исходной итерации
1	10	6.0	0.01	0.07	0.001	0.7	0.1
2	15	9.0	0.002	0.06	0.0001	0.5	0.05
3	18	12.0	0.002	0.049	0.0001	0.6	0.01
4	8	4.5	0.01	0.08	0.05	0.3	0.005
5	10	5.0	0.01	0.08	0.0001	0.4	0.1
6	16	11.8	0.002	0.054	0.001	0.5	0.005
7	18	11.0	0.003	0.01	0.0001	0.3	0.1
8	15	11.0	0.001	0.08	0.001	0.2	0.05
9	18	11.0	0.005	0.012	0.01	0.8	0.05

10	16	12.0	0.00107	0.07	0.00001	0.4	0.005
11	8	5.0	0.01	0.07	0.005	0.8	0.05
12	16	10.0	0.0011	0.085	0.00001	0.3	0.001
13	10	4.0	0.008	0.08	0.0001	0.5	0.05
14	18	9.5	0.006	0.01	0.0005	0.6	0.01
15	18	11.0	0.002	0.05	0.005	0.9	0.005
16	16	11.0	0.00095	0.078	0.000001	0.4	0.001
17	10	3.0	0.012	0.09	0.001	0.3	0.05
18	12	7.0	0.005	0.085	0.001	0.5	0.01
19	8	3.0	0.01	0.1	0.001	0.6	0.005
20	16	9.0	0.0018	0.1	0.001	0.5	0.01
21	8	4.0	0.01	0.09	0.005	0.4	0.005
22	18	10.2	0.005	0.01	0.0001	0.3	0.001
23	8	2.5	0.02	0.08	0.001	0.7	0.0005
24	10	4.5	0.01	0.09	0.005	0.3	0.005
25	12	6.0	0.006	0.09	0.001	0.8	0.0005
26	15	10.0	0.002	0.07	0.005	0.6	0.005
27	16	12.0	0.0015	0.05	0.005	0.2	0.0005
28	18	9.0	0.001	0.1	0.001	0.9	0.0001
29	8	4.5	0.008	0.1	0.0005	0.3	0.001
30	10	4.5	0.01	0.1	0.005	0.6	0.00005

В ходе этой работы вам необходимо получить и объяснить некоторые особенности функционирования сети с маркерным доступом. Эти особенности хорошо иллюстрируются зависимостями эффективности работы сети от исходных данных. Вам необходимо получить по 10-15 значе*ий исследуемого параметра и занести их в таблицу для дальнейшего построения графиков зависимостей. При этом значения величины, влияющей на исследуемый параметр, необходимо брать таким образом, чтобы он изменялся от своего минимально возможного значения до максимума. После этого необходимо объяснить полученные результаты.

В результате проведения серии экспериментов при различных исходных данных необходимо построить графики и исследовать следующие зависимости:

1. Среднего времени полного цикла обращения от коэффициента загрузки. Интенсивность обращения к моноканалу каждого из заданных абонентов определяется студентом самостоятельно из условия, что известны значения « n_i » и « $\sum \lambda_i$ », а также заданы следующие ограничения для λ_i : $0 < \lambda_i \leq 0,99$.

Например,

если $n=4$, а $\sum \lambda_i = 3.0$, то тогда исходные значения λ_i могут быть представлены следующим образом: $\lambda_1 = 0,9$; $\lambda_2 = 0,6$; $\lambda_3 = 0,7$; $\lambda_4 = 0,8$.

2. Среднего времени ожидания обслуживания в зависимости от числа абонентов при фиксированных значениях « T_0 », « τ » и « ρ ».

3. Среднего времени обработки пакета в зависимости от изменения значений T_0 .
4. Среднего времени обработки пакета в зависимости от изменения значений τ .
5. Среднего времени ожидания обслуживания в зависимости от значений суммарной интенсивности от всех абонентов.
6. Среднего времени пребывания пакета «i»-го сетевого адаптера в зависимости от того, какому абоненту «j» принадлежит в рассматриваемый момент маркер $R(i,j)$. (Последняя таблица результатов для «всех значений»).

Раскроем более подробно характеристики результатов моделирования первого варианта.

«Полный цикл обращения» - время, в течение которого в загруженной ЛВС маркер будет использован всеми сетевыми станциями и затем вернется в исходную.

Вероятность наличия маркера в любой произвольный момент времени у сетевого адаптера (CA_i) с номером i . (Слева от результатов в скобках указан номер сетевого адаптера).

«Вероятность наличия пакетов в канале» - вероятность того, что в любой произвольный момент времени в моноканале передается пакет, передаваемый от сетевого адаптера номер CA_i . Номер сетевого адаптера представлен слева от результатов в скобках.

«Среднее время обработки пакета» - среднее время передачи пакета данных от любого «i»-го сетевого адаптера CA_i в моноканале. Номер сетевого адаптера представлен слева от результатов в скобках.

$$(\tau_{ci} = Q_i T_0 + \tau)$$

«Среднее время ожидания обслуживания» - среднее время ожидания получения маркера для «i»-го сетевого адаптера CA_i .

«Среднее время пребывания пакета «i»-го сетевого адаптера CA_i в зоне моноканала » = «среднему времени обработки пакета» + «среднее время ожидания».

«t ожидания маркера» + «t передачи пакета» = «полное время ожидания передачи маркера» + передача пакета «i»-го сетевого адаптера CA_i , если в рассматриваемый момент времени маркер принадлежит «j»-му сетевому адаптеру. Слева в скобках (i,j) представлено:

«i» - номер рассматриваемого сетевого адаптера;

«j» - маркер принадлежит «j»-му сетевому адаптеру.

Содержание отчёта

Результаты исследования необходимо оформить в виде таблиц и графиков. Допускается для построения графиков использовать табличный процессор EXCEL. Пример оформления приведен в табл 7.

Таблица 7. Пример оформления результатов исследования.

Номер опыта	1	2	3	4	5									12
Среднее время ожидания обслуживания														
Суммарной интенсивность всех абонентов														

Контрольные вопросы

1. Дайте описание метода доступа к моноканалу в сетях с передачей маркера.
2. Охарактеризуйте особенности сетей с маркерным методом доступа.
3. Чем отличается алгоритм передачи маркера в ЛВС типа «шина» и в ЛВС типа «кольцо».
4. Объясните алгоритм контроля целостности маркера.
5. Назовите преимущества и недостатки маркерного метода доступа.
6. Опишите особенности функционирования приоритетного маркерного метода доступа.

Литература

1. Олифер В.Г., Олифер Н.А. Компьютерные сети. Принципы, технологии, протоколы. СПб: Издательство «Питер», 1999. -672с.
2. Флинт Д. Локальные сети ЭВМ. Архитектура, принципы построения, реализация. М.: Финансы и статистика, 1986,-360 с.
3. Блэк Ю. Сети ЭВМ: протоколы, стандарты, интерфейсы. М.: Мир, 1990, - 510с.
4. Климанов В.П. Методы разработки аналитических моделей для анализа локальных вычислительных сетей, используемых в управлении технологическими процессами. М.: МЭИ, 1995, -115с.
5. Нанс Б. Программирование в локальных сетях. Пермь:1992.-756с.
6. Шварц М. Сети связи: протоколы, моделирование и анализ. Ч 2. М.: Наука, 1992,- 272 с.

Контрольные задания для СРС

1. Опишите алгоритм доступа к среде технологии *Token Ring*
2. Из каких соображений выбирается максимальное время оборота маркера по кольцу?
3. Если бы вам пришлось выбирать, какую из технологий - *Ethernet* или *Token Ring* - использовать в сети вашего предприятия, какое решение вы бы приняли? Какие соображения привели бы в качестве обоснования этого решения?
4. В чем состоит сходство и различие технологий *FDDI* и *Token Ring*?
5. Какие элементы сети *FDDI* обеспечивают отказоустойчивость?

Методические указания к лабораторной работе №3

Тема: Исследование поведения характеристик надежности сети при введении в систему резервных элементов

Цель работы:

Исследование поведения характеристик надежности системы при различном способе резервирования при помощи аналитического подхода.

Содержание работы:

Потребность в любой технической системе определяется ее эффективностью. Полная эффективность системы определяется суммой эффективностей на всех ее жизненных фазах. Жизнь системы состоит из следующих фаз: создание, хранение, использование и утилизация. Рассмотрим более подробно эффективность использования системы. Данная фаза состоит из двух подфаз: работа системы и ее простой. Находящаяся в использовании система работает, если она выполняет возложенные на нее функции, то есть реагирует заданным образом на определенные действия среды, которая ее окружает. В противном случае система простаивает. Переход системы из рабочего состояния в состояние простоя называется отказом, из состояния простоя в рабочее состояние – восстановлением. Эффективность работы системы определяется разницей в ресурсах на ее выходе и входе. Эффективность простоя системы определяется потерями в связанных с ней ресурсах и затратами на ее восстановление. Научное направление, которое занялось вопросами исследования работоспособности систем, получило название: теория надежности.

Для обнаружения переходов системы между состояниями работы и простоя в нее вводится внешний контроль. Данный контроль может быть двух типов: непрерывный и периодический. При непрерывном типе контроля система контролируется постоянно. При периодическом типе контроля система контролируется через некоторые (в общем случае не фиксированные) промежутки времени. Система может иметь одну часть подсистем, охваченной непрерывным контролем, а другую – периодическим. Использование системы с точки зрения контроля ее работы распадается на следующие повторяющиеся стадии:

1. Нахождение системы в рабочем состоянии до обнаружения ее восстановления.
2. Нахождение системы в рабочем состоянии после обнаружения ее восстановления.
3. Нахождение системы в состоянии простоя до обнаружения ее отказа.
4. Нахождение системы в состоянии простоя после обнаружения ее отказа.

Под общей характеристикой надежности системы понимается вектор случайных величин $S=(S_1, S_2, S_3, S_4)$ времен ее нахождения в этих стадиях. S_1 называется временем периодического контроля работы системы; S_2 – време-

нем отказа системы; S_3 – временем периодического контроля простоя системы; S_4 – временем восстановления системы.

Величины, обратные средним значениям элементов вектора S называются интенсивностями:

$\eta = \frac{1}{S_1}$ — интенсивность периодического контроля работы системы;

$\lambda = \frac{1}{S_2}$ — интенсивность отказов системы;

$\sigma = \frac{1}{S_3}$ — интенсивность периодического контроля простоя системы;

$\mu = \frac{1}{S_4}$ — интенсивность восстановлений системы.

Практический интерес представляют вероятности нахождения системы в этих стадиях:

$$P_1 = \frac{\bar{S}_1}{\bar{S}_1 + \bar{S}_2 + \bar{S}_3 + \bar{S}_4}; P_2 = \frac{\bar{S}_2}{\bar{S}_1 + \bar{S}_2 + \bar{S}_3 + \bar{S}_4}; P_3 = \frac{\bar{S}_3}{\bar{S}_1 + \bar{S}_2 + \bar{S}_3 + \bar{S}_4}; P_4 = \frac{\bar{S}_4}{\bar{S}_1 + \bar{S}_2 + \bar{S}_3 + \bar{S}_4}.$$

P_2 , определяющая среднюю долю времени, в течение которого данная система работоспособна, также носит название коэффициента готовности системы в установившемся режиме K_T .

Кроме состояний полной работоспособности и полного простоя системы возможно введение ряда промежуточных состояний с переходами между ними или построение функциональной зависимости работоспособности системы от времени. Физическим смыслом модели при этом является точность исполнения системой своих функций. Для рассмотрения подобного вида моделей используется более сложный математический аппарат.

С точки зрения восстанавливаемости после отказа все технические системы можно разделить на два класса: восстанавливаемые и невосстанавливаемые. Восстанавливаемые системы разделяются на самовосстанавливающиеся и требующие ремонта. Восстанавливаемая система может обладать как любым из этих качеств, так и обоими качествами одновременно, но с различной интенсивностью восстановления. Для систем, требующих ремонта, существует такой параметр, как число бригад, которые одновременно ее ремонтируют N_b . Если число отказавших подсистем в системе больше, чем N_b , то, если ремонт беспriorитетный, подсистема, вышедшая из строя позже, не ремонтируется (но может самовосстанавливаться), пока не появится свободная бригада для ее ремонта. Если ремонт приоритетный и отказывает подсистема с более высоким приоритетом, то одна из бригад, ремонтирующих в текущий момент подсистемы с наиболее низким приоритетом, переключается на ремонт отказавшей подсистемы.

Для повышения надежности системы существуют два метода: повышение надежности отдельных ее подсистем или их резервирование. Выбор метода

повышения надежности для той или иной системы осуществляется по критерию эффективности получаемой системы. Повышение надежности отдельных подсистем достигается путем улучшения их физических характеристик. Использование метода резервирования дает возможность в случае обнаружения контролем отказа основной подсистемы (то есть той, от которой ожидают нужной реакции на среду) происходит переключение на резервную. Резервирование разделяют на два типа: "горячее" и "холодное". При использовании "горячего" резервирования процесс переключения происходит мгновенно. При использовании "холодного" резервирования – с некоторой временной задержкой, во время которой резервной подсистеме передаются необходимые данные. Величина, обратная среднему значению этой задержки, называется интенсивностью переключения между резервными подсистемами θ . Надежность устройства переключения при резервировании считается существенно более высокой, чем надежность других подсистем.

Для нахождения характеристик надежности системы, как и любых других ее характеристик, применяются два подхода: экспериментальный и аналитический. При использовании экспериментального подхода нужные характеристики системы получаются из результатов опыта. При использовании аналитического подхода для получения нужных характеристик системы применяется следующий алгоритм:

1. Разложение системы на составляющие ее подсистемы.
2. Определение связей между полученными подсистемами.
3. Построение зависимостей характеристик системы от характеристик ее подсистем.
4. Получение нужных характеристик подсистем при помощи экспериментального подхода либо при помощи их дальнейшего анализа.
5. Нахождение характеристик системы через характеристики ее подсистем.

Порядок выполнения работы:

В качестве рассматриваемой взята система, представляющая собой два резервных устройства, соединенных моноканалом:



над которой были выполнены 4 первых шага алгоритма анализа характеристик надежности.

Известными характеристиками подсистем являются:

- интенсивности отказов для первого и второго устройства и моноканала;

- интенсивности периодического контроля простоя для первого и второго устройства;
- интенсивности восстановлений для первого и второго устройства и моноканала;
- тип резервирования устройств (горячий или холодный);
- интенсивность переключения между устройствами в холодном резерве;
- количество ремонтных бригад (одна или две);

Оба устройства являются не самовосстанавливающимися. Контроль их работы считается непрерывным. Моноканал считается самовосстанавливающейся системой с непрерывным контролем работы и простоя.

Характеристики подсистем:

Подсистема	λ	σ	μ
Сервер базы данных	0.00004	0.125	0.5
Принтер	0.0002	0.125	0.5
Моноканал	0.00000 8	—	0.5

Задание

Довести исследование до конца, построив графики зависимостей вероятностей нахождения системы в различных стадиях (P_2, P_3, P_4) при изменении одной из характеристик ее подсистем (убедитесь, что $P_1=0$):

1. При горячем резервировании.
2. При холодном резервировании с интенсивностью переключения $\theta=0,5$.
3. При холодном резервировании с интенсивностью переключения $\theta=1,0$.
4. При холодном резервировании с интенсивностью переключения $\theta=1,5$.

Варианты заданий

№	Число бригад ремонта	Тип резервных устройств	Изменяемый параметр
1	Две бригады	Серверы базы данных	λ_1
2	Две бригады	Серверы базы данных	$\lambda_1=\lambda_2$
3	Две бригады	Серверы базы данных	σ_1
4	Две бригады	Серверы базы данных	$\sigma_1=\sigma_2$
5	Две бригады	Серверы базы данных	μ_1
6	Две бригады	Серверы базы данных	$\mu_1=\mu_2$
7	Две бригады	Серверы базы данных	$\lambda_{\text{моноканала}}$
8	Две бригады	Серверы базы данных	$\mu_{\text{моноканала}}$
9	Две бригады	Принтеры	λ_1
10	Две бригады	Принтеры	$\lambda_1=\lambda_2$
11	Две бригады	Принтеры	σ_1
12	Две бригады	Принтеры	$\sigma_1=\sigma_2$
13	Две бригады	Принтеры	μ_1
14	Две бригады	Принтеры	$\mu_1=\mu_2$
15	Две бригады	Принтеры	$\lambda_{\text{моноканала}}$
16	Одна бригада	Принтеры	$\mu_{\text{моноканала}}$
17	Одна бригада	Серверы базы данных	λ_1
18	Одна бригада	Серверы базы данных	$\lambda_1=\lambda_2$
19	Одна бригада	Серверы базы данных	σ_1
20	Одна бригада	Серверы базы данных	$\sigma_1=\sigma_2$
21	Одна бригада	Серверы базы данных	μ_1
22	Одна бригада	Серверы базы данных	$\mu_1=\mu_2$
23	Одна бригада	Серверы базы данных	$\lambda_{\text{моноканала}}$
24	Одна бригада	Серверы базы данных	$\mu_{\text{моноканала}}$
25	Одна бригада	Принтеры	λ_1

№	Число бригад ремонта	Тип резервных устройств	Изменяемый параметр
26	Одна бригада	Принтеры	$\lambda_1 = \lambda_2$
27	Одна бригада	Принтеры	σ_1
28	Одна бригада	Принтеры	$\sigma_1 = \sigma_2$
29	Одна бригада	Принтеры	μ_1
30	Одна бригада	Принтеры	$\mu_1 = \mu_2$
31	Одна бригада	Принтеры	$\lambda_{\text{моноканала}}$
32	Одна бригада	Принтеры	$\mu_{\text{моноканала}}$

Методика проведения работы

Запустите программу. В появившемся диалоговом окне:

Ввод параметров системы

Интенсивности отказов

Устройство 1 : 0.0002

Устройство 2 : 0.0002

Моноканал : 8e-06

Метод восстановления

Одна бригада :

Две бригады :

Интенсивности восстановлений

Устройство 1 : 0.5

Устройство 2 : 0.5

Моноканал : 0.5

Интенсивности контроля

Устройство 1 : 0.125

Устройство 2 : 0.125

Резервирование

Горячий резерв:

Холодный резерв:

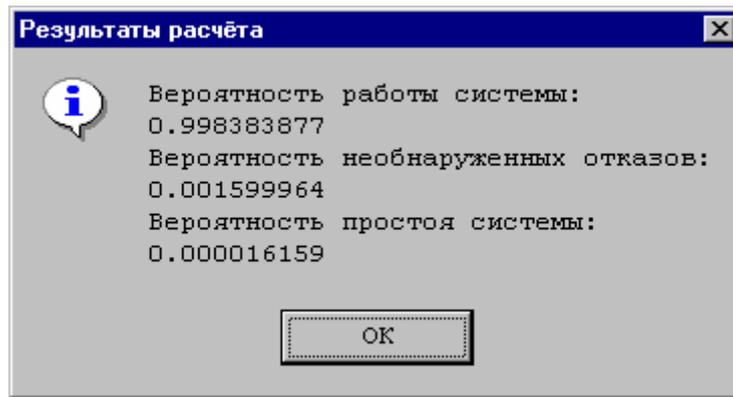
Интенсивность переключения:

Устройство 1 Устройство 2

Моноканал

Расчёт Отмена

введите значения параметров системы. После нажатия кнопки "Расчёт" появится окно сообщения со значениями вероятностей нахождения системы в различных стадиях:



Полученные зависимости вероятностей от исследуемого параметра оформите в виде графиков, используя для этой цели любое стандартное программное средство.

Содержание отчёта

Отчет о результатах исследования должен содержать:

1. Номер варианта.
2. Сформулированное задание.
3. Список исследованных значения изменяемого параметра.
4. Таблицу 4X3 графиков зависимостей вероятностей от исследуемого параметра.

Контрольные вопросы:

1. Чем определяется потребность в любой технической системе?
2. Как определяется полная эффективность системы?
3. Перечислите жизненные фазы технических систем.
4. На каких состояниях может находиться используемая система? Поясните их физический смысл. Как называются переходы между этими состояниями?
5. Какие существуют виды контроля систем?
6. Перечислите стадии использования системы с точки зрения контроля ее работы.
7. Что понимается под общей характеристикой надежности системы?
8. Определите физический смысл интенсивностей η , λ , σ , μ .
9. Напишите формулы для расчета вероятностей нахождения системы при различных стадиях ее использования.
10. Дайте определение коэффициента готовности системы в установившемся режиме.
11. Как подразделяются технические системы с точки зрения восстанавливаемости? Каким образом возможно их восстановление?
12. В какой последовательности происходит ремонт отказавших подсистем при недостатке бригад ремонта?

13. Какие методы повышения надежности систем вы знаете?
14. Как может осуществляться резервирование подсистем?
15. Какие подходы используются для нахождения характеристик систем? Поясните экспериментальный подход.
16. Напишите алгоритм получения характеристик систем при помощи аналитического подхода.

Литература:

1. Граф Ш., Гессель М. Схемы поиска неисправностей. Перевод с немецкого. Москва, Издательство Энергоатомиздат, 1989
2. Дружинин Г. В. Надежность автоматизированных систем. Издание третье, переработанное и дополненное. Москва, Издательство «Энергия», 1977
3. Ю. К. Беляев, В. А. Богатырев, В. В. Болотин и др.; Под ред. И. А. Ушакова. Надежность технических систем: Справочник. Москва, Издательство Радио и связь, 1985

Контрольные задания для СРС

1. *Структура сетей связи и особенности их функционирования*
2. *Методы надежностного анализа сетей*
3. *Методы обеспечения требуемого уровня надежности сетей*

Методические указания к лабораторной работе №4

Тема: Исследование вычислительных возможностей ЛВС и кластера.

Цель работы : Сравнительный анализ решения одного класса задач в ЛВС типа Ethernet, Fast Ethernet и Gigabit Ethernet, в Кластере и в МВС с Распределенной и Общей памятью.

Содержание работы:

Ethernet

Ethernet и Fast Ethernet – это самые распространенные на сегодняшний день стандарты локальных сетей. В сетях Ethernet используется метод доступа к среде передачи данных, называемый методом коллективного доступа с опознаванием несущей и обнаружением коллизий (CSMA/CD). Этот метод применяется исключительно в сетях с логической шиной. Все компьютеры такой сети имеют непосредственный доступ к общей шине, поэтому она может быть использована для передачи данных между любыми двумя узлами сети. Одновременно все компьютеры сети имеют возможность немедленно (с учетом задержки распространения сигнала по физической среде) получить данные, которые любой из компьютеров начал передавать в общую шину.

Все данные, передаваемые по сети, помещаются в кадры определенной структуры и снабжаются уникальным адресом станции назначения. Чтобы получить возможность передавать кадр, станция должна убедиться, что разделяемая среда свободна. Если среда свободна, то узел имеет право начать передачу кадра. Кадр всегда сопровождается *преамбулой*, которая состоит из 7 байт, состоящих из значений 10101010, и 8-го байта, равного 10101011. Преамбула нужна для вхождения приемника в побитовый и побайтовый синхронизм с передатчиком. Все станции распознают факт передачи кадра, и та станция, которой адресован этот кадр, записывает его содержимое в свой внутренний буфер, обрабатывает полученные данные, а затем посылает по кабелю кадр-ответ. Адрес станции-источника содержится в исходном кадре, поэтому станция-получатель знает, кому нужно послать ответ. Если во время передачи, другой узел попытается начать передачу, то обнаружит, что среда занята и будет ожидать окончания передачи.

После окончания передачи кадра все узлы сети обязаны выдержать технологическую паузу в 9,6 мкс. Эта пауза, называемая также *меж кадровым интервалом*, нужна для приведения сетевых адаптеров в исходное состояние, а также для предотвращения монопольного захвата среды одной станцией. После окончания технологической паузы узлы имеют право начать передачу своего кадра, так как среда свободна.

При описанном подходе возможна ситуация, когда две станции одновременно пытаются передать кадр данных по общей среде. Механизм прослушивания среды и пауза между кадрами не гарантируют от возникновения

такой ситуации, когда две и более станции одновременно решают, что среда свободна, и начинают передавать свои кадры.

Или же другой случай, одна станция уже начала передачу, а до другого сигнала еще не дошли. Говорят, что при этом происходит *коллизия*, так как содержимое обоих кадров сталкивается на общем кабеле и происходит искажение информации. Коллизия – это нормальная ситуация в работе сетей Ethernet. Чтобы корректно обработать коллизию, все станции одновременно наблюдают за возникающими на кабеле сигналами. Если передаваемые и наблюдаемые сигналы отличаются, то фиксируется *обнаружение коллизии*. Для увеличения вероятности скорейшего обнаружения коллизии всеми станциями сети станция, которая обнаружила коллизию, прерывает передачу своего кадра и усиливает ситуацию коллизии посылкой в сеть специальной последовательности из 32 бит, называемой *jam-последовательностью*. После этого обнаружившая коллизию передающая станция обязана прекратить передачу и сделать паузу в течении короткого случайного интервала времени. Затем она может снова предпринять попытку захвата среды и передачи кадра. Случайная пауза выбирается по следующему алгоритму:

$$\text{Пауза} = L \times (\text{интервал отсрочки}),$$

где интервал отсрочки равен 512 битовым интервалам (для скорости 10 Мбит/с величина битового интервала равна 0,1 мкс или 100 нс);

L – любое целое число, выбранное с равной вероятностью из диапазона $[0, 2^N]$, где N – номер повторной попытки передачи данного кадра: 1, 2, ..., 10. После 10-й попытки интервал, из которого выбирается пауза, не увеличивается. Таким образом, случайная пауза может принимать значения от 0 до 52,4 мс. Если 16 последовательных попыток передачи кадра вызывают коллизию, то передатчик должен прекратить попытки и отбросить этот кадр.

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

Четкое распознавание коллизий всеми станциями сети является необходимым условием корректной работы сети Ethernet. Для надежного распознавания коллизий должно выполняться следующее условие:

$$T_{\min} \geq PDV,$$

где T_{\min} – время передачи кадра минимальной длины, PDV – время, за которое сигнал коллизии успевает распространиться до самого дальнего узла сети. Так как в худшем случае сигнал должен пройти дважды между наиболее удаленными друг от друга станциями сети (в одну сторону проходит неискаженный сигнал, а на обратном пути распространяется уже искаженный коллизией сигнал), то это время называется *временем двойного оборота (Path Delay Value)*. При выполнении этого условия передающая станция должна

успевать обнаружить коллизию, которую вызвал переданный кадр, еще до того, как она закончит передачу этого кадра. Очевидно, что выполнение этого условия зависит, с одной стороны, от длины минимального кадра и пропускной способности сети, а с другой стороны, от длины кабельной системы сети и скорости распространения сигнала в кабеле. Все параметры протокола Ethernet подобраны таким образом, чтобы при нормальной работе узлов сети коллизии всегда четко распознавались.

В стандарте Ethernet принято, что минимальная длина поля данных кадра составляет 46 байт (что вместе со служебными полями дает минимальную длину кадра 64 байт, а вместе с преамбулой – 72 байт или 576 бит). Отсюда может быть определено ограничение на расстояние между станциями.

Итак, в 10-мегабитном Ethernet время передачи кадра минимальной длины (576 бит) равно 575 битовых интервалов, следовательно время двойного оборота должно быть меньше $575 \times 100\text{нс} = 57\,500\text{нс}$ или 57,5 мкс. Расстояние, которое сигнал может пройти за это время, зависит от типа кабеля и для толстого коаксиального кабеля равно примерно 13 280 м. Учитывая, что за это время сигнал должен пройти по линии связи дважды, расстояние между двумя узлами не должно быть больше 6 635 м. В стандарте величина этого расстояния выбрана существенно меньше, с учетом других, более строгих ограничений и равняется 2 500 м.

Fast Ethernet

Fast Ethernet – дальнейшее развитие технологии Ethernet. Основные общие характеристики и отличия (нас интересующие) следующие:

1. Производительность 100 Мбит/с;
2. Форматы кадров технологии Fast Ethernet не отличаются от форматов кадров технологии 10-мегабитного Ethernet;
3. Межкадровый интервал равен 0,96 мкс, а битовый равен 10 нс. Все временные алгоритмы доступа (интервал отсрочки, время передачи кадра минимальной длины и т.п.), измеренные в битовых интервалах, остались прежними.
4. Сокращение диаметра сети до 210 м.

Gigabit Ethernet

Gigabit Ethernet – дальнейшее развитие технологий Ethernet и Fast Ethernet. Основные общие характеристики и отличия (нас интересующие) следующие:

1. Производительность 1000 Мбит/с;
2. Форматы кадров технологии Gigabit Ethernet не отличаются от форматов кадров технологии 10-мегабитного и 100-мегабитного Ethernet, но при этом максимальный диаметр сети равен всего 25 м;

3. Межкадровый интервал равен 0,096 мкс, а битовый равен 1 нс. Все временные алгоритмы доступа (интервал отсрочки, время передачи кадра минимальной длины и т.п.), измеренные в битовых интервалах, остались прежними.

4. Сокращение диаметра сети до 25 м.

Для достижения большего диаметра сети (до 200м) разработчики стандарта увеличили минимальный размер кадра (без учета преамбулы) с 64 до 512 байт или до 4096 битовых интервалов.

	Ethernet	Fast Ethernet	Gigabit Ethernet
Битовая скорость	10 Мбит/с	100 Мбит/с	1000 Мбит/с
Битовый интервал (б.и.)	100 нс	10 нс	1 нс
Интервал отсрочки	512 б.и.	512 б.и.	512 б.и.
Межкадровый интервал	9,6 мкс	0,96 мкс	0,096 мкс
Максимальное число попыток передачи	16	16	16
Максимальное число возрастания диапазона паузы	10	10	10
Длина jam-последовательности	32 бита	32 бита	32 бита
Максимальная длина кадра (без преамбулы)	1518 байт	1518 байт	1518 байт
Минимальная длина кадра (без преамбулы)	64 байт	64 байт	64 (512) байт
Длина преамбулы	64 бит (8 байт)	64 бит (8 байт)	64 бит (8 байт)
Минимальная длина случайной паузы после коллизии	0 б.и.	0 б.и.	0 б.и.
Максимальная длина случайной паузы после коллизии	524 000 б.и.	524 000 б.и.	524 000 б.и.
Максимальное расстояние между станциями сети	2500 м	210м	25м (200м)
Максимальное число станций в сети	1024	∞	∞

Кластер

В данной лабораторной работе рассматривается и сравнивается не только работа локальных вычислительных сетей (ЛВС) стандарта Ethernet, Fast Ethernet и Gigabit Ethernet, но и кластера, работающего на сетях тех же стандартов.

В ЛВС все компьютеры соединены друг с другом с помощью общей шины, и компьютеры обмениваются между собой необходимой информацией только когда шина свободна. При большом количестве компьютеров и больших объемах передаваемой информации загруженность сети очень высока, и не всегда удастся одному компьютеру передать необходимую информацию для другого с первой попытки. В результате чего производительность всей системы резко падает. Кластер (в нашем случае) представляет из себя такой же набор компьютеров, каждый из которых подключен сразу к двум ЛВС – вертикальной и горизонтальной – как на шахматной доске, клетки – компьютеры, а каждая клетка входит в вертикальный столбец (буква) и в горизонтальную строку (цифра). Передача данных в кластере осуществляется следующим образом (на примере шахматной доски): чтобы передать информацию от компьютера В2 компьютеру D7, компьютер В2 выбирает из 2-х маршрутов: либо сначала от В2 к В7, а затем от В7 к D7, либо от В2 к D2, а затем от D2 к D7. Таким образом, в ЛВС содержащей 64 компьютера на одной шине сразу все 64 компьютера, а в кластере на любой шине всего 8 компьютеров. Передача данных в кластере осуществляется сразу по двум маршрутам (какой потребует меньше времени) и в два этапа – сначала от компьютера-источника к компьютеру-посреднику, а затем от компьютера-источника к компьютеру-цели. Этим достигается меньшее время передачи данных в кластере, меньшее количество коллизий и более высокая производительность всей системы.

Многопроцессорные Вычислительные Системы.

Данные системы были подробно рассмотрены в предыдущих лабораторных работах, поэтому вспомним о них вкратце.

МВС с Общей Памятью.

В МВС с *Общей Памятью* все процессоры имеют разделяемую общую память с единым адресным пространством. Соединение общей памяти со всеми вычислительными модулями (ВМ) осуществляется посредством шины. К шине подключены также внешние устройства ввода/вывода.

Все процессоры обладают одинаковыми возможностями доступа к общей памяти и внешним устройствам и функционируют под управлением одной операционной системы (ОС).

Обращение к памяти происходит с помощью команд “чтение” (load) и “запись” (store). Сама память строится по модульному принципу. Этот принцип предусматривает возможность введения на каждый модуль памяти своей шины.

После решения очередной подзадачи любой процессор сначала по шине передает всю информацию в общую память. После окончания передачи, данные для решения следующей подзадачи, он получает из общей памяти по шине передачи данных.

МВС с Распределенной Памятью.

В МВС с *Распределенной Памятью* все процессоры имеют свою собственную локальную память, время доступа к которой существенно меньше времени доступа к памяти другого процессора по шине. Если для решения подзадачи процессор имеет в своей локальной памяти все необходимые данные, то он незамедлительно начинает ее решение, используя свою собственную память. Если же данных не хватает, то он обращается к памяти другого процессора, где лежит необходимая информация и по общей шине получает эти данные.

Модели задач

Задачи представлены в виде *Базы данных* и рассредоточены по 3 степеням связности основных в них элементов – узлов. Это *задачи с сильной, средней и слабой степенью связности*.

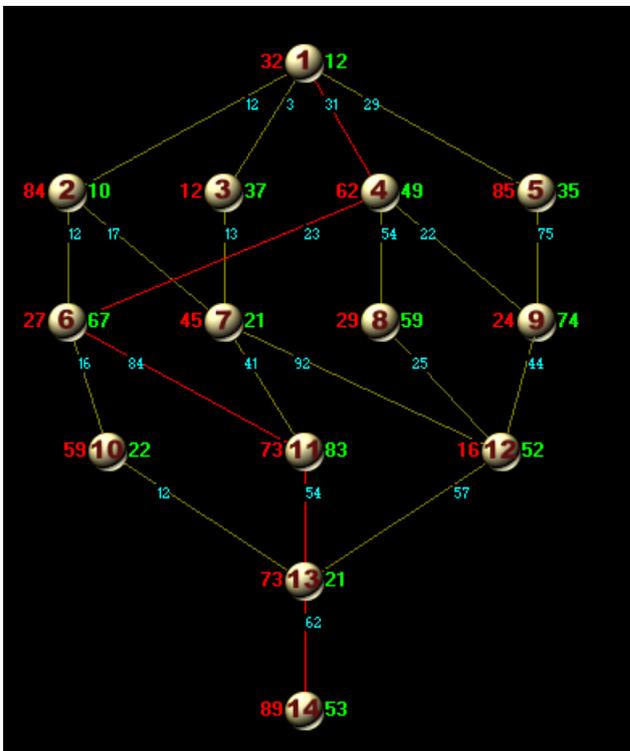
В качестве моделей задач в данном проекте используются однонаправленные (т.е. не имеющие обратных связей) графы. Каждая вершина этого графа – это подзадача, которая характеризуется количеством машинных тактов, необходимых для ее решения и объемом машинного кода самой подзадачи (в байтах). Так же, для каждой вершины определены входящие и исходящие ветви. Нагрузка этих ветвей определяет объем передаваемых данных. Входящие ветви определяют данные, необходимые для начала решения узла; исходящие ветви характеризуют данные, полученные в результате решения узла.

Данные о графе хранятся в специальном файле с расширением *.grf для ЛВС кластера и *.dat для МВС с общей и распределенной памятью. Информация в файле распределена следующим образом: в первой строке файла указано количество вершин в графе. Далее, в каждой строке указано, с какими вершинами связана данная вершина, номер которой совпадает с номером строки, и указан объем передаваемых данных по этим связям. В предпоследней строке содержится информация о количестве машинных тактов, необходимых для решения вершины. В последней строке указан размер выполняемого кода подзадачи.

```

14
0 12 3 31 29 0 0 0 0 0 0 0 0
0 0 0 0 12 17 0 0 0 0 0 0 0
0 0 0 0 0 13 0 0 0 0 0 0 0
0 0 0 0 0 23 0 54 22 0 0 0 0
0 0 0 0 0 0 0 75 0 0 0 0 0
0 0 0 0 0 0 0 0 16 84 0 0 0
0 0 0 0 0 0 0 0 0 41 92 0 0
0 0 0 0 0 0 0 0 0 0 25 0 0
0 0 0 0 0 0 0 0 0 0 44 0 0
0 0 0 0 0 0 0 0 0 0 0 12 0
0 0 0 0 0 0 0 0 0 0 0 54 0
0 0 0 0 0 0 0 0 0 0 0 57 0
0 0 0 0 0 0 0 0 0 0 0 62 0
0 0 0 0 0 0 0 0 0 0 0 0 0
12 10 37 49 35 67 21 59 74 22 83 52 21 53
32 84 12 62 85 27 45 29 24 59 73 16 73 89

```



Время передачи данных по шинам в МВС с Общей Памятью и Распределенной Памятью существенно меньше, времени передачи в ЛВС, поэтому размер выполняемого кода подзадачи для МВС с Общей Памятью и Распределенной Памятью не существенен и он в этих системах не учитывается – считается, что вся необходимая информация передается за указанное количество машинных тактов (в МВС сама операционная система управляет работой МВС, а в ЛВС каждый компьютер помимо данных получает еще и управляющий код для выполнения операции).

Порядок выполнения работы

Для работы с программой необходимо запустить файл *LaboratoryWork_7.exe*. Программа предложит ответить на 5 вопросов для проверки теоретической подготовки пользователя. При ответе менее чем на 3 вопроса программа предложит еще раз ознакомиться с теоретическим разделом и закончит свою работу. После этого пользователю при запуске опять будет предложено ответить на 5 вопросов, но уже отличных от предыдущих. При ответе на 3 и более вопроса программа считает, что пользователь готов к выполнению Лабораторной работы и предложит 3 варианта дальнейших действий на выбор: *Запуск Моделирования ЛВС и кластера*, *Запуск МВС с Распределенной памятью* и *Запуск МВС с Общей памятью*.

Процесс работы с Моделями МВС с Распределенной и Общей памятью ничем не отличается от рассмотренных моделей в Лабораторной работе №3 и №4 по курсу *Вычислительные системы и сети*, за исключением того, что дополнительным исходным параметром является тактовая частота процессора, а во временной диаграмме показывается не только время решения в *Машиных тактах*, но и в *наносекундах*.

При выборе *ЛВС и кластера*, пользователю сначала предлагается выбрать следующие параметры:

1. количество рабочих станций (от 1 до 1024 с шагом 1);
2. тактовую частоту процессоров станций (от 100 до 2000 МГц с шагом 50);
3. номинальную скорость передачи данных по сети (от 1 до 125 Мб/сек с шагом 1);
4. расстояние между станциями (от 1 до 2500 м с шагом 5);
5. стратегию выбора вершин.

Затем пользователь должен выбрать файл задачи, после чего появится изображение графа.

Далее пользователь может выбрать тип моделируемой системы (ЛВС или кластер) и нажать на кнопку *Моделирование*, для начала процесса моделирования работы системы. В результате на экран выводится время работы и количество коллизий моделируемой системы при решении выбранной задачи с заданными параметрами самой системы.

Варианты заданий

№	Кол-во рабочих станций	Тактовая частота	Скорость передачи данных	Расстояние между станциями	Стратегия выбора вершин	Степень связности задач
1	1-32, степень 2	100	1	1	Отсутствует	Слабая
2	1-32, степень 2	200	5	10	Максимальное время выполнения	Средняя

3	1-32, степень 2	300	10	20	Минимальное время выполне- ния	Сильная
4	1-32, степень 2	400	20	30	Максимальное количество по- следователей	Слабая
5	1-32, степень 2	500	30	40	Принадлежность критическому пути	Средняя
6	1-32, степень 2	600	40	50	Отсутствует	Сильная
7	1-32, степень 2	700	50	60	Максимальное время выполне- ния	Слабая
8	1-32, степень 2	800	60	70	Минимальное время выполне- ния	Средняя
9	1-32, степень 2	900	70	80	Максимальное количество по- следователей	Сильная
10	1-32, степень 2	1000	80	90	Принадлежность критическому пути	Слабая

Содержание отчёта

Результаты исследования необходимо оформить в виде таблиц и графиков. Допускается для построения графиков использовать табличный процессор EXCEL. Пример оформления приведен в табл 7.

Таблица 1. Пример оформления результатов исследования.

Кол-во станций: 1, локальная сеть

Номер опыта	1	2	3	4	5									20	С р е д н е е з н а ч е н и е
Кол-во коллизий															

Время моделтрования, нс																		
--------------------------------	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

Кол-во станций: 1, кластер

Номер опыта	1	2	3	4	5												20	Среднее значение
Кол-во коллизий																		
Время моделтрования, нс																		

.....

Кол-во станций: 32, локальная сеть

Номер опыта	1	2	3	4	5												20	Среднее значение
Кол-во коллизий																		
Время моделтрования, нс																		

Кол-во станций: 32, кластер

Номер опыта	1	2	3	4	5												20	Среднее значение
Кол-во коллизий																		
Время моделтрования, нс																		

На графики вывести зависимости кол-ва коллизий и времени моделирования (средние значения) от числа рабочих станций для ЛВС и кластера.

Контрольные вопросы

1. Дайте описание метода доступа к моноканалу CSMA/CD.
2. Охарактеризуйте особенности случайных методов доступа.
3. Структура кадра ETHERNET. Назначение его полей.
4. Что такое коллизия. Причины ее возникновения.
5. Опишите применяемый в ETHERNET алгоритм отсрочки доступа к моноканалу в случае возникновения коллизии.
6. Охарактеризуйте особенности известных методов захвата канала.
7. Назовите основные характеристики ЛВС со случайным доступом.
8. Назовите преимущества и недостатки рассмотренного метода доступа.
9. Что такое кластер?
10. Перечислите типы кластеров.
11. Связи процессоров в кластерной системе.

Литература

1. В.В.Воеводин, Вл.В.Воеводин - "Параллельные вычисления", издательство "БХВ", 2002.
2. В. Корнеев В., "Современные микропроцессоры", издательство "Нолидж", 2003.
3. Э. Таненбаум, "Архитектура компьютера", СПб, издательство «Питер», 2002.
4. В.Г.Олифер, Н.А.Олифер Компьютерные сети, издательство «Питер», 2000.
5. К. Хамахер, З. Вранешич, С. Заки, Организация ЭВМ. СПб, издательство «Питер», 2003 г.

Контрольные задания для СРС

1. Утром пчелиная матка созывает рабочих пчел и сообщает им, что сегодня им нужно собрать нектар ноготков. Рабочие пчелы вылетают из улья и летят в разных направлениях в поисках ноготков. Что это за система, SIMD или MIMD?
2. Рассмотрим мультипроцессор с общей шиной. Что произойдет, если два процессора попытаются получить доступ к глобальной памяти в один и тот же момент?
3. Предположим, что к общей шине подсоединено n процессоров. Вероятность того, что один из процессоров пытается использовать шину в данном цикле, равна p . Какова вероятность, что:
 - 1) шина свободна (0 запросов);
 - 2) совершается один запрос;
 - 3) совершается более одного запроса.

6 Тематический план самостоятельной работы студента с преподавателем

Наименование темы СРСП	Цель занятия	Форма проведения занятия	Содержание задания	Рекомендуемая литература
Техническая реализация и дополнительные функции коммутаторов	Углубление знаний по данной теме	Устный опрос, обсуждение докладов	Подготовка ответов на контрольные вопросы по данной теме	В.Г.Олифер, Н.А.Олифер Компьютерные сети, издательство "Питер", 2000
Принципы объединения сетей на основе протоколов сетевого уровня.	Углубление знаний по данной теме	Устный опрос, обсуждение докладов	Подготовка ответов на контрольные вопросы по данной теме	В.Г.Олифер, Н.А.Олифер Компьютерные сети, издательство "Питер", 2000
Адресация в IP-сетях	Углубление знаний по данной теме	Устный опрос, обсуждение докладов	Подготовка ответов на контрольные вопросы по данной теме	В.Г.Олифер, Н.А.Олифер Компьютерные сети, издательство "Питер", 2000
Протокол IP	Углубление знаний по данной теме	Устный опрос, обсуждение докладов	Подготовка ответов на контрольные вопросы по данной теме	В.Г.Олифер, Н.А.Олифер Компьютерные сети, издательство "Питер", 2000
Протоколы маршрутизации в IP-сетях	Углубление знаний по данной теме	Устный опрос, обсуждение докладов	Подготовка ответов на контрольные вопросы по данной теме	В.Г.Олифер, Н.А.Олифер Компьютерные сети, издательство "Питер", 2000
Основные характеристики маршрутизаторов и концентраторов	Углубление знаний по данной теме	Устный опрос, обсуждение докладов	Подготовка ответов на контрольные вопросы по данной теме	В.Г.Олифер, Н.А.Олифер Компьютерные сети, издательство "Питер", 2000

Наименование темы СРСП	Цель занятия	Форма проведения занятия	Содержание задания	Рекомендуемая литература
Глобальные сети. Основные понятия и определения	Углубление знаний по данной теме	Устный опрос, обсуждение докладов	Подготовка ответов на контрольные вопросы по данной теме	В.Г.Олифер, Н.А.Олифер Компьютерные сети, издательство "Питер", 2000
Глобальные связи на основе выделенных линий	Углубление знаний по данной теме	Устный опрос, обсуждение докладов	Подготовка ответов на контрольные вопросы по данной теме	В.Г.Олифер, Н.А.Олифер Компьютерные сети, издательство "Питер", 2000
Глобальные связи на основе сетей с коммутацией каналов	Углубление знаний по данной теме	Устный опрос, решение задач	Подготовка ответов на контрольные вопросы по данной теме	В.Г.Олифер, Н.А.Олифер Компьютерные сети, издательство "Питер", 2000
Компьютерные глобальные сети с коммутацией пакетов	Углубление знаний по данной теме	Устный опрос, решение задач	Подготовка ответов на контрольные вопросы по данной теме	В.Г.Олифер, Н.А.Олифер Компьютерные сети, издательство "Питер", 2000
Удаленный доступ	Углубление знаний по данной теме	Устный опрос, решение задач	Подготовка ответов на контрольные вопросы по данной теме	В.Г.Олифер, Н.А.Олифер Компьютерные сети, издательство "Питер", 2000

7 Материалы для контроля знаний студентов в период рубежного контроля и итоговой аттестации

7.1 Тематика письменных работ по дисциплине

Тематика рефератов

1. Суперкомпьютеры. Архитектура. Области применения.
2. Обзор микроархитектур современных десктопных процессоров.
3. Системы охлаждения в современных компьютерах.
4. Завод по производству памяти.
5. Корпорация Intel. Прошрое. Настоящее. Будущее.
6. Развитие индустрии жестких дисков за 50 лет (1956-2006).
7. Обзор устройств и технологий хранения данных на магнитной ленте.
8. Технологии защиты от ударов и тряски в новейших жестких дисках.
9. Новейшие технологии памяти(NRAM, FRAM, MRAM, CRAM и пр.).
10. Современная терминология 3D графики(A-Z).
11. 3D-технологии в играх(2006-2007).
12. Параметры современных ЖК-мониторов.
13. Технология LEP – мониторы, сделанные из пластика.
14. Выбор монитора с точки зрения безопасности для здоровья.
15. Оптические системы хранения : сегодня и завтра.
16. Принтер для офиса — критерии выбора.
17. Современные способы компьютерной печати.
18. Цифровая камера: анатомия и основные параметры.
19. Электропитание компьютера(БП, ИБП).
20. Мобильные компьютеры: ноутбуки и планшетные компьютеры.
21. КПК(PDA).
22. Периферия компьютера(клавиатура, мышь, трекбол, игровые устройства и тп.)
23. Модемы.
24. Технология ADSL .
25. Bluetooth: технология и ее применение

7.2 Вопросы (тестовые задания) для самоконтроля

Основные вопросы при рубежном контроле знаний

1. Чем можно объяснить тот факт, что глобальные сети появились раньше, чем локальные?
2. Поясните использование термина «сеть» в следующих предложениях:
 - сеть нашего предприятия включает *сеть* Ethernet и *сеть* Token Ring;
 - маршрутизатор - это устройство, которое соединяет *сети*;

- чтобы получить выход в Internet, необходимо получить у поставщика услуг Internet номер *сети*;
 - в последнее время *IP-сети* становятся все более распространенными;
 - гетерогенность корпоративной *сети* приводит к тому, что на первый план часто выходит проблема согласования *сетей*.
3. Всякое ли приложение, выполняемое в сети, можно назвать сетевым?
 4. Что общего и в чем отличие между взаимодействием компьютеров в сети и взаимодействием компьютера с периферийным устройством?
 5. Как распределяются функции между сетевым адаптером и его драйвером?
 6. Поясните значения терминов «клиент», «сервер», «редиректор».
 7. Назовите главные недостатки полносвязной топологии, а также топологий типа общая шина, звезда, кольцо.
 8. Какую топологию имеет односегментная сеть Ethernet, построенная на основе концентратора: общая шина или звезда?
 9. Какие из следующих утверждений верны:
 - A. разделение линий связи приводит к повышению пропускной способности канала;
 - B. конфигурация физических связей может совпадать с конфигурацией логических связей;
 - C. главной задачей службы разрешения имен является проверка сетевых имен и адресов на допустимость;
 - D. протоколы без установления соединений называются также дейтаграммными протоколами.
 10. Определите функциональное назначение основных типов коммуникационного оборудования - повторителей, концентраторов, мостов, коммутаторов, маршрутизаторов.
 11. В чем отличие логической структуризации сети от физической?
 12. Если все коммуникационные устройства в приведенном ниже фрагменте сети (рис. 1.34) являются концентраторами, то на каких портах появится кадр, если его отправил компьютер А компьютеру В? Компьютеру С? Компьютеру D?

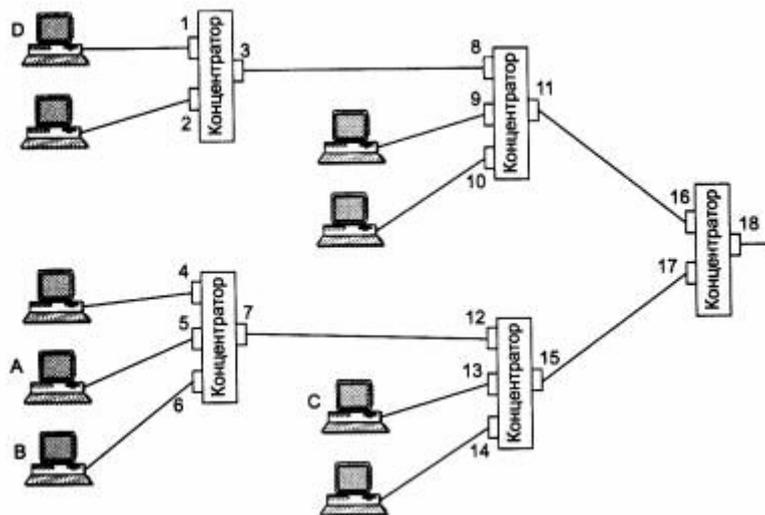


Рис. 1.34. Фрагмент сети

13. Если в предыдущем упражнении изменить условия и считать, что все коммуникационные устройства являются коммутаторами, то на каких портах появится кадр, посланный компьютером А компьютеру В? Компьютеру С? Компьютеру D?
14. Что такое «открытая система»? Приведите примеры закрытых систем.
15. Поясните разницу в употреблении терминов «протокол» и «интерфейс» применительно к многоуровневой модели взаимодействия устройств в сети.
16. Что стандартизует модель OSI?
17. Что стандартизует стек OSI?
18. Почему в модели OSI семь уровней?
19. Дайте краткое описание функций каждого уровня и приведите примеры стандартных протоколов для каждого уровня модели OSI.
20. Являются ли термины «спецификация» и «стандарт» синонимами?
21. Какая организация разработала основные стандарты сетей Ethernet и Token Ring?
22. Из приведенной ниже последовательности названий стандартных стеков коммуникационных протоколов выделите названия, которые относятся к одному и тому же стеку: TCP/IP, Microsoft, IPX/SPX, Novell, Internet, DoD, NetBIOS/SMB, DECnet.
23. В чем состоит отличие локальных сетей от глобальных на уровне служб? На уровне транспортной системы?
24. Назовите наиболее часто используемые характеристики производительности сети?
25. Что важнее для передачи мультимедийного трафика: надежность или синхронность?
26. Поясните значение некоторых сетевых характеристик, названия которых помещены в англоязычном написании:
 - availability;
 - fault tolerance;

- security;
- extensibility;
- scalability;
- transparency.

7.3 Экзаменационные билеты (тесты)

Тесты

1. Области применения МВС

- А) расчеты, требующие значительных вычислительных ресурсов
- В) - поиск в Интернете
- С) - поддержка работы электронной почты

2. Производительность МВС это:

- А) - количество операций, производимых за единицу времени
- В) - количество байтов информации, переданных в единицу времени
- С) - число импульсов, генерируемых в единицу времени

3. Пиковая производительность системы определяется:

- А) - временем выполнения тестовых задач
- В) - временем выполнения реальных задач
- С) - произведением пиковой производительности одного процессора на число процессоров в системе

4. Единица производительности МВС:

- А) - Мегагерц
- В) - Flops
- С) - MIPS

6. Классификация архитектур вычислительных систем Флина основывается

...

- А) - на понятии потока, под которым понимается последовательность элементов, команд или данных, обрабатываемая процессором.
- В) - на описании возможностей параллельной и конвейерной обработки информации вычислительной системой.
- С) - на выделении типичных способов компоновки вычислительных систем на основе фиксированного числа базисных блоков:

устройства управления, арифметико-логического устройства, памяти команд и памяти данных.

7. Какая из приведенных ниже архитектур отсутствует в классификации Флинна?

- A)- SIND
- B)- MISD
- C)- SIMD

8. Главная особенность архитектуры NUMA?

- A)- Неоднородный доступ к памяти
- B)- Сверхвысокая производительность
- C)- Наличие векторно-конвейерных процессоров

9. Вычислительные машины с какой архитектурой наиболее дешевы?

- A)- Кластерные системы
- B)- Параллельная архитектура с векторными процессорами
- C)- Массивно-параллельная архитектура

10. Система, главной особенностью является наличие общей физической памяти, разделяемой всеми процессорами называется ...

- A)- SMP
- B)- MPP
- C)- PVP

11. Кэши являются когерентными, если ...

- A)- все центральные процессоры получают одинаковые значения одних и тех же переменных в любой момент времени.
- B)- каждый следующий процессор получает данные только после обработки их предыдущим.
- C)- их объем совпадает.

12. Критическим параметром, влияющим на величину производительности кластерной системы, является ...

- A)- расстояние между процессорами (скорость обмена информацией между процессорами)

- В)- количество процессоров
- С)- быстродействие процессоров

13. Что в большей мере определяет производительность кластерной системы?

- А)- Способ соединения процессоров друг с другом.
- В)- Тип используемых в ней процессоров.
- С)- Операционная система.

14. Коммуникационной средой называется:

- А)- способ соединения процессоров между собой, с памятью и с внешними устройствами
- В) - организация процесса преобразования информации
- С) - совокупность устройств, реализующих арифметические и логические операции

15. Интерфейс SCI представляет собой ...

- А) - сетевую плату
- В) - комбинацию шины и локальной сети
- С) - коммуникационный процессор

16. Технология Myrinet основана на использовании ...

- А)- конвейерной обработки данных
- В)- скалярных процессоров
- С) - многопортовых коммутаторов

17. Технология Myrinet используется при построении ...

- А)- кластерных систем
- В)- VP систем
- С) - SMP систем

18. Узлы соединяются с помощью колец в коммуникационной технологии...

- А)- Myrinet
- В)- Raceway
- С)- SCI

19. Транспьютер представляет собой ...

- A) - кластерную систему
- B) - микροпроцессор
- C) - коммуникационную среду

20. Ассоциативный способ обработки данных предполагает

- A) - обработку только тех данных, которые удовлетворяют определенным критериям отбора
- B) - обработку всех данных
- C) - доступ к данным по указателям

21. Конвейерная технология предполагает ...

- A) - последовательную обработку команд
- B) - обработку команд, удовлетворяющих определенным критериям
- C) - обработку несколько команд одновременно.

22. Организация матричных процессов предполагает наличие

- A) - одного управляющего устройства и большого числа процессорных элементов, работающих параллельно
- B) - нескольких процессорных элементов, работающих последовательно
- C) - нескольких управляющих устройств, работающих параллельно

23. Биокомпьютинг это:

- A) - численное моделирование процессов, протекающих в биологических организмах
- B) - создание баз данных по вопросам биоинформатики
- C) - создание вычислительных систем на основе биологических объектов

24. Коммуникационные процессоры предназначены для:

- A) - оптимизации работы сети
- B) - конвейерной обработки данных
- C) - организации процесса преобразования информации

25. Процессоры баз данных предназначены для:

- создания пользовательских интерфейсов
- осуществления транзакций
- управления базами данных

26. Поточковые процессоры принадлежат к архитектуре:

- A) - SISD
- B) - MIMD
- C) - SIMD

27. Отличительной особенностью нейронных процессоров является...

- A) - принцип обработки информации
- B) - высокая тактовая частота
- C) - наличие когерентного кэша

28. Назовите два типа простых коммутаторов

- A) - с временным и пространственным разделением
- B) - параллельные и последовательные
- C) - цифровые и аналоговые
- D) - управляемые и неуправляемые

29. Другое название коммутаторов с временным разделением

- A) - мезонин
- B) - шина
- C) - колесо
- D) - кольцо

30. Какого алгоритма арбитража для коммутаторов с временным разделением не существует

- A) - статические приоритеты
- B) - динамические приоритеты
- C) - фиксированные временные интервалы
- D) - очередь FIFO
- E) - метод деревьев

31. Что является достоинством коммутаторов с пространственным разделением

- A) - высокая надежность
- B) - возможность одновременного контакта со всеми устройствами
- C) - простота соединения

32. Из каких коммутаторов состоит коммутатор Клоза, являющийся альтернативой прямоугольному коммутатору с $(m \times d)$ входами и $(m \times d)$ выходами

A) - формируется из двух каскадов коммутаторов: m коммутаторов $(d \times d)$ во входном каскаде, d коммутаторов $(m \times m)$ в выходном

B) - формируется из трех каскадов коммутаторов: m коммутаторов $(d \times d)$ во входном каскаде, m коммутаторов $(d \times d)$ в выходном и d промежуточных коммутаторов $(m \times m)$

C) - формируется из трех каскадов коммутаторов: d коммутаторов $(m \times m)$ во входном каскаде, m коммутаторов $(d \times d)$ в выходном и m промежуточных коммутаторов $(m \times d)$

D) - формируется из четырех каскадов коммутаторов: m коммутаторов $(d \times d)$ во входном каскаде, m коммутаторов $(d \times d)$ в выходном, d промежуточных коммутаторов $(m \times m)$ и m промежуточных коммутаторов $(d \times d)$

33. Основная особенность баньян-сетей

A) - состоит из трех каскадов простых коммутаторов

B) - состоит из коммутаторов 2×2

C) - существует только один путь от каждого входа к каждому выходу

D) - существует как минимум два пути от каждого входа к каждому выходу

34. Наиболее эффективный граф межмодульных связей с точки зрения организации обмена данными между вычислительными модулями

A) - полный

B) - связный

C) - остовное дерево

D) - планарный

35. Какова структура объединения блоков системы SPP1000

A) - кольцо (одномерный тор)

B) - двумерный тор

C) - куб

D) - гиперкуб

36. Размерность матрицы вычислительных узлов, составляющих структурный модуль системы МВС-100

A) - 2×2

B) - 4×4

- C) - 8 x 8
- D) - 12 x 12

37. Топология связей узлов в структурном модуле системы МВС-1000

- A) - кольцо (одномерный тор)
- B) - двумерный тор**
- C) - куб
- D) - гиперкуб

38. Система Cray T3E это...

- A) - Векторно-параллельная система
- B) - Масштабируемая параллельная система**
- C) - Кластерная система

39. Узлом кластера может считаться...

- A) - Отдельный процессор.
- B) - Процессор с локальной памятью.
- C) - Рабочая станция.**

40. Суперкластер SGI Altix 3000...

- A) - Основан на архитектуре глобальной разделяемой памяти.**
- B) - Это система массового параллелизма.
- C) - Это симметричная мультипроцессорная система.

41. Серверы Silicon Graphics POWER CHALLENGE работают под управлением ОС...

- A) - UNIX
- B) - IRIX**
- C) - SOLARIS

42. Системы с неоднородным доступом к памяти это...

- A) - SGI Origin2000**
- B) - NEC SX-6i
- C) - Hitachi SR8000

43. Пиковая производительность системы Cray T3E-1200 составляет...

- A) - Примерно 800 Гфлоп
- B) - Около 1 Тфлоп
- C) - Около 2 Тфлоп**

44. Базовый вычислительный блок системы МВС-1000 содержит...
- A) - 16 вычислительных модулей.
 - B) - 32 вычислительных модуля.**
 - C) - 64 вычислительных модуля.
45. Системы пакетной обработки строились на базе...
- A) Суперкомпьютера
 - B) Мэйнфрейма**
 - C) Персонального компьютера
46. Пробразом сети явились...
- A) Многотерминальные системы**
 - B) Многопроцессорные системы
 - C) Многопользовательские системы
47. Первые сети соединяли...
- A) Рабочие станции
 - B) Мэйнфрейма
 - C) Суперкомпьютеры**
48. Появление первых локальных сетей пришлось на..
- A) 60 годы 20 века**
 - B) 70 годы 20 века
 - C) 80 годы 20 века
49. Способ представления двоичных данных, применяемый в вычислительных сетях, называется...
- A) Демодуляция
 - B) Инфильтрация
 - C) Модуляция**
50. Способ организации физических связей в сети называется...
- A) Топонимика
 - B) Топография
 - C) Топология**
51. Конфигурация физических и логических связей сети...
- A) Всегда совпадают
 - B) Могут совпадать**
 - C) Никогда не совпадают
52. Способ организации физических связей сети, в котором каждый компьютер сети связан со всеми остальными называется...

- A) Полносвязным
- B) Общая шина
- C) Звезда

53. Способ организации физических связей сети, который получается из полносвязного, путем удаления некоторых возможных связей называется...

- A) Полносвязным
- B) Ячеистым
- C) Кольцевым

54. Способ организации физических связей сети, в котором компьютеры подключаются к одному коаксиальному кабелю называется...

- A) Общая шина
- B) Звезда
- C) Смешанным

55. Способ организации физических связей сети, в котором каждый компьютер подключается отдельным кабелем к общему устройству называется...

- A) Кольцевым
- B) Звезда
- C) Ячеистым

56. Способ организации физических связей сети, в котором данные передаются по кольцу от одного компьютера к другому называется...

- A) Звезда
- B) Концевым
- C) Кольцевым

57. Примером сети с разделяемыми линиями связи являются сети с топологией...

- A) Кольцо
- B) Общая шина
- C) Звезда

58. Разделяемые линии связи, используются в технологии..

- A) Ethernet
- B) Arcnet
- C) FDDI

59. Какое написание из нижеприведенных соответствует аппаратному адресу..

- A) 0081005e24a8
- B) ftp-arch1.ucl.ac.uk
- C) 192.168.0.1

60. Какое написание из нижеприведенных соответствует символному адресу...

- A) 230.141.111.67
- B) 00-1A-4D-25-FF-11
- C) www.3dnews.ru

60. Какое написание из нижеприведенных соответствует числовому составному адресу ...

- A) www.4click.kz
- B) 112.234.67.1
- C) 10000001000000000101111000100010010010101000

61. Физическая структуризация сети нужна для...

- A) Упрощения топологии сети
- B) Снятия ограничения на длину связи между узлами
- C) Снятия ограничения на виды трафика сети.

62. Устройство, применяемое для физической структуризации сети, называется...

- A) Коннектор
- B) Маршрутизатор
- C) Концентратор

63. Логическая структуризация сети нужна для...

- A) Перераспределения передаваемого трафика между различными физическими сегментами сети
- B) Снятия ограничения на количество узлов в сети
- C) Увеличения пропускной способности сети

64. Устройство, применяемое для логической структуризации сети, называется...

- A) Хаб
- B) Коммутатор
- C) Дешифратор

65. Нормализованные правила, определяющие последовательность и формат сообщений, которыми обмениваются сетевые компоненты, лежащие на одном уровне, но в разных узлах, называются...

- A) Протокол
- B) Интерфейс
- C) Спецификация

66. Правила, с помощью которых взаимодействуют модули, реализующие протоколы соседних уровней и находящиеся в одном узле называются...

- A) Протокол

- B) Интерфейс**
- C) Спецификация

67. Модель взаимодействия открытых систем скрывается под аббревиатурой...

- A) ISO
- B) OSI**
- C) SOI

68. В модели взаимодействия открытых систем...

- A) 7 уровней**
- B) 6 уровней
- C) 5 уровней

69. Физический уровень оперирует с...

- A) пакетами
- B) кадрами
- C) оптическим или электрическими сигналами**

70. Задачей канального уровня не является...

- A) проверка доступности среды передачи
- B) получение пакетов данных от вышележащего уровня**
- C) реализация механизмов обнаружения и коррекции ошибок

71. Сообщения сетевого уровня называются...

- A) пакетами**
- B) кадрами
- C) дейтаграммами

72. Примером протокола транспортного уровня является протокол...

- A) IP
- B) TCP**
- C) IPX

73. Задачей уровня представления является...

- A) организация сеансов обмена данными между оконечными машинами
- B) обеспечение преобразование данных**
- C) доступ приложений в сеть

74. К числу протоколов прикладного уровня не относится...

- A) FTP
- B) SMTP
- C) SSL**

75. Какой из уровней не является сетезависимым...
- A) физический
 - B) сеансовый**
 - C) сетевой
76. Основным протоколом глобальной сети Internet является...
- A) TCP/IP**
 - B) IPX
 - C) NetBIOS
77. Англиязычное сокращение WAN означает...
- A) городскую сеть
 - B) глобальную сеть**
 - C) локальную сеть
78. Интервал времени между возникновением запроса пользователя к какой-либо сетевой службе и получением ответа на этот запрос, это...
- A) время отклика
 - B) время запаздывания
 - C) время реакции**
79. Объем данных, переданных сетью или ее частью в единицу времени, это...
- A) Пропускная способность**
 - B) Скорость реакции
 - C) Задержка передачи
80. К задержке передачи чувствительны такие виды трафика как...
- A) передача файлов
 - B) передача голоса**
 - C) передача почты
81. availability означает...
- A) готовность**
 - B) безопасность
 - C) надежность
82. Расширяемость (extensibility) означает...
- A) что сеть позволяет наращивать количество узлов и протяженность связей в очень широких пределах, при этом производительность сети не ухудшается.
 - B) возможность сравнительно легкого добавления отдельных элементов сети (пользователей, компьютеров, приложений, служб), наращивания длины сегментов сети и замены существующей аппаратуры более мощной**

С) способность системы скрыть от пользователя отказ отдельных ее элементов

83. Возможность централизованно контролировать состояние основных элементов сети, это...

- А) Управляемость
- В) Администрируемость
- С) Контролируемость

84. Способность включать в себя самое разнообразное программное и аппаратное обеспечение, это

- А) Разнородность
- В) Совместимость
- С) Гетерогенность

85. В состав линии связи не входит...

- А) Модем
- В) Оконечное оборудование данных
- С) мультиплексор

86. Линии связи бывают...

- А) подводные
- В) кабельные
- С) подземные

87. Промежуточная аппаратура...

А) создает постоянный составной канала связи между двумя абонентами сети

В) непосредственно связывает компьютеры или локальные сети пользователя с линией связи

С) вырабатывает данные для передачи по линии связи

88. Линий связи не бывает...

- А) аналоговых
- В) цифровых
- С) смешанных

89. В аналоговых линиях применяется техника...

- А) частотного мультиплексирования
- В) амплитудного мультиплексирования
- С) фазового мультиплексирования

90. В аналоговых линиях применяется техника...

- А) опережающего мультиплексирования каналов
- В) временного мультиплексирования каналов
- С) пространственного мультиплексирования каналов

91. К основным характеристикам линий связи не относятся..
- A) амплитудно-частотная характеристика
 - B) полоса пропускания
 - C) защищенность
92. Амплитудно-частотная характеристика показывает...
- A) как затухает амплитуда синусоиды на выходе линии связи по сравнению с амплитудой на ее входе для всех возможных частот передаваемого сигнала
 - B) как возрастает амплитуда синусоиды на выходе линии связи по сравнению с амплитудой на ее входе для всех возможных частот передаваемого сигнала
 - C) как изменяется амплитуда синусоиды на выходе линии связи по сравнению с амплитудой на ее входе для всех возможных частот передаваемого сигнала
93. Полоса пропускания это непрерывный диапазон частот, для которого отношение амплитуды выходного сигнала ко входному превышает...
- A) 1
 - B) 0.5
 - C) 0.25
94. Полоса пропускания не зависит от...
- A) типа линии
 - B) протяженности
 - C) мощности передатчика
95. Если значимые гармоники выходят за границы полосы пропускания линии связи...
- A) сигнал будет значительно искажаться
 - B) сигнал не будет искажаться
 - C) сигнал будет неотличим от помех
96. Сколько единиц информации может нести сигнал, имеющий 4 различных состояния...
- A) 4
 - B) 2
 - C) 8
97. Количество изменений информационного параметра несущего периодического сигнала в секунду измеряется...
- A) Бодах
 - B) Герцах
 - C) Битах

98. Увеличение частоты несущего периодического сигнала...
- A) не влияет на пропускную способности линии связи
 - B) прямо пропорционально увеличению пропускной способности линии связи
 - C) увеличивает пропускную способность линии связи
99. Помехоустойчивость линии...
- A) определяет ее способность уменьшать уровень помех, создаваемых во внешней среде, на внутренних проводниках
 - B) определяют помехоустойчивость кабеля к внутренним источникам помех
 - C) вероятность искажения для каждого передаваемого бита данных
100. UTP это...
- A) Кабель на основе неэкранированной витой пары
 - B) Кабели на основе экранированной витой пары
 - C) коаксиальный кабель

Ключи правильных ответов

Номер вопроса	Правильный ответ
1	A
2	A
3	C
4	B
5	A
6	A
7	A
8	A
9	A
10	A
11	A
12	A
13	A
14	A
15	B
16	C
17	A
18	C
19	B
20	A
21	C
22	A
23	C
24	A
25	C
26	C
27	A
28	A
29	B
30	E
31	B
32	B
33	C
34	A
35	A
36	B
37	B
38	B
39	C
40	A
41	B

42	A
43	C
44	B
45	B
46	A
47	B
48	A
49	C
50	C
51	B
52	A
53	B
54	A
55	B
56	C
57	B
58	A
59	A
60	C
61	B
62	C
63	A
64	B
65	A
66	B
67	B
68	A
69	C
70	B
71	A
72	B
73	B
74	C
75	B
76	A
77	B
78	C
79	A
80	B
81	A
82	B
83	A
84	B

85	B
86	B
87	A
88	C
89	A
90	B
91	C
92	A
93	B
94	C
95	A
96	B
97	A
98	C
99	A
100	A

8 Методические указания для выполнения курсовой работы (проекта)

Не предусмотрен