

## ОГЛАВЛЕНИЕ

1. Особенности процессорных архитектур. CISC и RISC архитектура. Их краткая характеристика.	4
2. Виды обеспечения вычислительных систем. Определения. Примеры	5
3. Векторные и векторно-конвейерные вычислительные системы. Матричные вычислительные системы.	6
4. Метрики производительности конвейера	6
5. Классы конфликтов, возникающих в конвейерах и способы их устранения	7
6. Сигналы. Объем информации. Количество информации и энтропия. Свойства информации.	8
7. Устройство управления современного процессора. Определение. Микрокоманда. Микрооперация. Микропрограмма. Задачи, решаемые устройством управления	9
8. Организация ввода/вывода в вычислительной системе. Системные и локальные шины. Устройства ввода/вывода.	10
9. Классификация вычислительных систем. Альтернативная классификация.	11
10. Назначение, принципы построения и характеристики арифметико-логических устройств (АЛУ).	12
11. Многоуровневая организация ЭВМ. Структурная организация и архитектура вычислительных систем.	13
12. Основные архитектурные понятия. Типы команд. Типы данных Способы адресации.	14
13. Виртуальная память и организация защиты памяти.	15
14. Принципы фон-Неймана по построению вычислительных систем.	16
15. Конвейерная организация. Простейшая организация конвейера и оценка его производительности. Примеры.	17
16. Общие понятия и определения, структурная схема микропроцессора.	18
17. Классы конфликтов возникающих в конвейерах и способы их устранения.	19
18. Дисковые массивы и уровни RAID.	20
19. Иерархия памяти. Организация кэш-памяти. Принципы организации основной памяти в современных компьютерах.	21

20. Организация регистров современного процессора.	22
21. Фон-неймановская архитектура.	23
22. BIOS и UEFI. Определение. Состав. Предназначение.	24
23. Поколения вычислительных машин.	24
24. Классификация вычислительных систем по Флинну.	25
25. Состав системного блока современной рабочей станции. Единицы измерения рабочих частот процессоров и системных шин. Единицы измерения всех видов памяти.	26
26. Организация ввода/вывода в вычислительной системе. Системные и локальные шины. Устройства ввода/вывода.	27
27. Блоки управления командами. Структура устройства управления. Принципы организации систем прерываний. Процедура обслуживания прерываний.	27
28. Классификация ЭВМ по областям применения.	29
29. Цикл обработки команды современного процессора.	30
30. Многоуровневая организация ЭВМ. Многоступенчатая обработка.	31
31. Оценка производительности вычислительных систем.	32
32. Понятие архитектуры вычислительной системы.	33
33. Организация автоматической работы ЭВМ. Управляющие функции процессора. Общая организация выполнения программы на ЭВМ.	34
34. Память и запоминающие устройства. Виды и характеристики ЗУ: адресная, стековая и ассоциативная организация памяти.	35
35. Тестирование вычислительных систем.	36
36. Общие требования, предъявляемые к современным ЭВМ.	37
37. Эталонная модель взаимодействия открытых систем.	38
38. Многоуровневая организация управления в ИВС. Протоколы и интерфейсы. Способы и средства коммутации и передачи данных.	40
39. Функции сетевого и транспортного уровней, маршрутизация пакетов. Управление потоками передаваемых данных.	42
40. Количество информации и энтропия. Кодирование информации, способы контроля правильности передачи данных.	44

41. Структура и принципы построения ЛВС. Архитектура одноранговых сетей и сетей "клиент-сервер". **45**
42. Методы доступа: CSMA/CD, маркерные методы доступа. Сети Ethernet, Token Ring и FDDI. Высокоскоростные локальные сети. **47**
43. Структурная организация систем телекоммуникаций. Каналы передачи данных: аналоговые, цифровые; разделение каналов по времени и частоте. **49**
44. Характеристики проводных линий связи; спутниковые каналы; сотовые системы связи. Способы модуляции. **51**
45. Алгоритмы сжатия данных. Программное обеспечение телекоммуникаций. Протоколы TCP/IP, управления. Адресация в Интернете. **53**
46. Информационные услуги территориальных сетей. Технологии распределенных вычислений. Протоколы файлового обмена, электронной почты. **57**

## **1. Особенности процессорных архитектур. CISC и RISC архитектура. Их краткая характеристика.**

### **CISC**

CISC (англ. Complex Instruction Set Computer — «компьютер с полным набором команд») — тип процессорной архитектуры, в первую очередь, с нефиксированной длиной команд, а также с кодированием арифметических действий в одной команде и небольшим числом регистров, многие из которых выполняют строго определенную функцию.

Самый яркий пример CISC архитектуры — это x86 (он же IA-32) и x86\_64 (он же AMD64).

В CISC процессорах одна команда может быть заменена ей аналогичной, либо группой команд, выполняющих ту же функцию. Отсюда вытекают плюсы и минусы архитектуры: высокая производительность благодаря тому, что несколько команд могут быть заменены одной аналогичной, но большая цена по сравнению с RISC процессорами из-за более сложной архитектуры, в которой многие команды сложнее декодировать.

### **RISC**

RISC (англ. Reduced Instruction Set Computer — «компьютер с сокращённым набором команд») — архитектура процессора, в котором быстродействие увеличивается за счёт упрощения инструкций: их декодирование становится более простым, а время выполнения — меньшим. Первые RISC-процессоры не имели даже инструкций умножения и деления и не поддерживали работу с числами с плавающей запятой.

По сравнению с CISC эта архитектура имеет константную длину команды, а также меньшее количество схожих инструкций, позволяя уменьшить итоговую цену процессора и энергопотребление, что критично для мобильного сегмента. У RISC также большее количество регистров.

Примеры RISC-архитектур: PowerPC, серия архитектур ARM (ARM7, ARM9, ARM11, Cortex).

В общем случае RISC быстрее CISC. Даже если системе RISC приходится выполнять 4 или 5 команд вместо одной, которую выполняет CISC, RISC все равно выигрывает в скорости, так как RISC-команды выполняются в 10 раз быстрее.

## **2. Виды обеспечения вычислительных систем. Определения.**

### **Примеры**

Выделяются следующие основные виды обеспечения вычислительной системы: программное, техническое(аппаратное), математическое, информационное.

Аппаратное обеспечение включает компьютеры, внешние устройства и диагностическую аппаратуру, энергетическое оборудование, батареи и аккумуляторы.

Программное обеспечение – программа или множество программ, используемых для управления компьютером. Вычислительные машины в настоящее время не могут работать без соответствующего программного обеспечения. То есть для того чтобы компьютер выполнял те или иные задачи, на него должны быть установлены программы, содержащих алгоритмы выполнения этих задач.

### **3. Векторные и векторно-конвейерные вычислительные системы. Матричные вычислительные системы.**

Векторно-конвейерные вычислительные системы относятся к классу SIMD-систем. Основные принципы: конвейерная организация обработки потока команд; введение в систему команд набора векторных операций, которые позволяют оперировать с целыми массивами данных.

Матричные ВС обладают более широкими архитектурными возможностями чем конвейерные ВС. Они относятся к классу системы с массовым параллелизмом и следовательно не имеют принципиальных ограничений в наращивании своей производительности, а также предназначены для решения сложных задач, связанных с выполнением операций над векторами, матрицами и массивами данных

### **4. Метрики производительности конвейера**

Для того чтобы оценить эффект, достигаемый за счет конвейеризации вычислений, обычно используют три метрики; ускорение, эффективность и производительность.

Под ускорением понимается отношение времени обработки без конвейера и при его наличии.

Еще одной метрикой, характеризующей конвейерный процессор, является эффективность — доля ускорения, приходящаяся на одну ступень конвейера.

В качестве третьей метрики часто выступает пропускная способность или производительность.

## **5. Классы конфликтов, возникающих в конвейерах и способы их устранения**

Конфликты — это такие ситуации в конвейерной обработке, которые препятствуют выполнению очередной команды в предназначенном для нее такте. Конфликты снижают реальную производительность конвейера, которая могла быть достигнута в идеальном случае.

Существуют три класса конфликтов:

1. Структурные конфликты, которые возникают из-за конфликтов по вычислительным ресурсам, когда аппаратные средства процессора не могут поддерживать все возможные комбинации команд в режиме одновременного выполнения с совмещением. Чтобы устранить конфликт нужно увеличение времени такта до такой величины, которая позволила бы все этапы любой команды выполнять за один такт.
2. Конфликты по данным, возникающие в случаях, когда выполнение одной команды зависит от результата выполнения предыдущей команды. Устранение конфликта заключается в замораживании операций в конвейере путем блокировки выполнения любой команды, следующей за командой условного перехода, до тех пор, пока не станет известным направление перехода.
3. Конфликты по управлению, которые возникают при конвейеризации команд переходов и других команд, изменяющих значение счетчика команд. Уменьшение влияния конфликта обеспечивается методом, который называется пересылкой или продвижением данных.

Конфликты в конвейере приводят к необходимости приостановки выполнения команд. В простейших конвейерах, если приостанавливается какая-либо команда, то все следующие за ней команды также приостанавливаются. Команды, предшествующие приостановленной, могут продолжать выполняться, но во время приостановки не выбирается ни одна новая команда.

## 6. Сигналы. Объем информации. Количество информации и энтропия. Свойства информации.

Информация это - сообщения (содержание сигналов), которые человек воспринимает с помощью специальных устройств.

**Свойства:** достоверность, полнота, точность, актуальность, полезность, ценность, своевременность, понятность, доступность, краткость.

Сигнал – физический процесс, отображающий (несущий) передаваемое сообщение. Формируется путем изменения какого-либо параметра физического процесса по закону сообщения.

**Энтропия (информационная)** — мера хаотичности информации, неопределённость появления какого-либо символа первичного алфавита. При отсутствии информационных потерь численно равна количеству информации на символ передаваемого сообщения.

Так, возьмём, например, последовательность символов, составляющих какое-либо предложение на русском языке. Каждый символ появляется с разной частотой, следовательно, неопределенность появления для некоторых символов больше, чем для других. Если же учесть, что некоторые сочетания символов встречаются очень редко, то неопределённость еще более уменьшается.

Пусть сообщение состоит из одного символа. Если вероятности появления всех символов одинаковы и равны  $P = 1/m$ , то количество информации, которое переносит символ, можно выразить как

$$I_1 = \log m = -\log P$$

Здесь количество информации связано с *вероятностью* появления символа. В реальных сообщениях символы  $a_i \in A$  появляются с различными вероятностями  $P(a_i)$ , поэтому

$$I_2 = -\log P(a_i)$$

*Среднее* количество информации  $H(A)$ , которое приходится на *один символ* источника сообщений можно найти усреднением по всему объему алфавита



$$H(A) = - \sum_{i=1}^n P(a_i) \log P(a_i) \quad (1)$$

Эта величина называется *энтропией* источника дискретных сообщений. Формула (1) носит название формулы Шеннона.

## **7. Устройство управления современного процессора. Определение. Микрокоманда. Микрооперация. Микропрограмма. Задачи, решаемые устройством управления**

Устройство управления процессора реализует функции управления ходом вычислительного процесса, обеспечивая автоматическое выполнение команд программы. Работа программы представляет собой процесс выполнения последовательности командных циклов, каждый из которых в общем случае состоит из этапов или ступеней: выборки команды из памяти, ее декодирования, чтения операнда, исполнения операции, записи результата. Другими словами, процесс функционирования вычислительного устройства состоит из последовательности элементарных действий, выполняемых в его узлах. Такие элементарные преобразования информации, выполняемые в течение одного такта сигналов синхронизации, называются микрооперациями. Совокупность сигналов управления, вызывающих одновременно выполняемые микрооперации, образует микрокоманду.

В свою очередь, последовательность микрокоманд, определяющую содержание и порядок реализации машинного цикла, принято называть микропрограммой.

## **8. Организация ввода/вывода в вычислительной системе. Системные и локальные шины. Устройства ввода/вывода.**

Особенность данной организации ввода-вывода заключается в использовании общей информационной магистрали для передачи данных между процессором, памятью и контроллерами (устройствами) ввода-вывода (УВВ), подключенными к шине.

С целью снижения стоимости некоторые ЭВМ имеют единственную шину для памяти и устройств ввода-вывода. Такая шина называется **системной**. Примерами системных шин являются шины стандартов **ISA**, **EISA** или **MCA** персональных компьютеров недалекого прошлого.

Необходимость сохранения баланса производительности по мере роста быстродействия процессоров привела к двухуровневой организации шин в персональных компьютерах на основе локальной шины.

**Локальной шиной** называется шина, электрически выходящая непосредственно на контакты процессора. Она обычно объединяет процессор, память, схемы буферизации для системной шины и ее контроллер, а также некоторые вспомогательные схемы. Типичными примерами локальных шин являются **VL-Bus** и **PCI**.

К устройством ввода-вывода относится: монитор, клавиатура, мышь, принтер и т.д.

## 9. Классификация вычислительных систем. Альтернативная классификация.

Одна из первых классификаций, ссылки на которую наиболее часто встречаются в литературе, была предложена М. Флинном в конце 60-х годов прошлого века. Она базируется на понятиях двух потоков: команд и данных. На основе числа этих потоков выделяется четыре класса архитектур:

1. SISD (Single Instruction Single Data) - единственный поток команд и единственный поток данных. По сути дела, это классическая машина Фон Неймана. К этому классу относятся все однопроцессорные системы.
2. SIMD (Single Instruction Multiple Data) - единственный поток команд и множественный поток данных. Типичными представителями являются матричные компьютеры, в которых все процессорные элементы выполняют одну и ту же программу, применяемую к своим (различным для каждого ПЭ) локальным данным. Некоторые авторы к этому классу относят и векторно-конвейерные компьютеры, если каждый элемент вектора рассматривать как отдельный элемент потока данных.
3. MISD (Multiple Instruction Single Date) - множественный поток команд и единственный поток данных. М. Флинн не смог привести ни одного примера реально существующей системы, работающей на этом принципе. Некоторые авторы в качестве представителей такой архитектуры называют векторно-конвейерные компьютеры, однако такая точка зрения не получила широкой поддержки.
4. MIMD (Multiple Instruction Multiple Date) - множественный поток команд и множественный поток данных. К этому классу относятся практически все современные многопроцессорные системы.

Также существует классификация по Хокни, Фенга, Хэндлера, Шнайдера, Скилликорна

## **10. Назначение, принципы построения и характеристики арифметико-логических устройств (АЛУ).**

Арифметико-логическое устройство (АЛУ) (англ. arithmetic and logic unit, ALU) – устройство или же блок процессора, который предназначен для выполнения логических и математических операций над двоичными числами. АЛУ входит в состав любого микропроцессора.

*Пример работы АЛУ на операции сложения*

Функционально АЛУ состоит из двух регистров (Регистр1, Регистр 2), схемы управления и сумматора. Арифметическая операция выполняется по тактам:

- Значения операнда 1, участвующего в арифметической операции по шине данных поступает в Регистр 1
- Значения операнда 2, участвующего в арифметической операции по шине данных поступает в Регистр 2
- По шине инструкций поступает инструкция на выполнение операции в схему управления
- Данные из регистров поступают в сумматор, схема управления дает команду на выполнение сложения
- Результат сложения поступает в Регистр 1
- Признаки выполнения операции в АЛУ поступают в регистр флагов.

Характеристики АЛУ:

- Разрядность
- Состав операция
- Форматы обрабатываемых данных
- Способ построения и функционирования
- Быстродействие
- Надежность

## **11. Многоуровневая организация ЭВМ. Структурная организация и архитектура вычислительных систем.**

### **Многоуровневая организация вычислительных процессов.**

Структурная и функциональная сложность ЭВМ привела к необходимости иерархического подхода с выделением нескольких уровней организации. Существуют пять уровней организации вычислительных процессов в ЭВМ: концептуальный, языков высокого уровня, машинных команд, регистровых передач и логических вентилей.

На концептуальном уровне пользователь ЭВМ анализирует задачу, разрабатывает алгоритм ее решения, определяет содержимое обработки информации.

На уровне языков программирования высокого уровня изучается алгоритм решения задачи, составляется детальный проект решения (определяются структуры данных, содержание отдельных программных модулей, связи между ними), пишется программа на одном из языков высокого уровня.

На уровне машинных команд обеспечивается связь программных и аппаратных средств. На этом уровне составляется список команд, определяются способы кодирования кодов операций и адресов, число адресных полей и другие параметры, заложенные в структуру ЭВМ. При переходе на данный уровень программа пользователя, написанная на языке высокого уровня, преобразуется в программу, составленную из машинных команд с помощью компилятора соответствующего языка высокого уровня. Чем ближе состав операторов языка программирования к списку команд ЭВМ, тем проще компилятор и меньше затраты времени на компиляцию. В частности, минимальных затрат требует ассемблер - компилятор для перевода с языка Ассемблера в машинные коды.

На уровне регистровых передач осуществляются элементарные операции, выполняемые аппаратурой ЭВМ. Это операции преобразования информации, образующие функционально полный набор в структурах арифметики и булевой алгебры, а также операции чтения и записи в запоминающее устройство и регистры, операции коммутации, обеспечивающие передачу слов между отдельными модулями и блоками. Операции этого уровня представляют собой пересылки сигналов между регистрами через логические схемы, в частности через комбинационную схему АЛУ.

На уровне вентилей разрабатываются логические схемы при логическом проектировании аппаратуры ЭВМ. на уровне вентилей рассматриваются операции с отдельными двоичными переменными.

Процессор, ОЗУ, видеоподсистема, дисковая система, периферийные устройства и устройства ввода-вывода. Материнская плата, формируют основу вычислительной системы современного компьютера общего назначения

## **12. Основные архитектурные понятия. Типы команд. Типы данных Способы адресации.**

Методы адресации — в вычислительной технике способы указания на определенную ячейку (ячейки) памяти ЭВМ процессору с целью записи, чтения данных или передачи управления.

Задача адресации заключается в указании на текущую ячейку памяти, к которой происходит обращение процессора. Адрес текущей ячейки, как правило, записывается в один или несколько регистров процессора.

Методы адресации памяти имеют особое значение при программировании на языке низкого уровня (языке ассемблера).

Типы команд:

- Команды пересылки данных
- Арифметические команды
- Логические команды
- Команды переходов

### **13. Виртуальная память и организация защиты памяти.**

Виртуальная память делит физическую память на блоки и распределяет их между различными задачами. При этом она предусматривает также некоторую схему защиты, которая ограничивает задачу теми блоками, которые ей принадлежат. Большинство типов виртуальной памяти сокращают также время начального запуска программы на процессоре, поскольку не весь программный код и данные требуются ей в физической памяти, чтобы начать выполнение.

Кроме того, виртуальная память упрощает также загрузку программ, обеспечивая механизм автоматического перемещения программ, позволяющий выполнять одну и ту же программу в произвольном месте физической памяти.

## **14. Принципы фон-Неймана по построению вычислительных систем.**

1. **Принцип двоичности.** Для представления данных и команд используется двоичная система счисления.
2. **Принцип программного управления.** Программа состоит из набора команд, которые выполняются процессором друг за другом в определённой последовательности.
3. **Принцип однородности памяти.** Как программы (команды), так и данные хранятся в одной и той же памяти. Над командами можно выполнять такие же действия, как и над данными.
4. **Принцип адресуемости памяти.** Структурно основная память состоит из пронумерованных ячеек; процессору в произвольный момент времени доступна любая ячейка.
5. **Принцип последовательного программного управления.** Все команды располагаются в памяти и выполняются последовательно, одна после завершения другой.
6. **Принцип условного перехода.** Команды из программы не всегда выполняются одна за другой. Возможно присутствие в программе команд условного перехода, которые изменяют последовательность выполнения команд в зависимости от значений данных.



## **15. Конвейерная организация. Простейшая организация конвейера и оценка его производительности. Примеры.**

Конвейеризация (или конвейерная обработка) в общем случае основана на разделении подлежащей исполнению функции на более мелкие части, называемые ступенями, и выделении для каждой из них отдельного блока аппаратуры. Так, обработку любой машинной команды можно разделить на несколько этапов (несколько ступеней), организовав передачу данных от одного этапа к следующему. Производительность при этом возрастает благодаря тому, что одновременно на различных ступенях конвейера выполняются несколько команд.

Простейшая организация конвейера:

1. Выборка команды (чтение очередной команды из памяти и занесение ее в регистр команды)
2. Декодирование команды (определение кода операции и способов адресации операндов)
3. Вычисление адресов (вычисление адреса операнда)
4. Выборка операндов (извлечение операндов из памяти. Эта операция не нужна для операндов, находящихся в регистрах)
5. Исполнение команды (непосредственное выполнение команды)
6. Запись результата

Конвейеризация увеличивает пропускную способность процессора (количество команд, завершающихся в единицу времени), но она не сокращает время выполнения отдельной команды.

В качестве примера рассмотрим неконвейерную машину с пятью этапами выполнения операций, которые имеют длительность 50, 50, 60, 50 и 50 нс (наносекунд). Пусть накладные расходы на организацию конвейерной обработки составляют 5 нс. Тогда среднее время выполнения команды в неконвейерной машине будет равно 260 нс. Если же используется конвейерная организация, длительность такта будет равна длительности самого медленного этапа обработки плюс накладные расходы, т.е. 65 нс. Это время соответствует среднему времени выполнения команды в конвейере. Таким образом, ускорение, полученное в результате конвейеризации, будет равно:

$260/65=4$ , где 260 – среднее время выполнения в неконвейерном режиме, а 65 – среднее время выполнения команды в конвейерном режиме

## **16. Общие понятия и определения, структурная схема микропроцессора.**

В состав микропроцессора входят: арифметико-логическое устройство, блок управления и синхронизации, запоминающее устройство, регистры, шины передачи данных и команд.

Арифметико-логическое устройство служит для выполнения арифметических и логических операций над числами, представленными в двоичном коде.

Блок управления памятью — компонент аппаратного обеспечения компьютера, отвечающий за управление доступом к памяти, запрашиваемым центральным процессором.

Запоминающее устройство — устройство, предназначенное для записи и хранения данных. В основе работы запоминающего устройства может лежать любой физический эффект, обеспечивающий приведение системы к двум или более устойчивым состояниям. Устройство, реализующее компьютерную память.

Регистр — устройство для записи, хранения и считывания  $n$ -разрядных двоичных данных и выполнения других операций над ними. Регистр представляет собой упорядоченный набор триггеров, обычно D-триггеров, число которых соответствует числу разрядов в слове. С регистром может быть связано комбинационное цифровое устройство, с помощью которого обеспечивается выполнение некоторых операций над словами.

Компьютерная шина — соединение, служащее для передачи данных между функциональными блоками компьютера.

## **17. Классы конфликтов возникающих в конвейерах и способы их устранения.**

Конфликты — это такие ситуации в конвейерной обработке, которые препятствуют выполнению очередной команды в предназначенном для нее такте. Конфликты снижают реальную производительность конвейера, которая могла быть достигнута в идеальном случае.

**Существуют три класса конфликтов:**

1. Структурные конфликты, которые возникают из-за конфликтов по вычислительным ресурсам, когда аппаратные средства процессора не могут поддерживать все возможные комбинации команд в режиме одновременного выполнения с совмещением. Чтобы устранить конфликт нужно увеличение времени такта до такой величины, которая позволила бы все этапы любой команды выполнять за один такт.
2. Конфликты по данным, возникающие в случаях, когда выполнение одной команды зависит от результата выполнения предыдущей команды. Устранение конфликта заключается в замораживании операций в конвейере путем блокировки выполнения любой команды, следующей за командой условного перехода, до тех пор, пока не станет известным направление перехода.
3. Конфликты управления, которые возникают при конвейеризации команд переходов и других команд, изменяющих значение счетчика команд. Уменьшение влияния конфликта обеспечивается методом, который называется пересылкой или продвижением данных.

Конфликты в конвейере приводят к необходимости приостановки выполнения команд. В простейших конвейерах, если приостанавливается какая-либо команда, то все следующие за ней команды также приостанавливаются. Команды, предшествующие приостановленной, могут продолжать выполняться, но во время приостановки не выбирается ни одна новая команда.

## 18 Дисковые массивы и уровни RAID.

Одним из способов повышения производительности ввода/вывода является использование параллелизма путем объединения нескольких физических дисков в матрицу (группу) с организацией их работы аналогично одному логическому диску.

Что такое **RAID-массив** или просто **RAID**? Расшифровывается эта аббревиатура как "Redundant Array of Independent Disks" или "избыточный (резервный) массив независимых дисков". Говоря по-простому, **RAID-массив** это совокупность физических дисков, объединенных в один логический.

### Виды RAID

1. **RAID1:** Зеркальные диски. Зеркальные диски представляют традиционный способ повышения надежности магнитных дисков. Это наиболее дорогостоящий из рассматриваемых способов, так как все диски дублируются и при каждой записи информация записывается также и на проверочный диск.
2. **RAID 2:** Матрица с поразрядным расслоением. Один из путей достижения надежности при снижении потерь емкости памяти может быть подсказан организацией основной памяти, в которой для исправления одиночных и обнаружения двойных ошибок используются избыточные контрольные разряды.
3. **RAID 3:** Аппаратное обнаружение ошибок и четность
4. **RAID 4:** повышает производительность передачи небольших объемов данных за счет параллелизма, давая возможность выполнять более одного обращения по вводу/выводу к группе в единицу времени. RAID уровня 4 позволяли добиться параллелизма при считывании отдельных дисков, но запись по-прежнему ограничена возможностью выполнения одной операции на группу, так как при каждой операции должны выполняться запись и чтение контрольного диска.
5. **RAID 5:** Основным недостатком уровней RAID от 2-го до 4-го является невозможность производить параллельные операции записи, так как для хранения информации о четности используется отдельный контрольный диск. RAID 5 не имеет этого недостатка.
6. **RAID 6:** Двумерная четность для обеспечения большей надежности.

## **19. Иерархия памяти. Организация кэш-памяти. Принципы организации основной памяти в современных компьютерах.**

Основная память представляет собой следующий уровень иерархии памяти. Основная память удовлетворяет запросы кэш-памяти и служит в качестве интерфейса ввода/вывода, поскольку является местом назначения для ввода и источником для вывода.

Для оценки производительности основной памяти используются два основных параметра: **задержка** и **полоса пропускания**.

Традиционно задержка основной памяти имеет отношение к кэш-памяти, а полоса пропускания или пропускная способность относится к вводу/выводу. В связи с ростом популярности кэш-памяти второго уровня и увеличением размеров блоков у такой кэш-памяти, полоса пропускания основной памяти становится важной также и для кэш-памяти.

**Кэш-память** - промежуточный буфер с быстрым доступом, содержащий информацию, которая может быть запрошена с наибольшей вероятностью, и работающий на частоте ядра. В основу работы кэш-памяти положены два принципа: временной и пространственной локальности.

**Принцип временной локальности** заключается в том, что существует высокая вероятность того, что при считывании данных из памяти программа обратится к этим данным в течение некоторого небольшого интервала времени несколько раз.

**Принцип пространственной локальности** состоит в том, что есть высокая вероятность того, что программа обратится к нескольким последовательно расположенным ячейкам памяти.

Таким образом, в соответствии с принципом временной локальности информация хранится в кэш-памяти в течение небольших интервалов времени, а согласно принципу пространственной локальности в кэш-память целесообразно загружать несколько соседних ячеек.

## 20. Организация регистров современного процессора.

Регистры процессора, как сверхбыстродействующая память, предназначены для хранения операндов и команд, сведений о состоянии вычислительного процесса, обмена данными между МП, ОЗУ и портами, организации адресации и взаимодействия между блоками.

Регистры процессора разделяются по функциям:

- Программно доступные регистры - позволяют программисту, разрабатывающему программы на машинном языке или на языке ассемблера, минимизировать в программе обращение к оперативной памяти и, где это возможно, использовать для хранения промежуточных результатов быстродействующую внутреннюю память процессора.
- Регистры управления и состояния - используются для управления функционированием процессора. К этим регистрам иногда могут иметь доступ специальные привилегированные команды, которые используются только в программах операционной системы.

### Сегментные регистры процессора:

1. **CS – code segment** – сегмент команд. Определяет базовый адрес сегмента памяти, в котором хранится выполняемая программа.
2. **DS – data segment** – сегмент данных. Определяется базовый адрес сегмента памяти, в котором хранятся обрабатываемые данные.
3. **SS – stack segment** – сегмент стека. Определяет базовый адрес сегмента памяти, доступ к которому возможен с помощью стековой адресации.
4. **ES - extra segment** – дополнительный сегмент. Определяет базовый адрес сегмента памяти, в котором наряду с DS могут храниться обрабатываемые данные.

## 21. Фон-неймановская архитектура.

Архитектура фон Неймана — широко известный принцип совместного хранения программ и данных в памяти компьютера. Вычислительные системы такого рода часто обозначают термином «машина фон Неймана», однако, соответствие этих понятий не всегда однозначно. В общем случае, когда говорят об архитектуре фон Неймана, подразумевают физическое отделение процессорного модуля от устройств хранения программ и данных.

Принципы архитектуры фон Неймана:

- 1) **Принцип двоичности.** Для представления данных и команд используется двоичная система счисления.
- 2) **Принцип программного управления.** Программа состоит из набора команд, которые выполняются процессором друг за другом в определённой последовательности.
- 3) **Принцип однородности памяти.** Как программы (команды), так и данные хранятся в одной и той же памяти. Над командами можно выполнять такие же действия, как и над данными.
- 4) **Принцип адресуемости памяти.** Структурно основная память состоит из пронумерованных ячеек; процессору в произвольный момент времени доступна любая ячейка.
- 5) **Принцип последовательного программного управления.** Все команды располагаются в памяти и выполняются последовательно, одна после завершения другой.
- 6) **Принцип условного перехода.** Команды из программы не всегда выполняются одна за другой. Возможно присутствие в программе команд условного перехода, которые изменяют последовательность выполнения команд в зависимости от значений данных.

## **22. BIOS и UEFI. Определение. Состав. Предназначение.**

BIOS — базовая система ввода-вывода, (Basic Input-Output System) является важной частью процедур любого компьютера. По своей сути BIOS является посредником между компьютерным «железом» и операционной системой. Без BIOS операционная система не смогла бы связываться с «железом» и управлять им. BIOS, это программа низкого уровня, хранящаяся на чипе материнской платы вашего компьютера.

BIOS состоит из двух частей: постоянного запоминающего устройства (ПЗУ), где хранится информация, необходимая для запуска ПК (программы) и содержащая неизменяемые параметры системы (по умолчанию); оперативного запоминающего устройства (ОЗУ), где располагаются значения, изменяемые пользователем.

По сути, UEFI это тот же BIOS, только нового поколения. UEFI – более новое решение, он поддерживает жёсткие диски большего объёма, быстрее грузится, более безопасен – и, что очень удобно, обладает графическим интерфейсом и поддерживает мышь.

Основная цель разработки UEFI заключается в стандартизации взаимодействия операционной системы с микропрограммами платформы в ходе процесса загрузки.

## **23. Поколения вычислительных машин.**

1. «Нулевое» -- до 1940 года. Вычислительный элемент -- механический. Простые арифметические операции. Арифмометры, механические счетные машины.
2. «Первое» -- 1940--1960. Вычислительный элемент -- электронные лампы. Быстродействие -- 10 - 20 тысяч операций в секунду. «Большие» ЭВМ.
3. «Второе» - 1960--1964. Вычислительный элемент -- транзисторы. Быстродействие -- до 1--2 миллионов операций в секунду. Мини-ЭВМ.
4. «Третье» -- 1964--1971. Вычислительный элемент -- сверхинтегральные схемы. Быстродействие -- до 300 миллионов операций в секунду. Микро-ЭВМ, предназначенные для работы с одним пользователем.
5. «Четвертое» -- 1971 - по настоящее время. Вычислительный элемент -- микропроцессоры. Быстродействие -- миллиарды операций в секунду. Персональные ЭВМ.
6. «Пятое» -- настоящее время -- Нанотехнологии. Компьютеры на основе отдельных молекул и даже атомов. Нейросети, моделирующие структуру нервной системы человека. «Биологические компьютеры».



## 24. Классификация вычислительных систем по Флинну.

- 1) **SISD (single instruction, single data)** – системы с одиночным потоком команд и одиночным потоком данных. К этому классу относятся, прежде всего, классические последовательные машины (однопроцессорные ЭВМ), или иначе, машины фон-неймановского типа, например, PDP-11 или VAX 11/780.
- 2) **SIMD (single instruction, multiple data)** – системы с одиночным потоком команд и множественным потоком данных. В такой ВС имеется несколько одинаковых процессоров, которые выполняют команды из единственного потока команд.
- 3) **MISD (multiple instruction, single data)** – системы с множественным потоком команд и одиночным потоком данных. В такой системе единственный поток данных проходит через несколько процессоров, каждый из которых выполняет свою последовательность команд.
- 4) **MIMD (multiple instruction, multiple data)** - системы с множественным потоком команд и множественным потоком данных. Этот класс предполагает, что в вычислительной системе есть несколько устройств обработки команд, объединенных в единый комплекс, каждое из которых работает со своим потоком команд и данных. Основное отличие этих систем от многопроцессорных SIMD-машин состоит в том, что инструкции и данные связаны, потому что они относятся к одной и той же исполняемой задаче.

## 25. Состав системного блока современной рабочей станции. Единицы измерения рабочих частот процессоров и системных шин. Единицы измерения всех видов памяти.

### Состав системного блока:

1. Корпус
2. Блок питания
3. Процессор (CPU - центральный процессор) – это главный вычислительный элемент персонального компьютера. Все программы состоят из огромной последовательности микрокоманд, и именно процессор выполняет эти команды.
4. Материнская плата –объединяет все компоненты системного блока.
5. Корпусный вентилятор – используется для охлаждения системника.
6. Планки оперативной памяти (ОЗУ) - это быстродействующая память компьютера.
7. Видеокарта - устройство, которое обрабатывает и выводит графическую информацию на монитор. У многих CPU есть встроенная графика, но она слабая
8. Сетевая карта – элемент системного блока, необходимый для соединения компьютера с локальной сетью или сетью Интернет.
9. Оптический накопитель (CD/DVD) – устройство для чтения и записи оптических дисков.
10. Жесткий диск (harddisk, HDD, винчестер) - это устройство долговременной памяти.

**Тактовая частота** — показатель скорости выполнения команд центральным процессором. Такт — промежуток времени, необходимый для выполнения элементарной операции. Единицей одного такта принято считать 1 Гц (Герц). Это значит, что если частота равна 1 ГГц (Гига Герц), то ядро процессора выполняет 1 млрд. тактов.

### *Единицы измерения информации*

Байт	В	$2^0$	$2^0$
Килобайт	KB	1024 байта	$2^{10}$
Мегабайт	MB	1024 KB	$2^{20}$
Гигабайт	GB	1024 MB	$2^{30}$
Терабайт	TB	1024 GB	$2^{40}$
Петабайт	PB	1024 TB	$2^{50}$
Экзабайт	EB	1024 PB	$2^{60}$
Зеттабайт	ZB	1024 EB	$2^{70}$
Йоттабайт	YB	1024 ZB	$2^{80}$

## **26. Организация ввода/вывода в вычислительной системе. Системные и локальные шины. Устройства ввода/вывода.**

Назначение системы ввода-вывода – это, обеспечение центральной части машины с внешней средой представленными периферийными устройствами.

Вводом/выводом называют передачу данных между ядром ЭВМ, включающим в себя процессор и ОП, и периферийными устройствами.

Система ввода-вывода – это единственное средство общения ЭВМ с внешним миром. Ее возможности в серийных ЭВМ представляют собой один из важнейших параметров, определяющих выбор машины для конкретного применения.

Существует три режима ввода-вывода:

1. Программный ввод-вывод (нефорсированный).
2. Ввод-вывод по прерыванию (форсированный).
3. Прямой доступ к памяти (ПДП).

Шина - набор соединений, по которым передаются различные сигналы. Набор проводников (металлизированных дорожек на материнской плате), по которым передается информация в виде электрических сигналов.

Системная шина – это основная интерфейсная система компьютера.

Локальная шина - это скоростная шина, предназначенная для обмена информацией между быстродействующими периферийными устройствами (видеоадаптерами, сетевыми картами и др.) и системной шиной под управлением CPU.

## **27. Блоки управления командами. Структура устройства управления. Принципы организации систем прерываний. Процедура обслуживания прерываний.**

Прерыванием называется процесс, обуславливающий реакцию компьютера на некоторое событие, требующее немедленного его вмешательства. Процесс выполнения такой подпрограммы называется обработкой (обслуживанием) данного прерывания, а сама подпрограмма обслуживания прерывания часто называется обработчиком прерывания.

Выполнение машинной команды в процессоре, разбивается на этапы называемыми машинными циклами. Реализация этих этапов в процессоре осуществляется двумя способами управления: 1) микропрограмма ; 2) жесткая логика.

**Микропрограммное управление.**

Суть этого способа заключается в том, что блок управления содержит память, в которую записываются микропрограммы, в которых каждая микрокоманда управляет процессором в течение машинного такта.

Для этого в каждом такте процессора из памяти микропрограмм происходит чтение микрокоманды, которая содержит информацию, изменяющую состояния элементов в блоках процессора в течение машинного такта.

### **Жесткая логика**

RISC архитектура реализовала конвейерную обработку команд, архитектура в которой функциональные блоки процессора, отвечающие за выполнение этапов команд стали независимы друг от друга, а сами команды упростились, и необходимость в их интерпретации через средства микропрограммного управления отпала сама собой, а жесткая логика взяла на себя функции управления конвейером, устранение конфликтов, выполнение операций в функциональных блоках и реакции на все отклонения в процессе выполнения команд, представляя вышестоящему уровню архитектуры команд через систему прерывания всю информацию для обработки этих отклонений.

Во время выполнения ЭВМ текущей программы внутри машины и в связанной с ней внешней среде (например, в технологическом процессе, управляемом ЭВМ) могут возникать события, требующие немедленной реакции на них со стороны машины.

Реакция состоит в том, что машина прерывает обработку текущей программы и переходит к выполнению некоторой другой программы, специально предназначенной для данного события. По завершении этой программы ЭВМ возвращается к выполнению прерванной программы.

### **Основными функциями системы прерывания являются:**

1. запоминание состояния прерываемой программы и осуществление перехода к прерывающей программе;
2. восстановление состояния прерванной программы и возврат к ней.

## 28. Классификация ЭВМ по областям применения.

а) **ПК** — настольная микро-ЭВМ, имеющая эксплуатационные характеристики бытового прибора и универсальные функциональные возможности. ПК применяются как средства массовой автоматизации (в основном для создания на их основе автоматизированных рабочих мест) в социальной и производственных сферах деятельности в различных областях народного хозяйства и предназначенные для пользователей, не обладающих специальными знаниями в области вычислительной техники и программирования.

б) **Х-терминал** - комбинация бездисковых рабочих станций и стандартных ASCII-терминалов. Бездисковые рабочие станции часто применялись в качестве дорогих дисплеев и в этом случае не полностью использовали локальную вычислительную способность.

в) **Сервер** — компьютер, выделенный из группы ПК (или рабочих станций) для выполнения какой-либо сервисной задачи без непосредственного участия человека. Сервер и рабочая станция могут иметь одинаковую аппаратную конфигурацию, так как различаются лишь по участию в своей работе человека за консолью.

г) **Мейнфрейм** - большой универсальный высокопроизводительный отказоустойчивый сервер со значительными ресурсами ввода-вывода, большим объемом оперативной и внешней памяти, предназначенный для использования в критически важных системах с интенсивной пакетной и оперативной транзакционной обработкой

д) **Кластер** - два или более компьютеров (часто называемых узлами), объединяемые при помощи сетевых технологий на базе шинной архитектуры или коммутатора и предстающие перед пользователями в качестве единого информационно-вычислительного ресурса. Возможности масштабируемости кластеров позволяют многократно увеличивать производительность приложений для большого числа пользователей технологий. Кластеризация может осуществляться на разных уровнях компьютерной системы, включая аппаратное обеспечение, операционные системы, программы-утилиты, системы управления и приложения. Чем больше уровней системы объединены кластерной технологией, тем выше надежность, масштабируемость и управляемость кластера

е) **Суперкомпьютеры** - специализированная вычислительная машина, значительно превосходящая по своим техническим параметрам и скорости вычислений большинство существующих в мире компьютеров. Как правило, современные СК представляют собой большое число высокопроизводительных серверных компьютеров, соединенных друг с другом локальной высокоскоростной магистралью для достижения максимальной производительности в рамках подхода распараллеливания вычислительной задачи.

## **29. Цикл обработки команды современного процессора.**

Большинство современных процессоров для ПК в общем основано на той или иной версии циклического процесса последовательной обработки данных, изобретенного Джоном фон Нейманом. Отличительной особенностью архитектуры фон Неймана является то, что инструкции и данные хранятся в одной и той же памяти. В различных архитектурах и для различных команд могут потребоваться дополнительные этапы. Например, для арифметических команд могут потребоваться дополнительные обращения к памяти, во время которых производится считывание операндов и запись результатов.

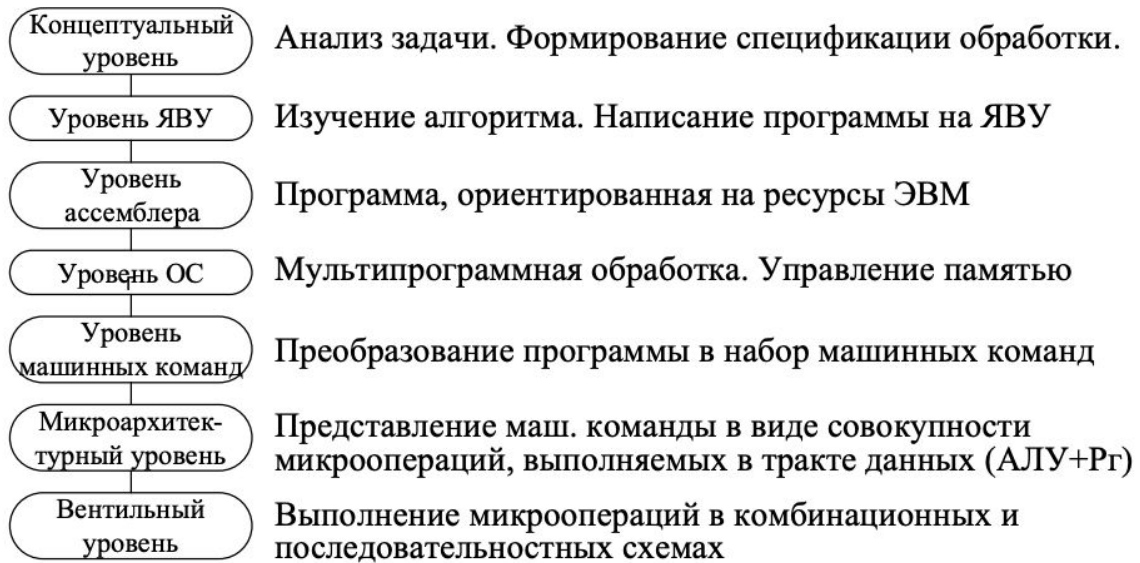
### **Этапы цикла выполнения:**

1. Процессор выставляет число, хранящееся в регистре счётчика команд, на шину адреса и отдаёт памяти команду чтения.
2. Выставленное число является для памяти адресом; память, получив адрес и команду чтения, выставляет содержимое, хранящееся по этому адресу, на шину данных и сообщает о готовности
3. Процессор получает число с шины данных, интерпретирует его как команду (машинную инструкцию) из своей системы команд и исполняет её
4. Если последняя команда не является командой перехода, процессор увеличивает на единицу (в предположении, что длина каждой команды равна единице) число, хранящееся в счётчике команд; в результате там образуется адрес следующей команды.

Данный цикл выполняется неизменно, и именно он называется процессом (откуда и произошло название устройства). Во время процесса процессор считывает последовательность команд, содержащихся в памяти, и исполняет их. Такая последовательность команд называется программой и представляет алгоритм работы процессора. Очередность считывания команд изменяется в случае, если процессор считывает команду перехода, — тогда адрес следующей команды может оказаться другим. Другим примером изменения процесса может служить случай получения команды останова или переключение в режим обработки прерывания.

### 30. Многоуровневая организация ЭВМ. Многоступенчатая обработка.

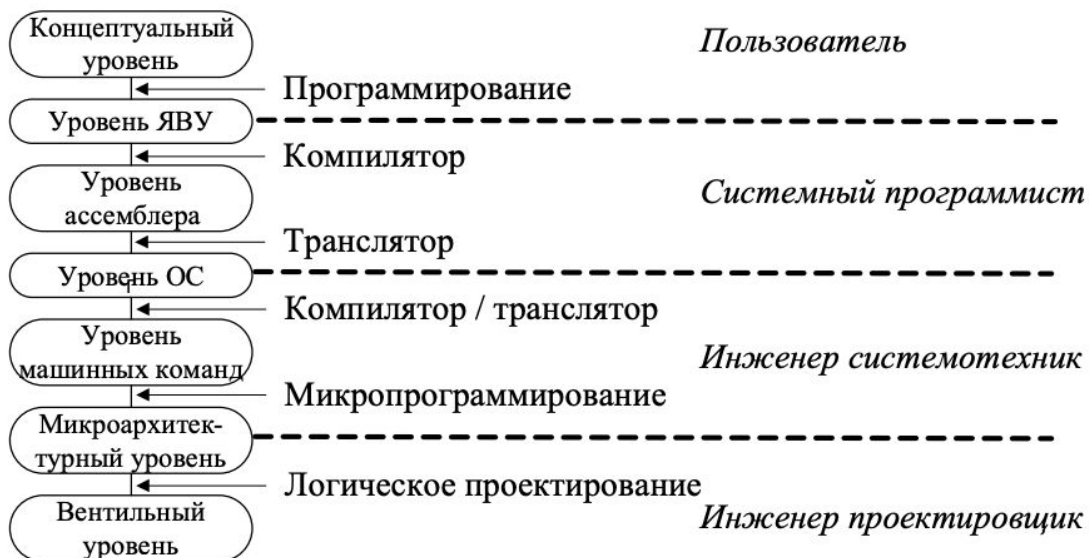
#### Многоступенчатая обработка



© С. Г. Мосин, 2007

53

#### Многоступенчатая обработка



© С. Г. Мосин, 2007

54

### 31. Оценка производительности вычислительных систем.

Единицей измерения производительности компьютера является время: компьютер, выполняющий тот же объем работы за меньшее время, является более быстрым. Время выполнения любой программы измеряется в секундах. Часто производительность измеряется как скорость появления некоторого числа событий в секунду, так что меньшее время подразумевает большую производительность. Еще один хороший способ – измерить время выполнения не одной, а нескольких программ, похожих на те, которые вы планируете запускать; это особенно важно, если ваша программа еще не написана или измерения проводит кто-то, у кого ее нет. Такие программы называются тестовым набором (benchmark), а полученные времена обычно публикуются, чтобы было ясно, насколько быстр компьютер. Время выполнения программы в секундах можно вычислить по формуле:

$$\text{Execution Time} = (\# \text{ instructions}) \left( \frac{\text{cycles}}{\text{instruction}} \right) \left( \frac{\text{seconds}}{\text{cycle}} \right)$$

Главная задача, стоящая перед разработчиком микроархитектуры – создать процессор, который обеспечивал бы наименьшее возможное время выполнения программ, в то же время удовлетворяя ограничениям по цене и/или энергопотреблению.



### **32. Понятие архитектуры вычислительной системы.**

**Архитектура ВС** — совокупность характеристик и параметров, определяющих функционально-логическую и структурную организацию системы. Понятие архитектуры охватывает общие принципы построения и функционирования. Компоновка основных частей ВС и связи между ними.

Под архитектурой ВС обычно понимается состав её компонент, принципы их взаимодействия, функции и характеристики (концептуальная структура ВС)

### **33. Организация автоматической работы ЭВМ. Управляющие функции процессора. Общая организация выполнения программы на ЭВМ.**

Каждая ЭВМ является автоматом, состоящим из памяти, образуемой внешним и оперативным запоминающими устройствами, устройства управления (УУ) и арифметические устройства (АУ), в котором могут выполняться некоторые действия или операции. Память имеет вид нумерованной последовательности ячеек, в каждой из которых хранится порция двоичной информации в виде серии нулей и единиц. Автоматическая работа ЭВМ, управляемая программой, состоит из последовательности тактов. На каждом такте УУ выбирает из предписанной ему ячейки памяти порцию информации. Эта порция трактуется как команда, т. е. предписание АУ выполнить некоторую операцию. Обычно в ЭВМ выполнение операции состоит в том, чтобы из определённых ячеек памяти взять хранящуюся там информацию, передать её в АУ для выполнения над ней нужного действия, результат которого отправить в указанную ячейку памяти, и сообщить УУ номер ячейки следующей команды. Отдельные действия, совершаемые ЭВМ, весьма просты — это арифметические и логические операции, операции сравнения, переписывания порции информации и т.п.

В состав процессора входят следующие устройства: **устройство управления (УУ), арифметико-логическое устройство (АЛУ), регистры процессорной памяти.** УУ управляет работой всех устройств компьютера по заданной программе. УУ извлекает очередную команду из регистра команд, определяет, что надо делать с данными, а затем задает последовательность действий выполнения поставленной задачи.

#### **Функции процессора ограничиваются реализацией следующих процедур:**

- Выборка команды из ОП и операндов, указанных в адресной части команды.
- Выполнение операции, заданной кодом операции, что сводится к выполнению: арифметических, логических операций, передачи команд на выполнение средствами ввода/вывода и формировании адреса следующей команды.

Для того чтобы программа могла быть выполненной, она должна быть помещена в оперативную память компьютера. Туда же должны быть помещены и исходные данные. Как правило, программа вводится в оперативную память с жесткого диска. Исходные данные вводятся с клавиатуры либо также с жесткого диска, куда они должны быть заранее помещены с помощью другой программы. Результаты своей работы программа помещает в определенную область оперативной памяти, куда они могут быть выведены на какое-либо внешнее устройство, которым может быть жесткий диск, экран дисплея, печатающее устройство



### 34. Память и запоминающие устройства. Виды и характеристики ЗУ: адресная, стековая и ассоциативная организация памяти.

ЗУ, как правило, содержит множество одинаковых запоминающих элементов, образующих запоминающий массив. Массив разделен на отдельные ячейки; каждая из них предназначена для хранения двоичного кода, число разрядов в котором определяется шириной выборки памяти (байт, машинное слово или несколько слов). Способ организации памяти зависит от методов размещения и поиска информации в запоминающем массиве. По этому признаку различают адресную, ассоциативную и стековую память.

а) **Адресная память** размещение и поиск информации в запоминающем массиве основаны на использовании адреса хранения слова. Другими словами, адресный поиск предполагает, что искомый операнд извлекается из ячейки памяти, номер которой формируется на основе информации в адресном поле команды. По коду адреса в регистре адреса блок адресной выборки формирует в соответствующей ячейке памяти сигналы, позволяющие произвести считывание или запись слова в ячейку.

б) **Ассоциативная память** поиск нужной информации производится не по адресу, а по ее содержанию (по ассоциативному признаку). При этом поиск по АП (или последовательно по отдельным его разрядам) происходит параллельно во времени для всех ячеек запоминающего массива. Память этого типа применяется в специализированных вычислительных машинах, машинах баз данных и при организации работы кэш-памяти.

в) **Стековая память**, так же как и ассоциативная, является безадресной. В ней ячейки образуют одномерный массив, в котором соседние ячейки связаны друг с другом разрядными цепями передачи слов. Запись нового слова производится в верхнюю ячейку (ячейку 0), при этом все ранее записанные слова (включая слово, находившееся в ячейке 0), сдвигаются вниз, в соседние ячейки с большими на 1 номерами. Считывание возможно только из верхней (нулевой) ячейки памяти, при этом, если

производится считывание с удалением, все остальные слова в памяти сдвигаются вверх, в соседние ячейки с большими номерами. В этой памяти порядок считывания слов соответствует правилу: последним поступил — первым обслуживается.

### **35. Тестирование вычислительных систем.**

**Результаты тестирования позволяют:**

- А) на стадии проектирования принимать решения о примерном составе будущей ВС;
- Б) на стадии отладки и ввода в эксплуатацию путем сравнения с аналогами принимать решение о достижении необходимых характеристик;
- В) на стадии эксплуатации/модификации выявлять узкие места системы.

**Требования к тестам:**

**Полнота.** Тест должен оценивать только те параметры, для оценки которых создавался. Выдаваемые результаты должны быть непротиворечивыми, лаконичными и легкими для понимания.

**Легкость в использовании.**

**Масштабируемость.** Тест должен быть доступен для большого числа разного по вычислительной мощности аппаратного обеспечения

**Переносимость.** Тест должен быть доступен для большого числа разного по архитектуре аппаратного обеспечения. Основной чертой переносимости является язык программирования, и, соответственно, наличие компилятора под данную платформу

**Репрезентативность.** Вне зависимости от платформы тест должен оценивать наиболее важные для большинства приложений пользователей характеристики (выборка характеристик должна быть показательной)

**Доступность.** Тест должен быть доступен, в том числе и его исходный код. При представлении результатов должна быть указана версия и все внесенные изменения

**Воспроизводимость.** При необходимости должна быть возможность повторить тест с получением аналогичных результатов. При публикации результатов необходимо предоставлять исчерпывающую информацию о программном и аппаратном обеспечении

### 36. Общие требования, предъявляемые к современным ЭВМ.

**А) Отношение стоимость-производительность.** Появление нового направления в вычислительной технике определяется требованием компьютерного рынка. Поэтому у разработчиков ЭВМ нет одной единой цели; при проектировании высокопроизводительной конструкции приходится игнорировать стоимостные характеристики. Другим примером служит низкостоймостная конструкция, где производительность принесена в жертву стоимости. Между двумя этими направлениями находятся конструкции, в которых разработчики находят баланс между стоимостными параметрами и производительностью.

**Б) Надежность и отказоустойчивость.** Важнейшей характеристикой ВС является **надежность**. Повышение надежности основано на принципе предотвращения неисправности путем снижения сбоев за счет: применения электронных схем и компонентов со сверхвысокой степенью интеграции; облегчения режима работы схем; обеспечения тепловых режимов; совершенствование методов сборки аппаратуры.

**Отказоустойчивость** – свойство ВС, обеспечивающее ей, как логической машине, возможность продолжения действий заданной программы, после возникновения неисправности. Введение отказоустойчивости требует дополнительного аппаратного и программного обеспечения. Направления, связанные с предотвращением неисправности, и с отказоустойчивостью – основные в проблеме надежности ВС. Понятие отказоустойчивости (надежности) включает не только аппаратные средства, но и программное обеспечение.

**В) Масштабируемость** – возможность наращивания числа и мощности процессоров, объема памяти и других ресурсов ВС. Масштабируемость должна обеспечиваться архитектурой и конструкцией ЭВМ, соответствующими средствами программного обеспечения. Добавление каждого нового процессора должно давать прогнозируемое увеличение производительности и пропускной способности при приемлемых затратах. В идеале добавление процессора к системе должно приводить к линейному росту ее производительности.

### **37. Эталонная модель взаимодействия открытых систем.**

В соответствии с этой моделью процесс передачи сообщений в системах связи последовательно разбивается на принципиально различающиеся операции. Каждую из этих операций относят к своему уровню. Уровни строятся по принципу строгой иерархии: на высшем уровне находятся источник и получатель информации -пользователи СС, на нижнем - среда распространения электромагнитных волн. Высший уровень управляет работой низшего. Каждому уровню соответствует свое техническое устройство или организационная единица СС, пользователь или должностное лицо, обеспечивающее функционирование СС. В некоторых СС часть этих устройств может отсутствовать либо выполнять не все функции некоторого уровня. В ЭМВОС выделяют 7 уровней:

- **пользовательский**
- **представительский**
- **сеансовый**
- **транспортный**
- **сетевой**
- **канальный**
- **физический**

Полную совокупность средств у одного пользователя, выполняющих операции различных уровней, называют станцией.

**На пользовательском уровне** происходят процессы обработки информации, передаваемой СС. Исполнителем функций этого уровня может быть как ЭВМ, так и человек. Устройства представительского уровня преобразуют сообщения из формы представления, удобной пользователю, к форме представления, удобной СС, и обратно. В частности, на этом уровне происходит сжатие информации, поскольку СС всегда удобно, чтоб сообщение занимало наименьший объем.

**Устройства сеансового уровня** обрамляют передаваемое сообщение служебной информацией с тем, чтобы количество топологических вариантов передачи было возможно большим. Выбор наилучшего варианта осуществляется устройствами нижних уровней. Таким образом, этот уровень отвечает за организацию сеанса связи.

**На транспортном уровне** принимается решение о перемещении данного сообщения к пользователю на уровне выбора необходимых сетей связи. Для этого решается задача межсетевой адресации сообщений и задача передачи сообщений между сетями различного рода, называемая задачей шлюзования.

**На сетевом уровне** решается задача наилучшей доставки сообщения к пользователю в рамках, одной сети связи. Для этого выбирается маршрут движений сообщения подсети, решается задача внутрисетевой адресации пользователей.

**Устройства канального уровня** обеспечивают защиту передаваемых сообщений от искажений, которые возникают вследствие изменения параметров сигналов в процессе распространения.

**Устройства физического уровня** обеспечивают преобразование передаваемого сообщения, в сигналы и восстановление сообщения по принятому сигналу.

Правила, по которым взаимодействуют устройства соседних уровней одной станции, называют **интерфейсом**.

Правила, по которым взаимодействуют устройства одинаковых уровней у различных станций, называют **протоколом**.

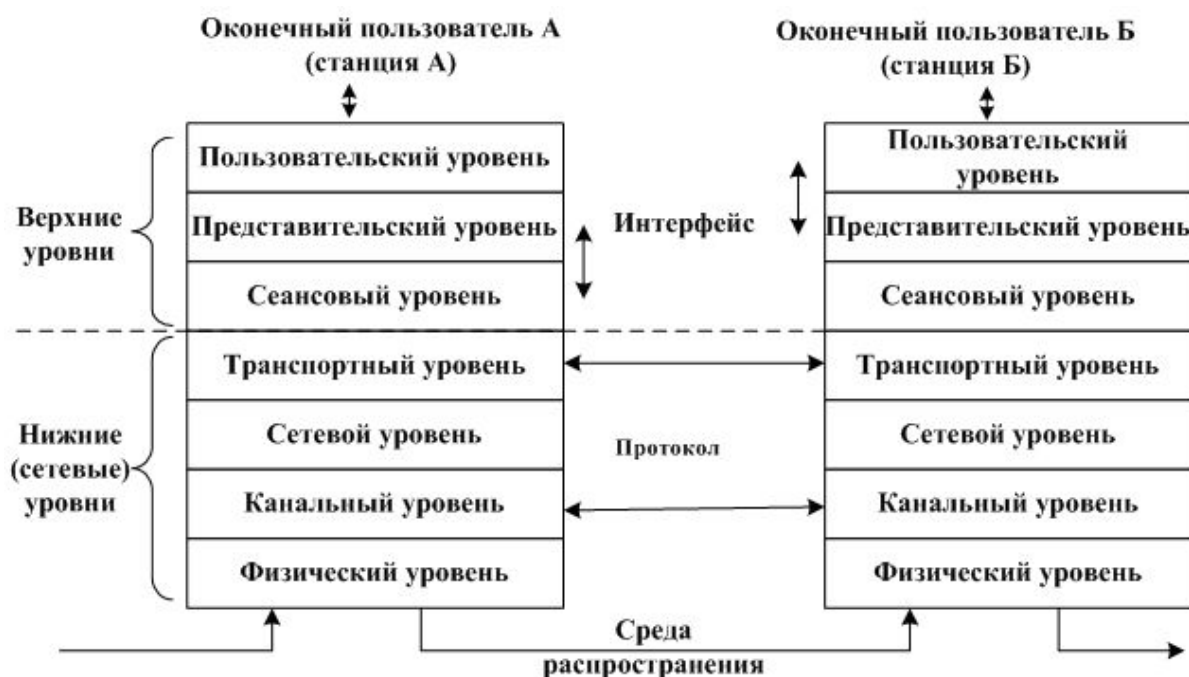


Рис. 1.4. Эталонная модель взаимодействия открытых систем (ЭМВОС)

### 38. Многоуровневая организация управления в ИВС. Протоколы и интерфейсы. Способы и средства коммутации и передачи данных.

В модели OSI различается два основных типа **протоколов**.

В протоколах с **установлением соединения** (connection-oriented network service, CONS) перед обменом данными отправитель и получатель должны сначала установить соединение и, возможно, выбрать протокол, который они будут использовать. После завершения диалога они должны разорвать это соединение.

Вторая группа протоколов - протоколы **без предварительного установления соединения** (connectionless network service, CLNS). Такие протоколы называются также дейтаграммными протоколами. Отправитель просто передает сообщение, когда оно готово. При взаимодействии компьютеров используются как те, так и другие протоколы.

**Интерфейс** — определяет последовательность и формат сообщений, которыми обмениваются сетевые компоненты, лежащие на соседних уровнях в одном узле. Интерфейс определяет набор услуг, предоставляемый данным уровнем соседнему уровню. В сущности, протокол и интерфейс выражают одно и то же понятие, но традиционно в сетях за ними закреплены разные области действия: **протоколы** определяют правила взаимодействия модулей одного уровня в разных узлах, а **интерфейсы** — модулей соседних уровней в одном узле. Средства каждого уровня должны отрабатывать, во-первых, собственный протокол, а во-вторых, интерфейсы с соседними уровнями.

Основная функция систем передачи данных в условиях функционирования вычислительных сетей заключается в организации быстрой и надежной передачи информации произвольным абонентам сети, а также в сокращении затрат на передачу данных. Важнейшая характеристика сетей передачи данных — время доставки информации — зависит от структуры сети передачи данных, пропускной способности линий связи, а также от способа соединения каналов связи между взаимодействующими абонентами сети и способа передачи данных по этим каналам.

В настоящее время различают системы передачи данных с **постоянным включением каналов** связи (некоммутируемые каналы связи) и **коммутацией** на время передачи информации по этим каналам. При коммутации абонентских пунктов и ЭВМ только на время передачи информации (т. е. нормальным режимом для которых является режим «off-line»). Принцип построения узла коммутации определяется способами организации прохождения информации в сетях передачи данных. Существуют три основных способа подготовки и передачи информации в сетях, основанных на коммутации: **каналов, сообщений и пакетов**.

**Коммутация каналов.** Способ коммутации каналов заключается в установлении физического канала связи для передачи данных непосредственно между абонентами



сети. При использовании коммутируемых каналов путь передачи данных образуется из самих каналов связи и устройств коммутации, расположенных в узлах связи.

Установление соединения заключается в том, что абонент посылает в канал связи заданный набор символов, прохождение которых по сети через соответствующие узлы коммутации вызывает установку нужного соединения с вызываемым абонентом. Этот транзитный канал образуется в начале сеанса связи, остается фиксированным на период передачи всей информации и разрывается только после завершения передачи информации.

**При коммутации сообщений** поступающая на узел связи информация передается в память узла связи, после чего анализируется адрес получателя. В зависимости от занятости требуемого канала сообщение либо передается в память соседнего узла, либо становится в очередь для последующей передачи. Таким образом, способ коммутации сообщений обеспечивает поэтапный характер передачи информации. Основное функциональное назначение центра коммутации сообщений — обеспечить автоматическую передачу информации от абонента к абоненту в соответствии с адресным признаком сообщения и требованиями к качеству и надежности связи.

**Коммутация пакетов** сочетает в себе ряд преимуществ методов коммутации каналов и коммутации сообщений. При коммутации пакетов перед началом передачи сообщение разбивается на короткие пакеты фиксированной длины, которые затем передаются по сети. В пункте назначения эти пакеты вновь объединяются в первоначальное сообщение, а так как их длительное хранение в запоминающем устройстве узла связи не предполагается, пакеты передаются от узла к узлу с минимальной задержкой во времени. В этом отношении указанный метод близок методу коммутации каналов. При коммутации пакетов их фиксированная длина обеспечивает эффективность обработки пакетов, предотвращает блокировку линий связи и значительно уменьшает емкость требуемой промежуточной памяти узлов связи. Кроме того, сокращается время задержки при передаче информации, т. е. скорость передачи информации превышает аналогичную скорость при методе коммутации сообщений.

Сети, обеспечивающие коммутацию каналов, сообщений и пакетов, называются **интегральными**. Они объединяют несколько коммутационных сетей. Часть интегральных каналов используется монопольно, т. е. для прямого соединения. Прямые каналы создаются на время проведения сеанса связи между различными коммутационными сетями. По окончании сеанса прямой канал распадается на независимые магистральные каналы.

### **39. Функции сетевого и транспортного уровней, маршрутизация пакетов. Управление потоками передаваемых данных.**

#### **Функции сетевого уровня:**

- Передача пакетов между конечными узлами в составных сетях. Сетевой уровень выступает в качестве координатора, организующего работу всех подсетей, лежащих на пути продвижения пакета по составной сети. Составная сеть (Интернет) – это совокупность нескольких сетей, называемых также подсетями (subnet), которые соединяются между собой маршрутизаторами.
- Выбор маршрута передачи пакетов, наилучшего по некоторому критерию.
- Согласование разных протоколов канального уровня, используемых в отдельных подсетях составной сети. Для перемещения данных в пределах подсетей сетевой уровень обращается к используемым в этих подсетях технологиям.
- На сетевом уровне выполняется одна из важнейших функций маршрутизатора – фильтрация трафика. Маршрутизаторы позволяют администраторам задавать разные правила фильтрации. Например, запретить прохождение в корпоративную сеть всех пакетов, кроме пакетов, поступающих из подсетей этого же предприятия. Фильтрация в данном случае происходит по сетевым адресам. Программное обеспечение маршрутизатора может реализовать различные дисциплины обслуживания очередей пакетов, а также различные варианты приоритетного обслуживания.
- На сетевом уровне проверяется контрольная сумма, и если пакет пришел поврежденным, то он отбрасывается (сетевой уровень коррекцией ошибок не занимается). Также проверяется время жизни пакета – не превышает ли оно допустимой величины (если превысило – то пакет отбрасывается).

#### **Транспортные функции** зависят от сетевого сервиса и включают:

- Отображения транспортного адреса на сетевом адресе;
- Мультиплексирование и расщепление транспортных соединений на сетевые соединения;
- Установление и расторжение транспортных соединений;
- Управление потоком на отдельных соединениях;
- Обнаружение ошибок и управление качеством сервиса;
- Исправление ошибок;
- Сегментирование, блокирование и сцепление;
- Передача срочных блоков данных.

**Маршрутизация** - процесс определения маршрута данных в сетях связи.

**Пакет** - это оформленный блок данных, передаваемый в сети пакетным режимом.

Пакет состоит из двух типов данных: управляющей информацией и пользовательской

**Управление потоком передачи данных** — в компьютерных сетях, механизм, который притормаживает передатчик данных при неготовности приёмника

**Различают три основных способа:**

1. Аппаратный, при котором сигналы «готов/занят» передаются по отдельным физическим линиям связи. Наиболее известна такая реализация в интерфейсе RS-232
2. Программный, при котором программный флажок «готов/занят» взводится и сбрасывается вставкой в поток данных специальной уникальной последовательности (XOn/XOff). Применяется в программных драйверах интерфейса RS-232 как альтернатива аппаратному контролю потока в случаях неполного соединительного кабеля
3. протокольный, при котором программный флажок «готов/занят» взводится и сбрасывается специальными соглашениями в рамках протокола обмена данными. На сегодня является практически единственным применяемым способом контроля потока. Наиболее известный пример — реализация контроля потока в протоколе TCP методом скользящего окна

#### **40. Количество информации и энтропия. Кодирование информации, способы контроля правильности передачи данных.**

**Количество информации в сообщении (элементе сообщения)** определяется по формуле:  $I = -\log_2 P$ , где  $P$  - вероятность появления сообщения (элемента сообщения). Из этой формулы следует, что единица измерения количества информации есть количество информации, содержащееся в одном бите двоичного кода при условии равной вероятности появления в нем 1 и 0. В то же время один разряд десятичного кода содержит  $I = -\log_2 P = 3,32$  единиц информации (при том же условии равновероятности появления десятичных символов, т.е. при  $P = 0,1$ )

**Энтропия** источника информации с независимыми и равновероятными сообщениями есть среднее арифметическое количество информации сообщений  $H = - \sum_{k=1..N} P_k \cdot \log_2 P_k$

где  $P_k$  - вероятность появления  $k$ -го сообщения. Другими словами, **Энтропия есть мера неопределенности ожидаемой информации.**

**Кодирование** - представление сообщения последовательностью элементарных символов. Примеры:

- EBCDIC (Extended Binary Coded Decimal Interchange Code) - символы кодируются восемью битами; популярен благодаря его использованию в IBM.
- ASCII (American Standards Committee for Information Interchange) - семибитовый двоичный код.

**Управление правильностью (помехозащищенностью)** передачи информации выполняется с помощью помехоустойчивого кодирования. Различают коды, обнаруживающие ошибки, и корректирующие коды, которые дополнительно к обнаружению еще и исправляют ошибки. Помехозащищенность достигается с помощью введения избыточности. Устранение ошибок с помощью корректирующих кодов (такое управление называют Forward Error Control) реализуют в симплексных каналах связи. В дуплексных каналах достаточно применения кодов, обнаруживающих ошибки (Feedback or Backward Error Control), так как сигнализация об ошибке вызывает повторную передачу от источника. Это основные методы, используемые в информационных сетях.

Простейшими способами обнаружения ошибок являются контрольное суммирование, проверка на нечетность. Однако они недостаточно надежны, особенно при появлении пачек ошибок. Поэтому в качестве надежных обнаруживающих кодов применяют циклические коды. Примером корректирующего кода является код Хемминга.

## 41. Структура и принципы построения ЛВС. Архитектура одноранговых сетей и сетей "клиент-сервер".

**Локальная вычислительная сеть (ЛВС)** - Группа ЭВМ, а также периферийное оборудование, объединенные одним или несколькими автономными (не арендуемыми) высокоскоростными каналами передачи цифровых данных (в том числе проводными, волоконно—оптическими, радио—СВЧ или ИК—диапазона) в пределах одного или нескольких близлежащих зданий. Служит для решения комплекса взаимосвязанных функциональных и/или информационных задач (например, в рамках какой—либо организации или ее автоматизированной системы), а также совместного использования объединенных информационных и вычислительных ресурсов. В зависимости от принципов построения ЛВС подразделяются на типа «**клиент—сервер**» и «**файл—сервер**» а также «**одноранговые**». ЛВС могут иметь в своем составе средства для выхода в распределенные и глобальные вычислительные сети.

Существуют две основные архитектуры сети: **одноранговая (peer-to-peer)** и **клиент/сервер (client/ server)**, причем вторая практически вытеснила первую. В **одноранговой сети** все компьютеры равны — имеют один ранг. Любой компьютер может выступать как в роли сервера, то есть предоставлять свои ресурсы (файлы, принтеры) другому компьютеру, так и в роли клиента, другими словами — использовать предоставленные ему ресурсы. **Одноранговые сети** преимущественно распространены в домашних сетях или небольших офисах. В самом простом случае для организации такой сети нужно всего лишь пара компьютеров, снабженных сетевыми платами и коаксиальный кабель. Когда сеть создана физически (компьютеры связаны с помощью коаксиального кабеля), нужно настроить сеть программно. Для этого необходимо, чтобы на компьютерах были установлены сетевые операционные системы (Linux, FreeBSD, Windows NT, Windows 98) или сетевые системы с поддержкой сетевых функций (Windows 95, Windows for Workgroups) Компьютеры в одноранговой сети объединяются в рабочие группы. Каждая рабочая группа имеет свой идентификатор — имя рабочей группы. Единственное ограничение доступа, которое возможно в одноранговой сети, это использование пароля для доступа к какому-нибудь ресурсу. Для того, чтобы получить доступ к этому ресурсу, например, принтеру, нужно знать пароль. Это называется управлением доступом на уровне ресурсов.

В **сети клиент/сервер** используется другой способ управления доступом — на уровне пользователей. В этом случае можно разрешить доступ к ресурсу только определенным пользователям. Например, ваш компьютер А через сеть могут использовать два пользователя: Иванов и Петров. К этому компьютеру подключен принтер, который можно использовать по сети. Но вы не хотите, чтобы кто угодно печатал на вашем принтере, и установили пароль для доступа к этому ресурсу. Если у вас **одноранговая сеть**, то любой, кто узнает этот пароль, сможет использовать ваш

принтер. В случае с сетью **клиент/сервер** вы можете разрешить использовать ваш принтер только Иванову или только Петрову (можно и обоим). Для получения доступа к ресурсу в сети клиент/сервер пользователь должен ввести свой уникальный идентификатор — имя пользователя (login — логин) и пароль. Использование логина и пароля для доступа к ресурсам называется **аутентификацией** пользователя.

**Аутентификация** — это проверка подлинности. После рассмотрения архитектуры **одноранговой** сети можно прийти к выводу, что единственное преимущество этой архитектуры — это ее простота и дешевизна. Сети **клиент/сервер** обеспечивают более высокий уровень производительности и безопасности. В отличие от одноранговой сети, в сети клиент/сервер существует один или несколько главных компьютеров — серверов. Все остальные компьютеры сети называются клиентами или рабочими станциями.

## **42. Методы доступа: CSMA/CD, маркерные методы доступа. Сети Ethernet, Token Ring и FDDI. Высокоскоростные локальные сети.**

В настоящее время наиболее распространенными методами доступа к локальной сети являются: **Множественный доступ с контролем несущей (с обнаружением коллизий (CSMA/CD)); с предотвращением коллизий (CSMA/CA); Доступ с передачей маркера; Доступ по приоритету запроса.**

**Метод CSMA/CD (множественный доступ с контролем носителя и обнаружением коллизий)** в настоящее время является одним из наиболее распространенных. Используется этот метод в архитектуре Ethernet. Все компьютеры такой сети имеют непосредственный доступ к шине для передачи и получения данных. Чтобы получить возможность передавать данные, компьютер должен убедиться, что разделяемая среда свободна от передачи других данных. Признаком незанятости среды является отсутствие на ней несущей частоты, которая равна 5-10 МГц. Если среда свободна, начинается передача кадра. Если во время передачи кадра рабочая станция обнаруживает другой сигнал, занимающий передающую среду, она останавливает передачу, посылает сигнал и ждёт в течение случайного промежутка времени, перед тем как снова отправить кадр. После задержки передача возобновляется. Реально конфликты приводят к уменьшению быстродействия сети только в том случае, если работает порядка 80 - 100 станций. **Коллизии CSMA/CD определяются** – на выходе источника. При этом методе обеспечивается повышение скорости передачи и коэффициента использования канала до 90 – 98% его пропускной способности. Конфликты могут быть обнаружены при квитировании пакетов, для чего все станции, кроме адресуемой, должны освобождать канал сразу после завершения передачи пакета. Все данные, передаваемые по сети, помещаются в кадры определенной структуры и снабжаются уникальным адресом станции назначения. Все станции, подключенные к кабелю, могут распознать факт передачи кадра, и та станция, которая узнает собственный адрес в заголовках кадра, записывает его содержимое в свой внутренний буфер, обрабатывает полученные данные, передает их вверх по своему стеку, а затем посылает по кабелю кадр-ответ. Адрес станции источника содержится в исходном кадре, поэтому станция-получатель знает, кому нужно послать ответ.

**Достоинства этого метода:** простота реализации; малое время задержки при небольшом количестве сообщений в сети; эффективность и независимость работы систем друг от друга; Возможно осуществление передачи данных между равноправными узлами, которая будет быстрее, чем в других методах. легкость включения новых станций, отсутствие управляющих кадров, разрешение конфликтов.

**Недостатки этого метода:** отсутствие гарантий на время доставки сообщений; рост конфликтных ситуаций (числа столкновений кадров) с увеличением интенсивности трафика (потока информации в канале). Сбой в передаче данных на определенный прибор будет обнаружен только тогда, когда станция запросит определенный ответ и

не получит его Конфигурация сети очень сложна. Не достигается четко определенное время отклика.

**К маркерным методам доступа относятся** два наиболее известных типа передачи данных по локальной сети: **маркерная шина** и **маркерное кольцо**. **Маркер** - управляющая последовательность бит, передаваемая компьютером по сети, в которую абоненты сети могут помещать свои информационные пакеты. Станция, имеющая данные для передачи, получив маркер, изымает его из кольца, тем самым получая право на передачу информации, заменяет его кадром данных установленного формата, содержащего следующие поля: начальный ограничитель, управление кадром, адрес назначения, адрес источника, данные, конечный ограничитель ED, статус кадра FS То есть, один из компьютеров создает специальный маркер (сообщение специального вида), который последовательно передается от одного компьютера к другому.

Название **Ethernet** (буквально «эфирная сеть» или «среда сети») отражает первоначальный принцип работы этой технологии: всё, передаваемое одним узлом, одновременно принимается всеми остальными (то есть имеется некое сходство с радиовещанием). В настоящее время практически всегда подключение происходит через коммутаторы, так что кадры, отправляемые одним узлом, доходят лишь до адресата (исключения составляют передачи на широковещательный адрес) — это повышает скорость работы и безопасность сети.

**Token Ring** — протокол передачи данных в локальной вычислительной сети (LAN) с топологией кольца и маркерным доступом. Станции в локальной вычислительной сети TR логически организованы в кольцевую топологию, с данными, передаваемыми последовательно от одной станции в кольце к другой. TR использует маркер, который так же перемещается по кольцу. Владение маркером предоставляет его обладателю право передавать данные.

**FDDI** (Волоконно-оптический распределенный интерфейс передачи данных) — стандарт передачи данных 1980-х годов для локальных сетей с расстояниями до 200 километров. Используется волоконно-оптические линии передачи, предоставляется скорость до 100 Мбит/с. Стандарт основан на протоколе Token Ring. Кроме большой территории, сеть FDDI способна поддерживать несколько тысяч пользователей.

В настоящее время в связи с увеличившимися объемами необходимой для передачи информации получили большое развитие сети с пропускной способностью свыше 100 Мбит/с. К таким сетям относится новое поколение сетей с топологией построения Ethernet – Gigabit Ethernet. Технология Gigabit Ethernet представляет собой дальнейшее развитие стандартов 802.3 для сетей Ethernet с пропускной способностью 10 и 100 Мбит/сек. Она призвана резко повысить скорость передачи данных, сохранив при этом совместимость с существующими сетями Ethernet, использующих метод случайного доступа к ЛВС.



### **43. Структурная организация систем телекоммуникаций. Каналы передачи данных: аналоговые, цифровые; разделение каналов по времени и частоте.**

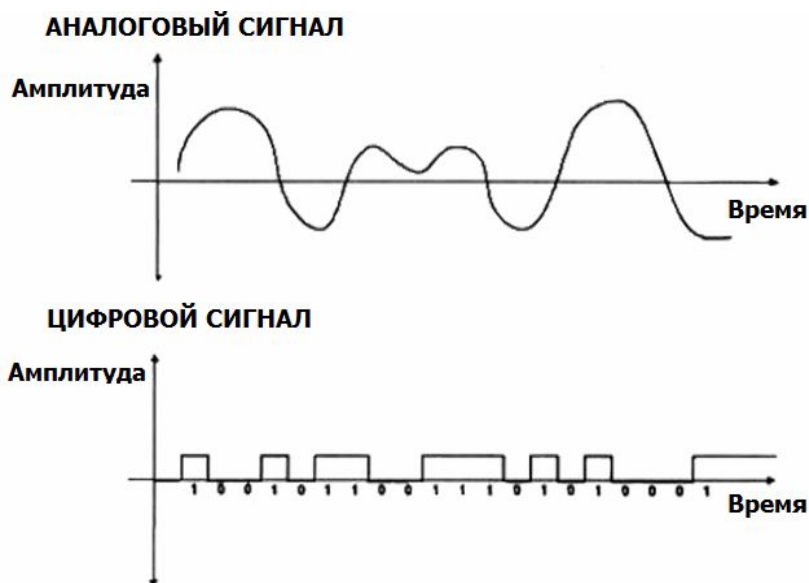
Основная функция телекоммуникационных систем (ТКС), заключается в организации оперативного и надежного обмена информацией между абонентами, а также в сокращении затрат на передачу данных. Это географически распределенная сеть, объединяющая в себе функции традиционных сетей передачи данных (СПД), телефонных сетей и предназначенная для передачи трафика различной природы, с разными вероятностно-временными характеристиками.

**Канал передачи данных** — это средства двухстороннего обмена данными, которые включают в себя линии связи и аппаратуру передачи (приема) данных. Каналы передачи данных связывают между собой источники информации и приемники информации.

В зависимости от формы информации, которую может передавать канал, различают **аналоговые** – кодирование на основе непрерывного синусоидального несущего сигнала и **цифровые** каналы - кодирование осуществляется на основе последовательности прямоугольных импульсов. Эти способы кодирования различаются шириной спектра передаваемого сигнала и сложностью аппаратуры для их реализации.

**Аналоговый канал** на входе (и, соответственно, на выходе) имеет непрерывный сигнал, те или иные характеристики которого (например, амплитуда или частота) несут передаваемую информацию.

**Цифровой канал** принимает и выдает данные в цифровой (дискретной, импульсной) форме.



**Разделение сигналов** – это обеспечение независимой передачи и приема многих сигналов по одной линии связи или в одной полосе частот, при которой, сигналы сохраняют свои свойства и не искажают друг друга. Коммутаторы должны обеспечивать использование соединяющих их каналов для одновременной передачи нескольких абонентских составных каналов. Для этого применяются разнообразные техники мультиплексирования абонентских каналов, среди которых **частотное мультиплексирование** и **мультиплексирование с разделением времени**.

**При частотном разделении** передача информации от нескольких источников сообщений по одной линии связи осуществляется одновременно на различных частотных диапазонах. Каждому каналу связи отводится определённый участок общей полосы частот, передаваемых по линии связи. Для первого канала связи выделяется полоса, для второго – и т.д. Соседние каналы связи разделяются между собой полосами непропускания, ширина которых зависит от качества фильтров, применяемых для разделения каналов.

**При временном разделении** каналов сообщения дискретизируются и передаются только их мгновенные значения, один раз за период повторения. Мгновенные значения каждого сообщения передаются короткими импульсами, поэтому по одной линии связи можно передавать последовательно во времени несколько сообщений. Для каждого канала связи выделяется определённый промежуток времени, являющийся частью периода повторения, в течение которого высылаются импульсы, модулированные информацией, передаваемой по данному каналу. Модуляция импульсов осуществляется по амплитуде, длительности или по фазе.

#### **44. Характеристики проводных линий связи; спутниковые каналы; сотовые системы связи. Способы модуляции.**

Для соединения компьютеров в локальную сеть обычно используют металлические (преимущественно медные) витые пары и коаксиальные кабели. Все более широкое распространение приобретают волоконно-оптические кабели.

##### **Витая пара**

Представляет собой пару переплетенных проводов. Для обеспечения требуемой скорости передачи данных по витой паре, она должна удовлетворять стандартам на площадь поперечного сечения провода, на количество витков на единицу длины и на расстояние от последнего витка до разъема. Существует две разновидности витой пары: экранированная (STP, Shielded Twisted Pair) и неэкранированная (UTP, Unshielded Twisted Pair). В основном используется более удобная при монтаже и дешевая неэкранированная витая пара.

##### **Коаксиальный кабель(КК)**

Состоит из двух концентрических проводников, разделенных слоем диэлектрика. Такой кабель меньше, чем витая пара, подвержен влиянию внешних электромагнитных помех. КК выпускается в нескольких вариантах, различающихся диаметром проводников. Наибольшее применение получил кабель с маркировкой RG-58 (толщина 4,95 мм, диаметр центрального проводника 0,81 мм, волновое сопротивление 50 Ом), так называемый "тонкий" коаксиальный кабель. Иногда можно встретить "толстый" (или обычный) коаксиальный кабель с маркировкой RG-8 (толщина ½ дюйма, диаметр центрального проводника 2,17 мм, волновое сопротивление 50 Ом). Сети, использующие коаксиальный кабель, обычно достигают пропускной способности 10 Мбит/с, хотя возможности такого типа кабеля гораздо выше.

Волоконно-оптический кабель передаёт данные посредством световых волн. Сердечник такого кабеля представляет собой тонкое кварцевое волокно, заключенное в пластиковую отражающую оболочку. ВОКи обладают наилучшими электромагнитными и механическими характеристиками, не подвержены влиянию электромагнитных помех, затрудняют перехват данных, но их монтаж наиболее сложен и трудоемок, требует применения специализированного дорогостоящего оборудования и квалифицированного персонала.

##### **Спутниковая связь (СС)**

Один из видов космической радиосвязи, основанный на использовании в качестве ретрансляторов искусственных спутников Земли, как правило специализированных спутников связи. СС осуществляется между так называемыми земными станциями, которые могут быть как стационарными, так и подвижными (наземными либо

установленными на летательных аппаратах). СС является развитием традиционной радиорелейной связи путём вынесения ретранслятора на очень большую высоту. Так как максимальная зона его видимости в этом случае — почти половина Земного шара, то необходимость в цепочке ретрансляторов отпадает — в большинстве случаев достаточно и одного. Особенностью СС является необходимость работать в условиях сравнительно малого отношения сигнал/шум. В связи с этим спутниковая связь плохо подходит для передачи аналоговых сигналов. Поэтому для передачи речи её предварительно оцифровывают, используя, например, импульсно-кодovou модуляцию. Для передачи цифровых данных по спутниковому каналу связи они должны быть сначала преобразованы в радиосигнал, занимающий определённый частотный диапазон. Для этого применяется модуляция (цифровая модуляция называется также манипуляцией). Наиболее распространёнными видами цифровой модуляции для приложений спутниковой связи являются фазовая манипуляция и квадратурная амплитудная модуляция. Например, в системах стандарта DVB-S2 применяются QPSK, 8-PSK, 16-APSK и 32-APSK.

### **Сотовая связь**

Один из видов мобильной радиосвязи, в основе которого лежит сотовая сеть. Ключевая особенность заключается в том, что общая зона покрытия делится на ячейки (соты), определяющиеся зонами покрытия отдельных базовых станций (БС). Соты частично перекрываются и вместе образуют сеть. На идеальной (ровной и без застройки) поверхности зона покрытия одной БС представляет собой круг, поэтому составленная из них сеть имеет вид шестиугольных ячеек (сот). Сеть составляют разнесённые в пространстве приёмопередатчики, работающие в одном и том же частотном диапазоне, и коммутирующее оборудование, позволяющее определять текущее местоположение подвижных абонентов и обеспечивать непрерывность связи при перемещении абонента из зоны действия одного приёмопередатчика в зону действия другого. Способом модуляции является частотная.

## **45. Алгоритмы сжатия данных. Программное обеспечение телекоммуникаций. Протоколы TCP/IP, управления. Адресация в Интернете.**

Сжатие данных повышает пропускную способность линии связи. Передающий модем автоматически сжимает данные, а принимающий их — восстанавливает. Модем, поддерживающий протокол сжатия, всегда пытается установить связь со сжатием данных, но если второй модем этот протокол не поддерживает, то и первый модем переходит на обычную связь без сжатия. Существует множество разных алгоритмов сжатия с потерей и без потерь данных с статистическими и адаптивными моделями. Их все можно классифицировать по определенному признаку. Каждое семейство алгоритмов сжимает текст определенной структуры с определенным коэффициентом сжатия и быстродействия.

### **Алгоритмы сжатия без потерь:**

- Преобразование Барроуза - Уилера - предварительная обработка данных для улучшения сжатия без потерь.
- Преобразование Шиндлера - модификация преобразования Барроуза – Уилера
- Алгоритм DEFLATE — популярный свободный алгоритм сжатия (используется в библиотеке zlib)
- Дельта-кодирование — эффективно для сжатия данных, в которых последовательности часто повторяются
- Инкрементное кодирование — дельта-кодирование, применяемое к последовательности строк
- Алгоритм сжатия RPPM
- Кодирование длин серий - последовательная серия одинаковых элементов заменяется на два символа: элемент и число его повторений
- Алгоритм SEQUITUR — сжатие без потерь, автоматическое адаптивное построение контекстно-свободной грамматики для обрабатываемых данных
- Вейвлет-кодирование на основе вложенных нуль-деревьев.
- Энтропийное кодирование — схема кодирования, которая присваивает коды символам таким образом, чтобы соотносить длину кодов с вероятностью появления символов
- Алгоритм Шеннона — Фано — самый простой алгоритм кодирования
- Алгоритм Хаффмана — алгоритм построения кода при помощи кодовых деревьев
- Адаптивное кодирование Хаффмана — техника адаптивного кодирования, основывающаяся на коде Хаффмана
- Усечённое двоичное кодирование — используется для однородного вероятностного распределения с конечным алфавитом
- Арифметическое кодирование — развитие энтропийного кодирования

- Адаптивное арифметическое кодирование — техника адаптивного кодирования, основывающаяся на арифметическом кодировании
- Кодирование расстояний — метод сжатия данных, который близок по эффективности к арифметическому кодированию
- Энтропийное кодирование с известными характеристиками
- Унарное кодирование — код, который представляет число  $n$  в виде  $n$  единиц с замыкающим нулём
- Дельта/гамма/омега-кодирование Элиаса — универсальный код, кодирующий положительные целые числа
- Кодирование Фибоначчи — универсальный код, который кодирует положительные целые числа в двоичные кодовые слова
- Кодирование Голомба — форма энтропийного кодирования, которая оптимальна для алфавитов с геометрическим распределением
- Кодирование Райса — форма энтропийного кодирования, которая оптимальна для алфавитов с геометрическим распределением

#### **Алгоритмы сжатия с потерями:**

- Линейное предсказывающее кодирование — сжатие с потерями, представляющее спектральную огибающую цифрового сигнала речи в сжатом виде
- А-закон — стандартный алгоритм компандирования. Применяется в РФ.
- Мю-закон — стандартный алгоритм компандирования
- Фрактальное сжатие — метод, использующий фракталы для сжатия изображений
- Трансформирующее кодирование — тип сжатия данных для «естественных» данных, таких как аудиосигналы или фотографические изображения
- Векторное квантование — техника, часто используемая в сжатии данных

Программное обеспечение станций, работающих в сети телекоммуникаций, имеет следующие характерные признаки. Оно создается большим коллективом специалистов (свыше 100 человек); имеет большой объем (более 100 000 строк программного текста); обновляется и дополняется в течение всего "жизненного цикла" системы.

**TCP/IP** - сетевая модель передачи данных, представленных в цифровом виде. Модель описывает способ передачи данных от источника информации к получателю. В модели предполагается прохождение информации через четыре уровня, каждый из которых описывается правилом (протоколом передачи). Наборы правил, решающих задачу по передаче данных, составляют стек протоколов передачи данных. Название TCP/IP происходит из двух важнейших протоколов семейства — Transmission Control Protocol (TCP) и Internet Protocol (IP), которые были первыми разработаны и описаны в данном стандарте. Набор интернет-протоколов — это модель и набор коммуникационных протоколов, используемых в Интернете и подобных компьютерных сетях. Он широко

известен как TCP/IP, поскольку базовые протоколы в пакете — это протокол управления передачей (TCP) и интернет-протокол (IP). Его иногда называют моделью Министерства обороны (МО), поскольку разработка сетевого метода финансировалась Министерством обороны Соединенных Штатов через DARPA. Набор интернет-протоколов обеспечивает сквозную передачу данных, определяющую, как данные должны пакетироваться, обрабатываться, передаваться, маршрутизироваться и приниматься. Эта функциональность организована в четыре слоя абстракции, которые классифицируют все связанные протоколы в соответствии с объемом задействованных сетей. От самого низкого до самого высокого уровня — это уровень связи, содержащий методы связи для данных, которые остаются в пределах одного сегмента сети (ссылка); интернет-уровень, обеспечивающий межсетевое взаимодействие между независимыми сетями; транспортный уровень, обрабатывающий связь между хостами; и прикладной уровень, который обеспечивает обмен данными между процессами для приложений.

Для отдельного компьютера или локальной сети, которые впервые подключаются к сети Интернет, специальная организация, занимающейся администрированием доменных имен, присваивает IP – номера. Первоначально в сети Internet применялись IP – номера, но когда количество компьютеров в сети стало больше чем 1000, то был принят метод связи имен и IP – номеров, который называется сервер имени домена (Domain Name Server, DNS). Сервер DNS поддерживает список имен локальных сетей и компьютеров и соответствующих им IP – номеров. В Интернете применяется так называемая доменная система имен. Каждый уровень в такой системе называется доменом. Типичное имя домена состоит из нескольких частей, расположенных в определенном порядке и разделенных точками. В Интернете доменная система имен использует принцип последовательных уточнений также как и в обычных почтовых адресах - страна, город, улица и дом, в который следует доставить письмо. Домен верхнего уровня располагается в имени правее, а домен нижнего уровня - левее. В нашем примере домены верхнего уровня info и ua указывают на то, что речь идет о принадлежности сайта [www.lessons-tva.info](http://www.lessons-tva.info) к тематическому домену верхнего уровня info, а сайта [uuu.zzz.ua](http://uuu.zzz.ua) к украинской (ua) части Интернета. Но в Украине множество пользователей Интернета, и следующий уровень определяет организацию, которой принадлежит данный адрес. В нашем случае это компания zzz. Интернет-адрес этой компании - [zzz.ua](http://zzz.ua). Все компьютеры, подключенные к Интернету в этой компании, объединяются в группу, имеющую такой адрес. Имя отдельного компьютера или сети каждая компания выбирает для себя самостоятельно, а затем регистрирует его в той организации Интернет, которая обеспечивает подключение. Это имя в пределах домена верхнего уровня должно быть уникальным. Далее следует имя хоста uuu, таким образом, полное имя домена третьего уровня: [uuu.zzz.ua](http://uuu.zzz.ua). В имени может быть любое число доменов, но чаще всего используются имена с количеством доменов от трех до пяти. Доменная система образования адресов гарантирует, что во всем Интернете больше не найдется другого компьютера с таким же адресом. Для доменов нижних уровней можно использовать любые адреса, но для доменов самого верхнего уровня

существует соглашение. В системе адресов Интернета приняты домены, представленные географическими регионами. Они имеют имя, состоящее из двух букв (Украина - ua; Франция - fr; Канада - ca; США - us; Россия – ru). Существуют и домены, разделенные по тематическим признакам (Учебные заведения - edu. Коммерческие организации – com)

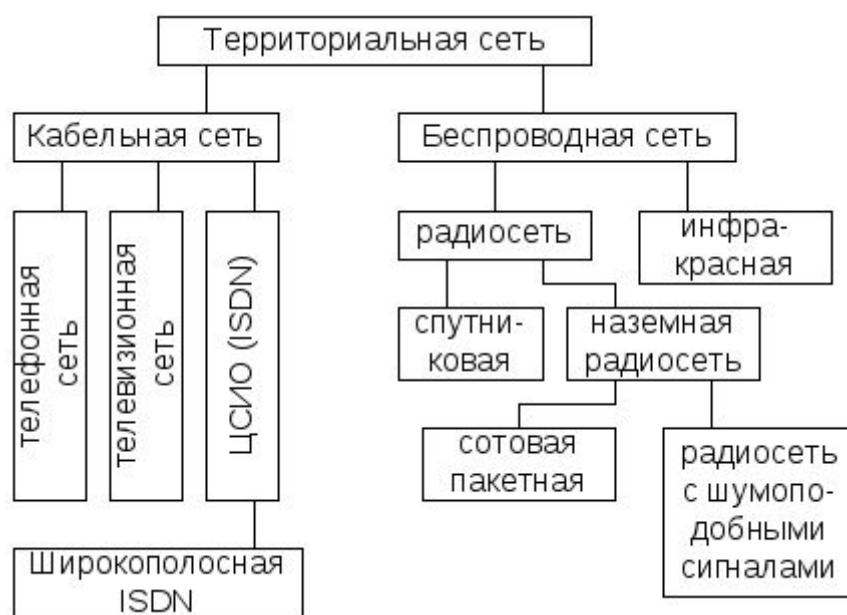
При работе в Internet используются не доменные имена, а универсальные указатели ресурсов, называемые URL (Universal Resource Locator). URL - это адрес любого ресурса (документа, файла) в Internet, он указывает, с помощью какого протокола следует к нему обращаться, какую программу следует запустить на сервере и к какому конкретному файлу следует обратиться на сервере. Общий вид URL: протокол://хост-компьютер/имя файла.



#### 46. Информационные услуги территориальных сетей. Технологии распределенных вычислений. Протоколы файлового обмена, электронной почты.

**Территориальная сеть** – это сеть, системы которой расположены в различных географических точках. Она охватывает большое пространство (от района до группы стран). В случае, если она охватывает континенты, то используется название **глобальной сети**. Характерной особенностью является применение протяжённых широкополосных каналов, большого числа узлов коммутации или спутников связи.

##### Классификация территориальных сетей:



Она должна удовлетворять следующим основным требованиям:

- Включать большое число абонентских систем (до нескольких тысяч);
- Покрывать большой географический район;
- Обеспечивать ширококовещание и доставку сообщений группам и отдельным адресатам;
- Иметь высокую пропускную способность (до десятков Гбит/с);
- Обладать большой надёжностью в работе;
- Гарантировать безопасность данных;
- Передавать разнообразные виды данных: тексты, звук, изображения.

**Распределенные вычисления** – применение доступных разнородных компьютерных ресурсов, соединенных между собой каналами связи общего назначения, для решения вычислительно сложных задач. Особенностью распределённых многопроцессорных ВС, в отличие от локальных суперкомпьютеров, является возможность неограниченного наращивания производительности за счёт масштабирования.

**Файловый обмен** - доступ к файлам, распределенным по различным компьютерам. В сети Internet на прикладном уровне используется протокол FTP. Доступ возможен в режимах off-line и on-line. В режиме off-line посылается запрос к FTP-серверу, сервер формирует и посылает ответ на запрос. В режиме on-line осуществляется интерактивный просмотр каталогов FTP-сервера, выбор и передача нужных файлов. На ЭВМ пользователя нужен FTP-клиент. Доступ к базам данных удаленных ЭВМ с эмуляцией удаленной машины на ЭВМ пользователя выполняется с помощью протокола Telnet. Протокол файлового обмена FTP. FTP довольно необычная процедура, так как поддерживает две логические связи между ЭВМ. Одна связь служит для удаленного доступа и использует протокол Telnet.

**Электронная почта** - наиболее популярный и быстроразвивающийся вид общения. Широко используются протоколы электронной почты UUCP (unix-to-unix copy protocol, RFC-976) и SMTP (simple mail transfer protocol; RFC-821, -822, -1351, -1352). Имея механизмы промежуточного хранения почты (spooling) и механизмы повышения надежности доставки, протокол smtp базируется на TCP-протоколе в качестве транспортного и допускает использование различных транспортных сред. Он может доставлять сообщения даже в сети, не поддерживающие протоколы TCP/IP. Главной целью протокола simple mail transfer protocol (SMTP, RFC-821, -822) служит надежная и эффективная доставка электронных почтовых сообщений. SMTP является довольно независимой подсистемой и требует только надежного канала связи. Средой для SMTP может служить отдельная локальная сеть, система сетей или весь Интернет.