

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ КЫРГЫЗСКОЙ РЕСПУБЛИКИ
БИШКЕКСКИЙ ТЕХНИЧЕСКИЙ КОЛЛЕДЖ

«Согласовано»
Директор _____ «_____»

«__» _____ 2023 г.



Рассмотрено на
Заседании методсовета
Протокол №__ от «__» ____ 2023 г
Председатель _____.

УЧЕБНЫЕ МАТЕРИАЛЫ

“ЭВМ и ПУ”

Специальность 230110 “Техническое обслуживание средств вычислительной техники и компьютерных сетей”

Трудоемкость –5 кредитов (150 ч)

Из них:

Аудиторная работа –76 ч.

Самостоятельная работа - 74 ч.

Составитель: Эрмат к.Зарина преподаватель _____

Оглавление

1. Глоссарий	3
2. Список сокращений	3
3. Рабочая учебная программа (силлабус)	4
4. Курс лекций (учебные материалы) по учебному модулю.....	6
Приложение 1 – Методические рекомендации по написанию и оформлению рефератов.....	75
Приложение 2 – Методические рекомендации по написанию опорного конспекта.....	80
Приложение 3– Перечень вопросов к экзамену.....	87

1. Глоссарий

Государственный образовательный стандарт – совокупность норм и правил, определяющая образовательный минимум содержания образовательных программ, базовые требования к качеству подготовки выпускников, предельно допустимую учебную нагрузку обучающихся;

Квалификация – формальный результат процесса оценки и признания, компетентным органом достижения индивидом результатов обучения по установленным стандартам;

Компетенция– заранее заданное социальное требование (норма) к образовательной подготовке ученика (обучаемого), необходимой для его эффективной продуктивной деятельности в определенной сфере;

Кредит – условная мера трудоемкости основной профессиональной образовательной программы;

Модуль– часть учебной дисциплины, имеющая определенную логическую завершенность по отношению к установленным целям и результатам обучения, воспитания;

Основная профессиональная образовательная программа– совокупность учебно-методической документации, регламентирующей цели, ожидаемые результаты, содержание и организацию реализации образовательного процесса по соответствующей специальности;

Профессиональный стандарт - основополагающий документ, определяющий в рамках конкретного вида профессиональной деятельности требования к ее содержанию и качеству и описывающий качественный уровень квалификации сотрудника, которому тот обязан соответствовать, чтобы по праву занимать свое место в штате любой организации, вне зависимости от рода ее деятельности;

Результаты обучения – компетенции, приобретенные в результате обучения по основной образовательной программе/модулю;

Учебный модуль - это логическое и приемлемое разделение труда в рамках профессии, работы или сферы труда, с четким началом и концом (закрытая и автономная единица как в отношении применения, так и оценки). Он представляет собой результат обучения и включает учебный материал, методы обучения и другие материалы, в том числе средства обучения и соответствующую техническую информацию;

Цикл дисциплин–часть образовательной программы или совокупность учебных дисциплин, имеющая определенную логическую завершенность по отношению к установленным целям и результатам обучения, воспитания.

2. Список сокращений

УМ – Учебный модуль;
ПЦК – Предметно-цикловая комиссия;
ТО- Техническое обслуживание ;
СРС - Самостоятельная работа студента;
ЭВМ- Электронно-вычислительная машина;
ПУ- Периферийные устройства

3. Рабочая учебная программа (силлабус)

1.Информация об учебном модуле

Наименование учебного модуля – «ЭВМ и ПУ»

Трудоемкость учебного модуля - 5 кредитов (150 часов)

Расписание учебного модуля -5 семестр – 4 часа в неделю аудиторных

2.Информация о преподавателе

Зарина Эрмат кызы преподаватель _____ -

3.Литература

Основная литература:

1. В.В. Косулин. – Казань: Казан. гос. энерг. ун-т, 2016. – 494 с.

Дополнительная литература:

Н. Н. Горнец, А. Г. Рошин. — М. : Издательский центр «Академия», 2012. — 240 с.

4.Пререквизиты учебного модуля:

Изучение данной дисциплины базируется на формировании системы теоретических знаний, практических умений в области Электронно-вычислительной системы и периферийных устройств.

5.Постреквизиты учебного модуля:

«Метод доступа к среде передачи, коммуникационные устройства, среда передачи, протокол, спецификация».

6.Описание учебного модуля

Цели обучения учебного модуля: обучение студентов принципам построения, назначения, функционирования и практического использования компьютера и эффективного управления процессами обработки данных в современных ЭВМ.

Результаты обучения учебного модуля: Знать назначение и функции компонентов сетей ЭВМ;

Сфера применения: взаимосвязь компьютерной техники как модель взаимодействия открытых систем, метод доступа к среде передачи;

Методы преподавания учебного модуля: лекция, практические занятия.

Методы изучения учебного модуля: выполнение практических заданий; самостоятельная работа студентов.

7. Тематический план по учебному модулю:

Темы	Всего	Аудиторная работа	Самостоятельная работа студента СРС
	часы	часы	часы
Раздел 1. Введение. Становление и эволюция вычислительной техники (ЭВМ)	4	2	2
Раздел 2. Представление числовой информации в ЭВМ	4	2	2
Раздел 3. Конструктивные узлы вычислительных машин. Счетчики	4+8	2	2
Раздел 4. Фон-неймановская вычислительная машина	4	4	2
Раздел 5. Организации шин вычислительной машины	6+8	6	2
Раздел 6. Принципы построения арифметико-логических устройств	6+4	4	2
Раздел 7. Системы памяти. Организация основной памяти	6	4	2
Раздел 8. Устройства основной памяти	8	4	2
Раздел 9. Принципы организации работы процессоров	8+8	4	4
Раздел 10. Периферийные устройства	6+2	4	2
Итого	150 (5 кредитов)	76	74

8. Политика курса:

Соблюдать общие этические нормы поведения в общественных местах; соблюдать правила поведения студентов в колледже; не пропускать занятий без уважительной причины; все пропуски занятий должны быть отработаны, включая пропуски по уважительным причинам (способ отработки определяет преподаватель); не опаздывать на урок более, чем на 5 минут; запрещается пользоваться телефоном во время урока (исключение составляет: использование телефона в качестве калькулятора; обучение в онлайн-режиме).

9. Оценивание:

- оценка «5 (*отлично*)» выставляется за глубокое и полное овладение содержанием учебного материала, в котором студент легко ориентируется; понятийным аппаратом, умение связать теорию с практикой, решать практические задачи, высказывать и обосновывать свои суждения. Отличная оценка предполагает грамотное, логическое изложение ответа (как в устной, так и в письменной форме), качественное внешнее оформление;

- оценка «**4(хорошо)**» выставляется студенту за полное освоение учебного материала, владение понятийным аппаратом, осознанное применение знаний для решения практических задач, грамотное изложение ответа, но в содержании и форме ответа имеются отдельные неточности;
- оценка «**3(удовлетворительно)**» выставляется студенту, который обнаружил знание и понимание основных положений учебного материала, но излагает его неполно, непоследовательно, допускает неточности в определении понятий, в применении знаний для решения практических задач, не умеет доказательно обосновывать свои суждения;
- оценка «**2(неудовлетворительно)**» выставляется студенту, имеющему разрозненные, бессистемные знания, не умеющему выделить главное и второстепенное, допускающему ошибки в определениях, понятиях, искажающему их смысл, беспорядочно и неуверенно излагающему материал (Приложение 2).

1. Курс лекций (учебные материалы) по учебному модулю

Расширенный тематический план по учебному модулю (календарный)

№ тем.	Наименование разделов и тем	Количество часов на тему			№ занятий	СРС задания по занятиям (урокам)	
		Всего	аудиторные	практич.			СРС
	Раздел 1. Введение. Становление и эволюция вычислительной техники (ЭВМ)						
1.1	Основные термины. Основные характеристики и требования современных ЭВМ	6	2		4	2(лк) 4(пр)	Урок 1 - Домашнее задание: Проработка конспекта пройденной темы.
1.2	Уровни детализации структуры вычислительной машины		1			1(лк)	Урок 2 - Домашнее задание: Проработка конспекта пройденной темы.
1.3	Эволюция средств вычислительной техники						
	Раздел 2. Представление числовой информации в ЭВМ						
2.1	Компоненты вычислительной сети и способы их взаимодействия.	5	1		4	1(лк) 4(пр)	Урок 3 - Домашнее задание: Ответить на контрольные вопросы
2.2	Топологии подключения. Логическая и физическая структуризация сети.	2	2			2(лк)	Урок 4 - Домашнее задание: Ответить на контрольные вопросы
	Раздел 3. Конструктивные узлы вычислительных машин. Счетчики						
3.1	Стандартизация компьютерных сетей.	10	2		2	2 (лк) 2(пр)	Урок 5 - Домашнее задание: Проработка конспекта пройденной темы
3.2	Эталонная модель взаимосвязи		2				Урок 6 - Домашнее задание:

	открытых систем.						Проработка конспекта пройденной
	Раздел 4. Фон-неймановская вычислительная машина						
4.1	Стеки коммуникационных протоколов.	5	1		4	1(лк) 4(пр)	Урок 7 - Домашнее задание: Проработка конспекта пройденной темы.
4.2	Линии связи.	1	1			1(лк)	Урок 8 - Домашнее задание: Проработка конспекта пройденной темы.
4.3	Методы и каналы передачи данных		1			1(лк)	Урок 9 - Домашнее задание: Проработка конспекта пройденной темы.
	Раздел 5. Организации шин вычислительной машины						
5.1	Стандарты и технологии локальных сетей	6	2		4	6 (лк)	Урок 10 - Домашнее задание: Ответить на контрольные вопросы
5.2	Локальная сеть Ethernet.		1			1(лк)	Урок 11 - Домашнее задание: Проработка конспекта пройденной темы.
5.3	Метод доступа к передающей среде и формат пакета сети Ethernet.		2			2(лк)	
5.4	Высокоскоростные варианты сети Ethernet		2			2(лк)	Урок 12 - Домашнее задание: Ответить на контрольные вопросы
5.5	Сеть Token Ring		2			2(лк)	Урок 13 - Домашнее задание: Проработка конспекта пройденной темы.
5.6	Распределенный волоконно-оптический интерфейс передачи данных FDDI		1			1(лк)	
	Раздел 6. Принципы построения арифметико-логических устройств						

6.1	Использование мостов в локальных сетях.	8	2		4	2 (лк) 4(пр)	Урок 14 - Домашнее задание: Проработка конспекта пройденной темы. Урок 15 - Домашнее задание: Ответить на вопросы заданные на рабочем листе.
6.2	Сетевые устройства	1	1			1(лк)	Урок 16 - Домашнее задание: Проработка конспекта пройденной темы.
	Раздел 7. Системы памяти. Организация основной памяти						
7.1	Использование мостов в локальных сетях.	3	1		2	1(лк) 2(пр)	Урок 17 - Домашнее задание: Проработка конспекта пройденной темы.
7.2	Управление сетью	1	1			1(лк)	Урок 18 - Домашнее задание: Ответить письменно: Ответить на вопросы заданные на рабочем листе
	Раздел 8. Устройства основной памяти						
8.1	Методика расчета конфигурации сети Ethernet	8		4	4	25(лк)	Урок 25 - Домашнее задание: Ответить письменно: Ответить на вопросы заданные на рабочем листе.
	Раздел 9. Принципы организации работы процессоров						
9.1	Эволюция компьютерных и телекоммуникационных технологий.	2		2			
	Раздел 10. Периферийные устройства						
10.1	От первых локальных сетей до современных сетевых технологий.	2		2			
	Итого:		Л (28)	Пр (8)	СРС (24)		

Раздел 1. Введение. Становление и эволюция вычислительной техники (ЭВМ)

Тема 1. СТАНОВЛЕНИЕ И ЭВОЛЮЦИЯ ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ Занятие 1 (лекция)

Ключевые вопросы: - признаки классификации сетей, особенности каждого типа.

Предыдущие знания обучающихся: Основы построения и архитектуры ЭВМ;

принципы построения, параметры и характеристики цифровых и аналоговых элементов ЭВМ

Результаты обучения: С помощью выбора методами элементной базы для построения различных архитектур вычислительных средств показать на практике.

Информация по теме:

В основе современных информационных технологий лежат аппаратные средства компьютерной техники. Поэтому вполне оправдано повышенное внимание, уделяемое изучению вычислительных машин и вычислительных сетей в рамках высшего образования. В рамках федерального государственного образовательного стандарта высшего образования по направлению подготовки 09.03.01 Информатика и вычислительная техника дисциплина «ЭВМ и периферийные устройства» содержит следующие аспекты: основные характеристики ЭВМ различных классов; организация процессора; организация памяти ЭВМ; выполнение команд; организация прерываний; периферийные устройства и т.д.

В результате изучения дисциплины «ЭВМ и периферийные устройства» формируются следующие компетенции или их составляющие: – разработка технических заданий на оснащение отделов, лабораторий, офисов компьютерным и сетевым оборудованием;

– способность участвовать в настройке и наладке программно- аппаратных комплексов;

– способность сопрягать аппаратные и программные средства в составе информационных и автоматизированных систем;

– способность подключать и настраивать модули ЭВМ и периферийного оборудования.

Целью данного издания является изложение теоретических основ, необходимых для освоения практических навыков в работе с ЭВМ и периферийными устройствами (ПУ), а также непосредственное получение этих практических навыков и освоение вышеназванных компетенций.

В результате освоения данной дисциплины студент должен:

знать:

– основы построения и архитектуры ЭВМ;

– принципы построения, параметры и характеристики цифровых и аналоговых элементов ЭВМ;

– современные технические средства взаимодействия с ЭВМ.

уметь:

– выбирать, комплексировать и эксплуатировать программно- аппаратные средства в создаваемых вычислительных и информационных системах и сетевых структурах;

— ставить и решать схемотехнические задачи, связанные с выбором системы элементов при заданных требованиях к параметрам (временным, мощностным, габаритным, надежностным);

— тестировать, испытывать и использовать аппаратные средства вычислительных и информационных систем;

владеть:

— методами выбора элементной базы для построения различных архитектур вычислительных средств.

Материал, изложенный в данном учебном пособии, ставит своей целью привить обучающимся вышеуказанные навыки.

1.1. Основные термины

Вычислительная машина – это комплекс технических и программных средств, предназначенный для автоматизации подготовки и решения задач пользователей.

Вычислительную систему определим как совокупность взаимосвязанных и взаимодействующих процессоров или вычислительных машин, периферийного оборудования и программного обеспечения, предназначенную для подготовки и решения задач пользователей.

Формально отличие вычислительной системы (ВС) от вычислительной машины (ВМ) выражается в количестве вычислителей. Множественность вычислителей позволяет реализовать в ВС параллельную обработку. С другой стороны, современные вычислительные машины с одним процессором также обладают определенными средствами распараллеливания вычислительного процесса. Иными словами, грань между ВМ и ВС часто бывает весьма расплывчатой, что дает основание там, где это целесообразно, рассматривать ВМ как одну из реализаций ВС. И напротив, вычислительные системы часто строятся из традиционных ВМ и процессоров, поэтому многие из положений, относящихся к ВМ, могут быть распространены и на ВС.

Архитектура вычислительной машины – это логическое построение вычислительной машины, т.е. то, какой машина представляется программисту. Подобную трактовку называют «узкой», и охватывает она перечень и формат команд, формы представления данных, механизмы ввода/вывода, способы адресации памяти и т.п. Такой подход не включает в рассмотрение вопросы физического построения вычислительных средств: состав устройств, число регистров процессора, ёмкость памяти, наличие специального блока для обработки вещественных чисел, тактовая частота центрального процессора и т.д. Этот круг вопросов принято определять понятием организация или *структурная организация*.

1.2. Уровни детализации структуры вычислительной машины

Вычислительная машина как законченный объект может рассматриваться на различных уровнях детализации. Таких уровней может быть достаточно много, однако сложившаяся практика ограничивает их число четырьмя (рис. 1.1).

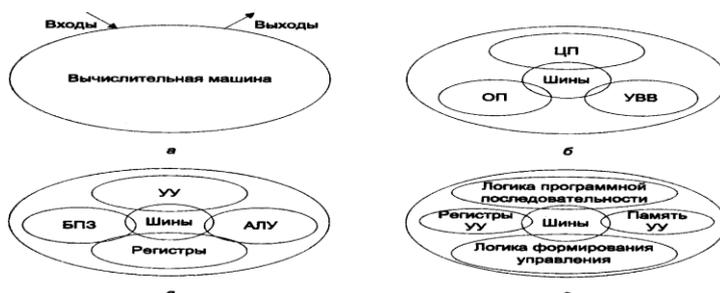


Рис. 1.1. Уровни детализации вычислительной машины:

а) уровень «черного ящика»; б) уровень общей архитектуры; в) уровень архитектуры центрального процессора; г) уровень архитектуры устройства управления

На *первом уровне* вычислительная машина рассматривается как устройство, способное хранить и обрабатывать информацию, а также обмениваться данными с внешним миром (рис. 1.1, а). ВМ представляется

«черным ящиком», который может быть подключен к коммуникационной сети и к которому, в свою очередь, могут подсоединяться устройства.

Уровень общей архитектуры (рис. 1.1, б) представляет ВМ в виде четырех составляющих: центрального процессора (ЦП), основной памяти (ОП), устройства ввода/вывода (УВВ) и системы шин.

На *третьем уровне* детализируется каждое из второго уровня. Для взят центральный процессор (рис. 1.1, в). В простейшем в нем можно выделить:

- арифметико-логическое устройство (АЛУ), обеспечивающее обработку целых чисел;
- блок обработки чисел в формате с плавающей запятой (БПЗ);
- регистры процессора, используемые для краткосрочного хранения команд, данных и адресов;
- устройство управления (УУ), обеспечивающее совместное функционирование устройств ВМ;
- внутренние шины.

На *четвертом уровне* детализируются элементы третьего уровня. На рис. 1.1, г раскрыта структура устройства управления. УУ представлено в виде четырех составляющих:

- логики программной последовательности – электронных схем, обеспечивающих выполнение команд программы в последовательности, предписываемой программой;
- регистров и дешифраторов устройства управления;
- управляющей памяти;
- логики формирования управления, генерирующей все необходимые управляющие сигналы.

1.3. Эволюция средств вычислительной техники

При описании эволюции средств вычислительной техники (ВТ) обычно используют один из двух подходов: хронологический или технологический. В первом случае это хронология событий, существенно повлиявших на становление ВТ, во втором – технологический подход, когда развитие вычислительной техники рассматривается в терминах архитектурных решений и технологий.

В качестве узловых моментов, определяющих появление нового поколения ВТ, обычно выбираются революционные идеи или технологические прорывы, кардинально изменяющие дальнейшее развитие средств автоматизации вычислений. Одной из таких идей принято считать концепцию вычислительной машины с хранимой в памяти программой, сформулированную Джоном фон Нейманом. Взяв ее за точку отсчета, историю развития ВТ можно представить в виде трех этапов:

1. Донеимановского периода.
2. Эры вычислительных машин и систем с фон-неймановской архитектурой.
3. Постнеймановской эпохи – эпохи параллельных и распределенных вычислений, где наряду с традиционным подходом все большую роль начинают играть отличные от фон-неймановских принципы организации вычислительного процесса.

Значительно большее распространение, однако, получила привязка поколений к смене технологий. Принято говорить о «механической» эре (нулевое поколение) и последовавших за ней пяти поколениях ВС. Первые четыре поколения традиционно связывают с элементной базой вычислительных систем: электронные: лампы, полупроводниковые приборы, интегральные схемы малой степени интеграции (ИМС), большие (БИС), сверхбольшие (СБИС) и ультрабольшие (УБИС) интегральные микросхемы. Пятое поколение в общепринятой интерпретации ассоциируют с интеллектуальными возможностями ВС. Работы по созданию ВС пятого поколения велись в рамках четырех достаточно независимых программ, осуществлявшихся учеными США, Японии, стран Западной Европы и стран Совета экономической взаимопомощи.

Ввиду того, что ни одна из программ не привела к ожидаемым результатам, разговоры о ВС пятого поколения понемногу утихают. Трактовка пятого поколения явно выпадает из «технологического» принципа. С другой стороны, причисление всех ВС на базе сверхбольших интегральных схем (СБИС) к четвертому поколению не отражает принципиальных изменений в архитектуре ВС, произошедших за последние годы.

1.4. Концепция машины с хранимой в памяти программой

В основе архитектуры современных ВМ лежит представление алгоритма решения задачи в виде программы последовательных вычислений. Согласно стандарту ISO 2382/1-84, программа для ВМ – это

«упорядоченная последовательность команд, подлежащая обработке».

Вычислительная машина, где определенным образом закодированные команды программы хранятся в памяти, называется *вычислительной машины с хранимой в памяти программой*. Идея создания такой ВМ принадлежит создателям вычислителя ENIAC Эккерту, Мочли и фон Нейману. Относительно авторства существует несколько версий, но поскольку в законченном виде идея впервые была изложена в 1945 г. в статье фон Неймана, именно его фамилия фигурирует в обозначении архитектуры подобных машин, составляющих подавляющую часть современного парка ВМ и ВС.

Сущность фон-неймановской концепции вычислительной машины можно свести к четырем принципам:

- двоичного кодирования;
- программного управления;
- однородности памяти;
- адресности.

1.4.1. Принцип двоичного кодирования

Согласно этому принципу, вся информация (данные, команды), кодируются двоичными цифрами 0 и 1. Каждый тип информации представляется двоичной последовательностью и имеет свой *формат*. Последовательность битов в формате, имеющая определенный смысл, называется *полем*. В числовой информации обычно выделяют поле знака и поле значащих разрядов. В формате команды можно выделить два поля (рис. 1.2): *поле кода операции* (КОп) и *поле адресов* (адресную часть – АЧ).

Код операции (КОп)	Адресная часть (АЧ)
-----------------------	------------------------

Рис. 1.2. Структура команды

Код операции представляет собой указание, какая операция должна быть выполнена, и задается с помощью r -разрядной двоичной комбинации.

Вид *адресной части* и число составляющих ее адресов зависят от типа команды:

- в командах преобразования данных АЧ содержит адреса объектов обработки (операндов) и результата;
- в командах изменения порядка вычислений – адрес следующей команды программы;
- в командах ввода/вывода – номер устройства ввода/вывода. Адресная часть также представляется двоичной последовательностью, длину которой обозначим через p . Таким образом, команда в вычислительной машине имеет вид $(r + p)$ -разрядной двоичной комбинации.

1.4.2. Принцип программного управления

Все вычисления, предусмотренные алгоритмом решения задачи, должны быть представлены в виде программы, состоящей из последовательности управляющих слов – команд. Каждая команда предписывает некоторую операцию из набора операций, реализуемых вычислительной машиной. Команды программы хранятся в последовательных ячейках памяти вычислительной машины и выполняются в естественной последовательности, то есть в порядке их положения в программе. При необходимости, с помощью специальных команд эта последовательность может быть изменена. Решение об изменении порядка выполнения команд программы принимается либо на основании анализа результатов предшествующих вычислений, либо безусловно.

1.4.3. Принцип однородности памяти

Команды и данные хранятся в одной и той же памяти и внешне в памяти неразличимы. Распознать их можно только по способу использования. Это позволяет производить над командами те же операции, что и над числами, и, соответственно, открывает ряд возможностей. Так, циклически изменяя адресную часть команды, можно обеспечить обращение к последовательным элементам массива данных. Такой прием носит название модификации команд и с позиций современного программирования не приветствуется. Более полезным является другое следствие принципа однородности, когда команды одной программы могут быть получены как результат исполнения другой программы. Эта возможность лежит в основе *трансляции* – перевода текста программы с языка высокого уровня на язык конкретной ВМ.

Концепция вычислительной машины, изложенная в статье фон Неймана, предполагает единую память для хранения команд и данных. Такой подход был принят в вычислительных машинах, создававшихся в Принстонском университете, из-за чего и получил название *принстонской архитектуры*. Практически одновременно в Гарвардском университете предложили иную модель, в которой ВМ имела отдельную память команд и отдельную память данных. Этот вид архитектуры называют *гарвардской архитектурой*. Долгие годы преобладающей была и остается принстонская архитектура, хотя она порождает проблемы пропускной способности тракта «процессор-память». В последнее время в связи с широким использованием кэш-памяти разработчики ВМ все чаще обращаются к гарвардской архитектуре.

1.4.4. Принцип адресности

Структурно основная память состоит из пронумерованных ячеек, причем процессору в произвольный момент доступна любая ячейка. Двоичные коды команд и данных разделяются на единицы информации, называемые словами, и хранятся в ячейках памяти, а для доступа к ним используются номера соответствующих ячеек – адреса.

1.5.

Принципы действия компьютера

1.5.1.

Фон-неймановская архитектура

Основное отличие персонального компьютера от больших машин, или так называемых мейнфреймов, состоит в том, что он позволяет одновременно использовать его ресурсы только одному пользователю. Он может выполнять одновременно несколько программ: обработки, вывода результатов, загрузки, поиска информации в сети и т.д. Кроме того, многие персональные машины используются в качестве серверов в сети, и их ресурсами (т.е. аппаратными и программными средствами) могут пользоваться несколько пользователей одновременно.

Структура самого компьютера за все время существования машин изменилась незначительно. Она по-прежнему строится на основе модели фон Неймана – ее основная память состоит из отдельных ячеек с последовательными номерами (или «адресами»), в которых могут храниться как коды отдельных команд (программа), так и данных. Однако технологический прогресс привел к объединению нескольких узлов и устройств в одной микросхеме.

Типичная фон-неймановская ВМ (рис. 1.3) содержит: память, устройство управления, арифметико-логическое устройство и устройство ввода/вывода.

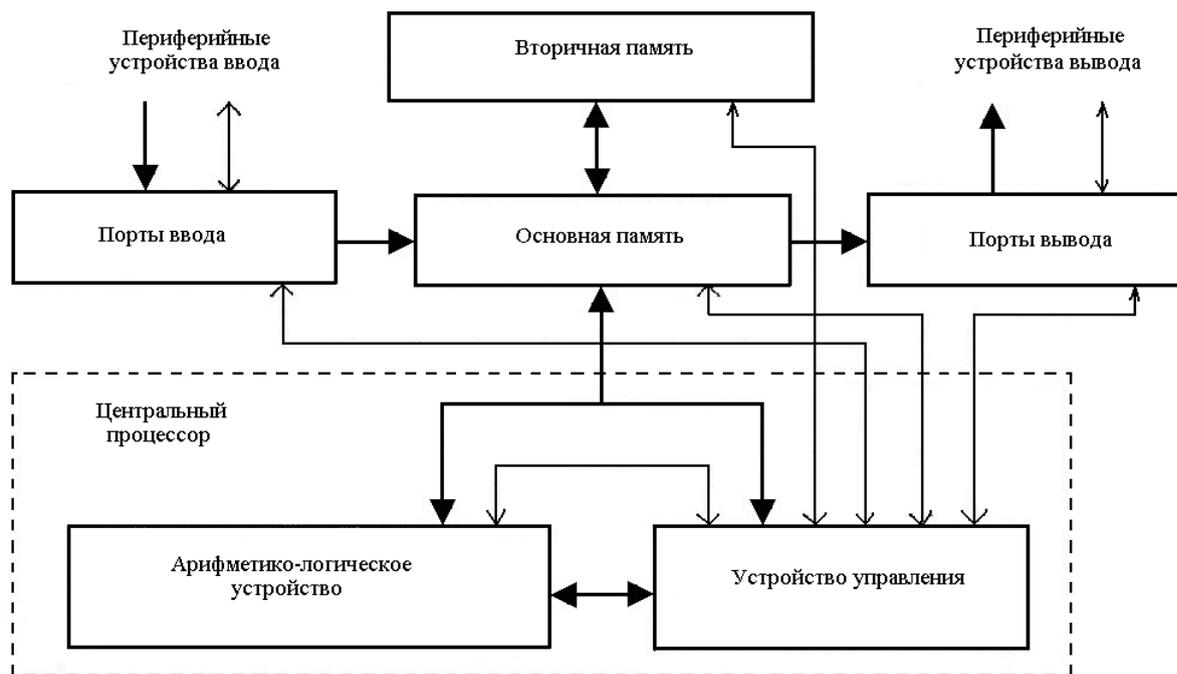


Рис. 1.3. Структура фон-неймановской вычислительной машины

В любой ВМ имеются средства для ввода программ и данных к ним. Информация поступает из подсоединенных к ЭВМ периферийных устройств (ПУ) ввода. Результаты вычислений выводятся на периферийные устройства вывода. Связь и взаимодействие ВМ и ПУ обеспечивают порты ввода и порты вывода. Термином *порт* обозначают аппаратуру сопряжения периферийного устройства с ВМ и управления им. Совокупность портов ввода и вывода называют *устройством ввода/вывода* (УВВ) или *модулем ввода/вывода ВМ* (МВВ).

Введенная информация сначала запоминается в основной памяти, а затем переносится во вторичную память, для длительного хранения. Чтобы программа могла выполняться, команды и данные должны располагаться в основной памяти (ОП), организованной таким образом, что каждое двоичное слово хранится в отдельной ячейке, идентифицируемой

адресом, причем соседние ячейки памяти имеют следующие по порядку адреса. Доступ к любым ячейкам запоминающего устройства (ЗУ) основной памяти может производиться в произвольной последовательности. Такой вид памяти известен как *память с произвольным доступом*. Основная память современных вычислительных машин в основном состоит из полупроводниковых оперативных запоминающих устройств (ОЗУ), обеспечивающих как считывание, так и запись информации. Для таких запоминающих характерна энергозависимость – хранимая информация теряется при отключении электропитания. Если необходимо, чтобы часть основной памяти была энергонезависимой, в состав основной памяти включают постоянные запоминающие устройства (ПЗУ), также обеспечивающие произвольный доступ. Хранящаяся в ПЗУ информация может только считываться (но не записываться).

Размер ячейки основной памяти обычно принимается равным 8 двоичным разрядам – байту. Для хранения больших чисел используются 2, 4 или 8 байтов, размещаемых в ячейках с последовательными адресами. В этом случае за адрес числа часто принимается адрес его младшего байта. Так, при хранении 32-разрядного числа в ячейках с адресами 200, 201, 202 и 203 адресом числа будет 200. Такой прием называют *адресацией по младшему байту* (little endian addressing). Возможен и противоположный подход – по меньшему из адресов располагается старший байт. Этот способ известен как *адресация по старшему байту* (big endian addressing). Адресация младшему байту характерна для микропроцессоров фирмы Intel и мини-ЭВМ фирмы DEC, а по старшему байту – для микропроцессоров фирмы Motorola и универсальных ЭВМ фирмы IBM. В принципе выбор порядка записи байтов существенен лишь при пересылке данных между вычислительными машинами с различными формами их адресации или при манипуляциях с отдельными байтами числа. В большинстве вычислительных машин предусмотрены специальные инструкции для перехода от одного способа к другому.

Для долговременного хранения больших программ и массивов данных в ВМ обычно имеется дополнительная память, известная как *вторичная*. Вторичная память энергонезависима и чаще всего реализуется на базе магнитных дисков. Информация в ней хранится в виде специальных программно поддерживаемых объектов – файлов.

Устройство управления (УУ) – часть вычислительной машины, организующая автоматическое выполнение программ (путем реализации функций управления) и обеспечивающая функционирование ВМ как единой системы. Устройство управления вычислительной машины следует рассматривать как совокупность элементов, между которыми происходит пересылка информации, в ходе которой эта информация может

подвергаться определенным видам обработки. Пересылка информации между любыми элементами ВМ инициируется своим сигналом управления (СУ), т.е. управление вычислительным процессом сводится к выдаче нужного набора сигналов управления в нужной временной последовательности. *Цепи СУ показаны на рис. 1.3 полутоновыми линиями.* Основной функцией УУ является формирование управляющих сигналов, отвечающих за извлечение команд из памяти в порядке, определяемом программой, и последующее исполнение этих команд. Кроме того, УУ формирует СУ для синхронизации и координации внутренних и внешних устройств ВМ.

Еще одной неотъемлемой частью ВМ является *арифметико-логическое устройство* (АЛУ). АЛУ обеспечивает арифметическую и логическую обработку двух входных переменных, в результате которой формируется выходная переменная. Функции АЛУ обычно сводятся к простым арифметическим и логическим операциям, а также операциям сдвига. Помимо результата операции АЛУ формирует ряд *признаков результата* (флагов), характеризующих полученный результат и события, произошедшие в процессе его получения (равенство нулю, знак, четность, перенос, переполнение и т.д.). Флаги могут анализироваться в УУ с целью принятия решения о дальнейшей последовательности выполнения команд программы.

УУ и АЛУ тесно взаимосвязаны и их обычно рассматривают как единое устройство, известное как *центральный процессор* (ЦП) или просто процессор. Помимо УУ и АЛУ, в процессор входит набор *регистров общего назначения* (РОН), служащих для промежуточного хранения информации в процессе ее обработки.

1.5.2. Структуры вычислительных машин

В настоящее время примерно одинаковое распространение получили два способа построения вычислительных машин: *с непосредственными связями и на основе шины.*

Типичным представителем первого способа может служить классическая фон-неймановская вычислительная машина (рис. 1.3). В ней между взаимодействующими устройствами (процессор, память, устройство ввода/вывода) имеются непосредственные связи. Особенности связей (число линий в шинах, пропускная способность и т.п.) определяются видом информации, характером и интенсивностью обмена. Достоинством архитектуры с непосредственными связями можно считать возможность развязки «узких мест» путем улучшения структуры

и характеристик только определенных связей, что экономически может быть наиболее выгодным решением. У фон-неймановских ВМ таким «узким местом» является канал пересылки данных между ЦП и памятью. Кроме того, ВМ с непосредственными связями плохо поддаются реконфигурации.

В варианте *с общей шиной* все устройства вычислительной машины подключены к магистральной шине, служащей единственным трактом для потоков команд, данных и управления (рис. 1.4). Наличие общей шины существенно упрощает реализацию ВМ, позволяет легко менять состав и конфигурацию машины. Благодаря этим свойствам *шинная архитектура* получила широкое распространение в мини-и микроЭВМ. Вместе с тем, именно с шиной связан и основной недостаток архитектуры: в каждый момент передавать информацию по шине может только одно устройство. Основную нагрузку на шину создают обмены между процессором и памятью, связанные с извлечением из памяти команд и данных и записью в память результатов вычислений. На операции ввода/вывода остается лишь часть пропускной способности шины. Практика показывает, что даже при достаточно быстрой шине для 90 % приложений этих остаточных ресурсов обычно не хватает, особенно в случае ввода или вывода больших массивов данных.

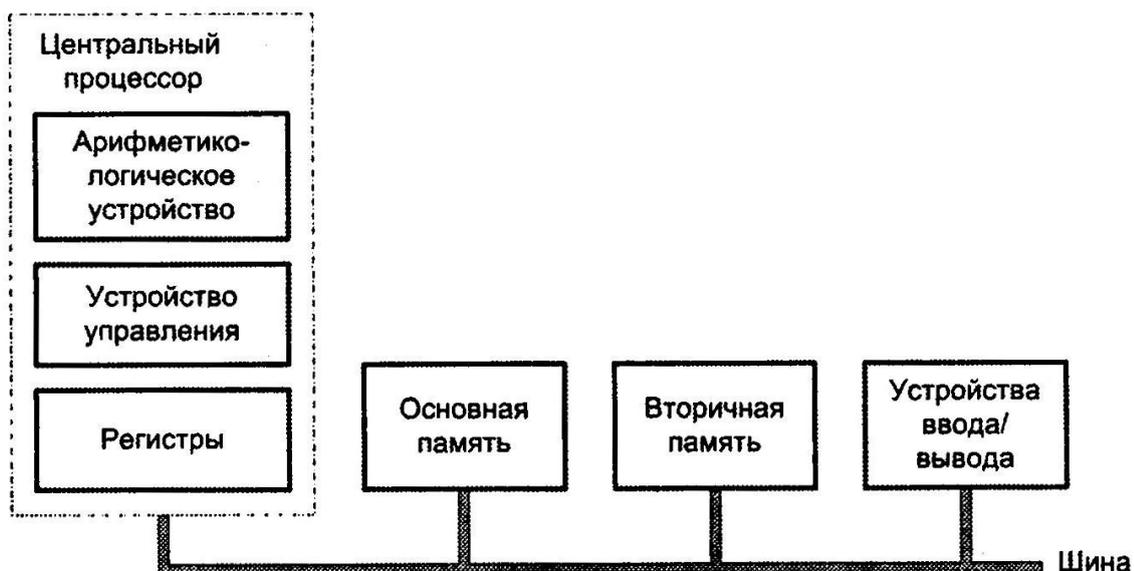


Рис. 1.4. Структура вычислительной машины на базе общей шины

При сохранении фон-неймановской концепции последовательного выполнения команд программы шинная архитектура в чистом ее виде оказывается недостаточно эффективной. Более распространена архитектура с иерархией шин, где помимо магистральной шины имеется

еще несколько дополнительных шин. Они могут обеспечивать непосредственную связь между устройствами с наиболее интенсивным обменом, например процессором и кэшпамятью. Другой вариант использования дополнительных шин – объединение однотипных устройств ввода/вывода с последующим выходом с дополнительной шины на магистральную. Все эти меры позволяют снизить нагрузку на общую шину и более эффективно расходовать ее пропускную способность.

В современных ВМ общая шина состоит из нескольких «подшин» – шины адреса, шины данных и шины управления. В персональных машинах для экономии места на системной плате шины адреса и данных иногда выполняют в виде одной разделяемой во времени шины; тогда адрес и данные по ней передаются только поочередно.

Базовым элементом компьютера является триггер. На его основе выполняются другие узлы компьютера.

Триггер – электронная схема, которая может находиться в одном из двух устойчивых состояний «0» и «1». Внешними сигналами можно переводить триггер из одного состояния в другое.

Регистр – это несколько определенным образом соединенных триггеров, т.е. можно записать двоичное слово в регистр, прочитать его, сдвинуть, инвертировать.

Счетчик позволяет определить число поступивших на него сигналов.

Он также строится на основе триггеров.

Логическая схема реализует определенную логическую функцию, т.е. формирует выходной сигнал при определенных комбинациях сигналов на ее входах.

Содержимое счетчика команд (СчК) процессора передается по адресной шине на регистр адреса (РгА) основной памяти (рис. 1.5). В момент включения компьютера в счетчике команд всегда находится один и тот же начальный адрес. Таким образом, запрашивается содержимое ячейки памяти с этим начальным адресом, принадлежащим BIOS. Как правило, эта ячейка содержит код команды безусловного перехода, служащей для изменения содержимого счетчика команд. Этот код передается на регистр команд (РгК) процессора по шине данных. Содержимое ячейки памяти поступает на регистр команд РгК, поскольку запрос к памяти произведен из счетчика команд; это *обязательное требование* для любого компьютера традиционной архитектуры.

Регистр команд процессора РгК, в свою очередь, состоит из нескольких регистров: регистра кода операции (РгКОП) и регистров адресов процессора (РгАП). Часть слова, попавшая в регистр кода операции, передается в блок управления (БУ), вырабатывающий последовательность управляющих сигналов.

Когда выполняется команда безусловного перехода, вторая адресная часть слова, попавшая в один из регистров адреса процессора, под управлением сигналов с БУ передается вновь на счетчик команд. Эта команда одноадресная, т.е. ее адресная часть содержит только один адрес. На этом и завершается ее выполнение. Блок управления формирует сигнал об окончании выполнения команды, а содержимое СчК вновь передается на RгА памяти, т.е. происходит запрос следующей команды.

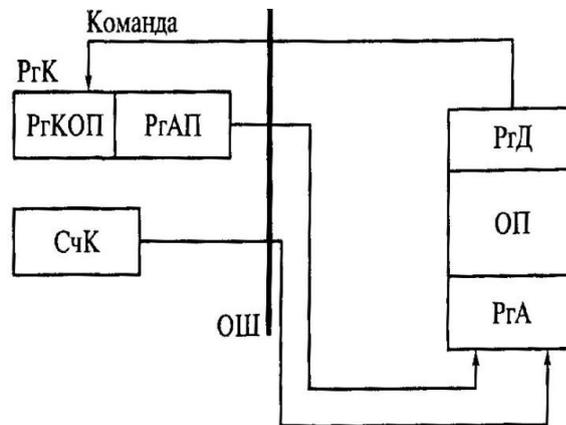


Рис. 1.5. Передача команд из оперативной памяти в центральный процессор

Таким образом, процедура обращения к памяти повторяется. Содержимое ячейки памяти, к которой произведено повторное обращение, рассматривается в качестве новой команды, т.е. вновь загружается на RгК процессора. Обычно вторая команда служит для начала загрузки ОЗУ с магнитного диска; она уже не является командой безусловного перехода. При ее выполнении под давлением кода операции (часть команды, попавшая на RгКОП) вырабатываются иные управляющие сигналы, а содержимое первого регистра RгАП, представляющего собой часть RгК, передается на адресный регистр памяти и рассматривается в качестве адреса первого операнда.

Для ОЗУ безразлично, откуда пришел запрос – из счетчика команд или адресного регистра, поэтому в регистре данных (RгД) памяти слово формируется так же, как и раньше. Однако в процессоре оно помещается на первый регистр данных АЛУ, поскольку запрос этого слова поступил из адресного регистра RгАП процессора. Затем блок управления формирует аналогичные сигналы для передачи на регистр адреса RгА основной памяти содержимого второго регистра адреса RгАП процессора; в результате содержимое ячейки памяти с адресом, находящимся в регистре адресов процессора RгАП, поступает на второй регистр данных арифметического устройства.

Затем блок управления вырабатывает сигналы в зависимости от кода операции в регистре кода операции RгКОП, подает их в АЛУ, которое выполняет соответствующую операцию, а ее результат помещает в выходной регистр-аккумулятор. После этого содержимое регистра-аккумулятора передается в ячейку памяти, адрес которой обычно находится в первом регистре адресов процессора RгАП, т.е. выполняется еще одно обращение к ОП. Информация из регистра-аккумулятора передается на шину данных, а адрес ячейки из регистра адресов процессора RгАП – на адресную шину. В зависимости от конструкции машины, числа адресов в выполняемой команде (адресности) и других особенностей, содержимое регистра-аккумулятора может сохраняться в нем, передаваться в ячейку ОП по адресу, находящемуся в первом или втором RгАП.

После сохранения содержимого регистра-аккумулятора к счетчику команд (СЧК) добавляется длина текущей команды в байтах (часто говорят

«единица»), чтобы обратиться к следующей ячейке памяти, и начинается новый цикл выполнения очередной команды.

Таким образом, выполнение программы происходит последовательно: каждый раз в машине реализуется лишь одна команда, попадающая в регистр команд из ОП. Чтобы увеличить производительность компьютера, нужно либо повысить скорость выполнения команды, либо выполнять несколько последовательных команд одновременно. Повышение скорости выполнения команды связано с улучшением технических характеристик и увеличением быстродействия всех компонентов, входящих в компьютер – ЦП, ОП, шин интерфейсов, устройств ввода-вывода. Но увеличение скорости выполнения команды принципиально ограничено – скорость распространения сигналов в машине не может превышать скорость света, а длина пути определяется числом вентилях и применяемой технологией. Второй путь, заключающийся в параллельном выполнении нескольких команд, наиболее перспективен.

1.6. Структуры вычислительных систем

Понятие «вычислительная система» предполагает наличие множества процессоров или законченных вычислительных машин, при объединении которых используется один из двух подходов.

В вычислительных системах с общей памятью (рис. 1.6) имеется общая основная память, совместно используемая всеми процессорами

системы. Связь процессоров с памятью обеспечивается с помощью коммуникационной сети, чаще всего вырождающейся в общую шину. Таким образом, структура вычислительной системы с общей памятью аналогична архитектуре с общей шиной, в силу чего ей свойственны те же недостатки. Применительно к вычислительным системам данная схема имеет дополнительное достоинство: обмен информацией между процессорами не связан с дополнительными операциями и обеспечивается за счет доступа к общим областям памяти.

Альтернативный вариант организации – распределенная система, где общая память вообще отсутствует, а каждый процессор обладает собственной локальной памятью (рис. 1.7). Часто такие системы объединяют отдельные ВМ. Обмен информацией между составляющими системы обеспечивается с помощью коммуникационной сети посредством обмена сообщениями.



Рис. 1.6. Структура вычислительной системы с общей памятью

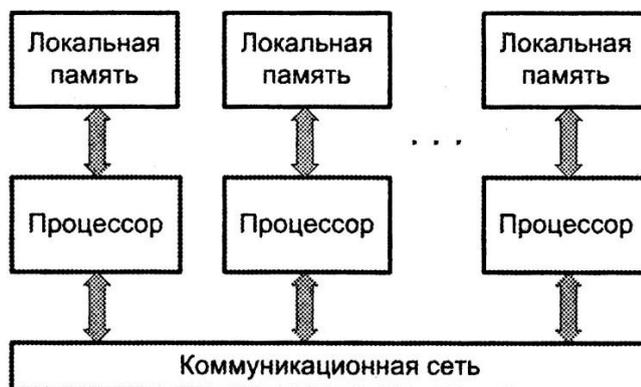


Рис. 1.7. Структура распределенной вычислительной системы

Подобное построение вычислительных систем снимает ограничения, свойственные для общей шины, но приводит к дополнительным издержкам на пересылку сообщений между процессорами или машинами.

1.7. Перспективы совершенствования архитектуры вычислительных машин и вычислительных систем

Совершенствование архитектуры вычислительных машин и систем началось с момента появления первых ВМ и не прекращается по сей день. Каждое изменение в архитектуре направлено на абсолютное повышение производительности или на более эффективное решение задач определенного класса. Эволюцию архитектур определяют самые различные факторы, главные из которых показаны на рис. 1.8.

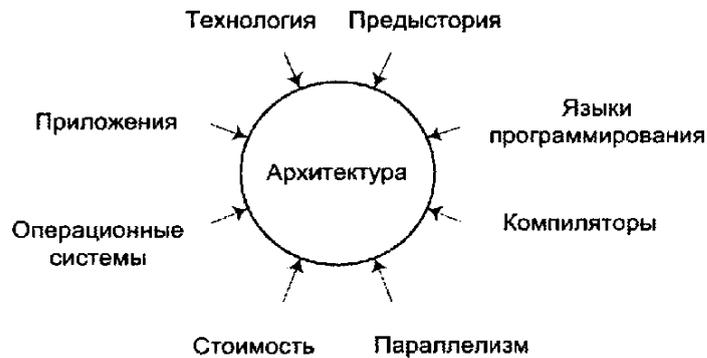


Рис. 1.8. Факторы, определяющие развитие архитектуры вычислительных систем

1.8. Надежность, производительность, быстродействие и его показатели

1.8.1. Надежность

Отказы компьютера могут оказать критическое влияние на работу информационно-справочной системы или системы управления. Например, задержка в выдаче управляющего воздействия в результате отказа компьютера приводит к потере управления, разрушению технологических установок, срыву выполнения задания, т.е. к значительным потерям, а в некоторых случаях и к экологическим катастрофам. Современные технологии не позволяют полностью избавиться от отказов в аппаратуре и ошибок в программах даже при значительных затратах, но ряд мер технического и организационного характера способны снизить их интенсивность.

Надежность – это свойство компьютера сохранять свою работоспособность, т.е. выполнять возложенные на него функции. Однако в работе компьютера возможны ошибки, которые подразделяют на систематические, возникающие в результате отказов, и случайные, возникающие в результате сбоев.

Отказ – это утрата возможности выполнения требуемой функции.

Сбой – кратковременное нарушение правильной работы аппаратуры.

1.8.2. Производительность

Способность машины выполнять некоторый объем работы по обработке данных за единицу времени называют *производительностью*. На нее оказывает влияние множество факторов: характер выполняемых задач, архитектура и параметры процессора, характеристики основной и внешней памяти, наличие дополнительных устройств обработки, быстродействие соединительных шин, способ подключения различных устройств и т.п. Оценка производительности проводится по стандартным методикам, служащим только для непосредственного сравнения различных компьютеров между собой.

Для оценки производительности часто используют понятие *времени прохождения задачи (времени ответа, времени выполнения)* – интервал от момента поступления задачи на выполнение до момента представления результатов ее решения пользователю. Это время включает в себя работу ЦП, обращения к оперативной памяти и дискам, операции ввода-вывода, накладные расходы, связанные с работой ОП, и т.п. В мультипрограммном режиме, когда процессор компьютера выполняет несколько программ, время прохождения задачи не будет постоянным.

1.8.3. Быстродействие и его показатели

Способность компьютера выполнять определенное число операций за единицу времени называют *быстродействием*. В большинстве случаев его можно определить по тактовой частоте генератора, которая стала важнейшей характеристикой быстродействия. При этом оценивается быстродействие только процессора. Время выполнения программы в ЦП зависит от трех параметров: длительности такта синхронизации (или тактовой частоты), числа тактов синхронизации, необходимого для выполнения каждой команды, и общего числа команд в программе. Для оценки быстродействия персональных компьютеров особенно часто используют частоту.

Кроме того, быстродействие можно оценить по следующим показателям:

- длительность выполнения операций определенного типа (обычно число наиболее коротких арифметических операций, выполняемых за секунду; это так называемое *пиковое быстродействие* процессора);
- средняя длительность выполнения операции из некоторого стандартного набора операций; это *номинальное быстродействие*;
- средняя длительность выполнения представительной задачи; при этом если в затратах времени учитывается только время обработки, то такую задачу принято называть ядром, а если учитывается и время на ввод-вывод, то эталонной задачей. Время на организацию вычислительного процесса не учитывается. Это *системная производительность*.

Быстродействие компьютера, имеющего традиционную архитектуру и призванного решать задачи с большим числом логических операций, принято оценивать числом MIPS (миллион инструкций в секунду). Применение MIPS в качестве универсального показателя наталкивается на ряд трудностей:

- каждый компьютер обладает структурой, ориентированной на обработку слов определенного формата и разрядности, т.е. инструкции в разных компьютерах определяют различный объем работы;
- этот показатель не учитывает сложность выполняемой команды. Поэтому при наличии в компьютере дополнительного сопроцессора, призванного выполнять операции с плавающей точкой, этот показатель снижается. При отсутствии математического сопроцессора операции над числами с плавающей точкой реализуются посредством подпрограмм, состоящих из нескольких достаточно простых команд целочисленной арифметики, и показатель MIPS имеет высокое значение. Сложная команда сопроцессора выполняется достаточно долго (хотя

и значительно быстрее, чем соответствующая подпрограмма), поэтому показатель MIPS снижается.

Контрольные вопросы

1. По каким признакам можно разграничить понятия «вычислительная машина» и «вычислительная система»?
2. В чем состоит различие между «узкой» и «широкой» трактовкой понятия «архитектура вычислительной машины»?
3. Какой уровень детализации вычислительной машины позволяет определить, можно ли данную ВМ причислить к фон-неймановским?
4. Какие закономерности в эволюции вычислительных машин породили появление нового научного направления – «Теория эволюции компьютеров»?
5. По каким признакам выделяют поколения вычислительных машин?
6. Поясните определяющие идеи для каждого из этапов эволюции вычислительной техники.
7. Какой из принципов фон-неймановской концепции вычислительной машины можно рассматривать в качестве наиболее существенного?
8. Оцените достоинства и недостатки архитектур вычислительных машин с непосредственными связями и общей шиной.
9. Сформулируйте основные тенденции развития интегральной схемотехники.
10. Охарактеризуйте основные направления в дальнейшем развитии архитектуры вычислительных машин и систем.

Раздел 2. ПРЕДСТАВЛЕНИЕ ЧИСЛОВОЙ ИНФОРМАЦИИ В ЭВМ Тема 2. СИСТЕМЫ СЧИСЛЕНИЯ

Занятие 2 (лекция)

В компьютере может храниться и обрабатываться информация различного характера: числа, адреса, команды, различные символы, графические изображения и т.д. Любая информация в компьютере представляется в числовой форме, при этом используются различные системы счисления.

Под *системой счисления* понимается способ представления чисел с помощью символов (цифр), имеющих определенное количественное значение. В любой системе счисления числа представляются в виде последовательности цифр.

В *непозиционных системах счисления* количественный эквивалент каждой цифры не зависит от ее положения в записи числа. Примером непозиционной системы, которая сохранилась до наших дней, может служить римская система счисления. В основе римской системы счисления лежат знаки I для числа 1, V для числа 5, X для 10, L для 50, а для обозначения чисел 100, 500 и 1000 стали применять первые буквы соответствующих латинских слов (Centum – сто, Demimille – половина тысячи, Mille – тысяча).

В *позиционных системах счисления* количественное значение цифры зависит от ее места в числе. В позиционных системах счисления каждая цифра имеет вес. Обычно вес старшей цифры по отношению к весу соседней младшей цифры больше в количество раз, равное основанию системы счисления. При этом для целых чисел вес младшего разряда в любой системе счисления равен единице.

Основные достоинства любой позиционной системы счисления – простота выполнения арифметических операций и ограниченное количество цифр, необходимых для записи любых чисел. Примером позиционной системы счисления служит десятичная система счисления, которой мы широко пользуемся.

В позиционной системе счисления любое вещественное число в развернутой форме может быть представлено в следующем виде:

где A – само число; q – основание системы счисления; a_i – цифры, принадлежащие алфавиту данной системы счисления; n – число целых разрядов числа; m – число дробных разрядов числа.

Так, десятичное число $A_{10} = 4718,63$ в развернутой форме запишется так:

$$A_{10} = 4 \cdot 10^3 + 7 \cdot 10^2 + 1 \cdot 10^1 + 8 \cdot 10^0 + 6 \cdot 10^{-1} + 3 \cdot 10^{-2}.$$

В современных компьютерах используются позиционные системы счисления с основаниями 2, 8, 10 и 16. В табл. 2.1 приведены возможные способы изображения первых 16 чисел во всех четырех системах счисления.

Таблица 2.1
Системы счисления

Двоичные числа D_2	Восьмеричные числа D_8	Десятичные числа D_{10}	Шестнадцатеричные числа D_{16}
0000	0	0	0
0001	1	1	1
0010	2	2	2
0011	3	3	3
0100	4	4	4
0101	5	5	5
0110	6	6	6
0111	7	7	7
1000	10	8	8
1001	11	9	9
1010	12	10	A
1011	13	11	B
1100	14	12	C
1101	15	13	D
1110	16	14	E
1111	17	15	F

При хранении и обработке информации внутри компьютера используется двоичная система счисления. Это объясняется необходимостью физического представления только двух цифр (0 и 1), простотой выполнения арифметических операций и возможностью осуществления любых преобразований информации с помощью двоичных логических элементов.

Шестнадцатеричная (и реже восьмеричная) система счисления используется для более компактного представления информации (по сравнению с двоичной системой) при вводе и выводе больших массивов двоичных данных. Это связано с простотой перехода от двоичной системы счисления к шестнадцатеричной (восьмеричной) и наоборот.

2.1. Формы представления чисел

В общем случае числа имеют знак (плюс или минус). Кроме того, число может включать в себя целую и дробную части. Специальные формы представления чисел позволяют кодировать знаки чисел и указывать положение точки (запятой), разделяющей целую и дробную части числа.

Для кодирования знака числа отводится специальный разряд, называемый *знаковым*. Под него обычно отводится старший разряд числа. Для положительных чисел в нем записывается цифра 0, для отрицательных – 1.

Для указания положения точки используют одну из двух форм: форму с фиксированной или форму с плавающей точкой.

2.1.1. Форма представления чисел с фиксированной точкой

Эта форма, называемая также естественной, предполагает, что все числа в компьютере могут быть только целыми или только дробными. В этом случае положение точки является стандартным для данного компьютера и не требует специального указания. Эта форма является простой, но приводит к некоторому усложнению программирования.

Если в компьютере для всех чисел положение точки зафиксировано справа от младшего цифрового разряда, то числа принимают только целые значения.

На рис. 2.1 представлено $(n + 1)$ -разрядное целое число. Один разряд занимает знак, остальные n разрядов используются как цифровые. Веса цифровых разрядов показаны в верхней части рисунка. В этом случае в компьютере могут быть представлены числа, модуль которых находится в диапазоне

$$2^0 \leq A \leq 2^n - 1.$$

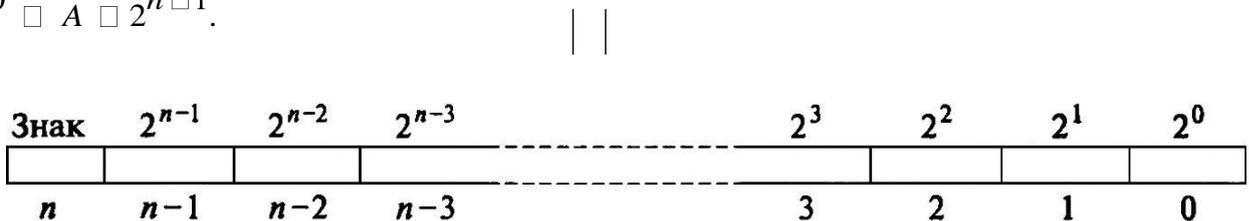


Рис. 2.1. Целое число в формате с фиксированной точкой

При этом точность представления чисел равна единице, так как числа могут быть только целыми.

Если точка зафиксирована слева от старшего цифрового разряда, то все числа могут быть только дробными. Формат дробного числа с фиксированной точкой (ФТ) представлен на рис. 2.2.



Рис. 2.2. Дробное число в формате с фиксированной точкой

Дробные числа с фиксированной точкой, имеющие n цифровых разрядов, представляются с точностью T^n (величина единицы младшего разряда дроби) в диапазоне

$$2^{\lfloor n \rfloor} \leq A \leq 1 - 2^{\lfloor n \rfloor}$$

В формате с фиксированной точкой могут представляться числа без знака. В этом случае все разряды являются цифровыми.

В современных микропроцессорах используется представление данных в форме целых чисел с фиксированной точкой. Форма дробных чисел с фиксированной точкой применяется для представления мантиисы числа в форме с плавающей точкой. Некоторые из целочисленных форматов микропроцессоров фирмы Intel представлены на рис. 2.3.

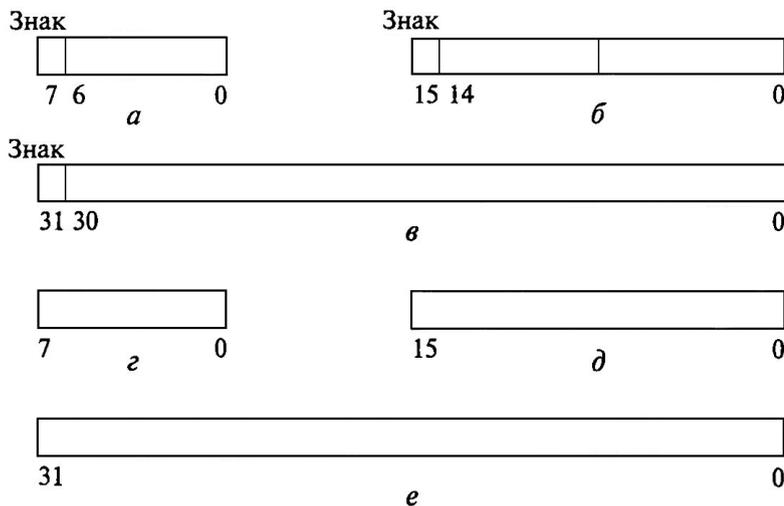


Рис. 2.3. Целочисленные форматы микропроцессоров фирмы Intel:

a – байт (целое со знаком); b – слово (целое со знаком); v – двойное слово (целое со знаком); z – байт (целое без знака); d – слово (целое без знака); e – двойное слово (целое без знака)

Стандартными форматами являются *байт*, *слово* и *двойное слово*.

При обработке мультимедийной информации используются не только отдельные целые числа, но и группы целых чисел, которые обрабатываются одновременно. При этом несколько малоразрядных чисел упаковываются в 64-разрядное слово. Упакованными могут быть восемь байтов, четыре слова или два двойных слова.

2.1.2. Форма представления чисел с плавающей точкой

Эта форма (нормальная или полулогарифмическая) позволяет представлять в компьютере любые (целые, дробные или смешанные) числа. Число в форме с плавающей точкой записывается в виде двух частей: мантиисы и порядка. Мантииса включает в себя значащие разряды числа, а порядок указывает положение точки. При этом мантииса записывается как дробное число с фиксированной точкой, а порядок – как целое число с фиксированной точкой.

Знак мантиисы является знаком всего числа, а знак порядка определяет, содержит ли число целую часть.

Значение числа с плавающей точкой определяется следующим образом:

$$A \approx m q^p$$

где m – мантииса числа; q – основание системы счисления; p – порядок числа.

Так, в десятичной системе счисления число $A_{10} = -123,456$ в форме с плавающей точкой может быть записано следующим образом:

$$m_A \approx 0,123456,$$

$$p_A \approx 3,$$

$$m_A \approx 0,0123456,$$

$$p_A \approx 4,$$

$$m_A \approx 0,00123456,$$

$$p_A \approx 5 \text{ и т.д.}$$

При заданных значениях мантиисы и порядка для определения значения числа нужно точку (запятую) в мантиисе перенести на количество разрядов, равное величине порядка, вправо для положительных порядков и влево для отрицательных. Например:

$$m_A \approx 0,87412456,$$

$$p_A \approx 5$$

$$A \approx 87412,456;$$

$$m_B \approx 0,12437696,$$

$$\approx$$

$$B \approx$$

$$0,0012437696.$$

$$p_A \approx 2$$

$$\approx$$

Число с плавающей точкой может быть нормализованным и ненормализованным. Число не нормализовано, если старшая цифра мантииссы равна нулю, т.е.

$$m \square 1 \quad | \quad q$$

Нормализованное число с плавающей точкой позволяет сохранить большее количество значащих цифр, поэтому в памяти числа хранятся в нормализованном виде. Нормализация выполняется путем сдвига мантииссы влево до тех пор, пока старший разряд мантииссы не станет равным единице. Так как значение мантииссы при этом увеличивается, для сохранения величины числа при сдвиге на каждый разряд значение порядка уменьшается на единицу.

Нормализованное число с плавающей точкой представляется с точностью 2^{-n} , где n – разрядность мантииссы. Диапазон чисел с плавающей точкой составляет:

$$\frac{1}{2} \square 2^{-k} \square 1 \square 2^n \square 2^k \square 1$$

где k – разрядность порядка.

Для увеличения диапазона чисел с плавающей точкой (за счет некоторого уменьшения точности) двоичная мантиисса может рассматриваться как шестнадцатеричное число. В этом случае каждая двоичная тетрада представляет одну шестнадцатеричную цифру, поэтому нормализация будет нарушена лишь тогда, когда четыре старших разряда мантииссы будут равны нулю. Например, если старшие разряды мантииссы имеют вид 0, 000101 ..., то мантиисса считается нормализованной, так как тетрада 0001 представляет шестнадцатеричную цифру 1. Появление незначащих нулей в мантииссе приводит к потере точности. Вместе с тем существенно увеличивается диапазон представления чисел:

$$\frac{1}{16} \square 16^{-k} \square 1 \square 16^n \square 16^k \square 1$$

Порядок числа может быть положительным или отрицательным. Для упрощения операций над порядками часто используют смещенный

порядок путем увеличения действительного порядка на величину 2^k ,

где k – число разрядов порядка. При этом смещенный порядок всегда является положительным числом и поэтому его знак не указывается.

Пример формата n -разрядного числа с плавающей точкой и смещенным порядком представлен на рис. 2.4. Для знака числа отводится старший разряд, k разрядов занимает смещенный порядок, а остальные $n - k - 1$ выделяются под мантиссу. Диапазон и точность представляемых в форме с плавающей точкой чисел зависят от формата.

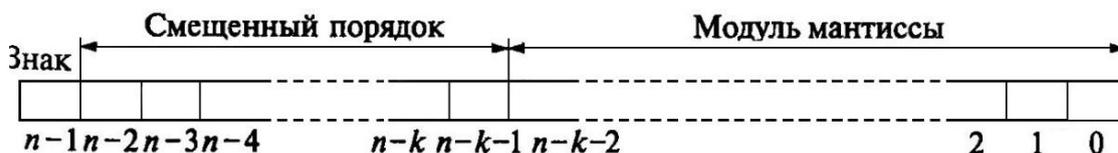


Рис. 2.4. Формат числа с плавающей точкой

Рекомендуемые стандартом основные форматы чисел с плавающей точкой представлены на рис. 2.5. Одинарный формат занимает 32 разряда, двойной – 64. Обычно для повышения точности используют способ скрытой единицы.

Суть способа заключается в том, что в нормализованном числе старший разряд мантиссы всегда равен единице, поэтому его можно не записывать, а подразумевать. Освободившийся разряд используется для записи дополнительного разряда мантиссы. Перед выполнением арифметических операций подразумеваемый разряд восстанавливается.

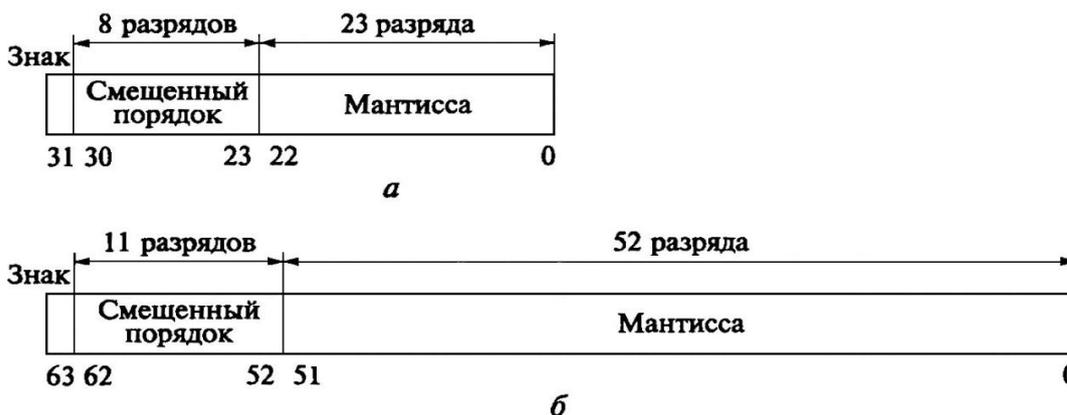


Рис. 2.5. Основные форматы чисел с плавающей точкой: a – одинарный; b – двойной

В одинарном формате под смещенный порядок отводится восемь разрядов, и под мантиссу – 24 (с учетом скрытой единицы). При этом диапазон представления чисел составляет:

$$10^{\square 38} \square A \square 10^{\square 38} \text{ в двойном}$$

формате смещенный порядок занимает 11 разрядов, мантисса – 53, а диапазон представления чисел составляет: $10^{\square 308} \square A \square 10^{\square 308}$.

Как и целые числа, числа с плавающей точкой могут быть записаны в упакованном формате. В микропроцессорах фирмы Intel четыре числа одинарной точности (по 32 разряда) упаковываются в группу длиной 128 разрядов. В такую же группу упаковываются два числа двойной точности (по 64 разряда).

2.1.3. Форматы двоично-десятичных чисел

Такие числа могут быть представлены в двух форматах: зонном и упакованном. В этих форматах каждая десятичная цифра и знак числа заменяются двоичной тетрадой в соответствии с используемым двоично-десятичным кодом. Количество цифр в числе может быть произвольным.

В *зонном* формате под каждую десятичную цифру отводится байт (рис. 2.6). В старшей тетраде записывается код зоны (например, код 1111), а в младшей – код цифры. В младшем байте вместо кода зоны записывается знак числа.

В *упакованном* формате байт содержит две цифры (рис. 2.7).

Младшая тетрада последнего байта содержит знак числа.

Байт		Байт		...	Байт		Байт	
Зона	Цифра	Зона	Цифра		Зона	Цифра	Знак	Цифра

a

Байт		Байт		Байт		Байт		Байт	
Зона	1	Зона	2	Зона	9	Зона	8	Минус	7
1111	0001	1111	0010	1111	1001	1111	1000	1101	0111

б

Рис. 2.6. Зонный формат двоично-десятичных чисел:

a – структура формата; *б* – пример записи числа – 12987

Байт		Байт		...	Байт		Байт	
Ци	Ци	Ци	Ци		Ци	Ци	Ци	Зн
фра	фра	фра	фра		фра	фра	фра	ак

a

Байт		Байт		Байт		Байт	
0	2	3	7	4	6	5	Плюс
000	001	001	011	010	011	010	1100
0	0	1	1	0	0	1	

б

Рис. 2.7. Упакованный формат двоично-десятичных чисел:

a – структура формата; *б* – пример записи числа +237465

2.2.

Машинные коды

Для упрощения арифметических операций числа записываются в специальной форме, называемой *машинными кодами*. Эти коды позволяют:

- свести операцию вычитания к операции сложения;
- автоматически получать знак суммы (разности);
- выявлять переполнение разрядной сетки.

Существуют следующие коды:

- прямой (ПК);
- обратный (ОК);
- дополнительный (ДК).

1. Что такое система счисления?
2. Что такое основание системы счисления?
3. Как зависит разрядность чисел от величины основания системы счисления?
4. Почему информация в компьютере представляется в двоичной системе счисления?
5. Какие системы счисления используются для представления информации в компьютере?
6. Для чего применяются двоично-десятичные коды?
7. Чем отличаются формы представления чисел с фиксированной и плавающей точками?
8. Как кодируются знаки чисел?
9. Можно ли различить форматы целых и дробных чисел с фиксированной точкой?
10. Какие элементы формата чисел с плавающей точкой вы знаете?
11. Чем отличается нормализованное число с плавающей точкой от ненормализованного числа?
12. Для чего применяются смещенные порядки?
13. Какой прием используется для расширения диапазона чисел с плавающей точкой?
14. В чем заключается способ скрытой единицы?
15. Охарактеризуйте форматы двоично-десятичных чисел.
16. Как кодируются знаки двоично-десятичных чисел?
17. Для чего используются машинные коды чисел?
18. Чем отличаются дополнительный и обратный коды отрицательных чисел?
19. Как из дополнительного кода отрицательного числа получить прямой код числа?
20. Какую позицию запятой в формате с фиксированной запятой можно считать общепринятой?
21. Чем в формате с фиксированной запятой заполняются избыточные старшие разряды?
22. Какое минимальное количество полей должен содержать формат с плавающей запятой?
23. Как в формате с плавающей запятой решается проблема работы с порядками, имеющими разные знаки?
24. От чего зависят точность и диапазон представления чисел в формате с плавающей запятой?

Ссылки на литературные источники, приведенные в рабочей программе дисциплины: пп. 1, 2 основной литературы, интернет-источники.

Вопросы для закрепления:

1. Что такое **Вычислительная сеть**?
2. Чем является любая вычислительная сеть?
3. Что такое **Прозрачность, Интегрируемость**?

Домашнее задание:

Проработка конспекта пройденной темы.

Литература:

1. Чекмарев Ю.В. Локальные вычислительные сети : учебное пособие / Чекмарев Ю.В.. — Саратов : Профобразование, 2017. — 200 с

Тема2. Основные характеристики и требования современной вычислительной сети.

1. **Требования к современным сетям**
2. **Основные показатели сети**
3. **Способы определения характеристик**
4. **Способы коммутации**

Занятие 2 (лекция)

Ключевые вопросы – производительность, надежность, управляемость, совместимость, интегрируемость, прозрачность, коммутация сообщений, каналов, пакетов.

Предыдущие знания обучающихся: Архитектура ЭВМ и ВС

Результаты обучения: – организация целенаправленной познавательной деятельности студентов по овладению программным материалом.

Задача – изучение основных характеристик современных вычислительных сетей, овладение специальной терминологией.

Информация по теме:

Вычислительная сеть создается для обеспечения потенциального доступа к любому ресурсу сети для любого пользователя сети. Качество доступа к ресурсу может быть описано многими показателями, выбор которых зависит от задач, стоящих перед сетью.

Производительность вычислительной сети может быть оценена с разных позиций. С точки зрения пользователя, важным числовым показателем производительности сети является время реакции системы, т.е. время между моментом возникновения запроса и моментом получения ответа. Пропускная способность определяется количеством информации, переданной через сеть или ее сегмент в единицу времени и измеряемая в битах в секунду. Пропускная способность может быть мгновенной, максимальной и средней.

Задержка передачи – задержка между моментом поступления пакета на вход какого-либо устройства и моментом появления его на выходе.

Надежность работы вычислительной сети определяется надежностью работы всех ее компонентов. Для технических устройств используются такие показатели надежности, как среднее время обработки на отказ, вероятность отказа, интенсивность отказов. Для оценки надежности сложных систем применяют *готовность* или *коэффициент готовности*.

Для оценки надежности передачи пакетов используются показатели вероятности потери пакета при его передаче, либо вероятности доставки пакета. В современных сетях важна и другая сторона надежности – безопасность – способность сети обеспечить защиту информации от несанкционированного доступа. Еще одной характеристикой надежности является *отказоустойчивость*.

Любая вычислительная сеть является развивающимся объектом, и не только в плане модернизации ее элементов, но и в плане ее физического расширения, добавления новых элементов сети (пользователей, компьютеров, служб). Существование таких возможностей, трудоемкость их осуществления входят в понятие *расширяемости*. Другой похожей характеристикой является масштабируемость сети, которая определяет возможность расширения сети без существенного снижения ее производительности.

Прозрачность сети предполагает скрытие (невидимость) особенностей сети от конечного пользователя. Другой важной стороной прозрачности сети является возможность распараллеливания работы между разными элементами сети.

Интегрируемость (или *совместимость*) означает возможность подключения к вычислительной сети разнообразного и разнотипного оборудования, программного обеспечения от разных производителей.

Под коммутацией данных понимают их передачу, при которой канал данных может использоваться попеременно. Различают коммутация сообщений, каналов, пакетов.

Ссылки на литературные источники, приведенные в рабочей программе дисциплины: пп. 1, 2 основной литературы, интернет-источники.

Вопросы для закрепления:

1. Что такое **Задержка передачи**?
2. Чем является любая вычислительная сеть?
3. Что такое **Прозрачность, Интегрируемость**?

Домашнее задание:

Проработка конспекта пройденной темы.

Литература:

2. Чекмарев Ю.В. Локальные вычислительные сети : учебное пособие / Чекмарев Ю.В.. — Саратов : Профобразование, 2017. — 200 с

Раздел 2. Принципы взаимодействия компьютеров в сети

Тема 3. Компоненты вычислительной сети и способы их взаимодействия.

1. Интерфейсы взаимодействие устройств
2. Взаимодействие двух компьютеров и программных компонентов
3. Функциональные группы устройств сети

Занятие 3(лекция)

Ключевые вопросы – интерфейс, протокол, рабочая станция, узел сети, сервер, сегмент сети, коммуникационные устройства.

Предыдущие знания обучающихся: Архитектура ЭВМ и ВС

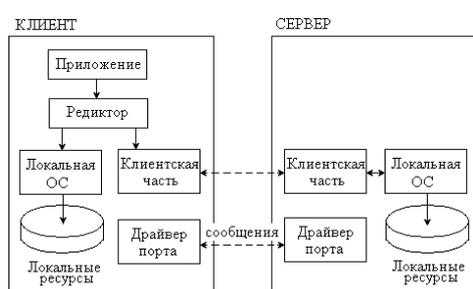
Результаты обучения: – организация целенаправленной познавательной деятельности студентов по овладению программным материалом.

Задача – изучение основных компонентов вычислительных сетей, овладение специальной терминологией.

Информация по теме:

Изучение сети в целом предполагает знание принципов работы ее отдельных компонентов: компьютеров, коммуникационного оборудования, операционных систем, сетевых приложений.

Для обмена данными между компьютером и периферийным устройством в компьютере предусмотрен внешний интерфейс, то есть набор проводов, соединяющих компьютер и периферийное устройство, а также набор правил обмена информацией по этим проводам (интерфейс или протокол). Примерами интерфейсов, используемых в компьютерах,



являются параллельный интерфейс Centronics, предназначенный, как правило, для подключения принтеров, и последовательный интерфейс RS-232C, через который подключаются мышь, модем и много других устройств. Со стороны ПУ интерфейс чаще всего реализуется аппаратным устройством управления, хотя встречаются и программно-управляемые периферийные устройства. Схема взаимодействия программных компонентов при

связи двух компьютеров приведена на рис.

Узел – любое устройство, подключенное к сети. Рабочая станция – это персональный компьютер, подключенные к сети, на котором пользователь сети выполняет свою работу. Удаленная рабочая станция – рабочая станция, подключенная к локальной сети через медленную линию связи, отличную от используемой в локальной сети.

Сервер сети – компьютер, подключенный к сети и выполняющий для пользователей сети определенные услуги. Часть сети, в которую не входит устройство расширения, принято называть сегментом сети.

Повторитель (хаб, концентратор) – устройство, позволяющее расширить сеть подключением дополнительных сегментов кабеля.

Мост – устройство, позволяющее объединить несколько сегментов, так что передача данных между станциями внутри одного сегмента не будет влиять на передачу данных в других сегментах. Коммутатор 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.

Мультиплексор – устройство, позволяющее мультиплексировать данные, приходящие одновременно от различных станций или сегментов и передавать их через передающую среду.

Маршрутизатор – устройство, соединяющее сети разного типа, но использующие одну сетевую операционную систему или протокол обмена данными.

Шлюз – устройство, позволяющее организовать обмен данными между сетевыми объектами, использующими различные протоколы обмена данными.

Домашнее задание:

Проработка конспекта пройденной темы.

Литература:

3. Чекмарев Ю.В. Локальные вычислительные сети : учебное пособие / Чекмарев Ю.В.. — Саратов : Профобразование, 2017. — 200 с

Тема 4. Топологии подключения. Логическая и физическая структуризация сети.

1. Понятие «топология». Логическая и физическая топология.
 2. Виды топологии сети
 3. Использование структурообразующего оборудования.
- Занятие 4(лекция)

Ключевые вопросы – ячеистая, шинная, кольцеобразная, звездообразные, сотовая топологии, методы логической и физической структуризации.

Предыдущие знания обучающихся: Архитектура ЭВМ и ВС

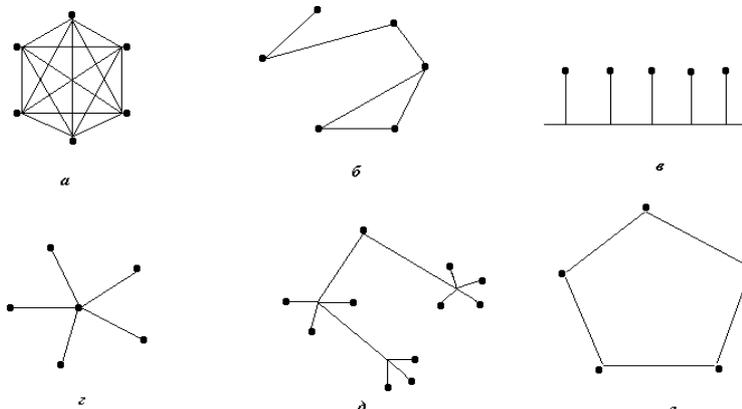
Результаты обучения – организация целенаправленной познавательной деятельности студентов по овладению программным материалом.

Задача – изучение основных принципов построения вычислительных сетей, овладение специальной терминологией.

Информация по теме:

Под топологией вычислительной сети понимается конфигурация графа, вершинам которого соответствуют компьютеры сети (иногда и другое оборудование, например концентраторы), а ребрам – физические связи между ними. Необходимо различать физическую и логическую топологии сети.

Конфигурация физических связей определяется электрическими соединениями компьютеров между собой. Логические связи представляют собой маршруты передачи данных между узлами сети и образуются путем соответствующей настройки коммуникационного оборудования.



При полно связной топологии (рис. а) каждый компьютер сети связан со всеми остальными. Ячеистая топология получается из полно связной путем удаления некоторых возможных связей

(рис. б). Шина (рис. в) – передаваемая информация может передаваться в обе стороны. В топологии звезда (рис. г) каждый компьютер подключается отдельным кабелем к общему устройству, называемому концентратором. Иногда имеет смысл строить сеть с использованием нескольких концентраторов,

иерархически соединенных друг с другом связями типа звезда (рис. д). В сетях с кольцевой конфигурацией (рис. е) данные передаются по кольцу, как правило, в одном направлении. В то время как небольшие сети, как правило, имеют типовую топологию, для крупных сетей характерно наличие произвольных связей между узлами. Их называют сетями смешанной топологии.

Сотовая топология метод разделения географической области на зоны, в каждой из которых обеспечивается обмен информацией между станциями сети, находящимися внутри зоны.

Физическая структуризация сети с помощью концентраторов полезна не только для увеличения расстояния между узлами сети, но и для повышения ее надежности.

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. Решение проблемы состоит в отказе от идеи использования единой однородной разделяемой среды. 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.

Домашнее задание:

Проработка конспекта пройденной темы.

Литература:

4. Чекмарев Ю.В. Локальные вычислительные сети : учебное пособие / Чекмарев Ю.В.. — Саратов : Профобразование, 2017. — 200 с

Раздел 3. Модель OSI

Тема 5. Стандартизация компьютерных сетей.

- 1. Виды стандартов сетей**
- 2. Ведущие организации по стандартизации компьютерных сетей**
- 3. Основные понятия открытых систем**

Занятие 5(лекция)

Ключевые вопросы – стандарт, открытая система, спецификация, организации по стандартизации сетей.

Предыдущие знания обучающихся: Архитектура ЭВМ и ВС

Результаты обучения – организация целенаправленной познавательной деятельности студентов по овладению программным материалом.

Задача – изучение основных принципов построения открытых сетей, овладение специальной терминологией.

Информация по теме:

Работы по стандартизации вычислительных сетей ведутся большим количеством организаций. В зависимости от их статуса различаются и виды стандартов: стандарты отдельных фирм, стандарты специальных комитетов и объединений, национальные стандарты, международные стандарты.

К организациям наиболее активно и успешно занимающихся разработкой стандартов в области вычислительных сетей относятся:

Международная организация по стандартизации (International Organization for Standardization (ISO);

Международный союз электросвязи (International Telecommunications Union, ITU) – организация, являющаяся специализированным органом ООН. Наиболее важную роль играет постоянно действующий в ней Международный комитет по телефонии и телеграфии (Consultative Committee on International Telegraphy and Telephony, CCITT). В

1993 году в результате реорганизации ITU он сменил название на сектор телекоммуникационной стандартизации ITU (ITU Telecommunications Standardization Sector – ITU-T).

Институт инженеров по электротехнике и радиоэлектронике – Institute of Electrical and Electronics Engineers, IEEE – национальная организация США, определяющая сетевые стандарты.

Ассоциация электронной промышленности (Electronic Industries Association, EIA) – промышленно-торговая группа производителей электронного и сетевого оборудования, национальная коммерческая ассоциация США.

Американский институт национальных стандартов (American National Standards Institute, ANSI) – представляет США в ISO.

В широком смысле открытой системой может называться любая система (компьютер, вычислительная сеть, аппаратные или программные продукты), которая построена в соответствии с открытыми спецификациями. Термин «спецификация» означает формализованное описание аппаратных или программных компонент, способ их функционирования, взаимодействия с другими компонентами, условий эксплуатации, ограничений и особых характеристик. Не всякая спецификация является стандартом. Открытые спецификации – опубликованные, общедоступные, соответствующие стандарту и принятые после всестороннего обсуждения.

Для обеспечения обмена данными между компьютерными сетями ISO совместно с ССИТТ разработала многоуровневый комплект протоколов, известный как эталонная модель взаимосвязи открытых систем (Open System Interconnection – OSI) (1977 г). Одна из основных ее идей – обеспечение относительно легкого и простого обмена информацией при использовании изготовленных разными фирмами аппаратных и программных средств, соответствующих стандартам OSI. Модель OSI касается только одного аспекта открытости – открытости взаимодействия устройств, связанных в вычислительную сеть. Ярким примером открытой системы является Internet.

Домашнее задание:

Проработка конспекта пройденной темы.

Литература:

5. Чекмарев Ю.В. Локальные вычислительные сети : учебное пособие / Чекмарев Ю.В.. — Саратов : Профобразование, 2017. — 200 с

Тема 6. Эталонная модель взаимосвязи открытых систем.

- 1. История создания модели OSI.**
- 2. Уровни модели OSI.**
- 3. Функции уровней, примеры протоколов**

Занятие 6 (лекция)

Ключевые вопросы – сетезависимые и сетезависимые уровни, единицы передаваемых данных.

Предыдущие знания обучающихся: Архитектура ЭВМ и ВС

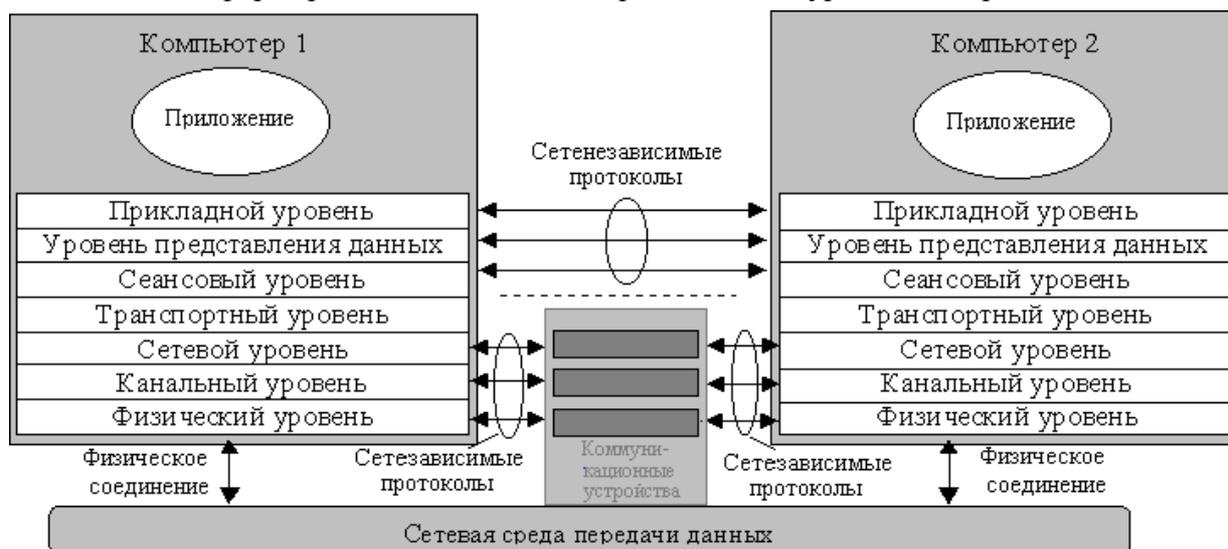
Результаты обучения – организация целенаправленной познавательной деятельности студентов по овладению программным материалом.

Задача – изучение основных принципов построения открытых сетей, овладение специальной терминологией.

Информация по теме:

Все задачи, которые необходимо решить для организации взаимодействия между объектами информационной системы, разделены на семь отдельных процедур или уровней, представленных на рисунке. Каждый уровень выполняет определенную логическую функцию и обеспечивает определенный набор услуг для расположенного над ним уровня.

Отдельные уровни модели OSI удобно рассматривать как группы программ, предназначенных для выполнения конкретных функций. При обращении некоторого сетевого приложения к прикладному уровню на основании запроса программное обеспечение формирует сообщение стандартного формата, состоящее из заголовка и поля данных. После формирования сообщения прикладной уровень направляет его вниз.



Протокол каждого уровня на основании информации, полученной из заголовка вышележащего уровня, выполняет требуемые действия и добавляет к сообщению собственную служебную информацию. Когда сообщение по сети поступает на станцию-адресат, оно принимается физическим уровнем и перемещается вверх. Последовательно каждый уровень анализирует и обрабатывает заголовок своего уровня, выполняя соответствующие данному уровню функции, а затем его удаляет.

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

Три верхних уровня – прикладной, представительский и сеансовый – ориентированы на приложения и мало зависят от технических особенностей построения сети. На протоколы этих уровней не влияют изменения в топологии сети, замена оборудования или переход на другую сетевую технологию.

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

В стандартах ISO для обозначения единиц данных, с которыми имеют протоколы разных уровней, используется общее название протокольный блок данных (*Protocol Data*

Unit, PDU). Для обозначения блоков данных определенных уровней используются специальные названия: кадр (frame), пакет (packet), дейтаграмма (datagram), сегмент (segment).

Домашнее задание:

Проработка конспекта пройденной темы.

Литература: Чекмарев Ю.В. Локальные вычислительные сети : учебное пособие / Чекмарев Ю.В.. — Саратов : Профобразование, 2017. — 200 с

Раздел 4. Типы и характеристики линий связи

Тема 7. Стеки коммуникационных протоколов.

1. Виды стандартных сетевых протоколов.
2. Соответствие протоколов уровням модели OSI.
3. Протокол TCP/IP – популярнейший протокол компьютерных сетей.

Занятие 7(лекция)

Ключевые вопросы – протоколы TCP/IP, IPX/SPX, NetBIOS/SMB, DECnet, SNA. Важнейшим направлением стандартизации в области вычислительных сетей является стандартизация коммуникационных протоколов.

Предыдущие знания обучающихся: Архитектура ЭВМ и ВС

Результаты обучения – организация целенаправленной познавательной деятельности студентов по овладению программным материалом.

Задача – изучение основных принципов построения открытых сетей, изучение стандартов, овладение специальной терминологией.

Информация по теме:

Наиболее популярными являются стеки: TCP/IP, IPX/SPX, NetBIOS/SMB, DECnet, SNA, OSI. Все стеки, кроме SNA, на физическом и канальном уровнях используют одни и те же хорошо стандартизированные протоколы Ethernet, Token Ring, FDDI и позволяют использовать во всех сетях одну и ту же аппаратуру. На верхних уровнях все стеки работают по своим собственным протоколам, которые часто не соответствуют рекомендуемой модели OSI разбиению на уровни. В таблице показано соответствие популярных стеков протоколов модели OSI.

модель OSI	IBM/ Microsoft	TCP/IP	Стек OSI
Прикладной	SMB	telnet FTP SMTP SNPT WWW	X.400 X.500 FTAM
Представительский			Представительский протокол модели OSI
Сеансовый	NetBIOS	TCP	Сеансовый протокол модели OSI
Транспортный			Транспортный протокол модели OSI
Сетевой		IP RIP OSPF	ES-ES IS-IS
Канальный	Ethernet, Token Ring, FDDI, Fast Ethernet, SLIP, X.25, ATM, PPP, 100VG-AnyLAN		
Физический	Коаксиальный кабель, экранированная и неэкранированная витая пара, волоконно-оптический кабель, радиоволны		

Стек TCP/IP – один из самых распространенных стеков транспортных протоколов вычислительных сетей. Основными протоколами стека являются, давшие ему название, протоколы IP и TCP. Они относятся соответственно к сетевому и транспортному уровням. IP обеспечивает продвижение пакета по составной сети, TCP гарантирует надежность его доставки. TCP/IP вобрал в себя популярные протоколы прикладного уровня – протокол пересылки файлов FTP, протокол эмуляции терминала telnet, почтовый протокол SMTP. Особенностью технологии TCP/IP является гибкая система адресации, позволяющая достаточно просто включать в интерес сети других технологий. Однако мощные функциональные возможности стека TCP/IP требуют для своей реализации высоких вычислительных затрат и предъявляют высокие требования к администрированию IP-сетей.

Ссылки на литературные источники, приведенные в рабочей программе дисциплины: пп. 1, 2 основной литературы, интернет-источники, периодические издания.

Домашнее задание:

Проработка конспекта пройденной темы.

Литература:Чекмарев Ю.В. Локальные вычислительные сети : учебное пособие / Чекмарев Ю.В.. — Саратов : Профобразование, 2017. — 200 с

Тема 8. Линии связи

1. Типы линий связи.
2. Характеристики кабельных линий связи и способы их определения.
3. Стандарты кабельных линий связи.
4. Стандарты беспроводной связи.

Занятие 8 (лекция)

Ключевые вопросы – среда передачи данных, кабельные системы, стандарты физического уровня.

Предыдущие знания обучающихся: Архитектура ЭВМ и ВС

Результаты обучения – организация целенаправленной познавательной деятельности студентов по овладению программным материалом.

Задача – изучение основных типов кабельной системы сетей, изучение их стандартов, овладение специальной терминологией.

Информация по теме:

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

В зависимости от среды передачи данных линии связи разделяют на проводные, кабельные (медные, волоконно-оптические), радиоканалы наземной и спутниковой связи.

В компьютерных сетях применяются типы кабеля: на основе скрученных пар медных проводов, коаксиальные с медной жилой и волоконно-оптические. Основные характеристики кабельных линий связи: амплитудно-частотная характеристика, полоса пропускания, затухание, помехоустойчивость, перекрестные наводки на ближнем конце линии, пропускная способность, достоверность передачи данных, удельная стоимость.

Новый стандарт EIA/TIA-568A коаксиальные кабели не описывает, как морально устаревшие.

Медный неэкранированный кабель UTP делится на 5 категорий (Category 1 - Category 5).

Основным стандартом неэкранированной витой пары STP является фирменный стандарт IBM, в котором кабели делятся не на категории, а на типы: Type 1, Type 2, . . . , Type 9.

В зависимости от распределения показателя преломления и от величины диаметра сердечника различают: многомодовое волокно со ступенчатым изменением показателя преломления, многомодовое волокно с плавным изменением показателя преломления, одномодовое волокно.

В спутниковых системах используются антенны СВЧ-диапазона частот для приема радиосигналов от наземных станций и ретрансляции этих сигналов обратно на наземные станции.

LMDS (Local Multipoint Distribution System) - это стандарт сотовых сетей беспроводной передачи информации для фиксированных абонентов.

Радиоканалы WiMAX (Worldwide Interoperability for Microwave Access) в отличие от традиционных технологий радиодоступа работают и на отраженном сигнале, вне прямой видимости базовой станции.

Радиоканалы MMDS (Multichannel Multipoint Distribution System). Эти системы способна обслуживать территорию в радиусе 50—60 км, при этом прямая видимость передатчика оператора является не обязательной.

Стандартом беспроводной связи для локальных сетей является технология Wi-Fi. Wi-Fi обеспечивает подключение в двух режимах: точка-точка (для подключения двух ПК) и инфраструктурное соединение (для подключения несколько ПК к одной точке доступа). Радиоканалы Bluetooth – это технология передачи данных на короткие расстояния (не более 10 м) и может быть использована для создания домашних сетей.

Ссылки на литературные источники, приведенные в рабочей программе дисциплины: пп. 1, 2 основной литературы, интернет-источники, периодические издания.

Домашнее задание:

Проработка конспекта пройденной темы.

Литература:Чекмарев Ю.В. Локальные вычислительные сети : учебное пособие / Чекмарев Ю.В.. — Саратов : Профобразование, 2017. — 200 с

Тема 9. Методы и каналы передачи данных

1. Методы передачи данных.
2. Аналоговые каналы передачи данных.
3. Способы модуляции. Модемы.
4. Цифровые каналы передачи данных.
5. Кодирование информации. Способы контроля передачи данных

Занятие 9(лекция)

Ключевые вопросы – аналоговые и цифровые каналы, потенциальное и импульсное кодирование, симплексная, дуплексная, полудуплексная передача, методы кодирования.

Предыдущие знания обучающихся: Архитектура ЭВМ и ВС

Результаты обучения – организация целенаправленной познавательной деятельности студентов по овладению программным материалом.

Задача – изучение основ передачи данных в компьютерных сетях.

Информация по теме:

Так как аппаратура передачи и приема данных работает с данными в дискретном виде (т.е. единицам и нулям данных соответствуют дискретные электрические сигналы), то при их передаче через аналоговый канал требуется преобразование дискретных данных в аналоговые (модуляция).

При приеме таких аналоговых данных необходимо обратное преобразование – демодуляция. Модуляция/демодуляция – процессы преобразования цифровой информации в аналоговые сигналы и наоборот. При модуляции информация представляется синусоидальным сигналом той частоты, которую хорошо передает канал передачи данных. К способам модуляции относятся: амплитудная модуляция; частотная модуляция; фазовая модуляция.

При передаче дискретных сигналов через цифровой канал передачи данных используется кодирование: потенциальное; импульсное. Потенциальное или импульсное кодирование применяется на каналах высокого качества, а модуляция на основе синусоидальных сигналов предпочтительнее в тех случаях, когда канал вносит сильные искажения в передаваемые сигналы.

При обмене данными между узлами вычислительных сетей используются три метода передачи данных: симплексная (однаправленная) передача (телевидение, радио); полудуплексная (прием/передача информации осуществляется поочередно); дуплексная (двунаправленная), каждый узел одновременно передает и принимает.

коды включают в себя также коды проверки ошибок и другую информацию.

Существует три принципиально различные схемы коммутации в вычислительных сетях: коммутация каналов; коммутация пакетов; коммутация сообщений.

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.

Для обнаружения искажений наиболее популярны методы, основанные на циклических избыточных кодах (CRC), которые выявляют многократные ошибки. Для восстановления кадров используется метод повторной передачи на основе квитанций. Этот метод работает по алгоритму с простоями источника, а также по алгоритму скользящего ок-

на. Для повышения полезной скорости передачи данных в сетях применяется динамическая компрессия данных на основе различных алгоритмов. Коэффициент сжатия зависит от типа данных и применяемого алгоритма и может колебаться в пределах от 1:2 до 1:8.

Ссылки на литературные источники, приведенные в рабочей программе дисциплины: пп. 1, 2 основной и дополнительной литературы, интернет-источники.

Домашнее задание:

Проработка конспекта пройденной темы.

Литература:Чекмарев Ю.В. Локальные вычислительные сети : учебное пособие / Чекмарев Ю.В.. — Саратов : Профобразование, 2017. — 200 с

Раздел 5. Базовые стандарты и технологии локальных сетей

Тема 10. Стандарты и технологии локальных сетей.План

1. История создания стандартов
2. Структура и содержание стандартов IEEE 802.x

Занятие 10 (лекция)

Ключевые вопросы – комитет 802 по стандартизации локальных сетей, структура его стандартов.

Предыдущие знания обучающихся: Архитектура ЭВМ и ВС

Результаты обучения – организация целенаправленной познавательной деятельности студентов по овладению программным материалом.

Задача – изучение стандартов и технологий организации локальных вычислительных сетей.

Информация по теме:

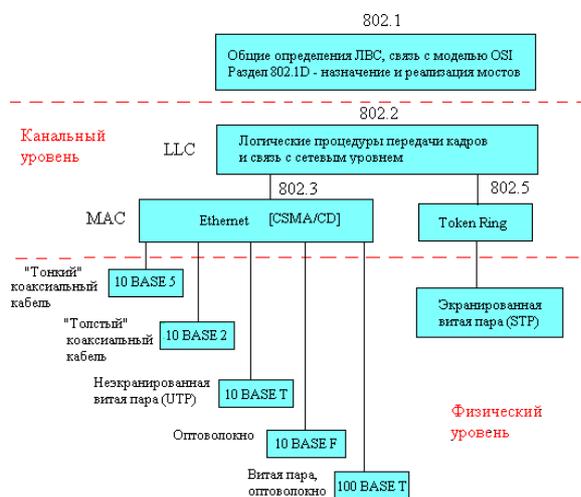
В 1980 году в институте IEEE был организован комитет 802 по стандартизации локальных сетей. Результатом его работы стало семейство стандартов IEEE 802.x, содержащих рекомендации по проектированию нижних уровней локальных сетей.

Стандарты IEEE 802 имеют достаточно четкую структуру, приведенную на рисунке.

Специфика локальных сетей нашла свое отражение в разделении канального уровня на два подуровня: логической передачи данных (Logical Link Control – LLC),управления доступом к среде (MediaAccess Control,MAC).

Уровень MAC появился из-за существования в ЛВС разделяемой среды передачи данных.

Этот уровень обеспечивает корректное совместное использование общей среды,



предоставляя ее в соответствии с определенным алгоритмом в распоряжении той или иной станции сети. После того как доступ к среде получен, ею может пользоваться более высокий уровень – LLC, организующий передачу логических единиц данных с различным уровнем качества транспортных услуг. В современных ЛВС получили распространение несколько протоколов уровня MAC, реализующих различные алгоритмы доступа к передающей среде. Эти протоколы полностью определяют специфику отдельных технологий.

Стандарты 802.3, 802.4, 802.5 и 802.12 описывают технологии локальных сетей, полученные в результате улучшений фирменных технологий, легших в их основу. К другим известным стандартам относятся:

802.6 – Metropolitan Area Network , MAN – сети мегаполисов;

802.8 – Fiber Optic Technical Advisory Group – техническая консультационная группа по волоконно-оптическому кабелю;

802.9 – Integrated Voice and data Networks – интегрированная среда передачи голоса и данных;

802.10 – Network Security – сетевая безопасность;

802.11 – Wireless Networks – беспроводные сети.

Ссылки на литературные источники, приведенные в рабочей программе дисциплины: пп. 1, 2 основной литературы, интернет-источники.

Домашнее задание:

Проработка конспекта пройденной темы.

Литература: Чекмарев Ю.В. Локальные вычислительные сети : учебное пособие / Чекмарев Ю.В. — Саратов : Профобразование, 2017. — 200 с

Тема 11. Локальная сеть Ethernet.

1. История создания сети Ethernet.
2. Основные технологии.
3. Принципы подключения устройств

Занятие 11 (лекция)

Ключевые вопросы - стандарт IEEE 802.3, технологии Ethernet.

Предыдущие знания обучающихся: Архитектура ЭВМ и ВС

Результаты обучения – организация целенаправленной познавательной деятельности студентов по овладению программным материалом.

Задача – изучение стандартов и технологий организации ЛВС.

Информация по теме:

Ethernet – самый распространенный в настоящее время стандарт локальных сетей. Ethernet версии 1 появилась в 1980 г. В 1982 году была опубликована спецификация на Ethernet версии 2.0. Обе версии используются до сих пор, причем между ними существуют различия и по интерфейсу, и по уровню сигналов. На базе Ethernet версии 2 институтом IEEE был разработан стандарт IEEE 802.3. В 1995 году был принят стандарт Fast Ethernet, который во многом не является самостоятельным стандартом. Его описание 802.3u является дополнительным разделом к основному стандарту. Аналогично, принятый в 1998 году стандарт Gigabit Ethernet описан в разделе 802.z основного документа. В таблице приведены данные «классических» технологий сети Ethernet.

Ethernet	Thick Wire Ethernet	Thin Wire Ethernet	UTP Ethernet	Fiber Optic Ethernet	Broadband Ethernet
IEEE 802.3	10BASE-5	10BASE-2	10BASE-T	10BASE-F	10BROAD36
Скорость передачи данных, Мбит/с	10	10	10	10	10
Метод передачи сигналов	Одно-Полосной	Одно-полосной	Одно-полосной	Одно-полосной	Широко-полосной
Длина сегмента кабеля, м	500	185	100	2000	1800
Максим. расстояние между узлами сети (при использовании повторителей), м	2500	925	500	2500	3600
Максимальное число станций на сегменте	100	30	1024	1024	100
Максим. число повторителей Между любыми станциями сети	4	4	4	4	4
Тип кабеля	коаксиальный, «толстый» RG-8 или RG-11	коаксиальный, «тонкий» RG-58	Неэкранированная витая пара категории 3, 4, 5	Многомодовый волоконно-оптический кабель	75 Омный коаксиальный, «толстый»

АИ - Адрес Источника - адрес передающей станции.

Поле Типа Пакета определяет тип Ethernet пакета.

Поле Длины Пакета определяет в IEEE 802.3 количество байт данных в поле данных. Количество символов-заполнителей (PAD) не входит в это число.

Поле Данных содержит данные и символы-заполнители в IEEE 802.3, данные - в Ethernet.

ПОО Поле Обнаружения Ошибок служит для определения достоверности полученной информации.

IEEE 802.3 определяет, что после поля адреса источника следует двухбайтное поле длины пакета, а в Ethernet - поле типа пакета.

Домашнее задание:

Проработка конспекта пройденной темы.

Литература: Чекмарев Ю.В. Локальные вычислительные сети : учебное пособие / Чекмарев Ю.В.. — Саратов : Профобразование, 2017. — 200 с

Тема 13. Высокоскоростные варианты сети Ethernet

1. История создания высокоскоростных вариантов
2. Технология Fast Ethernet
3. Технология Гигабит Ethernet
4. Технология 10 Гигабит Ethernet

Занятие 13 (лекция)

Ключевые вопросы – история создания и современные высокоскоростные технологии.

Предыдущие знания обучающихся: Архитектура ЭВМ и ВС

Результаты обучения- организация целенаправленной познавательной деятельности студентов по овладению программным материалом.

Задача – изучение принципов взаимодействия устройств в сети Ethernet.

Информация по теме:

К ранним высокоскоростным вариантам относятся: коммутированная Ethernet, дуплексная Ethernet, 100BaseVG AnyLAN. Первые две использовались лишь в качестве временного решения, в 100BaseVG AnyLAN используется другой метод доступа – обработка

запросов по приоритету, что усложняет совместимость с существующими сетями Ethernet.

Технология 100BaseX или Fast Ethernet реализуется с помощью дуплексной передачи сигналов по традиционной для Ethernet схеме доступа с контролем несущей и обнаружением коллизий в комбинации с уровнем зависимости от физической среды. Она принята в 1995 году в качестве дополнения 802.3u к существующему стандарту 802.3.

100BASE-T - общий термин для обозначения стандартов, использующих в качестве среды передачи данных витую пару. Длина сегмента до 100 метров. Включает в себя стандарты 100BASE-TX, 100BASE-T4 и 100BASE-T2.

100BASE-TX, IEEE 802.3u – развитие стандарта 10BASE-T для использования в сетях топологии "звезда". Задействована витая пара категории 5, фактически используются только две неэкранированные пары проводников, поддерживается дуплексная передача данных, расстояние до 100 м.

Более сложная структура физического уровня технологии Fast Ethernet вызвана тем, что в ней используется три варианта кабельных систем: волоконно-оптический многомодовый кабель (2 волокна), 2-х парная витая пара категории 5; 4-х парная витая пара категории 3. Fast Ethernet всегда имеет иерархическую древовидную структуру, построенную на концентраторах, подобно технологиям 10Base-T и 10Base-F. Диаметр сети сокращен до 200 метров, что объясняется уменьшением времени передачи кадра минимальной длины.

При использовании коммутаторов протокол Fast Ethernet может работать в полнодуплексном режиме, в котором нет ограничения на длину сети, а существуют лишь ограничения на длину физических сегментов. Формат кадра не отличается от формата 10-мегабитного Ethernet.

Gigabit Ethernet – поддерживает скорость обмена до 1 Гбит/с. Сеть Gigabit Ethernet представляет собой ЛВС, в которой применяется многомодовый оптоволоконный кабель, но IEEE изучает методы ее применения на сетях с UTP 5 категории. Топология звездообразная. Оптоволоконный кабель используется для магистрали, коаксиальный кабель для отводов к концентраторам.

1000BASE-T, IEEE 802.3ab – стандарт, использующий витую пару категорий 5е. В передаче данных участвуют все 4 пары. Скорость передачи данных – 250 Мбит/с по одной паре. Используется метод кодирования PAM5, частота основной гармоники 62,5 МГц.

Новый стандарт 10 Гигабит Ethernet включает в себя семь стандартов физической среды для LAN, MAN и WAN. В настоящее время он описывается поправкой IEEE 802.3ae и должен войти в следующую ревизию стандарта IEEE 802.3.

Ссылки на литературные источники, приведенные в рабочей программе дисциплины: пп. 1, 2 основной литературы, интернет-источники, периодические издания.

Домашнее задание:

Проработка конспекта пройденной темы.

Литература:Чекмарев Ю.В. Локальные вычислительные сети : учебное пособие / Чекмарев Ю.В.. — Саратов : Профобразование, 2017. — 200 с

Тема 14. Сеть Token Ring

1. История создания стандарта Token Ring. Основные характеристики
2. Метод доступа станций к передающей среде
3. Взаимодействие устройств в сети, способы их подключения
4. Типы пакетов, форматы

Занятие 14 (лекция)

Ключевые вопросы – маркерное кольцо, однократно и двукратно подключаемые устройства.

Предыдущие знания обучающихся: Архитектура ЭВМ и ВС

Результаты обучения – организация целенаправленной познавательной деятельности студентов по овладению программным материалом.

Задача – изучение принципов взаимодействия устройств в сети Token Ring.

Информация по теме:

Крупные предприятия имеют сети смешанной структуры. Доминирует среди них Ethernet, но присутствует и большое количество сетей Token Ring (маркерное кольцо).

Этот стандарт разработан фирмой IBM в 1984 году. В качестве передающей среды применяется неэкранированная или экранированная витая пара (UTP или STP) или оптоволокно. Скорость передачи данных 4 Мбит/с или 16 Мбит/с. В качестве метода управления доступом станций к передающей среде используется метод маркерное кольцо (Token Ring). Основные положения этого метода:

- 1) устройства подключаются к сети по топологии кольцо;
- 2) все устройства, подключенные к сети, могут передавать данные, только получив разрешение на передачу (маркер);
- 3) в любой момент времени только одна станция в сети обладает таким правом.

На базе IBM Token Ring в 1985 году комитетом IEEE 802 был разработан стандарт IEEE 802.5. Этот стандарт в отличие от IBM Token Ring не определяет среду передачи данных и топологии подключения устройств.

В IBM Token Ring используются три основных типа пакетов: пакет управление/данные, маркер, пакет сброса.

Станция, которая приняла маркер, получает право на передачу, и может передавать данные. Для этого станция удаляет маркер из кольца, формирует пакет данных и передает его следующей станции. В сети Token Ring все станции принимают и ретранслируют все пакеты, проходящие по кольцу. При приеме станция сравнивает поле адреса приемника пакета с собственным адресом. Если адреса не совпадают, то пакет передается далее по кольцу без изменений. Если адреса совпадают, или принят пакет с широковещательным адресом, то содержимое пакета копируется, а по результатам приема данных вносятся изменения в поле статуса пакета. Затем пакет передается далее по сети, и таким образом возвращается на станцию - отправитель. Получив пакет, станция-отправитель проверяет поле статуса пакета. Если при приеме пакета станцией-приемником была обнаружена ошибка, выполняется повторная передача пакета. Если ошибок при приеме не было или станция-приемник в данный момент не работает, станция - отправитель удаляет пакет из кольца, формирует маркер и передает его следующей станции. Такой алгоритм доступа применяется в сетях Token Ring со скоростью передачи 4 Мбит/с, описанный в стандарте 802.5. В сетях Token Ring 16 Мбит/с используется алгоритм раннего освобождения маркера.

Логически сеть Token Ring представляет собой кольцо, а физически – звезду, в которой устройства подключаются к сети через специальный концентратор.

Технология позволяет использовать различные типы кабелей: STP Type1, UTP Type 3 и 6, а также волоконно-оптический кабель. Максимальное расстояние от станции

до MSAU – 100 м для STP и 45 м для UTP. Возможно также расширение сети с помощью мостов и повторителей. Сеть Token Ring может включать 260 узлов.

Ссылки на литературные источники, приведенные в рабочей программе дисциплины: пп. 1, 2 основной литературы, интернет-источники, периодические издания.

Домашнее задание:

Проработка конспекта пройденной темы.

Литература:Чекмарев Ю.В. Локальные вычислительные сети : учебное пособие / Чекмарев Ю.В.. — Саратов : Профобразование, 2017. — 200 с

Тема15. Распределенный волоконно-оптический интерфейс передачи данных FDDI.

1. История создания стандарта FDDI. Основные характеристики
2. Метод доступа станций к передающей среде
3. Взаимодействие устройств в сети, способы их подключения
4. Типы пакетов, форматы

Занятие 15(лекция)

Ключевые вопросы – маркерное кольцо, однократно и двукратно подключаемые устройства.

Предыдущие знания обучающихся: Архитектура ЭВМ и ВС

Результаты обучения: организация целенаправленной познавательной деятельности студентов по овладению программным материалом.

Задача – изучение принципов взаимодействия устройств в сети FDDI.

Информация по теме:

Стандарт был разработан в середине 80-х годов Национальным Американским Институтом Стандартов (ANSI) и получил номер ANSI X3T9.5.

FDDI обеспечивает передачу данных со скоростью 100 Мбит/с по двойному волоконно-оптическому кольцу на расстояние до 100 км.

Стандарт FDDI определяет кольцевую топологию с маркерным доступом, способную охватить большую площадь. Этот стандарт обеспечивает совместимость с сетями Token Ring, поскольку форматы кадров у них одинаковы.

Стандарт FDDI определяет перечень компонентов сети, который включает однократно подключенную станцию (SAS – Single Attached Station), двукратно подключенную станцию (DAS – Dual Attached Station) и концентраторы проводных линий. Соединения SAS с концентраторами имеют топологию звезды.

Интерфейс двукратного подключения обеспечивает отказоустойчивость системы благодаря своей избыточности. В случае разрыва кабеля сеть выполняет “заворачивание” – включает второе кольцо для обхода отказавшей станции. Сеть продолжает работать, но ее производительность падает.

Существуют также версии FDDI на медных проводах. «Медные» варианты FDDI не являются как таковыми стандартами и не могут гарантировать совместимость. Второе ограничение в “медных” версиях и даже в самом FDDI заключается в том, что здесь используется иная топология(двойное кольцо) и и другой размер пакета, нежели например в Ethernet.

Максимальное количество станций двойного подключения в кольце – 500. Максимальные расстояния между соседними узлами для многомодового кабеля – 2 км, для витой пары UTP категории 5 – 100 м, а для многомодового волокна зависит от его качества.

Максимальный размер кадра – 4500 байтов. Формат кадра FDDI:

Преамбула	Началн. разделитель	Контроль кадра	Адрес получателя	Адрес отправителя	Данные	Последовательность проверки кадра	Конечный разделитель	Статус кадра
-----------	---------------------	----------------	------------------	-------------------	--------	-----------------------------------	----------------------	--------------

Поле начального разделителя обозначает начало кадра. Поле контроля кадра содержит данные о режиме передачи (синхронная или асинхронная), размере адреса и типе кадра (кадр LLC или управляющий кадр MAC). В поле последовательности проверки кадра для контроля ошибок используется 32-разрядный циклический избыточный код. Поле конечного разделителя означает конец кадра. Поле статуса кадра используется для индикации обнаружения ошибки.

Ссылки на литературные источники, приведенные в рабочей программе дисциплины: пп. 1, 2 основной литературы, интернет-источники, периодические издания.

Домашнее задание:

Проработка конспекта пройденной темы.

Литература: Чекмарев Ю.В. Локальные вычислительные сети : учебное пособие / Чекмарев Ю.В.. — Саратов : Профобразование, 2017. — 200 с

Тема 16. Использование мостов в локальных сетях.

1. Построение ЛВС с помощью мостов и коммутаторов.
2. Алгоритмы маршрутизации.

Занятие 16(лекция)

Ключевые вопросы – алгоритм работы прозрачного моста, алгоритм с маршрутизацией от источника, алгоритм мостового дерева.

Предыдущие знания обучающихся: Архитектура ЭВМ и ВС

Результаты обучения – организация целенаправленной познавательной деятельности студентов по овладению программным материалом.

Задача – изучение принципов работы и преимущества использования мостов в локальных сетях.

Информация о теме:

Мосты функционируют на подуровне управления доступом к среде передачи канального уровня модели OSI.

При проектировании сетей мосты являются необходимыми элементами, с их помощью обеспечивается повышение эффективности, безопасности и дальности.

Чаще всего мосты устанавливаются в целях повышения эффективности. Администратор сети может воспользоваться мостом для уменьшения перегрузки и повышения быстродействия: большая сеть делится на несколько подсетей, соединенные мостами. Две небольшие сети будут работать быстрее, поскольку трафик локализуется в пределах подсети, кроме этого они более просты в управлении и обслуживании. Использование мостов приводит к повышению эффективности работы сети еще и потому, что разработчик может использовать разные топологии среды передачи, а затем соединить эти сети посредством мостов. Поскольку комитет IEEE 802 разработал для различных сетевых архитектур общий уровень управления логическим каналом, то существует возможность объединения, например, двух сетей Token Ring, разделенных ЛВС Ethernet. ЛВС Ethernet может пересылать пакеты так же, как почтальон может доставлять письма, написанные на иностранном языке, если конверты (пакеты) оформлены в соответствии со стандартом.

Мосты часто используют в целях повышения безопасности, т.к. программируя мосты на передачу только тех пакетов, которые содержат определенные адреса отправителя и получателя, можно ограничить круг рабочих станций, которые могут посылать и принимать информацию из другой подсети, тем самым предотвращая несанкционированный доступ. Здесь часто используют интеллектуальные мосты.

Мосты можно также использовать и в целях повышения отказоустойчивости системы, если с помощью внутренних мостов связать два файловых сервера, которые постоянно будут подстраховывать друг друга, причем при этом снизится уровень трафика.

Мосты увеличивают дальность охвата сети. Поскольку мост ретранслирует пакет в широкополосном режиме, то он функционирует как повторитель.

Алгоритмы работы мостов:

алгоритм работы прозрачного моста, в котором мост строит адресную таблицу на основании пассивного наблюдения за трафиком.

Алгоритм мостового дерева разработан фирмами DEC и Vitalink, а в последствии принят комитетом IEEE 802.1 как стандарт. Этот алгоритм применяется для объединения мостами нескольких Ethernet сетей при возможности существования более одного неупущенного пути.

Алгоритм с маршрутизацией от источника используется в сетях Token Ring и FDDI, в нем ответственность за разработку для кадра полного маршрута возлагается на рабочую станцию – источник. Станция-отправитель помещает в посылаемый в другое кольцо кадр всю адресную информацию о промежуточных мостах и кольцах. При получении каждого пакета мост просматривает поле маршрутной информации на наличие в нем своего идентификатора.

Домашнее задание:

Проработка конспекта пройденной темы.

Литература: Чекмарев Ю.В. Локальные вычислительные сети : учебное пособие / Чекмарев Ю.В.. — Саратов : Профобразование, 2017. — 200 с

Тема 17. Сетевые устройства

Занятие 17 (лекция)

Ключевые вопросы – Сетевые устройства и их назначения.

Предыдущие знания обучающихся: Архитектура ЭВМ и ВС

Результаты обучения – организация целенаправленной познавательной деятельности студентов по овладению программным материалом.

Задача – изучение принципов работы и преимущества использования сетевых устройств.

Информация о теме:

При объединении большого числа рабочих станций наиболее целесообразным является создание в отдельных местах локальных сетей, которые затем объединять между собой. Для соединения между собой различных локальных сетей требуются устройства, которые управляют потоками информации:

- Концентратор (Hub)
- Коммутатор (Switch)
- Маршрутизатор (Router)
- Мост (Bridge)
- Повторитель (Repeater)
- Точка доступа (Access Point)
- Брандмауэр (Firewall).

Самый простой способ построения локальных сетей - это использование **повторителей**, которые реализуют сетевое соединение путём повторения электрического сигнала «один в один». Бывают однопортовые повторители и многопортовые. К портам присоединяются кабели. При этом повторитель должен принимать сигнал, далее распознавать его первоначальный вид, и генерировать на выходе его точную копию. При этом может возникнуть проблема, при которой по двум и более портам приходят пакеты в одно и то же время. Другая проблема - безопасность - все пакеты доходят до всех компьютеров сети, поэтому существует возможность несанкционированного доступа к информации. И, наконец, ещё одной проблемой является то, что копирование пакетов повышает нагрузку на сеть, причём весьма существенно (весь трафик сегмента сети поступает к каждому из компьютеров и тем самым загружает сеть).

Коммутаторы умеют конфигурировать персонально каждый сегмент сети и устанавливать соответствующий режим работы. При приёме/передачи пакетов данных они не отправляют его сразу во все выходные порты, а лишь в те, которые подсоединены к устройствам готовым его принять. Коммутатор хранит в памяти таблицу коммутации, в которой указывается соответствие MAC-адреса (уникальный идентификатор, присваиваемый каждой единице оборудования компьютерных сетей) узла порту коммутатора. Коммутаторы производят передачу на основании MAC-адресов устройств по таблицам поиска интерфейсов связанных с соответствующим MAC-адресом. Обычно коммутаторы используют в сетях с простой топологией в виде звезды, при которой рабочая станция связана напрямую в дуплексном режиме со всеми другими рабочими станциями. Такое решение в настоящее время является наиболее распространённым в локальных сетях (при определённых ограничениях на число рабочих станций).

Маршрутизаторы сетевое устройство, пересылающее пакеты данных между различными сегментами сети и принимающее решения на основании информации о топологии сети. Обычно маршрутизатор использует IP адрес получателя, указанный в пакетах данных, и определяет по таблице маршрутизации наиболее подходящий путь, по которому следует передать

данные. Если в таблице маршрутизации для адреса нет описанного маршрута, пакет отбрасывается. Маршрутизатор объединяет по крайней мере две различные сети.

Мост - сетевое оборудование, предназначенное для объединения сегментов (подсети) компьютерной сети разных топологий и архитектур. В общем случае коммутатор (свитч) и мост аналогичны по функциональности; разница заключается во внутреннем устройстве: мосты обрабатывают трафик, используя центральный процессор, коммутатор же использует коммутационную матрицу (аппаратную схему для коммутации пакетов). Мост работает в OSI модели на 2 уровне (MAC-уровень) и прозрачен для сетевых устройств более высокого уровня.

Точка доступа - устройство для объединения компьютеров в единую беспроводную сеть.

Брандмауэр (сетевой экран) есть устройство, препятствующее несанкционированному перемещению данных между сетями. Также сетевые экраны часто называют фильтрами, так как их основная задача - не пропускать (фильтровать) пакеты, не подходящие под критерии, определённые в конфигурации. Современные брандмауэры позволяют настраивать сеть. Например, они могут закрывать порты, которые реально не используются. Брандмауэр может быть как устройством, так и программой, защищающей вход отдельного устройства сети.

Вопросы для закрепления:

1. Что такое концентратор, коммутатор, маршрутизатор, мост?
2. Чем является Бранмауэр?
3. Что такое Точка доступа?.

Домашнее задание:

Проработка конспекта пройденной темы.

Литература:

6. Чекмарев Ю.В. Локальные вычислительные сети : учебное пособие / Чекмарев Ю.В.. — Саратов : Профобразование, 2017. — 200 с

Тема18 . Глобальные сети.

1. Принципы организации глобальных сетей. Классификация.
2. Отличие глобальных сетей от локальных
3. Организация корпоративных сетей.

Занятие 18(лекция)

Ключевые вопросы – технологии глобальных сетей, магистральные сети и сети доступа.

Предыдущие знания обучающихся: Архитектура ЭВМ и ВС

Результаты обучения – организация целенаправленной познавательной деятельности студентов по овладению программным материалом.

Задача – изучение принципов работы глобальных сетей.

Информация о теме:

Глобальные сети обычно создаются крупными телекоммуникационными компаниями для оказания платных услуг абонентам. Такие сети называют публичными или общественными. Существуют также такие понятия, как оператор сети и поставщик услуг сети. Гораздо реже глобальная сеть полностью создается какой-нибудь крупной корпорацией (такой, например, как

Dow Jones или «Транснефть») для своих внутренних нужд. В этом случае сеть называется частной. Очень часто встречается и промежуточный вариант - корпоративная сеть пользуется услугами или оборудованием общественной глобальной сети, но дополняет эти услуги или оборудование своими собственными. Наиболее типичным примером здесь является аренда каналов связи, на основе которых создаются собственные территориальные сети.

Кроме вычислительных глобальных сетей существуют и другие виды территориальных сетей передачи информации. В первую очередь это телефонные и телеграфные сети, работающие на протяжении многих десятков лет, а также телексная сеть. Ввиду большой стоимости глобальных сетей существует долговременная тенденция создания единой глобальной сети, которая может передавать данные любых типов. Существенного прогресса в этой области не достигнуто, хотя технологии для создания таких сетей начали разрабатываться достаточно давно - первая технология для интеграции телекоммуникационных услуг ISDN стала развиваться с начала 70-х годов. Пока каждый тип сети существует отдельно и наиболее тесная их интеграция достигнута в области использования общих первичных сетей - сетей PDH и SDH, с помощью которых сегодня создаются постоянные каналы в сетях с коммутацией абонентов. Тем не менее, каждая из технологий, как компьютерных сетей, так и телефонных, старается сегодня передавать «чужой» для нее трафик с максимальной эффективностью, а попытки создать интегрированные сети на новом витке развития технологий продолжают под преемственным названием Broadband ISDN (B-ISDN), то есть широкополосной (высокоскоростной) сети с интеграцией услуг. Сети B-ISDN будут основываться на технологии ATM, как универсальном транспорте, и поддерживать различные службы верхнего уровня для распространения конечным пользователям сети разнообразной информации - компьютерных данных, аудио- и видеoinформации, а также организации интерактивного взаимодействия пользователей.

Хотя в основе локальных и глобальных вычислительных сетей лежит один и тот же метод - метод коммутации пакетов, глобальные сети имеют достаточно много отличий от локальных сетей. Эти отличия касаются как принципов работы (например, принципы маршрутизации почти во всех типах глобальных сетей, кроме сетей TCP/IP, основаны на предварительном образовании виртуального канала), так и терминологии.

Технологии глобальных сетей определяют два типа интерфейса: «пользователь-сеть» (UNI) и «сеть-сеть» (NNI).

Глобальные компьютерные сети работают на основе технологии коммутации пакетов, кадров и ячеек.

Глобальные сети делятся на магистральные сети и сети доступа.

Вопросы для закрепления:

1. Каков принцип работы глобальных сетей?
2. Чем является ATM?
3. Что такое UNI и NNI?.

Домашнее задание:

Проработка конспекта пройденной темы.

Литература:

7. Чекмарев Ю.В. Локальные вычислительные сети : учебное пособие / Чекмарев Ю.В.. —

Тема19. Управление сетью

1. Протоколы управления.
2. Сетевые операционные системы.

Занятие 19(лекция)

Ключевые вопросы – технологии глобальных сетей, магистральные сети и сети доступа.

Предыдущие знания обучающихся: Архитектура ЭВМ и ВС

Результаты обучения – организация целенаправленной познавательной деятельности студентов по овладению программным материалом.

Задача – изучение принципов управления работой вычислительных сетей.

Информация о теме:

Типичная сеть состоит из множества устройств различных производителей. Управлять такой сетью возможно только при наличии стандартного, не зависящего от производителя протокола управления. Самым популярным стандартным протоколом управления в современных сетях является простой протокол управления сетью (Simple Network Management Protocol, SNMP). Широкое распространение он получил в силу своей гибкости и расширяемости — SNMP позволяет описывать объекты для самых разных устройств.

Обычно при использовании SNMP присутствуют управляемые и управляющие системы. В состав управляемой системы входит компонент, называемый *агентом*, который отправляет отчёты управляющей системе. По существу SNMP агенты передают управленческую информацию на управляющие системы как переменные (такие как «свободная память», «имя системы», «количество работающих процессов»).

Управляющая система может получить информацию через операции протокола GET, GETNEXT и GETBULK. Агент может самостоятельно без запроса отправить данные, используя операцию протокола TRAP или INFORM. Управляющие системы могут также отправлять конфигурационные обновления или контролирующие запросы, используя операцию SET для непосредственного управления системой. Операции конфигурирования и управления используются только тогда, когда нужны изменения в сетевой инфраструктуре. Операции мониторинга обычно выполняются на регулярной основе. Переменные, доступные через SNMP, организованы в иерархии. Эти иерархии и другие метаданные (такие, как тип и описание переменной) описываются Базами Управляющей Информации.

Сетевая операционная система обеспечивает поддержку сетевого оборудования, сетевых протоколов, протоколов маршрутизации, фильтрации сетевого трафика, доступа к удалённым ресурсам и пр.

Примерами сетевых операционных систем служат: Novell NetWare, Microsoft Windows (95, NT и более поздние), различные UNIX системы, такие как Solaris, FreeBSD, различные GNU/Linux системы и др. Сетевая ОС предоставляет виртуальную вычислительную систему не полностью скрывающую распределенную природу своего реального прототипа. Пользователю необходимо помнить, что при работе с сетевыми ресурсами необходимо выполнить особые операции. В распределенных ОС сетевые ресурсы предоставляются пользователю в виде единой централизованной виртуальной машины.

Сетевые средства делятся на три компонента: средства предоставления локальных ресурсов и услуг в общее пользование – серверная часть; средства запроса доступа к удаленным ресурсам сети – клиентская часть; транспортные средства, которые совместно с коммуникационной системой обеспечивают передачу сообщений между компьютерами сети. Совокупность серверной и клиентской частей, предоставляющих доступ к конкретному типу ресурса компьютера, называются сетевой службой. В сетях с выделенным сервером

используют специальные варианты ОС, оптимизированные для работы либо с серверами, либо с клиентами.

Вопросы для закрепления:

1. Какие виды протоколов есть?
2. Что обеспечивает сетевая система?
3. Что такое UNIX система?

Домашнее задание:

Проработка конспекта пройденной темы.

Литература:

8. Чекмарев Ю.В. Локальные вычислительные сети : учебное пособие / Чекмарев Ю.В.. — Саратов : Профобразование, 2017. — 200 с

Раздел 8. Проектирование ЛВС
Тема 20. Методика расчета конфигурации сети ETHERNET
Занятие 20 (практическое)

Ключевые вопросы – методика расчета конфигурации сети.

Предыдущие знания обучающихся: Архитектура ЭВМ и ВС

Результаты обучения – организация целенаправленной познавательной деятельности студентов по овладению программным материалом.

Задача – изучение принципов управления работой вычислительных сетей.

Информация о теме:

Гарантию корректной работы сети дает соблюдение многочисленных ограничений, установленных для различных стандартов физического уровня сетей Ethernet.

Наиболее часто приходится проверять ограничения, связанные с длиной сегмента кабеля, а также количеством повторителей и общей длиной сети. Правило «5-4-3» для коаксиальных сетей и «4-х хабов» для сетей на основе витой пары и оптоволокна 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 коаксиального и волоконно-оптического, на которые правила о количестве повторителей не рассчитаны. При этом максимальная длина каждого сегмента физического уровня должна строго соответствовать стандарту.

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

- 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, PVV) 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.

Расчет PDV

Для упрощения расчетов обычно используются справочные данные IEEE, содержащие значения задержек распространения сигналов в повторителях, приемопередатчиках и различных физических средах. В таблице 6 приведены данные, необходимые для расчета значения PDV для всех физических стандартов сетей Ethernet. Битовый интервал обозначен как bt.

Тип сегмента	База левого сегмента, bt	База промежуточного сегмента, bt	База правого сегмента, bt	Задержка среды на 1 м, bt	Максимальная длина сегмента, м
10Base-5	11,8	46,5	169,5	0,0866	500
10Base-2	11,8	46,5	169,5	0,1026	185
10Base-T	15,3	42,0	165,0	0,113	100
10Base-FB	—	24,0	-	0,1	2000
10Base-FL	12,3	33,5	156,5	0,1	2000
AUI (> 2 м)	0	0	0	0,1026	2+48

Таблица 1. Данные для расчета значения PDV

Комитет 802.3 максимально упростил выполнение расчетов. Данные, приведенные в таблице, включают сразу несколько этапов прохождения сигнала, такие как задержки, вносимые повторителем, состоят из задержки входного трансивера, задержки блока повторения и задержки выходного трансивера. В таблице все эти задержки представлены одной величиной, названной базой сегмента. Складывать задержки дважды, вносимые ка-белем, не нужно – в таблице даются удвоенные величины задержек для каждого типа ка-беля.

Использование табличных понятий левый сегмент, правый сегмент и промежуточный сегмент рассмотрено на примере сети, приведенной на рисунке 6.

Левым сегментом называется сегмент, в котором начинается путь сигнала от выхода передатчика конечного узла. На рисунке это сегмент 1. Затем сигнал проходит через промежуточные сегменты 2 – 5 и приходит до наиболее удаленного узла сегмента 6, который называется правым. Именно здесь в худшем случае происходит столкновение кадров и возникает коллизия, что и подразумевается в таблице.

С каждым сегментом связана постоянная задержка, названная базой, которая зависит от типа сегмента и от положения сегмента на пути сигнала (левый, промежуточный или правый). База правого сегмента, в котором возникает коллизия, намного превышает базу левого и промежуточных сегментов.

С каждым сегментом также связана задержка распространения сигнала вдоль кабеля сегмента, зависящая от длины сегмента и вычисляемая путем умножения времени распространения сигнала по одному метру кабеля (в битовых интервалах) на длину кабеля в метрах. Расчет заключается в вычислении задержек, вносимых каждым отрезком кабеля (приведенная в таблице задержка сигнала на 1 м кабеля умножается на длину сегмента), а затем суммировании этих задержек с базами левого, промежуточных и правого сегментов. Общее значение PDV не должно превышать 575.

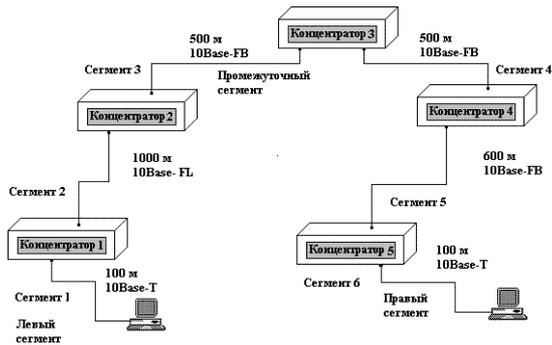


Рисунок 1. Пример сети Ethernet

Так как левый и правый сегменты имеют различные величины базовой задержки, то в случае различных типов сегментов на удаленных краях сети необходимо выполнить расчеты дважды: один раз принять в качестве левого сегмента одного типа, а во второй — сегмент другого типа. Результатом можно считать максимальное значение PDV. В примере крайние сегменты сети принадлежат к одному типу — стандарту 10Base-T, следовательно, двойной расчет не требуется. Если же сегменты были разного типа, то в первом случае нужно было бы принять в качестве левого сегмента между станцией и концентратором 1, а во втором считать левым сегмент между станцией и концентратором 5.

Приведенная на рисунке сеть в соответствии с правилом 4-х хабов не является корректной — в сети между узлами сегментов 1 и 6 имеется 5 хабов, хотя не все сегменты являются сегментами 10Base-FB. Кроме того, общая длина сети равна 2800 м, что нарушает правило 2500 м.

Расчет значения PDV для рассматриваемого примера:

- Левый сегмент 1: $15,3 \text{ (база)} + 100 \times 0,113 = 26,6.$
- Промежуточный сегмент 2: $33,5 + 1000 \times 0,1 = 133,5.$
- Промежуточный сегмент 3: $24 + 500 \times 0,1 = 74,0.$
- Промежуточный сегмент 4: $24 + 500 \times 0,1 = 74,0.$
- Промежуточный сегмент 5: $24 + 600 \times 0,1 = 84,0.$
- Правый сегмент 6: $165 + 100 \times 0,113 = 176,3.$

Сумма всех составляющих дает значение PDV, равное 568,4.

Полученное значение PDV меньше максимально допустимой величины 575, и, следовательно, сеть проходит по критерию времени двойного оборота сигнала, несмотря на то, что ее общая длина составляет больше 2500 м, а количество повторителей — больше 4-х.

Расчет PVV

Чтобы признать конфигурацию сети корректной, нужно рассчитать также уменьшение межкадрового интервала повторителями, то есть величину PVV.

Для расчета PVV также можно воспользоваться значениями максимальных величин уменьшения межкадрового интервала при прохождении повторителей различных физических сред, рекомендованными IEEE и приведенными в таблице 2.

Тип сегмента	Передающий сегмент, bt	Промежуточный сегмент, bt
10Base-5 или 10Base-2	16	11
10Base-FB	—	2
10Base-FL	10,5	8
10Base-T	10,5	8

Таблица 2. Сокращение межкадрового интервала повторителями

В соответствии с этими данными рассчитаем значение PVV для примера.

Левый сегмент 1 10Base-T: сокращение в 10,5 bt.

Промежуточный сегмент 2 10Base-FL: 8.

Промежуточный сегмент 3 10Base-FB: 2.

Промежуточный сегмент 4 10Base-FB: 2.

Промежуточный сегмент 5 10Base-FB: 2.

Сумма этих величин дает значение PVV, равное 24,5, что меньше предельного значения в 49 битовых интервала. В результате приведенная в примере сеть соответствует стандартам Ethernet по всем параметрам, связанным и с длинами сегментов, и с количеством повторителей.

Если расчеты показывают, что сеть неработоспособна, то для преодоления этих ограничений предлагаются следующие методы:

1. Уменьшение длины кабелей с целью снижения задержки прохождения сигнала по сети (если возможно).
2. Уменьшение количества концентраторов для снижения задержек (если возможно).
3. Выбор кабеля с наименьшей задержкой. Кабели различных марок имеют разные задержки, то есть разные скорости распространения сигнала. Различия могут достигать 10%.
4. Разбиение сети на две части или более с помощью коммутатора – более радикальный метод. Коммутатор снижает требования к сети во столько раз, на сколько сегментов (*зон конфликта*) он разбивает сеть. Для каждой новой части сети требуется произвести расчет работоспособности еще раз. Сегмент, который присоединяет коммутатор, также входит в *зону конфликта*, и его надо учитывать при расчетах.

Выбор конфигурации Fast Ethernet

Для определения работоспособности сети Fast Ethernet стандарт IEEE 802.3 предлагает две модели, называемые Transmission System Model 1 и Transmission System Model 2. Первая модель основана на нескольких несложных правилах. Она исходит из того, что все компоненты сети (в частности, кабели) имеют наихудшие из возможных временные характеристики, поэтому всегда дает результат со значительным запасом. Вторая модель использует систему точных расчетов с реальными временными характеристиками кабелей. В связи с этим ее применение позволяет иногда преодолеть жесткие ограничения первой модели.

Правила модели 1

В соответствии с первой моделью, при выборе конфигурации надо руководствоваться следующими принципами:

Сегменты, выполненные на электрических кабелях (витых парах) не должны быть длиннее 100 метров. Это относится к кабелям всех категорий – 3, 4 и 5, к сегментам 100BASE-T4 и 100BASE-TX.

Сегменты, выполненные на оптоволоконных кабелях, не должны быть длиннее 412 метров.

Если используются адаптеры с внешними (выносными) трансиверами, то трансиверные кабели (МП) не должны быть длиннее 50 сантиметров.

Модель 1 выделяет три возможные конфигурации сети Fast Ethernet:

1. Соединение двух абонентов (узлов) сети напрямую, без репитера или концентратора (рисунок 2). Абонентами при этом могут выступать не только компьютеры, но и сетевой принтер, порт коммутатора, моста или маршрутизатора. Такое сопряжение называ-

ется соединением DTE - DTE или двухточечным.

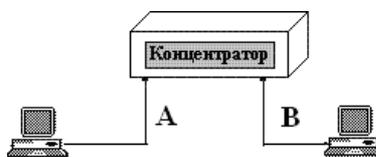


Рисунок 2. Двухточечное соединение компьютеров без концентратора

2. Соединение двух абонентов сети с помощью одного концентратора (репитера) класса I или класса II (рисунок 3).

Рисунок 3. Соединение с одним концентратором

3. Соединение двух абонентов сети с помощью двух концентраторов класса II (рисунок 4). При этом предполагается, что для связи концентраторов всегда используется электрический кабель длиной не более 5 метров.

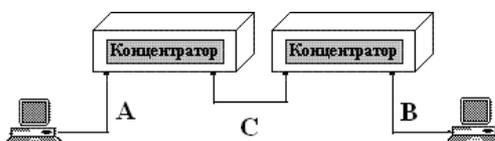


Рисунок 4. Соединение с двумя концентраторами

Концентраторы класса II имеют меньшую задержку, поэтому их может быть два. Использование трех концентраторов в соответствии с моделью 1 не допускается.

В случае выбора первой конфигурации (двухточечной) правила модели 1 предельно просты: электрический кабель не должен быть длиннее 100 метров, полудуплексный оптоволоконный – не более 412 метров, полнодуплексный оптоволоконный – 2000 метров (при этом задержка сигнала в кабеле не имеет значения, так как метод CSMA/CD не работает). В случае применения второй конфигурации (с одним концентратором) надо ограничивать длину кабелей А и В сети в соответствии с таблицей 3.

Вид кабеля А	Вид кабеля В	Класс концентратора	Макс. длина кабеля А, м	Макс. длина кабеля В, м	Макс. размер сети, м
TX, T4	TX, T4	I или II	100	100	200
TX	FX	I	100	160,8	260,8
T4	FX	I	100	131	231
FX	FX	I	136	136	272
TX	FX	II	100	208,8	308,8
T4	FX	II	100	204	304
FX	FX	II	160	160	320

Таблица 3. Максимальная длина кабелей в конфигурации с одним концентратором

В случае выбора третьей конфигурации сети (с двумя концентраторами) надо ограничивать длину кабелей А и В в соответствии с таблицей 4. При этом по умолчанию предполагается, что кабель С имеет длину 5 метров.

Вид кабеля А	Вид кабеля В	Макс. длина кабеля А, м	Макс. длина кабеля В, м	Макс. размер сети, м
TX, T4	TX, T4	100	100	205
TX	FX	100	116,2	221,2
T4	FX	136,3	136,3	241,3
FX	FX	114	114	233

Таблица 4. Максимальная длина кабелей в конфигурации с двумя концентраторами

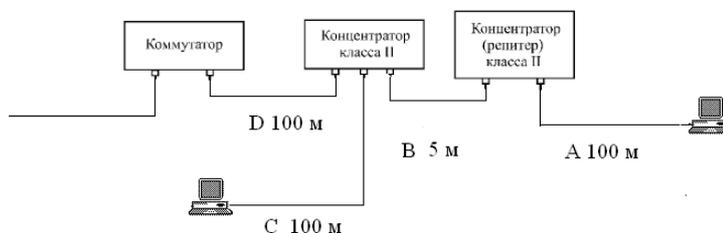
В обеих конфигурациях с концентраторами при использовании одновременно электрического и оптоволоконного кабелей можно за счет уменьшения длины электрического кабеля увеличить длину оптоволоконного. Причем уменьшению длины электрического кабеля на 1 метр соответствует увеличение длины оптоволоконного кабеля на 1,19 метра. Например, уменьшив кабель TX на 10 метров, можно увеличить кабель FX на 11,9 метра, и его предельная длина составит при двух концентраторах 128,1 метра. Немного увеличится и предельный размер сети (в нашем примере на 1,9 метра).

В случае использования двух оптоволоконных кабелей можно уменьшать один из кабелей за счет увеличения другого. При уменьшении одного кабеля на 10 метров можно увеличить другой тоже на 10 метров. Если же используется два электрических кабеля, то увеличивать один из них за счет уменьшения другого нельзя, так как их длина в принципе не может превышать 100 метров из-за затухания сигнала в кабеле.

Концентратор класса II в принципе не может одновременно поддерживать сегменты с разными методами кодирования TX/FX и T4. Поэтому варианты, соответствующие вторым снизу строкам обеих таблиц 3 и 4 никогда не реализуются на практике, но стандарт все же дает цифры и для них.

Во всех перечисленных случаях под размером сети понимается размер *зоны конфликта* (области коллизии). При этом надо учитывать, что включение в сеть одного коммутатора позволяет увеличить полный размер сети вдвое.

Пример сети максимальной конфигурации в соответствии с первой моделью для витой пары показан на рисунке 5. Здесь максимальный размер *зоны конфликта* складывается из сегментов А, В и С, то есть составляет: $100 + 5 + 100 = 205$ метров, что удовлетво-



условию работоспособности сети (таблица 4, верхняя строка).

Рисунок 5. Пример максимальной конфигурации сети Fast Ethernet

Сегмент D также входит в *зону конфликта*, так как коммутатор является полноправным передатчиком пакетов сети. Длина сегмента D не может превышать 100 метров, чтобы суммарная длина сегментов А, В и D не была больше все тех же 205 метров. Сегменты, отделенные от рассматриваемой *зоны конфликта* коммутатором, никак не влияют на ее работоспособность.

Расчет по модели 2

Вторая модель для сети Fast Ethernet, как и в случае Ethernet, основана на вычислении суммарного двойного времени прохождения сигнала по сети. В отличие от второй модели, используемой для оценки конфигурации Ethernet, здесь не проводится расчетов величины сокращения межпакетного интервала. Это связано с тем, что даже максимальное количество репитеров и концентраторов, допустимых в Fast Ethernet (два), в принципе не может вызвать недопустимого сокращения межпакетного интервала.

Для расчетов в соответствии со второй моделью сначала надо выделить в сети путь с максимальным двойным временем прохождения и максимальным числом репитеров

(концентраторов) между компьютерами, то есть путь максимальной длины. Если таких путей несколько, то расчет должен производиться для каждого из них.

Расчет в данном случае ведется на основании таблицы 5.

Тип сегмента	Задержка на метр	Макс. задержка
Два абонента TX/FX	-	100

Два абонента T4	-	138
Один абонент T4 и один TX/FX	-	127
Сегмент на кабеле категории 3	1,14	114 (100 м)
Сегмент на кабеле категории 4	1,14	114 (100 м)
Сегмент на кабеле категории 5	1,112	111,2 (100 м)
Экранированная витая пара	1,112	111,2 (100 м)
Оптоволоконный кабель	1,0	412 (412 м)
Репитер (концентратор) класса I	-	140
Репитер (концентратор) класса II с портами TX/FX	-	92
Репитер (концентратор) класса II с портами T4	-	67

Таблица 5. Двойные задержки компонентов сети Fast Ethernet

Для вычисления полного двойного (кругового) времени прохождения для сегмента сети необходимо умножить длину сегмента на величину задержки на метр, взятую из второго столбца таблицы. Если сегмент имеет максимальную длину, то можно сразу взять величину максимальной задержки для данного сегмента из третьего столбца таблицы.

Затем задержки сегментов, входящих в путь максимальной длины, надо просуммировать и прибавить к этой сумме величину задержки для приемопередающих узлов двух абонентов (три верхние строчки таблицы) и величины задержек для всех репитеров (концентраторов), входящих в данный путь (три нижние строки таблицы).

Суммарная задержка должна быть меньше, чем 512 битовых интервалов. При этом надо помнить, что стандарт IEEE 802.3u рекомендует оставлять запас в пределах 1 – 4 битовых интервалов для учета кабелей внутри соединительных шкафов и погрешностей измерения. Лучше сравнивать суммарную задержку с величиной 508 битовых интервалов, а не 512 битовых интервалов.

Все задержки, приведенные в таблице, даны для наихудшего случая. Если известны временные характеристики конкретных кабелей, концентраторов и адаптеров, то практически всегда предпочтительнее использовать именно их. В ряде случаев это может дать заметную прибавку к допустимому размеру сети.

Пример расчета по второй модели для сети, показанной на рисунке 5. Здесь существуют два максимальных пути: между компьютерами (сегменты А, В и С) и между верхним (по рисунку) компьютером и коммутатором (сегменты А, В и D). Оба эти пути включают в себя два 100-метровых сегмента и один 5-метровый. Предположим, что все сегменты представляют собой 100BASE-TX и выполнены на кабеле категории 5. Для двух 100-метровых сегментов (максимальной длины) из таблицы следует взять величину задержки 111,2 битовых интервалов.

Для 5-метрового сегмента при расчете задержки, умножается 1,112 (задержка на метр) на длину кабеля (5 метров): $1,112 * 5 = 5,56$ битовых интервалов.

Величина задержки для двух абонентов TX из таблицы – 100 битовых интервалов.

Из таблицы величины задержек для двух репитеров класса II – по 92 битовых интервала.

Суммируются все перечисленные задержки: $111,2 + 111,2 + 5,56 + 100 + 92 + 92 = 511,96$ это меньше 512, следовательно, данная сеть будет работоспособна, хотя и на пределе, что не рекомендуется.

Для гарантии лучше несколько уменьшить длину кабелей или взять кабели, имеющие меньшую задержку. Например, при использовании кабеля AT&T 1061 ($NVP = 0,7, t_3$

= 0,477) получаются следующие величины задержек для 100-метровых сегментов: $(0,477 * 2) * 100 = 95,4$ битовых интервалов (умножение на два необходимо, чтобы получить двойное время прохождения), а для 5-метрового сегмента – 4,77 битовых интервалов. Суммарная задержка при этом составит: $95,4 + 95,4 + 4,77 + 100 + 92 + 92 = 483,57$, то есть гораздо меньше 512 и даже 508, что означает полностью работоспособную сеть.

Пользуясь моделью 2, можно обойти некоторые ограничения модели 1, так как модель 1 строится из расчета на наихудший случай. Например, в сети может присутствовать больше двух концентраторов класса II или больше одного концентратора класса I, а кабель, соединяющий концентраторы, может быть длиннее 5 метров.

На рисунке 6 оказана сеть, содержащая три концентратора класса II, соединенных между собой отрезками кабеля длиной по 10 метров. Компьютеры соединены с концентраторами сегментами 100BASE-TX длиной по 50 метров.

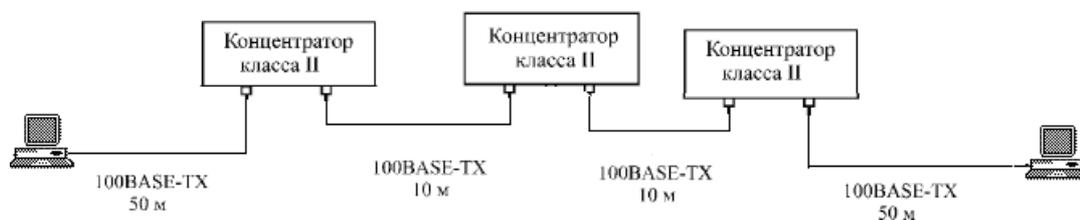


Рисунок 6. Пример работоспособной конфигурации сети, нарушающей правила модели 1

Расчет двойного времени прохождения для этого случая.

1. Каждый из трех концентраторов класса II с портами TX даст задержку 92 битовых интервала. Суммарная задержка концентраторов составит 276 битовым интервалам.
2. Для двух соединительных кабелей между концентраторами задержка равна $2 * 1,112 * 10 = 22,24$ битовых интервала.
3. Для двух сегментов TX по 50 метров задержка составит $2 * 1,112 * 50 = 111,2$ битовых интервала.
4. Для двух абонентов TX задержка будет равна 100 битовым интервалам.
5. Итого суммарная задержка: $276 + 22,24 + 111,2 + 100 = 509,44$ битовых интервала.

Данная сеть работоспособна. Но при этом надо учитывать, что каждый дополнительный концентратор класса II уменьшает общую допустимую длину кабеля на $92/1,112 = 82,7$ метра. Сеть с четырьмя концентраторами не будет иметь смысла, так как на задержку в кабеле уже не остается почти никакого запаса (четыре концентратора дадут суммарную задержку в $92 * 4 = 368$ битовых интервалов).

Какова будет максимальная величина сети Fast Ethernet? Для этого надо взять сеть с одним концентратором класса II и два сегмента 100BASE-FX. Элементарный расчет показывает, что при одинаковых сегментах длина каждого из них может достигать 160 метров (рисунок 7), а общая длина сети составит 320 метров.



Рисунок 7. Сеть Fast Ethernet максимальной длины

Расчет двойного времени прохождения для этого случая будет выглядеть так:

$$92 + 100 + 2 * 1,0 * 160 = 512$$

Получается, что сеть работоспособна, хотя и на пределе. В данном случае важна только суммарная длина обоих кабелей. При уменьшении длины какого-нибудь из сегментов можно без потери работоспособности увеличить на точно такую же величину длину другого сегмента.

Если в приведенной конфигурации используется концентратор класса I, а не концентратор класса II, то допустимая суммарная длина сегментов сокращается с 320 метров до 272 метров (расчет для этого случая очевиден). А согласно стандарту запаса лучше уменьшить суммарную длину кабеля на 1 – 4 метра, что даст снижение круговой задержки на 1 – 4 битовых интервала.

В заключение следует отметить, что модель 2 целесообразно применять в основном при наличии в сети оптоволоконных сегментов. На электрическом кабеле даже при большом желании довольно трудно создать сеть значительного размера.

Задание

На рисунке в вариантах задания представлен план расположения компьютеров в зданиях. Требуется предложить проект ЛВС по следующему плану:

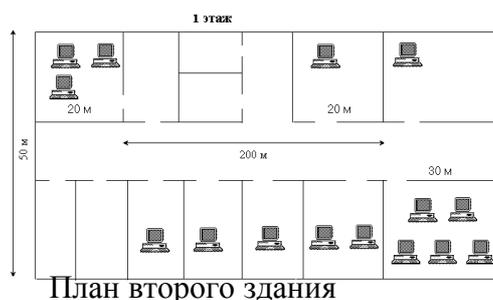
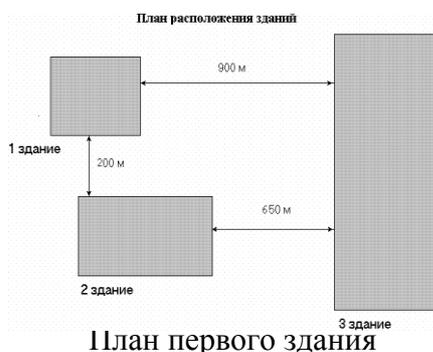
1. Выбрать топологию ЛВС. Обосновать выбор.
2. Выбрать оптимальную конфигурацию сети Ethernet.
3. Нарисовать функциональную схему ЛВС и составить перечень аппаратных средств.
4. Произвести ориентировочную трассировку кабельной системы и выполнить расчет длины кабельного соединения для выбранной топологии с учетом переходов между этажами.
5. С целью выполнения ограничений на длину кабеля и количества станций в сегменте выбранной конфигурации установить необходимые коммуникационные устройства.
6. Оценить проектируемую сеть по критерию времени двойного оборота (рассчитать PDV) и характеристики уменьшения межкадрового интервала повторителями (PVV).

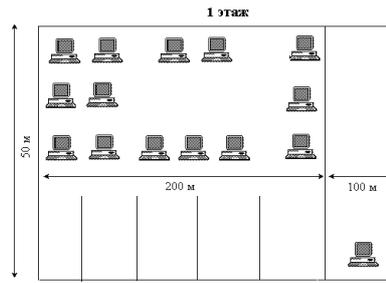
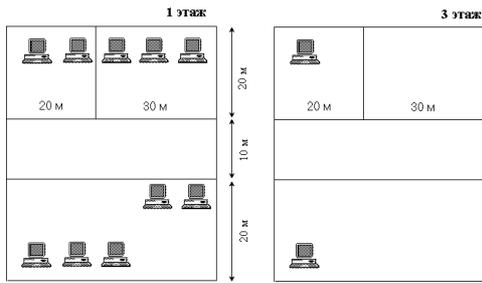
Вариант 1

Первое из зданий трехэтажное. Компьютеры расположены в отделах на первом и третьем этажах. Общее число компьютеров – 12 (10 на первом этаже и 2 на втором).

Второе и третье здания одноэтажные. В каждом из них – 15 компьютеров.

План третьего здания





Раздел 9. Классификация, требования, основные компоненты сетей ЭВМ Тема 20. Эволюция компьютерных и телекоммуникационных технологий.

Занятие 21 (практическое)

Цель работы:

Изучить принципы выбора конфигурации физических и логических связей, разработки схем адресации устройств.

Содержание:

Системы пакетной обработки. Многотерминальные системы – прообраз сети. Первые глобальные сети.

Теоретические основы.

Связь компьютера с периферийными устройствами. Частным случаем связи "точка-точка" является соединение компьютера с периферийным устройством. Для обмена данными компьютер и периферийное устройство (ПУ) оснащены внешними интерфейсами или портами (рис. 1). В данном случае к понятию "интерфейс" относятся:

- электрический разъем;
- набор проводов, соединяющих устройства;
- совокупность правил обмена информацией по этим проводам.

Со стороны компьютера логикой передачи сигналов на внешний интерфейс управляют:

- **контроллер ПУ** — аппаратный блок, часто реализуемый в виде отдельной платы;
- **драйвер ПУ** – программа, управляющая контроллером периферийного устройства.

Со стороны ПУ интерфейс чаще всего реализуется аппаратным устройством управления ПУ, хотя встречаются и программно-управляемые периферийные устройства.

Обмен данными между ПУ и компьютером, как правило, является двунаправленным. Так, например, даже принтер, который представляет собой устройство вывода информации, возвращает в компьютер данные о своем состоянии.

Рассмотрим последовательность действий, которые выполняются в том случае, когда некоторому приложению требуется напечатать текст на принтере. Со стороны компьютера в выполнении этой операции принимает участие, кроме уже названных **контроллера, драйвера и приложения**, еще один важнейший компонент — **операционная система**. Поскольку все операции ввода-вывода являются привилегированными, все приложения при выполнении операций с периферийными устройствами используют ОС как арбитра. Итак, последовательность действий такова:

1. Приложение обращается с запросом на выполнение операции печати к операционной системе. В запросе указываются: адрес данных в оперативной памяти, идентифицирующая информация принтера и операция, которую требуется выполнить.
2. Получив запрос, операционная система анализирует его, решает, может ли он быть выполнен, и если решение положительное, то запускает соответствующий драйвер, передавая ему в качестве параметров адрес выводимых данных. Дальнейшие действия, относящиеся к операции ввода-вывода, со стороны компьютера реализуются совместно драйвером и контроллером принтера.
3. Драйвер передает команды и данные контроллеру, который помещает их в свой внутренний буфер. Пусть, например, драйвер загружает значение некоторого байта в буфер контроллера ПУ.

4. Контроллер перемещает данные из внутреннего буфера во внешний порт.
5. Контроллер начинает последовательно передавать биты в линию связи, представляя каждый бит соответствующим электрическим сигналом. Чтобы сообщить устройству управления принтера о том, что начинается передача байта, перед передачей первого бита данных контроллер формирует стартовый сигнал специфической формы, а после передачи последнего информационного бита — стоповый сигнал. Эти сигналы синхронизируют передачу байта. Кроме информационных бит, контроллер может передавать бит контроля четности для повышения достоверности обмена.
6. Устройство управления принтера, обнаружив на соответствующей линии стартовый бит, выполняет подготовительные действия и начинает принимать информационные биты, формируя из них байт в своем приемном буфере. Если передача сопровождается битом четности, то выполняется проверка корректности передачи: при правильно выполненной передаче в соответствующем регистре устройства управления принтера устанавливается признак завершения приема информации. Наконец, принятый байт обрабатывается принтером — выполняется соответствующая команда или печатается символ.

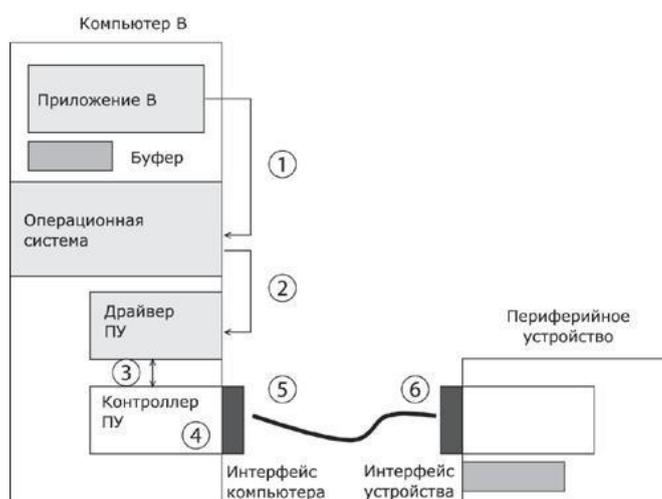


Рис. 3.1. Связь компьютера с периферийным устройством.

Обязанности между драйвером и контроллером могут распределяться по-разному, но чаще всего контроллер поддерживает набор простых команд, служащих для управления периферийным устройством, а на драйвер обычно возлагаются наиболее сложные функции реализации обмена. Драйвер, задавая ту или иную последовательность команд, определяет тем самым логику работы периферийного устройства. Для одного и того же контроллера можно разработать различные драйверы, которые с помощью одного и того же набора доступных команд будут реализовывать разные алгоритмы управления одним и тем же ПУ. Возможно распределение функций между драйвером и контроллером (УУ).

Функции, выполняемые драйвером:

- ведение очередей запросов;
- буферизация данных;
- подсчет контрольной суммы последовательности байтов;
- анализ состояния ПУ;
- загрузка очередного байта данных (или команды) в регистр контроллера;
- считывание байта данных или байта состояния ПУ из регистра контроллера.

Функции, выполняемые контроллером:

- преобразование байта из регистра (порта) в последовательность бит;
- передача каждого бита в линию связи;
- обрамление байта стартовым и стоповым битами – синхронизация;
- формирование бита четности;
- установка признака завершения приема/передачи байта.

Связь двух компьютеров.

А теперь предположим, что пользователь другого компьютера хотел бы распечатать текст. Сложность состоит в том, что к его компьютеру не подсоединен принтер, и требуется воспользоваться тем принтером, который связан с другим компьютером (рис. 2).



Рис. 2. Взаимодействие двух компьютеров.

Программа, работающая на одном компьютере, не может получить непосредственный доступ к ресурсам другого компьютера — его дискам, файлам, принтеру. Она может только "попросить" об этом другую программу, выполняемую на том компьютере, которому принадлежат эти ресурсы. Эти "просьбы" выражаются в виде сообщений, передаваемых по каналам связи между компьютерами. Такая организация печати называется удаленной.

Механизм обмена байтами между двумя компьютерами определен. Теперь нужно договориться о правилах обмена сообщениями между приложениями А и В. Приложение В должно "уметь" расшифровать получаемую от приложения А информацию. Для этого программисты, разработавшие приложения А и В, строго оговаривают форматы сообщений, которыми будут обмениваться приложения, и их семантику. Вернемся к последовательности действий, которые необходимо выполнить для распечатки текста на принтере "чужого" компьютера.

- Приложение А формирует очередное сообщение (содержащее, например, строку, которую необходимо вывести на принтер) приложению В, помещает его в буфер оперативной памяти и обращается к ОС с запросом на передачу содержимого буфера на компьютер В.
- ОС компьютера А обращается к **драйверу СОМ-порта**, который инициирует работу контроллера.
- Действующие с обеих сторон пары драйверов и контроллеров СОМ-порта последовательно, байт за байтом, передают сообщение на компьютер В.
- Драйвер компьютера В периодически выполняет проверку на наличие признака завершения приема, устанавливаемого контроллером при правильно выполненной передаче данных, и при его появлении считывает принятый байт из буфера контроллера в оперативную память, тем самым делая его доступным для программ компьютера В. В некоторых случаях драйвер вызывается асинхронно, по прерываниям от контроллера. Аналогично реализуется и передача байта в другую сторону — от компьютера В к компьютеру А.
- Приложение В принимает сообщение, интерпретирует его, и в зависимости от того, что в нем содержится, формирует запрос к своей ОС на выполнение тех или иных действий с принтером. В нашем примере сообщение содержит указание на печать текста, поэтому ОС передает драйверу принтера запрос на печать строки.
- Далее выполняются все действия 1-6, описывающие выполнение запроса приложения к ПУ в соответствии с рассмотренной ранее схемой "локальная ОС — драйвер ПУ — контроллер ПУ — устройство управления ПУ" (см. предыдущий раздел). В результате строка будет напечатана.

Контрольные вопросы:

1. Связь компьютера с периферийными устройствами.

2. Связь двух компьютеров.

Работа с литературой:

Рекомендуемые источники информации (№ источника)			
Основная	Дополнительная	Методическая	Интернет-ресурсы
1-2	1	1-3	1-2

Раздел 10. Принципы взаимодействия компьютеров в сети

Тема 21. От первых локальных сетей до современных сетевых технологий.

Занятие 21 (практическое)

Цель работы:

Изучить принципы выбора конфигурации физических и логических связей, разработки схем адресации устройств.

Содержание:

Роль персональных компьютеров в эволюции компьютерных сетей. Эволюция сетевых операционных систем.

Теоретические основы.

Клиент, редиректор и сервер.

Можно представить, что любая программа, которой потребуется печать на "чужом" принтере, должна включать в себя функции, подобные тем, которые выполняет приложение А. Но нагружать этими стандартными действиями каждое приложение — текстовые и графические редакторы, системы управления базами данных и другие приложения — не очень рационально (хотя существует большое количество программ, которые действительно самостоятельно решают все задачи по обмену данными между компьютерами, например Kermit — программа обмена файлами через СОМ-порты, реализованная для различных ОС, Norton Commander 3.0 с его функцией Link). Гораздо выгоднее создать специальный программный модуль, который (вместо приложения А) будет выполнять формирование сообщений-запросов к удаленной машине и прием результатов для всех приложений. Такой служебный модуль называется **клиентом**.

На стороне же компьютера В (на месте приложения В) должна работать другая специализированная программа — **сервер**, постоянно ожидающий прихода запросов на удаленный доступ к принтеру (или файлам, расположенным на диске) этого компьютера. Сервер, приняв запрос из сети, обращается к локальному ПУ, возможно, с участием локальной ОС. Очень удобной и полезной функцией клиентской программы является способность отличить запрос к удаленному файлу от запроса к локальному файлу. Если клиентская программа умеет это делать, она сама распознает и перенаправляет (redirect) запрос к удаленной машине. Отсюда и название, часто используемое для клиентской части — **редиректор**. Иногда функции распознавания выделяются в особый программный модуль, в этом случае редиректором называют не всю клиентскую часть, а только этот модуль. Схема взаимодействия клиента и сервера с приложениями и локальной операционной системой приведена на рис. 3.

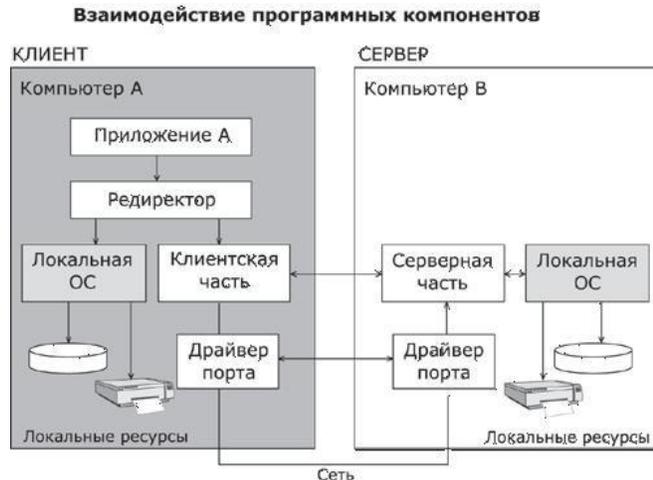


Рис. 3. Взаимодействие программных компонентов при связи двухкомпьютеров.

Для того, чтобы компьютер мог работать в сети, его операционная система должна быть дополнена клиентским и /или серверным модулем, а также средствами передачи данных между компьютерами. В результате такого добавления операционная система компьютера становится **сетевой ОС**.

Задача 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, связанные с **физической передачей** сигналов по линиям связи.

При соединении "точка-точка" на первый план выходит задача физической передачи данных по линиям связи. Эта задача среди прочего включает:

- **кодирование** и **модуляцию** данных;
- взаимную **синхронизацию** передатчика одного компьютера с приемником другого;
- подсчет **контрольной суммы** и передача ее по линиям связи после каждого байта или после некоторого блока байтов.

В вычислительной технике для представления данных используется двоичный код. Представление данных в виде электрических или оптических сигналов называется кодированием. Существуют различные способы кодирования двоичных цифр 1 и 0, например, потенциальный 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 использоваться для кодирования данных и при их передаче 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 существенно большим искажениям прямоугольных импульсов

(например, "заваливанию" фронтов), чем внутри компьютера. Поэтому 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. 3.4). При модуляции дискретная информация представляется синусоидальным сигналом той частоты, которую хорошо передает имеющаяся линия связи.

При передаче сигналов приходится еще решать проблему взаимной синхронизации передатчика одного компьютера с приемником другого. При организации взаимодействия модулей внутри компьютера она решается очень просто, так как в этом случае все модули синхронизируются от общего тактового генератора. Проблема синхронизации при связи компьютеров может решаться разными способами, как с помощью обмена специальными тактовыми синхроимпульсами по отдельной линии, так и посредством периодической синхронизации заранее обусловленными кодами или импульсами характерной формы, отличной от формы импульсов данных.

Несмотря на принятые меры (выбор соответствующей скорости обмена данными, линий связи с определенными характеристиками, способа синхронизации приемника и передатчика), существует вероятность искажения некоторых бит передаваемых данных. Для более надежной передачи данных часто используется стандартный прием — подсчет контрольной суммы и передача ее по линиям связи после каждого байта или после некоторого блока байтов. Часто в протокол обмена данными включается как обязательный элемент сигнал-квитанция, которая подтверждает правильность приема данных и посылается от получателя отправителю.

Контрольные вопросы:

1. Клиент, редиректор и сервер.
2. Задача физической передачи данных по линиям связи.

Работа с литературой:

Рекомендуемые источники информации (№ источника)			
Основная	Дополнительная	Методическая	Интернет-ресурсы
1-2	1	1-3	1-2

МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО САМОСТОЯТЕЛЬНОЙ РАБОТЕ СТУДЕНТОВ

Самостоятельная работа – планируемая учебная, учебно-исследовательская, научно-исследовательская работа студентов, выполняемая во внеаудиторное время по заданию и при методическом руководстве преподавателя, но без его непосредственного участия (при частичном непосредственном участии преподавателя, оставляющем ведущую роль за работой студентов).

Самостоятельная работа является важным видом учебной и научной деятельности студента и играет значительную роль в рейтинговой технологии обучения. Государственным стандартом предусматривается, как правило, 50% часов из общей трудоемкости дисциплины на самостоятельную работу студентов. Количество часов самостоятельной работы и ее распределение между дидактическими единицами приведено в рабочей программе.

Задачами самостоятельной работы являются:

систематизация и закрепление полученных теоретических знаний и практических умений студентов;

углубление и расширение теоретических знаний;

формирование умений использовать нормативную, правовую, справочную документацию и специальную литературу;

развитие познавательных способностей и активности студентов: творческой инициативы, самостоятельности, ответственности и организованности;

формирование самостоятельности мышления, способностей к саморазвитию, самосовершенствованию и самореализации;

развитие исследовательских умений;

использование собранного и полученного в ходе самостоятельных занятий для эффективной подготовки к итоговым зачетам и экзаменам.

В процессе самостоятельной работы студент приобретает навыки самоорганизации, самоконтроля, самоуправления, становится активным самостоятельным субъектом учебной деятельности.

Выполняя самостоятельную работу под контролем преподавателя, студент должен:

– освоить минимум содержания, выносимый на самостоятельную работу студентов и предложенный преподавателем в соответствии с Государственными образовательными стандартами высшего профессионального образования по данной дисциплине.

– планировать самостоятельную работу в соответствии с графиком самостоятельной работы, предложенным преподавателем.

– самостоятельную работу студент должен осуществлять в организационных формах, предусмотренных учебным планом и рабочей программой преподавателя.

– выполнять самостоятельную работу и отчитываться по ее результатам в соответствии с графиком представления результатов, видами и сроками отчетности по самостоятельной работе студентов.

Основной формой самостоятельной работы студента является изучение конспекта лекций, их дополнение, рекомендованной литературы, подготовка отчетов к лабораторным работам. Но для успешной учебной деятельности, ее интенсификации, необходимо учитывать следующие субъективные факторы:

Начинать самостоятельные внеаудиторные занятия следует с первых же дней семестра, пропущенные дни будут потеряны безвозвратно, компенсировать их позднее усиленными занятиями без снижения качества работы и ее производительности невозможно. Первые дни семестра очень важны для того, чтобы включиться в работу, установить определенный порядок, равномерный ритм на весь семестр.

КОНТРОЛЬ ЗНАНИЙ

Текущий контроль знаний предполагает защиту лабораторных работ и выполнение тестовых заданий. В тестовые задания входят вопросы по лекционному материалу и вопросы самостоятельной работы, приведенные в рабочей программе.

Вариант теста текущего контроля

Вариант 1

1. Какие функции выполняются на транспортном уровне эталонной модели взаимодействия открытых систем (OSI)?

а) определение начала и окончания сеанса связи; определение времени, длительности и режима сеанса связи; определение точек синхронизации для промежуточного контроля и восстановления при передаче данных;

б) контроль последовательности передачи данных; установка соответствия между транспортными (логическими) и сетевыми адресами абонентов; обнаружение и обработка ошибок передачи данных;

в) определение маршрутизации в сети и связь между сетями; обеспечение независимости высших уровней от используемой для передачи информации физической среды; обнаружение и обработка ошибок передачи данных;

г) определение метода доступа к среде передачи данных; определение логической топологии сети передачи данных; определение физической адресации.

2. *Какое устройство называется повторителем?* а) организовать обмен данными между сетевыми объектами, использующими различные протоколы обмена данными;

б) расширить сеть подключением дополнительных сегментов кабеля;

в) объединить несколько сегментов, так что передача данных между станциями внутри одного сегмента не будет влиять на передачу данных в других сегментах;

г) соединять сети разного типа, использующие одну сетевую операционную систему или протокол обмена данными.

3. Какую из ситуаций называют коллизией?

а) когда станция, желающая передать пакет, обнаруживает, что в данный момент другая станция уже заняла передающую среду;

б) когда в данных, переданных станцией, обнаружена ошибка;

в) когда две рабочие станции одновременно передают данные в разделяемую передающую среду;

г) когда передача данных завершена станцией успешно.

4. Какое название носит сети метод доступа станций к передающей среде в сети Ethernet?

а) маркерное кольцо

б) множественный доступ с контролем несущей и обнаружением коллизий

в) маркерная шина

г) раннее освобождение маркера

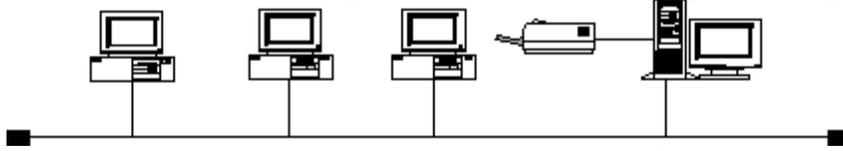
5. При каком способе обмена информацией между узлами сети данные передаются в одном направлении?

- а) при симплексной передаче; б) при дуплексной передаче;
 в) при полудуплексной передаче; г) при полусимплексной передаче

6. Как называется характеристика сети, предполагающая скрытие особенностей работы сети от конечного пользователя?

- а) интегрируемость; б) прозрачность;
 в) надежность; с) масштабируемость.

7. По какой топологии изображено подключение устройств на рисунке?



- а) звезда; б) шина; в) ячеистая топология; г) кольцо.

8. Какой из перечисленных вариантов характеристик соответствует технологии 10BASE-F?

	а	б	в	г
Длина сегмента кабеля, м	500	185	100	2000
Тип кабеля	50 Омный коаксиальный, «толстый»	50 Омный коаксиальный, «тонкий»	Витая пара (UTP)	Оптоволокно
Топология подключения устройства	Шина	Шина	Звезда	Звезда

9. Какое из утверждений о логическом адресе узла сети верное?

- а) не определяется используемым протоколом обмена данными;
 б) определяется стандартом локальной сети;
 в) не может быть изменен после подключения устройств к сети;
 г) может быть изменен в процессе работы.

10. Поставьте соответствия между типом коммуникационного устройства и уровнями модели OSI, на котором оно может работать

- 1) повторитель а) физический, канальный, сетевой, транспортный
 2) мост б) физический, канальный
 3) шлюз в) физический
 4) маршрутизатор г) над уровнями модели OSI

11. Какое устройство изолирует трафик одной подсети от трафика другой подсети, но не может быть использован в сети с замкнутым контуром?

- а) мост б) шлюз в) маршрутизатор г) повторитель

12. Что используется в одномодовом кабеле в качестве источника излучения света? _____

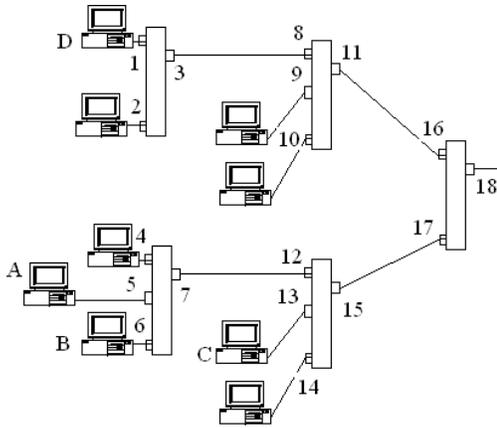
13. Какие из перечисленных пар сетевых технологий совместимы по форматам кадров и позволяют обрабатывать составную сеть без необходимости транслирования кадров?

- а) Token Ring – Fast Ethernet
 б) FDDI – Ethernet
 в) Ethernet – Fast Ethernet
 г) Token Ring - FDDI

14. Поставьте соответствия между названиями стандартов и технологий,

устройства на схеме сети являются концентраторами, то на каких портах появится кадр, если его отправить с компьютера A компьютеру D?

- а) на всех портах б) 5,6, 7 в) 4,5,6,7
 г) 4,5,6,7,12, 13,14,15, 17,18



25. Если в предыдущем примере считать, что все устройства являются коммутаторами, то на каких портах появится кадр, если его отправить с компьютера A компьютеру C?