



Министерство образования Республики Беларусь

Учреждение образования
«Гомельский государственный технический
университет имени П. О. Сухого»

Институт повышения квалификации
и переподготовки кадров

Кафедра «Информатика»

А. В. Ковалев

ТЕХНИЧЕСКИЕ СРЕДСТВА ИНФОРМАЦИОННЫХ СИСТЕМ

**КУРС ЛЕКЦИЙ
по одноименной дисциплине
для слушателей специальности
1-40 01 73 «Программное обеспечение
информационных систем»
заочной формы обучения**

Гомель 2015

УДК 004.2(075.8)
ББК 32.973я73
К56

*Рекомендовано научно-методическим советом
факультета автоматизированных и информационных систем
ГГТУ им. П. О. Сухого
(протокол № 5 от 29.12.2014 г.)*

Рецензенты: зав. каф. «Информационные технологии» ГГТУ им. П. О. Сухого
канд. физ.-мат. наук, доц. К. С. Курочка

Ковалев, А. В.

К56 Технические средства информационных систем : курс лекций по одноим. дисциплине для слушателей специальности 1-40 01 73 «Программное обеспечение информационных систем» заоч. формы обучения / А. В. Ковалев. – Гомель : ГГТУ им. П. О. Сухого, 2015. – 145 с. – Систем. требования: PC не ниже Intel Celeron 300 МГц ; 32 Mb RAM ; свободное место на HDD 16 Mb ; Windows 98 и выше ; Adobe Acrobat Reader. – Режим доступа: <https://elib.gstu.by>. – Загл. с титул. экрана.

Курс лекций предназначен для изучения основных элементов технического обеспечения информационных систем: поколений и тенденций развития электронной вычислительной техники, особенностей построения вычислительных систем, периферийных устройств, систем защиты информации. Рассмотрены особенности применения элементов технических средств в зависимости от сферы использования в информационных системах.

Для слушателей специальности 1-40 01 73 «Программное обеспечение информационных систем» заочной формы обучения ИПК и ПК.

**УДК 004.2(075.8)
ББК 33.973я73**

© Учреждение образования «Гомельский
государственный технический университет
имени П. О. Сухого», 2015

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	6
1. ТЕХНИКО-ЭКСПЛУАТАЦИОННЫЕ ХАРАКТЕРИСТИКИ ЭВМ	13
2. КЛАССИФИКАЦИЯ ЭВМ	18
2.1. Классификация ЭВМ по назначению	18
2.2. Классификация ЭВМ по функциональным возможностям	19
3. ФУНКЦИОНАЛЬНАЯ И СТРУКТУРНАЯ ОРГАНИЗАЦИЯ ЭВМ	23
3.1. Связь между функциональной и структурной организацией ЭВМ	23
3.2. Обобщенная структура ЭВМ и пути её развития	24
3.3. Обработка подсистема	25
3.4. Подсистема памяти	27
3.5. Подсистема ввода-вывода	27
3.6. Подсистема управления и обслуживания	28
4. МНОГОУРОВНЕВАЯ КОМПЬЮТЕРНАЯ ОРГАНИЗАЦИЯ	28
5. АРХИТЕКТУРЫ ЭВМ	35
5.1. SISD-компьютеры	36
5.2. Компьютеры с CISC архитектурой	37
5.3. Компьютеры с RISC архитектурой	38
5.4. Компьютеры с суперскалярной обработкой	38
5.5. SIMD-компьютеры	39
5.6. Матричная архитектура	41
5.7. Векторно-конвейерная архитектура	41
5.8. MISD компьютеры	42
5.9. MIMD компьютеры	43
5.10. Многопроцессорные вычислительные системы	44
5.11. Многопроцессорные вычислительные системы с общей шиной	45
5.12. Многопроцессорные вычислительные системы с многоходовыми модулями ОП	46
5.13. Многомашинные вычислительные системы (ММВС)	47

5.14. ММР архитектура	49
6. СТРУКТУРА И ФОРМАТЫ КОМАНД ЭВМ	50
6.1. Форматы команд ЭВМ	51
1.1. Способы адресации	53
1.2. Классификация способов адресации по наличию адресной информации в команде	55
1.3. Классификация способов адресации по кратности обращения в память	56
1.4. Классификация по способу формирования исполнительных адресов ячеек памяти	58
2. ТИПЫ ДАННЫХ	61
3. ПРОЦЕССОРЫ	67
3.1. Логическая структура ЦП	68
3.2. Структурная схема процессора	70
4. РЕГИСТРОВЫЕ СТРУКТУРЫ ЦЕНТРАЛЬНОГО ПРОЦЕССОРА	73
4.1. Основные функциональные регистры	73
4.2. Регистры процессора обработки чисел с плавающей точкой	75
5. ОБЩИЕ ПРИНЦИПЫ ОРГАНИЗАЦИИ СИСТЕМЫ ПРЕРЫВАНИЯ ПРОГРАММ	76
6. МАТЕРИНСКАЯ ПЛАТА И ЕЕ ОСНОВНЫЕ ИНТЕРФЕЙСЫ	83
6.1. Основные функции Северного моста	86
6.2. Интерфейсы связи с южным мостом	93
6.3. Основные функции Южного моста	94
6.4. Интерфейсы связи с периферийными устройствами, устройствами ввода - вывода и другими ЭВМ	95
6.5. Интерфейсы шин связи южного моста с жесткими дисками.	102
6.6. Интерфейсы связи с медленными компонентами материнской платы	106
6.7. BIOS (Basic Input-Output System)	107

7. УСТРОЙСТВА ВВОДА/ВЫВОДА ИНФОРМАЦИИ В ИНФОРМАЦИОННЫХ СИСТЕМАХ	108
7.1. Устройства ввода	108
7.2. Устройства вывода	112
7.2.1. Мониторы и проекторы	112
7.2.2. Принтеры	115
8. ОСОБЕННОСТИ ПОСТРОЕНИЯ СЕТЕЙ	126
9. ВЫЧИСЛИТЕЛЬНЫЕ СИСТЕМЫ	137
10. СИСТЕМЫ ЗАЩИТЫ ИНФОРМАЦИИ	140
Список литературы	145

ВВЕДЕНИЕ

Орудия и средства (инструменты) производства являются необходимым компонентом любой технологии. Не являются исключением и информационные технологии, инструментальную базу которых образуют технические, программные и лингвистические средства.

Инструментальные средства информационных технологий – совокупность технических программных и лингвистических средств, обеспечивающих реализацию информационных процессов.

В составе технического обеспечения информационных технологий (с некоторой долей условности) различают следующие группы средств:

- компьютерная техника (ЭВМ и периферийные устройства), обеспечивающая электронное представление информации и автоматизацию всех информационных процессов;
- телекоммуникационные средства и системы, обеспечивающие передачу информации на расстояние;
- полиграфическая, копировальная и множительная техника, предназначенная для копирования и тиражирования информации; средства записи и воспроизведения аудиовизуальной информации (фото-, теле- видео-, киноизображения и звука);
- оргтехника (офисная техника), предназначенная для механизации и автоматизации конторского труда и управленческой деятельности.

Условность подобной классификации связана с нарушением единства основания и принципа непересекаемости делений: одни и те же средства (например, компьютерные) представлены во всех пяти группах; а копировально-множительная техника и средства связи широко используются в офисе.

Имеет смысл классифицировать технические средства в разрезе информационных процессов, для реализации которых они предназначены.

1. Средства сбора (регистрации) и ввода (записи) информации:

- персональные компьютеры – средства ввода текстовой, табличной, графической, аудиовизуальной и иной информации и записи ее на машиночитаемые носители;
- сканеры – средства оптического ввода – автоматического считывания текста или изображения на бумажном носителе с последую-

щим преобразованием его в формат, доступный для обработки и хранения в ЭВМ;

- дигитайтеры – средства бесклавиатурного ввода текста и графических изображений в ЭВМ;

- пишущие машины (механические, электрические, электронные) – средства изготовления бумажных (тестовых и табличных) документов;

- оргавтоматы – комплекс электромеханических и электронных средств автоматизации процесса составления, редактирования и изготовления текстовых и табличных документов;

- диктофоны – средства записи звуковой (преимущественно речевой) информации на различные носители (плёночные, магнитные, оптические) часто с целью преобразования ее в текстовую информацию;

 - магнитофоны – средства записи аудиальной информации;

 - фото-, кино-, теле-, видеокамеры – средства записи статических и движущихся изображений и аудиовизуальной информации;

 - измерительная техника (датчики, приборы, установки) – средства фиксации и измерения сигнала, извещающего о наступлении контролируемых событий и др.

2. Средства семантической и технической обработки информации:

- компьютеры (микрокомпьютеры, персональные, портативные, карманные, большие, сверхбольшие) – средства автоматизированной обработки цифровой информации;

- монтажное оборудование – средства обработки (монтажа) аудиальной, визуальной, аудиовизуальной, мультимедийной информации (цифровые и аналоговые устройства монтажа звука и изображения, монтажные столы);

- средства репрографии и оперативной полиграфии – оборудование для копирования и тиражирования документов (средства фотокопирования, диазокопирования, электрофотографии, термографии, электронно-искрового копирования, ризографического копирования, микрофильмирования; оборудование для гектографической, трафаретной, офсетной печати);

- средства технической обработки носителей информации (фальцевальные, перфорирующие и резательные машины, машины для уничтожения бумаг и др.);

- средства технической обработки документов (скрепляющее, склеивающее и переплетное оборудование, машины для нанесения защитных покрытий на документы);

- средства технической обработки корреспонденции (конвертовскрывающие, адресовальные, штемпелевальные, маркировальные машины и устройства, машины для уничтожения бумаг и т. п.) и др.

3. Средства хранения информации:

- компьютеры – средства хранения электронных документов и данных (серверы БД, файловые серверы, серверы приложений и др., локальные компьютеры);

- носители информации (бумажные, пленочные, магнитные, оптические, голографические, микроносители, перфоносители);

- канцелярские средства хранения документов (мультифоры, папки, планшеты, контейнеры и др.);

- картотеки (плоские, вертикальные, элеваторные, вращающиеся и др.) и картотечное оборудование;

- офисная мебель (шкафы, столы, стеллажи, сейфы и др.).

4. Средства поиска информации:

- автоматизированные ИПС (электронные каталоги, банки данных, электронные библиотеки, Web-ресурсы Интернет и др.);

- механизированные ИПС – ИПС, основанные на использовании перфо- и микроносителей информации, осуществляющие поиск методом механической сортировки записей и кодов специальными устройствами (счетно-перфорационные машины, считывающие устройства, селекторы);

- ручные ИПС (карточные каталоги и картотеки, справочно-поисковый аппарат печатных изданий и др.).

5. Средства передачи информации:

- локальные, региональные, глобальные, корпоративные вычислительные сети – средства электронной связи, передачи на расстояние компьютерной информации;

- средства (аппаратура) электрической, радио-, телевизионной связи (телефонные, телеграфные, факсимильные аппараты, радио, телевизионные передатчики и приемники и др.).

- каналы связи – средства передачи акустических, оптических и электрических сигналов – делятся на беспроводные (радиосвязь, спутниковая связь) и проводные (кабельная связь: коаксиальный кабель, незащищенная витая пара, защищенная витая пара, оптоволоконный кабель);

- транспортные средства – средства механической доставки документов (тележки для перевозки документов внутри помещений, лифтовое оборудование, транспортеры, конвейеры, пневматическая почта, автомобильный и иной транспорт и др.).

6. Средства вывода информации: видеомониторы, мультимедийные проекторы, плазменные панели – средства отображения электронной информации;

- принтеры (матричные, струйные, лазерные) – печатающие устройства, обеспечивающие перенос машиночитаемой текстовой, числовой и графической информации на бумажный носитель;

- плоттеры (графопостроители) – устройства, обеспечивающие перенос машиночитаемой графической информации на бумажный носитель;

- аудиотехника – средства вывода звуковой информации (радиоприемники, проигрыватели, магнитофоны, аудиоплееры, музыкальные центры и др.);

- видеотехника – средства вывода аудиовизуальной информации (телевизоры, домашние кинотеатры, кинопроекторная аппаратура, видеосистемы, DVD-плееры и др.).

Оценивая состояние и тенденции развития технической базы информационных технологий, специалисты отмечают:

1) приоритетное внимание разработчиков и рост спроса на цифровые устройства в сравнении с аналоговыми (так, в ведущих странах мира рост числа домашних компьютеров превышает рост числа телевизоров);

2) число компьютеров в личном пользовании становится сопоставимым с числом машин, используемых на предприятиях и в организациях;

3) динамичное развитие сетей спутникового и кабельного телевидения и информационных каналов связи;

4) опережающее развитие системы компьютерных телекоммуникаций, мобильной телефонной связи в сравнении с другими способами дистанционной передачи информации.

Этапами развития информационной системы являются: формирование концепции, разработка технического предложения (задания) на ИС, проектирование ИС, изготовление ИС, ввод системы в эксплуатацию. Иногда второй и частично третий этапы называют этапами системного проектирования, последние два, а иногда и три, включая проектирование, – этапами реализации ИС

Моделью жизненного цикла ИС называют некоторую структуру, определяющую последовательность процессов, действий и задач, которые реализуются на протяжении ее жизненного цикла, а также взаимосвязи между этими процессами, действиями и задачами.

Спиральная модель разработки ИС в отличие от каскадной основана на итерационном процессе разработки информационной системы. При этом возрастает значимость начальных этапов жизненного цикла – анализа и проектирования, на которых осуществляются проверка и обоснованность оригинальности, жизнеспособности и реализуемости выбранных решений, а также создание прототипов ИС.

Основой понятия «информационный процесс» является термин «процесс», который означает, что что-то происходит, совершается, т. е. изменяется с течением времени. В природе нескончаемо что-нибудь происходит. Естественным изменениям, т. е. таким процессам, как старение, выветривание, эрозия, подвержены даже такие объекты, которые нам кажутся очень стабильными, неизменными, например, скалы и горы. То же самое относится и к процессу существования живого существа.

С точки зрения информационной системы в целом, система восприятия осуществляет первичную обработку, собираемой извне информации. В свою очередь, для системы восприятия первичную обработку информации производит система сбора информации. Нередко на практике встречаются информационные системы, не обладающие развитой системой восприятия информации. В последнем случае система восприятия представляет собой просто систему сбора информации.

Обработка информации – процессы преобразования формы и (или) содержания документов или данных. Цель этих преобразований – изменение состояния информации, придание ей новых свойств. Различают техническую и семантическую обработку информации.

Хранение информации – процесс обеспечения сохранности документов или данных, удовлетворительного состояния материального носителя, защиты от несанкционированного доступа и недозволенного использования.

Одним из основных элементов технических средств является ЭВМ, которая фактически воспроизводит процесс ручного механического счета и управления объектами. Рассмотрим основные термины.

ЭВМ – это совокупность технических и программных средств, предназначенных для автоматизированной обработки информации (дискретных сообщений) по требуемому алгоритму.

Алгоритм - набор правил, позволяющий решить любую конкретную задачу из определенного класса. С помощью алгоритма задают последовательность действий, которые надо совершить для получения искомого решения. Алгоритм решения задачи вводится в ЭВМ в виде программы, написанной на каком-либо языке программирования, и хранится в памяти ЭВМ в форме машинных кодов команд, осуществляющих управление ходом вычислительного процесса.

Памятью ЭВМ называется совокупность устройств, служащих для запоминания, хранения и выдачи информации.

Языки программирования – это формальные языки, служащие средством общения между человеком и ЭВМ. Языки программирования низкого уровня максимально приближены к машинным кодам. Характерной особенностью так называемых языков программирования высокого уровня является совпадение результата действия оператора с общепринятым смысловым или бытовым значением команды.

Любая ЭВМ содержит следующие основные подсистемы: обрабатывающую подсистему (операционные устройства и устройства управления); подсистему памяти; подсистему ввода-вывода; подсистему управления и обслуживания.

Перед началом решения задачи в *подсистему памяти* через *подсистему ввода/вывода* записываются кодированные данные и программа. В ходе вычислительного процесса по управляющим сигналам *устройств управления* при выполнении очередной команды программы считывается код команды, по нему определяется местонахождение данных, после чего они извлекаются из соответствующего запоминающего устройства. *Обрабатывающая система* выполняет над ними указанную в коде команды операцию и записывает результат на хранение в запоминающее устройство. Затем определяется местонахождение кода следующей команды и цикл повторяется.

В памяти ЭВМ хранятся также вспомогательные и служебные программы, составляющие программное обеспечение (ПО) ЭВМ

Операционные устройства (АЛУ) выполняют арифметические и логические операции над поступающими двоичными кодами (команд и данных), причем любой процессор в состоянии выполнить ограниченный набор команд, входящий в так называемую *систему команд ЭВМ*.

Устройство управления под воздействием поступающих данных автоматически координирует работу всех устройств ЭВМ посредством своевременной выдачи на них управляющих сигналов.

В составе ЭВМ выделяется *центральная обрабатывающая часть (ядро ЭВМ)* и *периферия* (все подсистемы ввода/вывода и внешняя память).

Периферийное оборудование подключается к центральному ядру через каналы ввода/вывода (КВВ) и шинные интерфейсы. КВВ представляют собой специализированные процессоры, управляющие обменом между устройствами ввода/вывода и выделенной областью памяти. Среди КВВ выделилось специализированное так называемое интерфейсное оборудование, обеспечивающее согласование периферии с центральным ядром. В результате ЭВМ приобрела иерархическую структуру, в которой центральная обрабатывающая часть, КВВ, интерфейсы и периферия находятся на разных уровнях иерархии.

Таким образом, к общим принципам построения современных ЭВМ относится следующее:

- Обеспечение максимального удобства в работе пользователей и эффективной эксплуатации оборудования. Это обеспечивается за счет введения в ЭВМ подсистемы управления и обслуживания;

- Возможность мультипрограммной работы. Для мультипрограммной работы ЭВМ оснащена: (средствами распределения памяти между программами, системой защиты памяти, системой прерываний и приоритетов, датчиком времени (таймером). В ОЗУ таких ЭВМ хранится одновременно несколько параллельно выполняемых программ, и имеются средства защиты этих программ и приписанных к ним массивов от взаимного влияния. ОС расширена дополнительными управляющими программами, обеспечивающими различные режимы мультипрограммной работы (пакетный, разделения во времени, интерактивный и т. д.), изменения числа задач, приоритетов, возможность работы с машинными словами разной длины, операции с числами в двоично-десятичном коде и т.п.;

- Иерархическая организация структуры ЭВМ, ее памяти, ПО и управления вычислительным процессом;

- Модульный принцип построения ЭВМ (из логически законченных блоков) позволяет наращивать вычислительную мощность и по мере необходимости подключать дополнительные устройства или устанавливать более совершенные, благодаря чему облегчается адаптация ЭВМ в зависимости от требований пользователя;

- Возможность адаптации, развития, модернизации и наращивания технических средств.

В наибольшей степени эти принципы используются в ЭВМ общего назначения, чья архитектура характеризуется:

- Универсальностью - обеспечивает возможность одинаково эффективного решения задач различных классов;

- Совместимостью: совместимость различных ЭВМ достигается за счет соответствующего выбора аппаратных средств, ОС, алгоритмических языков, пользовательских средств:

- *информационная совместимость* предполагает использование общих кодов и единых форматов представления данных, совпадение характеристик носителей информации и шин;

- *программная совместимость* чаще всего носит иерархический характер (снизу вверх – от младших моделей к старшим) и подразумевает единство логической структуры: систем адресации, команд, способов подключения периферийных устройств.;

- Развитостью программного обеспечения;

- Агрегатностью технических средств (при этом вся система разбивается на большое число конструктивных модулей, каждый из которых выполняет собственные функции);

- Широкой номенклатурой периферийных устройств;

- Высокой технологичностью, унификацией и стандартизацией оборудования.

1. ТЕХНИКО-ЭКСПЛУАТАЦИОННЫЕ ХАРАКТЕРИСТИКИ ЭВМ

Основным техническим параметром ЭВМ является ее быстродействие. **Быстродействие ЭВМ** - среднестатистическое число операций (кроме операций ввода, вывода и обращения к внешним запоминающим устройствам), выполняемых вычислительной машиной в единицу времени (номинальное быстродействие); один из основных параметров ЭВМ, характеризующий её производительность.

Различают следующие виды быстродействия:

- **пиковое** (предельное) - это быстродействие процессора без учета времени обращения к оперативной памяти (ОП) за операндами;

- **номинальное** - быстродействие процессора с ОП;

-системное - быстродействие базовых технических и программных средств, входящих в комплект поставки ЭВМ;

Методы определения быстродействия разделяются на три основных группы:

-расчетные, основанные на информации, получаемой теоретическим или эмпирическим путем;

-экспериментальные, основанные на информации, получаемой с использованием аппаратно-программных измерительных средств;

-имитационные, применяемые для сложных ЭВМ.

Для каждого вида быстродействия применяются следующие традиционные методы их определения.

Пиковая производительность (быстродействие) определяется средним числом команд типа «регистр-регистр», выполняемых в одну секунду без учета их статистического веса в выбранном классе задач.

Номинальная производительность (быстродействие) определяется средним числом команд, выполняемых подсистемой «процессор-память» с учетом их статистического веса в выбранном классе задач. Она рассчитывается, как правило, по формулам и специальным методикам, предложенным для процессоров определенных архитектур, и измеряется с помощью разработанных для них измерительных программ, реализующих соответствующую эталонную нагрузку.

Для данных типов производительностей используются следующие единицы измерения:

MIPS (Mega Instruction Per Second) - миллион целочисленных операций в секунду;

MFLOPS (Mega Floating Operations Per Second) - миллион операций над числами с плавающей запятой в секунду;

GFLOPS (Giga Floating Operations Per Second) - миллиард операций над числами с плавающей запятой в секунду;

TFLOPS (Tera Floating Operations Per Second) - триллион операций над числами с плавающей запятой в секунду.

Системная производительность измеряется с помощью синтезированных типовых (тестовых) оценочных программ, реализованных на унифицированных языках высокого уровня. Унифицированные тестовые программы используют типичные алгоритмические действия, характерные для реальных применений, и штатные компиляторы ЭВМ. Они рассчитаны на использование базовых технических средств и позволяют измерять производительность для расширенных конфигураций технических средств. Результаты оценки системной

производительности ЭВМ конкретной архитектуры приводятся относительно базового образца, в качестве которого используются ЭВМ, являющиеся промышленными стандартами систем ЭВМ различной архитектуры. Результаты оформляются в виде сравнительных таблиц, двумерных графиков и трехмерных изображений.

Очень часто при сравнении компьютеров пользуются отношением производительности к стоимости.

Другая не менее важная техническая характеристика ЭВМ - ёмкость памяти - определяется максимальным количеством данных, которые могут в ней храниться. Ёмкость измеряется в двоичных единицах (битах), машинных словах, но большей частью в байтах. Часто ёмкость памяти выражают через число $K = 1024$. Т.о. для измерения ёмкости различных типов запоминающих устройств используют следующие обозначения:

1 байт = 8 бит,

1 Кбит (килобит) = 1024 бит,

1 Кбайт (килобайт) = 1024 байт,

1 Мбайт (Мегабайт) = 1024 Кбайт,

1 Гбайт (гигабайт) = 1024 Мбайт,

1 Тбайт (терабайт) = 1024 Гбайт.

К другим технико-эксплуатационным характеристикам ЭВМ относятся:

- разрядность обрабатываемых слов и кодовых шин интерфейса;
- типы системного и локальных интерфейсов;
- тип и ёмкость оперативной памяти;
- тип и ёмкость накопителя на жестком магнитном диске;
- тип и ёмкость накопителя на гибком магнитном диске;
- тип и ёмкость кэш-памяти;
- тип видеоадаптера и видеомонитора;
- наличие средств работы в компьютерной сети;
- наличие и тип программного обеспечения;
- надёжность ЭВМ (свойство ЭВМ при заданных условиях выполнять требуемые функции в течение заданного периода времени);
- стоимость;
- габариты и масса.

История развития ЭВМ условно делится на отрезки, называемые поколениями ЭВМ. ЭВМ, принадлежащие к различным поколениям, отличаются элементной базой, функционально-логической организа-

цией, конструктивно-технологическим исполнением, программным обеспечением, техническими и эксплуатационными характеристиками, степенью доступа к ЭВМ со стороны пользователя.

Смене поколений ЭВМ сопутствуют изменения технических характеристик:

- быстродействия;
- емкости памяти;
- надежности;
- стоимости.

Основная тенденция развития – стремление уменьшить трудоемкость подготовки программ, облегчить связь операторов и машин, повысить эффективность использования ЭВМ.

Первое поколение (1949-1958)

Основным активным элементом ЭВМ первого поколения является *электронная лампа*. (Остальные элементы: резисторы, конденсаторы, трансформаторы).

Для построения оперативной памяти применялись ферритовые сердечники. В качестве устройств ввода/вывода (УВВ) сначала использовалась стандартная телеграфная аппаратура, а затем специально для ЭВМ были разработаны электромеханические УВВ на перфокартах и перфолентах.

Машины этого поколения характеризуются:

- огромными размерами;
- малым быстродействием;
- малой емкостью оперативной памяти (ОП);
- невысокой надежностью;
- недостаточно развитым программным обеспечением (ПО).

Второе поколение (1959-1963)

Основной активный элемент ЭВМ второго поколения – *транзистор*.

Все показатели улучшены по сравнению с I поколением: уменьшены размеры, стоимость, масса и потребляемая мощность, повышена надежность и быстродействие, увеличен объем памяти.

Отличительные черты поколения:

- применение печатного монтажа;
- дифференциация по применению (специализация);
- в программном обеспечении (ПО) – появление алгоритмических языков;

- появление многопрограммных ЭВМ (совместная реализация программ за счет организации параллельной работы основных устройств ЭВМ);

- применение УВВ на магнитных носителях (магнитные ленты, барабаны, диски).

Третье поколение (1964-1976)

Характеризуется широким применением *интегральных схем* (ИС) с многослойным печатным монтажом. ИС (кристалл) - это законченный функциональный блок, соответствующий сложной транзисторной схеме, вытравленной на поверхности кремниевого кристалла. Позднее стали применяться ИС малой (10-100 компонентов на кристалл) и средней (100-1000 компонентов на кристалл) степени интеграции

Отличительные черты поколения (при улучшении основных показателей по сравнению со вторым поколением) :

- увеличение количества используемых УВВ;
- ПО получило дальнейшее развитие, особенно операционные системы (используются различные режимы работы: пакетный, разделения времени, запрос-ответ и т.п.);
- возможность удаленного доступа пользователей к ЭВМ, находящихся на значительных расстояниях;
- виртуальное использование ЭВМ в режиме разделения времени (вследствие различия инерционности человека и машины у пользователя создается впечатление, что ему одному предоставлено машинное время).

- применение методов автоматического проектирования;
- тенденция к унификации ЭВМ;
- основной носитель информации – магнитный диск.

Четвёртое поколение (1977-до настоящего времени)

Характеризуется применением *больших интегральных схем* (БИС) и *сверхбольших интегральных схем* (СБИС). Высокая степень интеграции способствует высокой плотности компоновки электронной аппаратуры, повышению быстродействия, снижению стоимости.

Отличительные черты поколения (при дальнейшем улучшении основных показателей):

- тенденция к унификации ЭВМ и развитию мини- и микро-ЭВМ;
- использование быстродействующих систем памяти и МОП-технологий;
- создание машин, представляющих единую систему (ЕС ЭВМ);

- появление первых персональных компьютеров и рабочих станций;

- основной носитель информации – гибкий магнитный диск.

Пятое поколение (настоящее время)

Зарождается в недрах четвертого поколения ЭВМ и в значительной мере определяется результатами работы Японского комитета по научным исследованиям в области ЭВМ. Согласно этому проекту ЭВМ пятого поколения (кроме высокой производительности и надежности при более низкой стоимости, вполне обеспечиваемые СБИС и другими новейшими технологиями) должны удовлетворять следующим качественно новым функциональным требованиям:

- обеспечить простоту применения ЭВМ путем реализации систем ввода/вывода информации голосом; диалоговой обработки информации с использованием естественных языков; возможности обучаемости, ассоциативных построений и логических выводов;

- упростить процесс создания программных средств путем автоматизации синтеза программ по спецификациям исходных требований на естественных языках;

- улучшить основные характеристики и эксплуатационные качества ВТ для удовлетворения различных социальных задач, улучшить соотношения затрат и результатов, быстродействия, легкости, компактности ЭВМ; обеспечить их разнообразие, высокую адаптируемость к приложениям и надежность в эксплуатации.

Учитывая сложность реализации поставленных перед пятым поколением задач, вполне возможно разбиение его на более обозримые и лучше ощущаемые этапы, первый из которых во многом реализован в рамках настоящего четвертого поколения.

2. КЛАССИФИКАЦИЯ ЭВМ

2.1. Классификация ЭВМ по назначению

По назначению ЭВМ можно разделить на три группы: **универсальные** (общего назначения), **проблемно-ориентированные** и **специализированные**.

Универсальные ЭВМ предназначены для решения самых различных видов задач: научных, инженерно-технических, экономических, информационных, управленческих и других задач. В качестве универсаль-

ных ЭВМ используются различные типы компьютеров, начиная от супер-ЭВМ и кончая персональными ЭВМ. Решаемые на этих компьютерах задачи отличаются сложностью алгоритмов и объемами обрабатываемых данных. Причем одни универсальные ЭВМ могут работать в многопользовательском режиме (в вычислительных центрах коллективного пользования, в локальных компьютерных сетях и т.д.), другие - в однопользовательском режиме.

Проблемно-ориентированные ЭВМ служат для решения более узкого круга задач, связанных, как правило, с управлением технологическими объектами; регистрацией, накоплением и обработкой относительно небольших объемов данных; выполнением расчетов по относительно несложным алгоритмам. На проблемно-ориентированных ЭВМ, в частности, создаются всевозможные управляющие вычислительные комплексы.

Специализированные ЭВМ используются для решения еще более узкого круга задач или реализации строго определенной группы функций. Такая узкая ориентация ЭВМ позволяет четко специализировать их структуру, во многих случаях существенно снизить их сложность и стоимость при сохранении высокой производительности и надежности их работы.

2.2. Классификация ЭВМ по функциональным возможностям

По функциональным возможностям и размерам ЭВМ можно разделить на супер-ЭВМ, большие, малые и микро-ЭВМ.

Исторически первыми появились большие ЭВМ, элементная база которых прошла путь от электронных ламп до интегральных схем со сверхвысокой степенью интеграции.

Супер-ЭВМ - мощные, высокоскоростные вычислительные машины (системы) с производительностью от сотен миллионов до триллионов операций с плавающей точкой в секунду. Супер-ЭВМ выгодно отличаются от больших универсальных ЭВМ по быстродействию числовой обработки, а от специализированных машин, обладающих высоким быстродействием в сугубо ограниченных областях, возможностью решения широкого класса задач с числовыми расчетами.

При производительности порядка нескольких GFLOPS можно еще обойтись одним векторно-конвейерным процессором (однопроцессорные супер-ЭВМ). Создание высокопроизводительной супер-ЭВМ с быстродействием порядка TFLOPS по современной технологии

на одном процессоре не представляется возможным. Это связано с ограничением, обусловленным конечным значением скорости распространения электромагнитных волн (300 000 км/сек), так как время распространения сигнала на расстояние нескольких миллиметров (линейный размер стороны микропроцессора) при быстродействии 100 млрд. оп/с становится соизмеримым с временем выполнения одной операции. Поэтому супер-ЭВМ с такой производительностью создаются в виде высокопараллельных многопроцессорных вычислительных систем.

Большие ЭВМ часто называют мэйнфреймами (Mainframe). Они поддерживают многопользовательский режим работы (обслуживают одновременно от 16 до 1000 пользователей).

Основные направления эффективного применения мэйнфреймов - это решение научно-технических задач, работа в вычислительных системах с пакетной обработкой информации, работа с большими базами данных, управление вычислительными сетями и их ресурсами. Последнее направление - использование мэйнфреймов в качестве больших серверов вычислительных сетей - часто отмечается специалистами среди наиболее актуальных.

Малые ЭВМ (мини-ЭВМ) - надежные, недорогие и удобные в эксплуатации компьютеры, обладающие несколько более низкими по сравнению с мэйнфреймами возможностями. В многопользовательском режиме поддерживаются 16 - 512 пользователей.

Основные их особенности:

- широкий диапазон производительности в конкретных условиях применения,
- аппаратная реализация большинства системных функций ввода-вывода информации,
- простая реализация многопроцессорных и многомашинных систем,
- высокая скорость обработки прерываний,
- возможность работы с форматами данных различной длины.

К достоинствам мини-ЭВМ можно отнести:

- 1) специфическую архитектуру с большой модульностью;
- 2) лучшее, чем у мэйнфреймов, соотношение производительность/стоимость;
- 3) широкая номенклатура периферийных устройств;
- 4) повышенную точность вычислений.

Мини-ЭВМ успешно применяются:

- в качестве управляющих вычислительных комплексов.

- вычислений в многопользовательских вычислительных системах,
- в системах автоматизированного проектирования,
- в системах моделирования и искусственного интеллекта,

Микро-ЭВМ по назначению можно разделить на универсальные и специализированные.

Универсальные многопользовательские ЭВМ - это мощные микро ЭВМ, используемые в компьютерных сетях, оборудованные несколькими видеотерминалами и функционирующие в режиме разделения времени, что позволяет эффективно работать на них сразу нескольким пользователям. Это **универсальные серверы (Server)** компьютерных сетей, обрабатывающие запросы от всех станций сети, выделенный для обработки запросов от всех станций вычислительной сети, предоставляющий этим станциям доступ к общим системным ресурсам (вычислительным мощностям, базам данных, библиотекам программ, принтерам, факсам и др.) и распределяющий эти ресурсы.

Эту интенсивно развивающуюся группу компьютеров обычно относят к микро-ЭВМ, но по своим характеристикам мощные серверы скорее можно отнести к малым ЭВМ и даже к мэйнфреймам, а супер серверы приближаются к супер-ЭВМ.

Универсальные однопользовательские ЭВМ или персональные компьютеры (ПК) должны удовлетворять требованиям общедоступности и универсальности применения и иметь следующие характеристики:

- малую стоимость, находящуюся в пределах доступности для индивидуального покупателя;
- автономность эксплуатации без специальных требований к условиям окружающей среды;
- гибкость архитектуры, обеспечивающую ее адаптивность к разнообразным применениям в сфере управления, науки, образования, в быту;
- «дружественность» операционной системы и прочего программного обеспечения для пользователя;
- высокую надежность работы (более 5000 ч. наработки на отказ).

Специализированные ЭВМ ориентированы на решение определенного (постоянного) класса задач в течение периода своей эксплуатации. Ориентация специализированных ЭВМ осуществляется различными способами:

- специальной аппаратной организацией самих ЭВМ или их внешних связей;

- созданием для ЭВМ специального программного обеспечения;

- введением дополнительных аппаратных блоков, расширяющих те или иные функции, возлагаемые на ЭВМ,

- и др.

Можно выделить следующие основные области применения специализированных ЭВМ:

1) промышленное производство и транспорт;

2) военная техника и оборона;

3) непромышленная сфера.

Специализированные серверы используются для устранения наиболее «узких» мест в работе сети, а именно: создания и управления базами и архивами данных, поддержка многоадресной факсимильной связи и электронной почты, управления многопользовательскими терминалами (принтером, плоттером и др.

Файл-сервер используется для работы с файлами данных, имеет объемные дисковые ЗУ.

Архивационный сервер (сервер резервного копирования) предназначен для резервного копирования информации, использует накопители на магнитной ленте (стримеры) со сменными картриджами.

Факс-сервер, почтовый сервер - выделенные компьютеры для организации эффективной многоадресной факсимильной связи или электронной почты.

Встраиваемые микро-ЭВМ входят составным элементом в промышленные и транспортные системы, технические устройства и аппараты, бытовые приборы. Они способствуют существенному повышению их эффективности функционирования, улучшению технико-экономических и эксплуатационных характеристик.

Специализированные однопользовательские ЭВМ или рабочие станции (Work station), - это однопользовательская система с мощным процессором и многозадачной ОС, имеющая развитую графику с высоким разрешением, большую дисковую и оперативную память и встроенные сетевые средства.

Рабочие станции появились на рынке ЭВМ почти одновременно с ПК и находились впереди по своим вычислительным возможностям. Переломным моментом в развитии рабочих станций стало появление новой архитектуры микропроцессоров RISC, позволившей резко поднять производительность ЭВМ. Современные рабочие станции сопос-

тавимы, а иногда даже превосходят ПК по своим характеристикам. Современная рабочая станция - это не просто большая вычислительная мощность, это тщательно сбалансированные возможности всех подсистем машины, чтобы ни одна из них не стала «узким местом», сводя на нет преимущества других. Всё это в значительной мере и определяло их область применения и проблемную ориентацию: автоматизированное проектирование, банковское дело, управление производством, разведка и добыча нефти, связь, издательская деятельность и др.

3. ФУНКЦИОНАЛЬНАЯ И СТРУКТУРНАЯ ОРГАНИЗАЦИЯ ЭВМ

3.1. Связь между функциональной и структурной организацией ЭВМ

Существуют два взгляда на построение и функционирование ЭВМ. Первый - взгляд пользователя, не интересующегося технической реализацией ЭВМ и озабоченного только получением некоторого набора функций и услуг, обеспечивающих эффективное решение его задач; второй - разработчика ЭВМ, усилия которого направлены на рациональную техническую реализацию необходимых пользователю функций. С учетом этого обстоятельства и вводятся понятия "функциональная и структурная организация" компьютера.

Функциональная организация ЭВМ - это абстрактная модель ЭВМ, описывающая функциональные возможности машины и предоставляемые ею услуги. Функциональная организация ЭВМ в значительной степени определяется предъявляемыми к ней требованиями, уровнем подготовки потенциальных пользователей, типом решаемых ими задач, потребностями в развитии компьютера (по емкости ЗУ, разрядности, составу периферийных устройств и др.).

Предусматриваемые абстрактной моделью функции ЭВМ реализуются на основе реальных, физических средств (устройств, блоков, узлов, элементов) в рамках определенной структуры.

В общем случае под **структурной организацией ЭВМ** понимается некоторая физическая модель, устанавливающая состав, порядок и принципы взаимодействия основных функциональных частей машины (без излишних деталей их технической реализации).

По степени детальности различают структурные схемы, составленные на уровне устройств, блоков, узлов, элементов.

Устройство - наиболее крупная функциональная часть ЭВМ, состоящая из элементов, узлов, блоков и выполняющая глобальные операции над кодированными данными (запоминание, обработку, преобразование).

Блок - функциональный компонент ЭВМ, состоящий из элементов и узлов и выполняющий операции над машинными словами или управляющий такими операциями (пример: сумматор, блок регистров).

Узел - часть машины, состоящая из нескольких более простых элементов и представляющая собой сборочную единицу (логическая схема).

Элемент, простейшее устройство ЭВМ, выполняющее одну операцию над входными сигналами. (пример – логический элемент).

Блоки и устройства часто изготавливаются в виде самостоятельных конструктивных модулей.

Функциональная организация ЭВМ играет ведущую роль и в значительной степени определяет структурную организацию машины, хотя и не дает жестких ограничений на конечную техническую реализацию структурных элементов. Одна и та же функция может быть реализована на совершенно разных технических средствах.

3.2. Обобщенная структура ЭВМ и пути её развития

Развитие архитектуры неизбежно ведет к развитию структуры ЭВМ. Реализация принципов интеллектуализации, которые все больше определяют развитие архитектуры ЭВМ, возможна при совершенствовании структурной организации, обеспечивающей повышение эффективности вычислительного процесса и, как следствие этого, рост производительности ЭВМ. В конечном счете, условием и критерием развития структуры является рост производительности ЭВМ.

Основной тенденцией в развитии структуры ЭВМ является разделение функций системы и максимальная специализация подсистем для выполнения этих функций.

Обобщенная структура ЭВМ приведена на рис.1.2.

Обобщенная структура ЭВМ состоит из следующих составных частей:

- обрабатывающей подсистемы;
- подсистемы памяти;

- подсистемы ввода-вывода;
- подсистемы управления и обслуживания.

Для каждой подсистемы выделены основные направления их развития.

3.3. Обработка подсистема

Развитие обрабатывающей подсистемы в большей степени, чем всех остальных подсистем, идет по пути разделения функций и повышения специализации составляющих ее устройств. Создаются специальные средства, которые осуществляют функции управления системой, освобождая от этих функций средства обработки. Такое распределение функций сокращает эффективное время обработки информации и повышает производительность ЭВМ. В то же время средства управления, как и средства обработки, становятся более специализированными. Устройство управления памятью реализует эффективные методы передачи данных между средствами обработки и подсистемой памяти. Меняются функции центрального устройства управления. С одной стороны, ряд функций передается в другие подсистемы (например, функции ввода-вывода), с другой - развиваются средства организации параллельной обработки нескольких команд (суперскалярная обработка) с одновременным повышением темпа исполнения последовательности команд. Для повышения темпа выполнения последовательности команд применяются методы конвейерной обработки наряду с совершенствованием алгоритмов диспетчеризации и исполнения команд. Бурно развивается управление межпроцессорным обменом как эффективное средство передачи информации между несколькими центральными процессорами, входящими в состав вычислительной системы или комплекса.

Операционные устройства (АЛУ) обрабатывающей подсистемы, кроме традиционных средств скалярной (суперскалярной) и логической обработки, все шире стали включать специальные средства векторной обработки. При этом время выполнения операций можно резко сократить как за счет использования арифметического конвейера (одного или нескольких), так и за счет сокращения такта работы конвейера. Возможности задач к распараллеливанию алгоритма счета снимают принципиальные ограничения к организации параллельной обработки информации и использованию структур с глубокой конвейеризацией. В устройствах скалярной обработки все шире появля-

ются специальные операционные блоки, оптимизированные на эффективное выполнение отдельных операций.

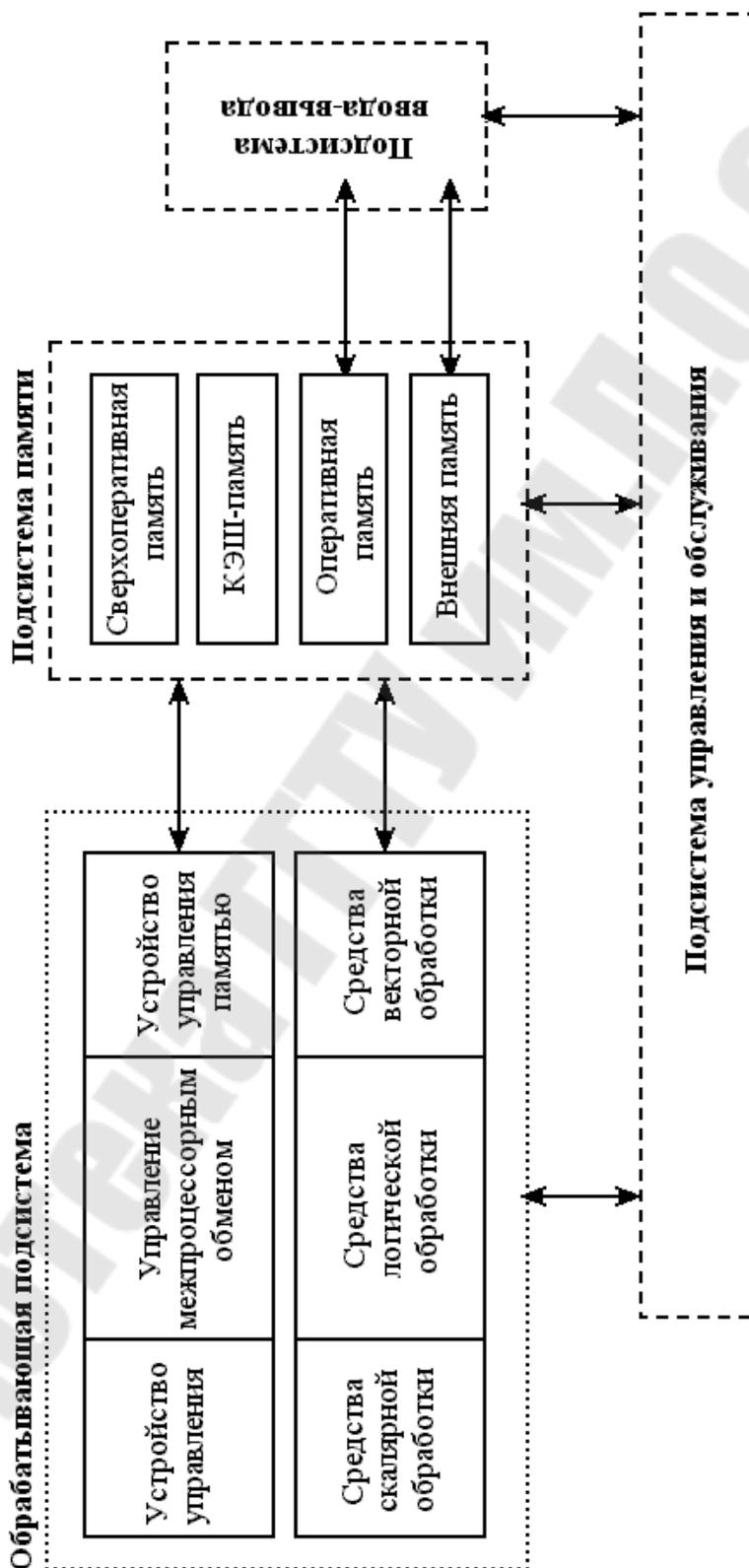


Рисунок 1 - Обобщенная структура ЭВМ

3.4. Подсистема памяти

Подсистема памяти современных компьютеров имеет иерархическую структуру, состоящую из нескольких уровней:

- сверхоперативный уровень (локальная память процессора, кэш-память первого и второго уровня);
- оперативный уровень (оперативная память, дисковый кэш);
- внешний уровень (внешние ЗУ на дисках, лентах и т.д.).

Каждый уровень состоит из запоминающих устройств, обладающих различным быстродействием и емкостью. Чем выше уровень, тем выше быстродействие соответствующей памяти, но меньше её емкость.

Эффективными методами повышения производительности ЭВМ являются увеличение количества регистров общего назначения процессора, использование многоуровневой кэш-памяти, увеличение объема и пропускной способности оперативной памяти, буферизация передачи информации между ОП и внешней памятью. Увеличение пропускной способности оперативной памяти достигается за счет увеличения их расслоения и секционирования.

3.5. Подсистема ввода-вывода

В состав подсистемы ввода-вывода входит набор специализированных устройств, между которыми распределены функции ввода-вывода, что позволяет свести к минимуму потери производительности системы при операциях ввода-вывода. Эти устройства можно условно разделить на критичные и некритичные по быстродействию. К критичным по быстродействию устройствам относятся обработчики команд ввода-вывода и контроллеры интерфейсов. Эти устройства определяют пропускную способность подсистемы ввода-вывода. Некритичные по быстродействию устройства управляют распределением линий в подсистеме ввода-вывода.

Основными направлениями развития подсистем ввода-вывода являются канальная технология ввода-вывода, матричная топология коммутации периферийных устройств (ПУ), увеличение количества и пропускной способности каналов.

3.6. Подсистема управления и обслуживания

Подсистема управления и обслуживания - это совокупность аппаратно-программных средств, предназначенных для обеспечения максимальной производительности, заданной надежности, ремонтпригодности, удобства настройки и эксплуатации. Она обеспечивает проблемную ориентацию и заданное время наработки на отказ, подготовку и накопление статистических сведений о загрузке и прохождении вычислительного процесса, выполняет функции "интеллектуального" интерфейса с различными категориями обслуживающего персонала, осуществляет инициализацию, тестирование и отладку. Подсистема управления и обслуживания позволяет поднять на качественно новый уровень эксплуатацию современных ЭВМ.

При разработке структуры ЭВМ все подсистемы должны быть сбалансированы между собой. Только оптимальное согласование быстродействия обрабатывающей подсистемы с объемами и скоростью передачи информации подсистемы памяти, с пропускной способностью подсистемы ввода-вывода позволяет добиться максимальной эффективности использования ЭВМ.

4. МНОГОУРОВНЕВАЯ КОМПЬЮТЕРНАЯ ОРГАНИЗАЦИЯ

Существует огромная разница между тем, что удобно людям, и тем, что могут компьютеры. Люди хотят сделать X, но компьютеры могут сделать только Y. Из-за этого возникает проблема. Цель данного курса — объяснить, как решить эту проблему, используя максимум возможностей аппаратного уровня ВМ.

Вышеупомянутую проблему можно решить двумя способами. Оба способа подразумевают разработку новых команд, более удобных для человека, чем встроенные машинные команды. Эти новые команды в совокупности формируют язык, который мы будем называть Я 1. Встроенные машинные команды тоже формируют язык, и мы будем называть его Я 0. Компьютер может выполнять только программы, написанные на его машинном языке Я 0. Два способа решения проблемы различаются тем, каким образом компьютер будет выполнять программы, написанные на языке Я 1, — ведь в конечном итоге компьютеру доступен только машинный язык Я 0.

Первый способ выполнения программы, написанной на языке Я 1, подразумевает замену каждой команды эквивалентным набором

команд на языке Я 0. В этом случае компьютер выполняет новую программу, написанную на языке Я 0, вместо старой программы, написанной на Я 1. Эта технология называется трансляцией.

Второй способ означает создание программы на языке Я 0, получающей в качестве входных данных программы, написанные на языке Я 1. При этом каждая команда языка Я 1 обрабатывается поочередно, после чего сразу выполняется эквивалентный ей набор команд языка Я 0. Эта технология не требует составления новой программы на Я 0. Она называется интерпретацией, а программа, которая осуществляет интерпретацию, называется интерпретатором.

Между трансляцией и интерпретацией много общего. В обоих подходах компьютер в конечном итоге выполняет набор команд на языке Я 0, эквивалентных командам Я 1. Различие лишь в том, что при трансляции вся программа Я 1 переделывается в программу Я 0, программа Я 1 отбрасывается, а новая программа на Я 0 загружается в память компьютера и затем выполняется.

При интерпретации каждая команда программы на Я 1 перекодирована в Я 0 и сразу же выполняется. В отличие от трансляции, здесь не создается новая программа на Я 0, а происходит последовательная перекодировка и выполнение команд. С точки зрения интерпретатора, программа на Я 1 есть не что иное, как «сырые» входные данные. Оба подхода широко используются как вместе, так и по отдельности. Впрочем, чем мыслить категориями трансляции и интерпретации, гораздо проще представить себе существование гипотетического компьютера или виртуальной машины, для которой машинным языком является язык Я 1.

Между языком и виртуальной машиной существует важная зависимость. Каждая машина поддерживает какой-то определенный машинный язык, состоящий из всех команд, которые эта машина может выполнять. В сущности, машина определяет язык. Сходным образом язык определяет машину, которая может выполнять все программы, написанные на этом языке. Машину, определяемую тем или иным языком, очень сложно и дорого конструировать из электронных схем, однако представить себе такую машину мы можем. Компьютер для работы с машинным языком С или С++ был бы слишком сложным, но в принципе его можно разработать, учитывая высокий уровень современных технологий. Однако существуют веские причины не создавать такой компьютер — это крайне неэффективное, по сравнению с другими, решение. Действительно, технология должна быть не только

осуществимой, но и рациональной. Те, кто хочет понять, как в действительности работает компьютер, должны изучить все уровни. Также должны быть знакомы со всеми уровнями разработчики новых компьютеров или новых уровней (то есть новых виртуальных машин).

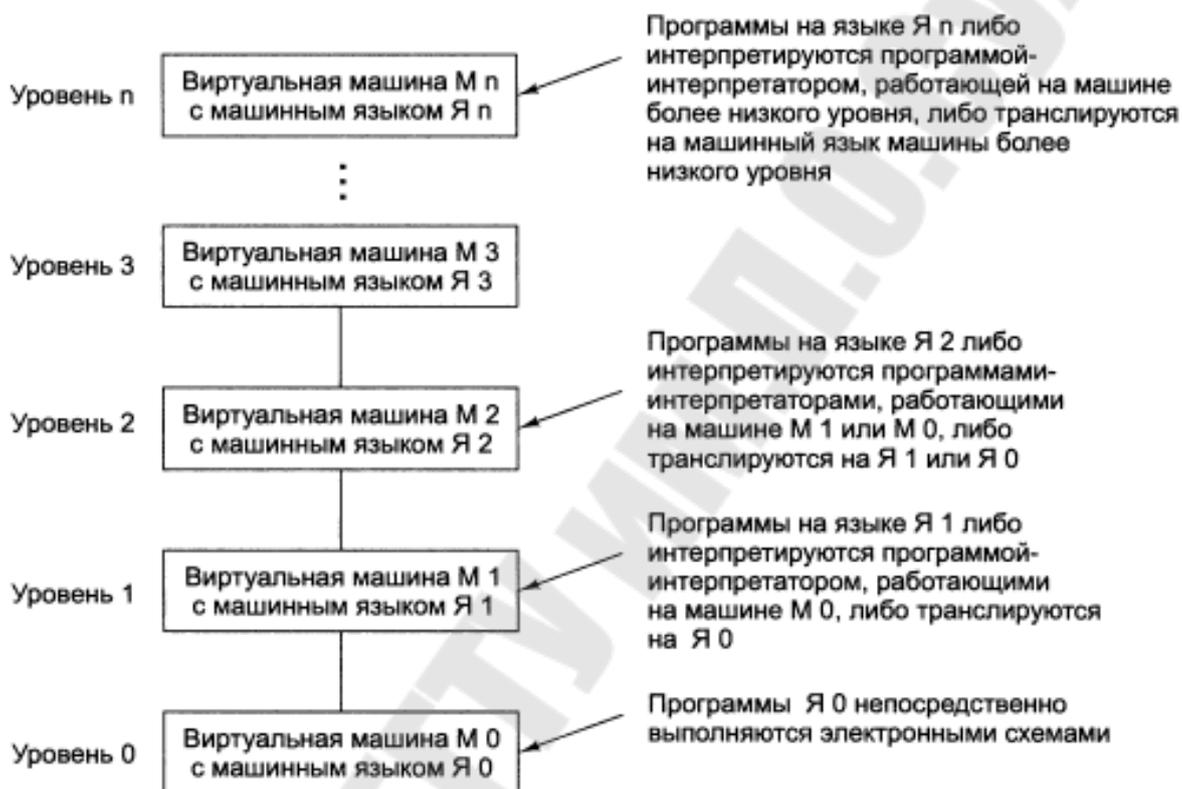


Рисунок 2 - Многоуровневая машина

Большинство современных компьютеров состоит из двух и более уровней. Существуют машины даже с шестью уровнями (рис. 3). Уровень 0 — это аппаратное обеспечение машины. Электронные схемы на уровне 1 выполняют машинно-зависимые программы. Ради полноты нужно упомянуть о существовании еще одного уровня, который расположен ниже нулевого. Этот уровень не показан на рис. 3, так как он попадает в сферу электронной техники и, следовательно, не рассматривается в этом курсе. Он называется уровнем физических устройств. На этом уровне находятся транзисторы, которые для разработчиков компьютеров являются примитивами. Объяснить, как работают транзисторы, — задача физики.

На самом нижнем уровне из тех, что мы будем изучать, а именно, на цифровом логическом уровне, объекты называются вентилями. Хотя вентили состоят из аналоговых компонентов, таких как транзи-

сторы, они могут быть точно смоделированы как цифровые устройства. У каждого вентиля есть один или несколько цифровых входов (сигналов, представляющих 0 или 1). Вентиль вычисляет простые функции этих сигналов, такие как И или ИЛИ. Каждый вентиль формируется из нескольких транзисторов. Несколько вентилях формируют 1 бит памяти, который может содержать 0 или 1. Биты памяти, объединенные в группы, например, по 16, 32 или 64, формируют регистры. Каждый регистр может содержать одно двоичное число до определенного предела. Из вентилях также может состоять сам компьютер.

Следующий уровень называется уровнем микроархитектуры. На этом уровне находятся совокупности 8 или 32 регистров, которые формируют локальную память и схему, называемую АЛУ (арифметико-логическое устройство). АЛУ выполняет простые арифметические операции. Регистры вместе с АЛУ формируют тракт данных, по которому поступают данные. Тракт данных работает следующим образом. Выбирается один или два регистра, АЛУ производит над ними какую-либо операцию, например сложения, после чего результат вновь помещается в один из этих регистров.

На некоторых машинах работа тракта данных контролируется особой программой, которая называется микропрограммой. На других машинах тракт данных контролируется аппаратными средствами. Поскольку сейчас тракт данных обычно контролируется аппаратным обеспечением, мы изменили название, чтобы точнее отразить смысл.

На машинах, где тракт данных контролируется программным обеспечением, микропрограмма — это интерпретатор для команд на уровне 2. Микропрограмма вызывает команды из памяти и выполняет их одну за другой, используя при этом тракт данных. Например, при выполнении команды ADD она вызывается из памяти, ее операнды помещаются в регистры, АЛУ вычисляет сумму, а затем результат переправляется обратно. На компьютере с аппаратным контролем тракта данных происходит такая же процедура, но при этом нет программы, интерпретирующей команды уровня 2.

Уровень 2 мы будем называть уровнем архитектуры набора команд. Каждый производитель публикует руководство для компьютеров, которые он продает, под названием «Руководство по машинному языку X», «Принципы работы компьютера Y» и т. п. Подобное руководство содержит информацию именно об этом уровне. Описываемый в нем набор машинных команд в действительности выполняется

микропрограммой-интерпретатором или аппаратным обеспечением. Если производитель поставляет два интерпретатора для одной машины, он должен издать два руководства по машинному языку, отдельно для каждого интерпретатора.

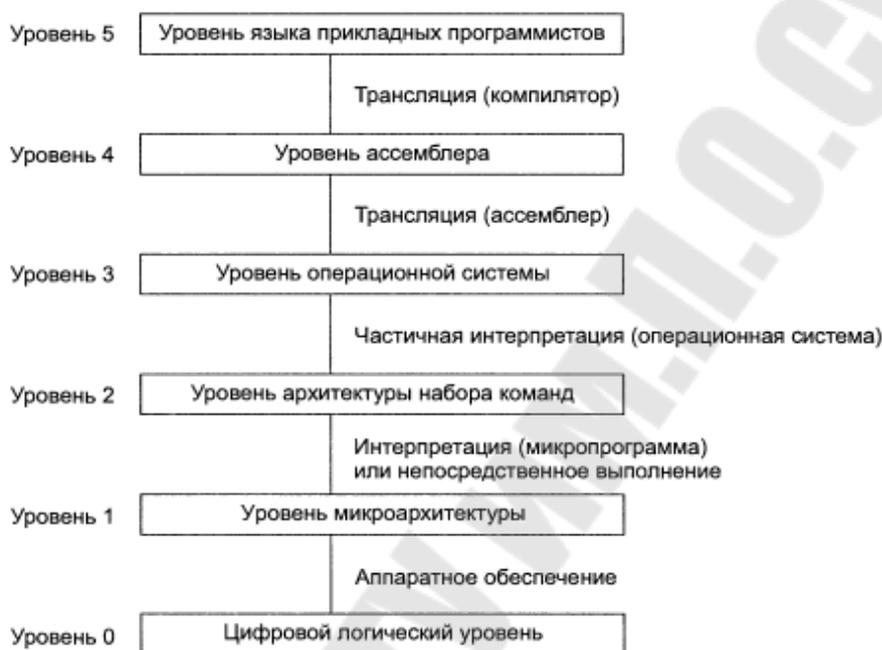


Рисунок 3 - Шестиуровневый компьютер

Следующий уровень обычно является гибридным. Большинство команд в его языке есть также и на уровне архитектуры набора команд (команды, имеющиеся на одном из уровней, вполне могут быть представлены и на других уровнях). У этого уровня есть некоторые дополнительные особенности: новый набор команд, другая организация памяти, способность выполнять две и более программы одновременно и некоторые другие. При построении уровня 3 возможно больше вариантов, чем при построении уровней 1 и 2.

Новые средства, появившиеся на уровне 3, выполняются интерпретатором, который работает на втором уровне. Этот интерпретатор был когда-то назван операционной системой. Команды уровня 3, идентичные командам уровня 2, выполняются микропрограммой или аппаратным обеспечением, но не операционной системой. Другими словами, одна часть команд уровня 3 интерпретируется операционной системой, а другая часть — микропрограммой. Вот почему этот уровень считается гибридным. Мы будем называть этот уровень уровнем операционной системы.

Между уровнями 3 и 4 есть существенная разница. Нижние три уровня задуманы не для того, чтобы с ними работал обычный программист. Они изначально ориентированы на интерпретаторы и трансляторы, поддерживающие более высокие уровни. Эти трансляторы и интерпретаторы составляют так называемые системные программистами, которые специализируются на разработке новых виртуальных машин.

Уровни с четвертого и выше предназначены для прикладных программистов, решающих конкретные задачи. Еще одно изменение, появившееся на уровне 4, — механизм поддержки более высоких уровней. Уровни 2 и 3 обычно интерпретируются, а уровни 4, 5 и выше обычно, хотя и не всегда, транслируются.

Другое отличие между уровнями 1, 2, 3 и уровнями 4, 5 и выше — особенность языка. Машинные языки уровней 1, 2 и 3 — цифровые. Программы, написанные на этих языках, состоят из длинных рядов цифр, которые воспринимаются компьютерами, но малопонятны для людей. Начиная с уровня 4, языки содержат слова и сокращения, понятные человеку.

Уровень 5 обычно состоит из языков, разработанных для прикладных программистов. Такие языки называются языками высокого уровня. Существуют сотни языков высокого уровня. Наиболее известные среди них — C, C++, Java, LISP и Prolog. Программы, написанные на этих языках, обычно транслируются на уровень 3 или 4. Трансляторы, которые обрабатывают эти программы, называются компиляторами. Отметим, что иногда также имеет место интерпретация. Например, программы на языке Java сначала транслируются на язык, напоминающий ISA и называемый байт-кодом Java, который затем интерпретируется.

В некоторых случаях уровень 5 состоит из интерпретатора для конкретной прикладной области, например символической логики. Он предусматривает данные и операции для решения задач в этой области, выраженные при помощи специальной терминологии.

Таким образом, компьютер проектируется как иерархическая структура уровней, которые надстраиваются друг над другом. Каждый уровень представляет собой определенную абстракцию различных объектов и операций. Рассматривая компьютер подобным образом, мы можем не принимать во внимание ненужные нам детали и, таким образом, сделать сложный предмет более простым для понимания.

Набор типов данных, операций и характеристик каждого отдельно взятого уровня называется **архитектурой**. Архитектура связана с программными аспектами. Например, сведения о том, сколько памяти можно использовать при написании программы, — часть архитектуры. Аспекты реализации (например, технология, применяемая при реализации памяти) не являются частью архитектуры. Изучая методы проектирования программных элементов компьютерной системы, мы изучаем компьютерную архитектуру. На практике термины «компьютерная архитектура» и «компьютерная организация» употребляются как синонимы.

В самых первых компьютерах граница между аппаратным и программным обеспечением была очевидна. Со временем, однако, произошло значительное размывание этой границы, в первую очередь благодаря тому, что в процессе развития компьютеров уровни добавлялись, убирались и сливались друг с другом. Аппаратное и программное обеспечение логически эквивалентны. Другими словами аппаратное обеспечение — это всего лишь «окаменевшее» программное обеспечение.

В те времена, когда ЭВМ только появились, принципы работы с ними сильно отличались от современных. Одним компьютером пользовалось большое количество людей. Рядом с машиной лежал листок бумаги, и если программист хотел запустить свою программу, он записывался на какое-то определенное время, скажем, на среду с трех часов ночи до пяти утра.

В 60-е годы человек попытался ускорить дело, автоматизировав работу оператора. Программа под названием операционная система загружалась в компьютер на все время его работы. Программист приносил пачку перфокарт со специализированной программой, которая выполнялась операционной системой. В последующие годы операционные системы все больше и больше усложнялись. К уровню архитектуры набора команд добавлялись новые команды, приспособления и особенности, и в конечном итоге сформировался новый уровень. Некоторые команды нового уровня были идентичны командам предыдущего, но некоторые (в частности, команды ввода-вывода) полностью отличались. Эти новые команды тогда назывались макросами операционной системы, или вызовами супервизора. Сейчас обычно используется термин системный вызов.

Мы рассмотрели развитие компьютеров, чтобы показать, что граница между аппаратным и программным обеспечением постоянно

смещается. Сегодняшнее программное обеспечение может быть за-
трашным аппаратным обеспечением и наоборот. Также обстоит дело и
с уровнями — между ними нет четких границ.

Для программиста не важно, как на самом деле выполняется ко-
манда (за исключением, может быть, скорости выполнения). Про-
граммист, работающий на уровне архитектуры системы, может ис-
пользовать команду умножения, как будто это команда аппаратного
обеспечения, и даже не задумываться об этом. То, что для одного че-
ловека — программное обеспечение, для другого — аппаратное.

5. АРХИТЕКТУРЫ ЭВМ

Под **архитектурой ЭВМ** понимается функциональная и структур-
ная организация машины, определяющая методы кодирования данных,
состав, назначение, принципы взаимодействия технических средств и
программного обеспечения.

Можно выделить следующие важные для пользователя группы
характеристик ЭВМ, определяющих её архитектуру:

- 1) характеристики и состав модулей базовой конфигурации ЭВМ;
- 2) характеристики машинного языка и системы команд (количест-
во и номенклатура команд, их форматы, системы адресации, наличие
программно-доступных регистров в процессоре и т.п.), которые опре-
деляют алгоритмические возможности процессора ЭВМ;
- 3) технические и эксплуатационные характеристики ЭВМ;
- 4) состав программного обеспечения ЭВМ и принципы его взаи-
модействия с техническими средствами ЭВМ.

К наиболее общему принципу классификации ЭВМ и систем по
типам архитектуры следует отнести разбиение их на однопроцессор-
ные и многопроцессорные архитектуры (см. рис.4).

Исторически первыми появились однопроцессорные архитекту-
ры. Классическим примером однопроцессорной архитектуры является
архитектура фон Неймана со строго последовательным выполнением
команд: процессор по очереди выбирает команды программы и также
по очереди обрабатывает данные (программа и данные хранятся в
единственной последовательно адресуемой памяти).

По мере развития вычислительной техники архитектура фон
Неймана обогатилась сначала конвейером команд, а затем много-

функциональной обработкой получила обобщенное название компьютера с одним потоком команд и одним потоком данных.

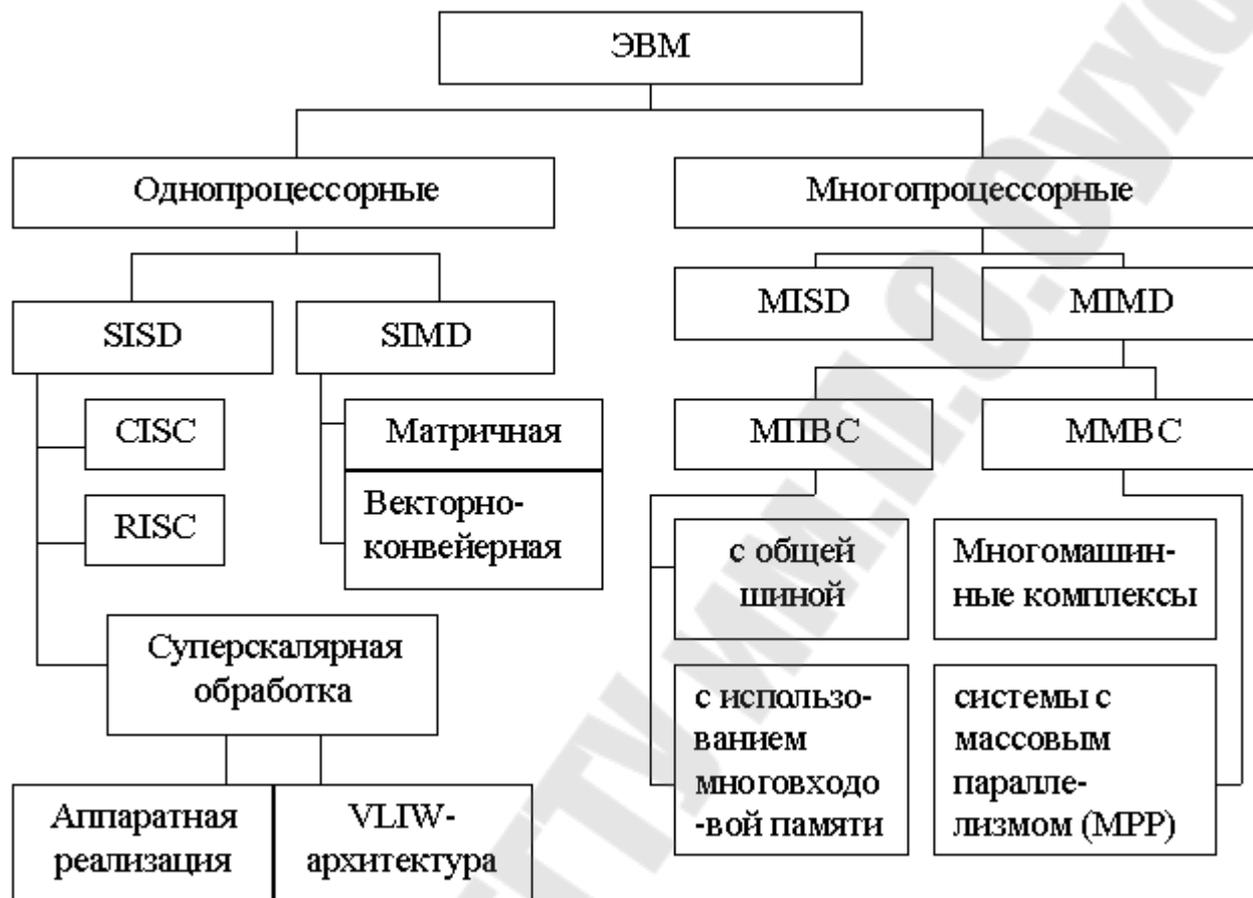


Рисунок 4 - Архитектуры ЭВМ

Поток команд - это последовательность команд, выполняемых ЭВМ (системой), а **поток данных** - последовательность данных (исходная информация и промежуточные результаты решения задачи), обрабатываемых под управлением потока команд.

5.1. SISD-компьютеры



Рисунок 5 - SISD- архитектура

SISD (Single Instruction Single Data) или ОКОД - один поток команд, один поток данных. SISD компьютеры это обычные, "традиционные" последовательные компьютеры, в которых в каждый момент времени выполняется лишь одна операция над одним элементом данных (числовым или каким-либо другим значением). При работе такой системы в мультипрограммном режиме, когда совместно решаются несколько задач (программы и исходные данные по каждой из них хранятся в оперативной памяти), обеспечивается параллельная работа устройств системы, происходит разделение времени и оборудования между совместно выполняемыми программами. Но в каждый данный момент операционное устройство (АЛУ), поскольку оно является единственным, занимается обработкой информации по какой-то одной команде, т. е. одновременное преобразование информации в АЛУ по нескольким командам, принадлежащим разным участкам одной и той же программы или разным программам, невозможно. Основная масса современных ЭВМ функционирует в соответствии с принципом фон Неймана и имеет архитектуру класса SISD. Данная архитектура породила CISC, RISC и архитектуру с суперскалярной обработкой.

5.2. Компьютеры с CISC архитектурой

Компьютеры с CISC (Complex Instruction Set Computer) **архитектурой** имеют комплексную (полную) систему команд, под управлением которой выполняются всевозможные операции типа «память-память», «память-регистр», «регистр-память», «регистр-регистр». Данная архитектура характеризуется:

- большим числом команд (более 200);
- переменной длиной команд (от 1 до 11 байт);
- значительным числом способов адресации и форматов команд;
- сложностью команд и многотактностью их выполнения;
- наличием микропрограммного управления, что снижает быстродействие и усложняет процессор.

Обмен с памятью в процессе выполнения команды делает практически невозможной глубокую конвейеризацию арифметики, т.е. ограничивается тактовая частота процессора, а значит, и его производительность.

Большинство компьютеров типа IBM PC относились к CISC архитектуре, например, компьютеры с микропроцессорами 8080, 80486, 80586 (товарная марка Pentium).

5.3. Компьютеры с RISC архитектурой

Компьютеры с RISC (Reduced Instruction Set Computer) **архитектурой** содержат набор простых, часто употребляемых в программах команд. Основными являются операции типа «регистр-регистр».

Данная архитектура характеризуется:

- сокращенным числом команд;
- тем, что большинство команд выполняется за один машинный такт;
- постоянной длиной команд;
- небольшим количеством способов адресации и форматов команд;
- тем, что для простых команд нет необходимости в использовании микропрограммного управления;
- большим числом регистров внутренней памяти процессора.

Компьютеры с RISC-архитектурой «обязаны» иметь преимущество в производительности по сравнению с CISC компьютерами, за которое приходится расплачиваться наличием в программах дополнительных команд обмена регистров процессора с оперативной памятью.

5.4. Компьютеры с суперскалярной обработкой

Еще одной разновидностью однопоточковой архитектуры является **суперскалярная обработка**. Смысл этого термина заключается в том, что в аппаратуру процессора закладываются средства, позволяющие одновременно выполнять две или более скалярные операции, т.е. команды обработки пары чисел. Суперскалярная архитектура базируется на многофункциональном параллелизме и позволяет увеличить производительность компьютера пропорционально числу одновременно выполняемых операций. Способы реализации суперскалярной обработки могут быть разными.

Аппаратная реализация суперскалярной обработки применяется как в CISC, так и в RISC - процессорах и заключается в чисто аппаратном механизме выборки из буфера инструкций (или кэша инст-

рукций) несвязанных команд и параллельном запуске их на исполнение. Этот метод хорош тем, что он «прозрачен» для программиста, составление программ для подобных процессоров не требует никаких специальных усилий, ответственность за параллельное выполнение операций возлагается в основном на аппаратные средства.

VLIW-архитектуры суперскалярной обработки. Вторым способом реализации суперскалярной обработки заключается в кардинальной перестройке всего процесса трансляции и исполнения программ. Уже на этапе подготовки программы компилятор группирует несвязанные операции в пакеты, содержимое которых строго соответствует структуре процессора. Например, если процессор содержит функционально независимые устройства (сложения, умножения, сдвига и деления), то максимум, что компилятор может «уложить» в один пакет - это четыре разнотипные операции; (сложение, умножение, сдвиг и деление). Сформированные пакеты операций преобразуются компилятором в командные слова, которые по сравнению с обычными инструкциями выглядят очень большими. Отсюда и название этих суперкоманд и соответствующей им архитектуры - VLIW (Very Large Instruction Word - очень широкое командное слово). По идее, затраты на формирование суперкоманд должны окупаться скоростью их выполнения и простотой аппаратуры процессора, с которого снята вся «интеллектуальная» работа по поиску параллелизма несвязанных операций. Однако практическое внедрение VLIW-архитектуры затрудняется значительными проблемами эффективной компиляции.

Архитектуры класса SISD охватывают те уровни программного параллелизма, которые связаны с одиночным потоком данных. Они реализуются многофункциональной обработкой и конвейером команд.

Параллелизм циклов и итераций тесно связан с понятием множественности потоков данных и реализуется векторной обработкой. В таксономии компьютерных архитектур М. Флина выделена специальная группа однопроцессорных систем с параллельной обработкой потоков данных – SIMD.

5.5. SIMD-компьютеры

SIMD (Single Instruction Stream - Multiple Data Stream) или ОКМД - один поток команд и множество потоков данных. SIMD компьютеры

состоят из одного командного процессора (управляющего модуля), называемого контроллером, и нескольких модулей обработки данных, называемых процессорными элементами.

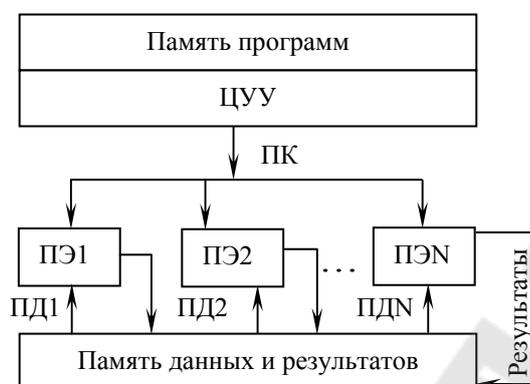


Рисунок 6 - SIMD- архитектура

Управляющий модуль принимает, анализирует и выполняет команды. Если в команде встречаются данные, контроллер рассылает на все процессорные элементы команду, и эта команда выполняется на нескольких или на всех процессорных элементах.

Все процессорные элементы идентичны и каждый из них представляет собой совокупность управляюще-обрабатывающего органа (быстродействующего процессора) и процессорной памяти небольшой емкости. Процессорные элементы выполняют операции параллельно над разными потоками данных (ПД) под управлением общего потока команд (ПК), вследствие чего такие ЭВМ называются системами с общим потоком команд. В любой момент в каждом процессоре выполняется одна и та же команда, но обрабатываются различные данные. Реализуется синхронный параллельный вычислительный процесс.

Одним из преимуществ данной архитектуры считается то, что в этом случае более эффективно реализована логика вычислений. До половины логических инструкций обычного процессора связано с управлением выполнением машинных команд, а остальная их часть относится к работе с внутренней памятью процессора и выполнению арифметических операций. В SIMD компьютере управление выполняется контроллером, а "арифметика" отдана процессорным элементам. Возможны два способа построения компьютеров этого класса. Это матричная структура ЭВМ и векторно-конвейерная обработка.

5.6. Матричная архитектура

Суть **матричной структуры** заключается в том, что имеется множество процессорных элементов, исполняющих одну и ту же команду над различными элементами вектора (поток данных), объединенных коммутатором. Каждый процессорный элемент включает схемы местного управления, операционную часть, схемы связи и собственную оперативную память. Изменение производительности матричной системы достигается за счет изменения числа процессорных элементов.

Основные их преимущества - высокая производительность и экономичность. Недостатки матричных систем, ограничивающие области их применения, заключаются в жесткости синхронного управления матрицей процессорных элементов и сложности программирования обмена данными между процессорными элементами через коммутатор.

Они применяются главным образом для реализации алгоритмов, допускающих параллельную обработку многих потоков данных по одной и той же программе (одномерное и двумерное прямое и обратное преобразования Фурье, решение систем дифференциальных уравнений в частных производных, операций над векторами и матрицами и др.). Матричные системы довольно часто используются совместно с универсальными однопроцессорными ЭВМ. Примером векторных супер-ЭВМ с матричной структурой является знаменитая в свое время система ILLIAC-IV.

5.7. Векторно-конвейерная архитектура

В отличие от матричной, **векторно-конвейерная структура** компьютера содержит конвейер операций, на котором обрабатываются параллельно элементы векторов и полученные результаты последовательно записываются в единую память. При этом отпадает необходимость в коммутаторе процессорных элементов, служащем камнем преткновения в матричных компьютерах.

Векторно-конвейерную структуру имеют однопроцессорные супер-ЭВМ серии VP фирмы Fujitsu; серии S компании Hitachi; C90, M90, T90 фирмы Cray Research; Cray-3, Cray-4 фирмы Cray Computer и т.д. Общим для всех векторных суперкомпьютеров является нали-

чие в системе команд векторных операций, допускающих работу с векторами определенной длины, допустим, 64 элемента по 8 байт. В таких компьютерах операции с векторами обычно выполняются над векторными регистрами.

5.8. MISD компьютеры

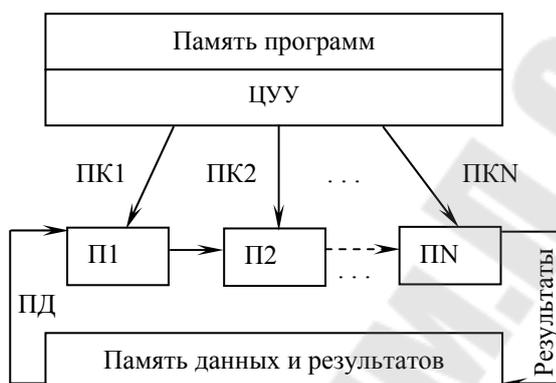


Рисунок 7 - MISD-архитектура

MISD (Multiple Instruction Stream - Single Data Stream) или МКОД - множество потоков команд и один поток данных. MISD компьютеры представляет собой, как правило, регулярную структуру в виде цепочки последовательно соединенных процессоров П1, П2, ..., ПN, образующих **процессорный конвейер** (рис. 7). В такой системе реализуется принцип конвейерной (магистральной) обработки, который основан на разбиении всего процесса на последовательно выполняемые этапы, причем каждый этап выполняется на отдельном процессоре. Одианный поток исходных данных для решения задачи поступает на вход процессорного конвейера. Каждый процессор решает свою часть задачи, и результаты решения в качестве исходных данных передает на вход последующего процессора. К каждому процессору подводится свой поток команд, т. е. наблюдается множественный поток команд ПК1, ПК2, ..., ПКN.

Вычислительных машин такого класса практически нет и трудно привести пример их успешной реализации. Один из немногих - систолический массив процессоров, в котором процессоры находятся в узлах регулярной решетки, роль ребер которой играют межпроцессорные соединения. Все процессорные элементы управляются общим тактовым генератором. В каждом цикле работы каждый процессор-

ный элемент получает данные от своих соседей, выполняет одну команду и передает результат соседям.

В дальнейшем для MISD нашлась ещё одна адекватная организация вычислительной системы - распределенная мультипроцессорная система с общими данными. Наиболее простая и самая распространенная система этого класса - обычная локальная сеть персональных компьютеров, работающая с единой базой данных, когда много процессоров обрабатывают один поток данных. Впрочем, тут есть одна тонкость. Как только в такой сети все пользователи переключаются на обработку собственных данных, недоступных для других абонентов сети, MISD - система превращается в систему с множеством потоков команд и множеством потоков данных, соответствующую MIMD-архитектуре.

5.9. MIMD компьютеры

MIMD (Multiple Instruction Stream - Multiple Data Stream) или МКМД - множество потоков команд и множество потоков данных. Эта категория архитектур вычислительных машин наиболее богата, если иметь в виду примеры ее успешных реализаций. В неё попадают симметричные параллельные вычислительные системы, рабочие станции с несколькими процессорами, кластеры рабочих станций и т.д. Уже довольно давно появились компьютеры с несколькими независимыми процессорами, но вначале на таких компьютерах был реализован только параллелизм заданий, то есть на разных процессорах одновременно выполнялись разные и независимые программы.

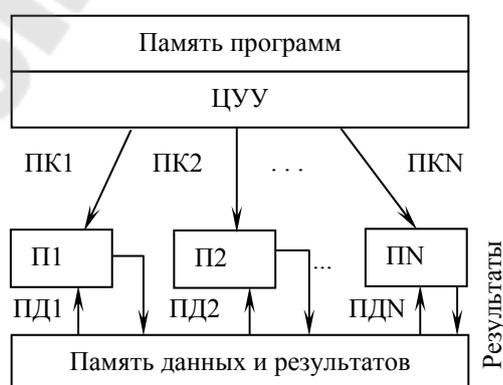


Рисунок 8 - MIMD-архитектура

Так как только MIMD-архитектура включает все уровни параллелизма от конвейера операций до независимых заданий и программ, то любая вычислительная система этого класса в частных приложениях может выступать как SISD и SIMD-система. Например, если многопроцессорный комплекс выполняет одну-единственную программу без каких-либо признаков векторного параллелизма данных, то в этом конкретном случае он функционирует как обычный SISD-компьютер, и весь его потенциал остается невостребованным. Таким образом, употребляя термин «MIMD», надо иметь в виду не только много процессоров, но и множество вычислительных процессов, одновременно выполняемых в системе. MIMD-системы по способу взаимодействия процессоров (рис. 8) делятся на системы с сильной и слабой связью.

5.10. Многопроцессорные вычислительные системы

Сильно-связанные вычислительные системы или многопроцессорные вычислительные системы (МПВС) основаны на объединении процессоров на общем поле оперативной памяти. Это поле называется разделяемой памятью (Shared Memory). Управление обеспечивается одной общей операционной системой. При этом достигаются более быстрый обмен информацией между процессорами, чем между ЭВМ в многомашинных вычислительных системах (комплексах), и более высокая суммарная производительность системы. Иногда их называют «истинными» мультипроцессорами.

В таких системах, как правило, число параллельных процессов невелико и управляет ими централизованная операционная система. Процессы обмениваются информацией через общую оперативную память. При этом возникают задержки из-за межпроцессорных конфликтов. При создании больших мультипроцессорных ЭВМ (мэйнфреймов, суперЭВМ) предпринимаются огромные усилия по увеличению пропускной способности оперативной памяти. В результате аппаратные затраты возрастают чуть ли не в квадратичной зависимости, а производительность системы упорно «не желает» увеличиваться пропорционально числу процессоров. Так, сложнейшие средства снижения межпроцессорных конфликтов в оперативной памяти суперкомпьютеров серии CRAY X-MP/Y-MP позволяют получить коэффициент ускорения не более 3,5 для четырехпроцессорной конфигурации системы.

По топологии межмодульных функциональных и управляющих связей и организации работы выделяются два типа многопроцессорных систем МКМД

- с общей шиной;
- с использованием многовходовой памяти (многошинные-многовходовые вычислительные системы).

5.11. Многопроцессорные вычислительные системы с общей шиной

В МПВС с общей шиной (Shared Memory Processing – мультипроцессоры с разделением памяти, SMP-архитектура) все функциональные модули (процессоры П1, П2, ..., ПМ, модули памяти МП1, МП2, МПК, устройства ввода-вывода УВВ1, УВВ2, УВВМ) подсоединены к одной общей шине межмодульных связей, ширина которой может быть от одного бита до нескольких байтов. Между модулями системы нет постоянных связей, информация между ними передается в режиме разделения времени. В каждый данный момент времени по шине передается лишь один пакет информации от какого-то одного источника. Другие источники информации должны ожидать, пока не освободится системная шина.

Основные преимущества системы с общей шиной межмодульных связей:

- система характеризуется относительно низкой функциональной сложностью и невысокой стоимостью;
- в системе легко осуществляется реконфигурация структуры путем добавления или удаления функциональных модулей.
- Недостатки таких систем:
 - ограничение производительности системы пропускной способностью общей шины;
 - ухудшение общей производительности системы при ее расширении путем добавления модулей;
 - потери производительности системы, связанные с разрешением конфликтов, которые возникают в случае, когда несколько модулей одновременно претендуют на занятие общей шины для передачи информации. Эти потери можно снизить, если, кроме общей памяти, доступной всем процессорам, каждый из них снабдить местной (локальной, процессорной) памятью для хранения информации, участвующей в ближайшем ряду вычислений. При наличии местной памяти частота

обращения процессоров к общей памяти уменьшается, следовательно, уменьшается частота возникновения конфликтов и потери времени на их разрешение;

- отказ общей шины приводит к выходу из строя всей системы.

- Организация связей между элементами системы на основе общей шины является одним из распространенных способов построения не только многопроцессорных, но и многомашинных вычислительных комплексов небольшой мощности.

5.12. Многопроцессорные вычислительные системы с многовходовыми модулями ОП

В МПВС с многовходовыми модулями ОП или симметричных МПВС взаимные соединения выполняются с помощью индивидуальных шин, подключающих каждый процессор и каждое устройство ввода-вывода к отдельному входу оперативной памяти. Для этого необходимо, чтобы модули ОП имели по несколько входов и снабжались управляющими схемами для разрешения конфликтов в случаях, когда два или более процессора или устройства ввода-вывода требуют доступа к одному и тому же модулю памяти в пределах одного временного цикла. Число подключаемых элементов системы к одному модулю памяти ограничивается числом его входов.

При построении общего поля ОП МПВС целесообразной оказывается реализация метода расслоения оперативной памяти, при которой ячейки с соседними адресами оказываются расположенными в соседних модулях. Обязательными условиями применения этого метода являются модульность структуры ОП и наличие для каждого модуля памяти автономного блока управления памятью.

Преимущества МПВС с многовходовыми модулями ОП:

- скорость передачи информации значительно выше, чем в МПВС с общей шиной;

- система может работать и в режиме однопроцессорной конфигурации.

- Недостатки таких систем следующие:

- большое число линий связи и разъемов, усложняющих конструкцию системы и снижающих ее надежность;

- оперативная память, составленная из многовходовых модулей, является дорогостоящей.

Принципы построения МПВС с многовходовыми модулями ОП используются в мэйнфреймах

5.13. Многомашинные вычислительные системы (ММВС)

В системах типа МКМД реализуется асинхронный вычислительный процесс, при котором каждый процессор системы выполняет свою программу (или свой участок сложной программы) с собственными данными. В таких системах происходит постоянное распараллеливание вычислений. Две основных причины создания этого типа ВС – дублирование важных блоков вычислений или модулей ВС и повышение производительности систем.

Вычислительные системы со слабой связью или распределенные вычислительные системы, как правило, представляются многомашинными комплексами, в которых отдельные компьютеры объединяются либо с помощью сетевых средств, либо с помощью общей внешней памяти (обычно — дисковые накопители большой емкости). Каждая ЭВМ системы имеет свою оперативную память и работает под управлением своей операционной системы. Каждая машина использует другую как канал или устройство ввода-вывода. Обмен информацией между машинами происходит в результате взаимодействия их операционных систем.

ММВС строится из логически независимых компонентов: процессоров, устройств оперативной памяти, каналов ввода-вывода, ВЗУ, устройств управления ВЗУ, устройств ввода-вывода, устройств управления УВВ. Логическая независимость процессоров системы определяется возможностью их независимого функционирования. Для остальных компонентов эта независимость определяется возможностью их подсоединения к одному или к нескольким процессорам ММВС.

Связь между машинами (процессорами) ММВС может осуществляться на уровне любого из его логически независимых компонентов с помощью специальных мультисистемных средств или средств комплексирования. Такая связь должна быть достаточно гибкой и обеспечивать независимость функционирования различных модулей системы и их взаимодействие с различной скоростью, соответствующей скорости обмена информацией между элементами системы. Для этого на разных уровнях комплексирования применяются различные по тактовой частоте, разрядности, пропускной способности шины интерфейсов.

Очевидно, что система со всеми возможными уровнями связей будет наиболее совершенной, гибкой и надежной в функционировании. Но с другой стороны, система с полными связями получается сложной по своей структуре и организации функционирования. В каждом конкретном многомашинном комплексе не обязательно реализуются все уровни комплексирования.

Рассмотрим уровни связи вычислительной системы в порядке возрастания скорости обмена информацией.

Межмашинная связь на уровне внешних устройств используется главным образом для организации общего поля внешней памяти. Такая связь организуется через каналы ввода-вывода этих устройств и шинные интерфейсы. При этом обычно часть ВЗУ остается в индивидуальном пользовании отдельных машин. Преимуществом комплексирования на уровне ВЗУ является значительное увеличение объемов информации (данных и программ), одновременно доступных процессорам ММВС.

Наибольшее распространение получила взаимодействие вычислительных средств на уровне канал-канал через адаптер канал-канал. Адаптер подключается к двум каналам, причем функционально он рассматривается как устройство управления ввода-вывода для каждого из каналов, а каждая из связанных адаптером машин по отношению друг к другу является внешним устройством. В отличие от любого другого устройства управления внешними устройствами адаптер не управляет устройствами ввода-вывода, а только осуществляет связь между каналами и синхронизирует их работу. Адаптер обеспечивает быстрый обмен информацией между каналами, а следовательно, и между ОП взаимодействующих процессоров, если общее поле оперативной памяти не организовано.

Взаимодействие на уровне ОП осуществляется для создания общего поля оперативной памяти, что значительно ускоряет обмен информацией между процессорами и повышает возможности функционирования системы. Наибольшая оперативность обмена информацией достигается при реализации именно такого уровня связи. Построение ММВС с общим полем ОП связано с необходимостью применения многоходовых модулей памяти. При этом существенно усложняются структура и функции устройства управления общей оперативной памятью. С помощью УУ должны решаться такие задачи, как реализация установленной очередности обращения к модулям памяти со стороны процессоров, решение конфликтных ситуаций, воз-

никающих при одновременном обращении к одному и тому же модулю памяти со стороны обоих процессоров, поддержание когерентности памяти. Наибольшие затруднения связаны с созданием программных средств, обеспечивающих функционирование системы с общим полем ОП.

Взаимодействие на уровне процессоров через интерфейс межпроцессорной связи осуществляется с целью синхронизации единого вычислительного процесса путем передачи между процессорами сигналов внешних прерываний и команд прямого управления.

Для оценки эффективности взаимодействия вычислительных средств системы на различных уровнях могут привлекаться такие показатели эффективности ВС, как время реакции системы на запросы с учетом их приоритетов, пропускная способность, время на решение заданного набора задач.

5.14. ММР архитектура

Другим примером реализации слабосвязанной архитектуры являются системы с массовым параллелизмом (МРР), состоящие из десятков, сотен, а иногда и тысяч процессорных узлов. Строгой границы не существует, однако считается, что при числе процессоров 128 и более система относится к МРР-архитектуре. Большинство МРР-систем имеют как логически так и физически распределенную между процессорами память. Каждый узел такой системы содержит процессор и модуль памяти, в котором хранится процесс - совокупность команд, исходных и промежуточных данных вычислений, а также системные идентификаторы процесса. Узлы массово-параллельной системы объединяются коммутационными сетями самой различной формы - от простейшей двумерной решетки до гиперкуба или трехмерного тора. В отличие от архитектуры фон Неймана, передача данных между узлами коммутационной сети происходит по готовности данных процесса, а не под управлением некоторой программы. Отсюда еще одно название подобных систем - «системы с управлением потоком данных» (иногда просто «потокосые машины»).

К достоинствам данной архитектуры относится то, что она использует стандартные микропроцессоры и обладает неограниченным быстродействием (порядка TFLOPS).

Однако есть и недостатки — программирование коммутаций процессов является слабо автоматизированной и очень сложной процедурой. Так что для коммерческих задач и даже для подавляющего большинства инженерных приложений системы с массовым параллелизмом недоступны.

Многомашинные вычислительные системы создаются и на базе мини- и микро-ЭВМ.

Сравнивая между собой ММВС и МПВС, можно отметить, что в МПВС достигается более высокая скорость обмена информацией между элементами системы и поэтому более высокая производительность, более высокая реакция на возникающие в системе и ее внешней среде нестандартные ситуации, более высокая надежность и живучесть (МПВС сохраняет работоспособность, пока работоспособны хотя бы по одному модулю каждого типа устройств). С другой стороны, построение ММВС из серийно выпускаемых ЭВМ с их стандартными операционными системами значительно проще, чем построение МПВС, требующих преодоления определенных трудностей, связанных главным образом с организацией общего поля оперативной памяти и созданием единой операционной системы. Разница организации МММД-систем с сильной и слабой связью проявляются ещё и при обработке приложений, отличающихся интенсивностью обменов между взаимодействующими процессами. Вычислительные сети являются дальнейшим развитием вычислительных систем распределенного типа. Они представляют собой новый, более совершенный этап в использовании средств вычислительной техники - переход к коллективному их использованию.

6. СТРУКТУРА И ФОРМАТЫ КОМАНД ЭВМ

Обработка информации в ЭВМ осуществляется путём программного управления.

Программа представляет собой алгоритм обработки информации, записанной в виде последовательности команд, которые должны быть выполнены машиной для получения результата.

Команда ЭВМ представляет собой код, определяющий операцию вычислительной машины и данные, участвующие в операции. В явной и неявной форме команда содержит также информацию об адресе, по которому помещается результат операции, и об адресе следующей команды.

По характеру выполняемых операций можно выделить следующие группы команд:

- команды арифметических операций для чисел с фиксированной или плавающей запятой;
- команды десятичной арифметики;
- команды передачи данных (MOV AX, 0FFFh);
- команды операций ввода/вывода (IN, OUT);
- команды логических операций (AND, OR, NOT);
- команды передачи управления (управление циклом – LOOP, условные переходы – JAE, JB);
- команды задания режима работы машины и др.

6.1. Форматы команд ЭВМ

В команде, как правило, содержатся не сами операнды, а информация об объектах адресах ячеек памяти или регистрах, в которых они находятся. Код команды можно представить состоящим из нескольких полей, каждое из которых имеет свое функциональное назначение.

В общем случае команда состоит из:

- операционной части (содержит код операции);
- адресной части (содержит адресную информацию о местонахождении обрабатываемых данных и месте хранения результатов).

В свою очередь, эти части, что особенно характерно для адресной части, могут состоять из нескольких полей.

Структура команды определяется составом, назначением и расположением полей в коде.

Форматом команды называется заранее оговоренная структура полей ее кода с разметкой номеров разрядов (бит), определяющих границы отдельных полей команды, или с указанием числа разрядов (бит) в определенных полях, позволяющая ЭВМ распознавать составные части кода.

Пример формата команды процессора i486.

- mod r/m - спецификатор режима адресации;
- r/m - регистр памяти;
- SS - масштабный множитель для режима масштабирования индексной адресации;

КОП	Байты адресации			Смещение	Операнд
	mod r/m	sib			
		SS	index		
1 или 2 байта	0 или 1 байт			0,1,2, или 4 байта	0,1,2, или 4 байта

- КОП - код операции;
index - определяет индексный регистр;
base - определяет базовый регистр.

Важной и сложной проблемой при проектировании ЭВМ является выбор структуры и форматов команды, т.е. ее длины, назначения и размерности отдельных ее полей. Естественно стремление разместить в команде в возможно более полной форме информацию о предписываемой командой операции. Однако в условиях, когда в современных ЭВМ значительно возросло число выполняемых различных операций и соответственно команд (в компьютерах с CISC-архитектурой более 200 команд) и значительно увеличилась емкость адресуемой основной памяти (32, 64 Мб), это приводит к недопустимо большой длине формата команды.

Вместе с тем, для упрощения аппаратуры и повышения быстродействия ЭВМ длина формата команды должна быть по возможности короче, укладываться в машинное слово или полуслово. Решение проблемы выбора формата команды значительно усложняется в микропроцессорах, работающих с коротким словом.

Проследим изменения классических структур команд.

Чтобы команда содержала в явном виде всю необходимую информацию о задаваемой операции, она должна, как это показано на рис. 9 (б), содержать следующую информацию: A1, A2 - адреса операндов, A3 - адрес результата, A4 - адрес следующей команды (принудительная адресация команд).

Такая структура приводит к большей длине команды и неприемлема для прямой адресации операндов основной памяти. В компьютерах с RISC-архитектурой четырехадресные команды используются для адресации операндов, хранящихся в регистровой памяти процессора.

Можно установить, как это принято для большинства машин, что после выполнения данной команды, расположенной по адресу K (и занимающей L ячеек), выполняется команда из (K+L)-ой ячейки. Такой порядок выборки команды называется естественным. Он наруша-

ется только специальными командами (передачи управления). В таком случае отпадает необходимость указывать в команде в явном виде адрес следующей команды.

В трехадресной команде (рис. 9, в) первый и второй адреса указывают ячейки памяти, в которых расположены операнды, а третий определяет ячейку, в которую помещается результат операции.

Можно условиться, что результат операции всегда помещается на место одного из операндов, например первого. Получим двухадресную команду (рис. 9, г), т.е. для результата используется подразумеваемый адрес.

В одноадресной команде (рис. 9, д) подразумеваемые адреса имеют уже и результат операции и один из операндов. Один из операндов указывается адресом в команде, в качестве второго используется содержимое регистра процессора, называемого в этом случае регистром результата или аккумулятором. Результат операции записывается в тот же регистр.

Наконец, в некоторых случаях возможно использование безадресных команд (рис. 9, е), когда подразумеваются адреса обоих операндов и результата операции, например, при работе со стековой памятью.

С точки зрения программиста, наиболее естественны и удобны трехадресные команды. Обычно в ЭВМ используется несколько структур и форматов команд разной длины. Приведенные на рис. 9. структуры команд достаточно схематичны. В действительности адресные поля команд большей частью содержат не сами адреса, а только информацию, позволяющую определить действительные (исполнительные) адреса операндов в соответствии с используемыми в командах способами адресации

1.1. Способы адресации

Существует два различных принципа поиска операндов в памяти: ассоциативный и адресный.

Ассоциативный поиск операнда (поиск по содержанию ячейки) предполагает просмотр содержимого всех ячеек памяти для выявления кодов, содержащих заданный командой ассоциативный признак. Эти коды и выбираются из памяти в качестве искомых операндов.

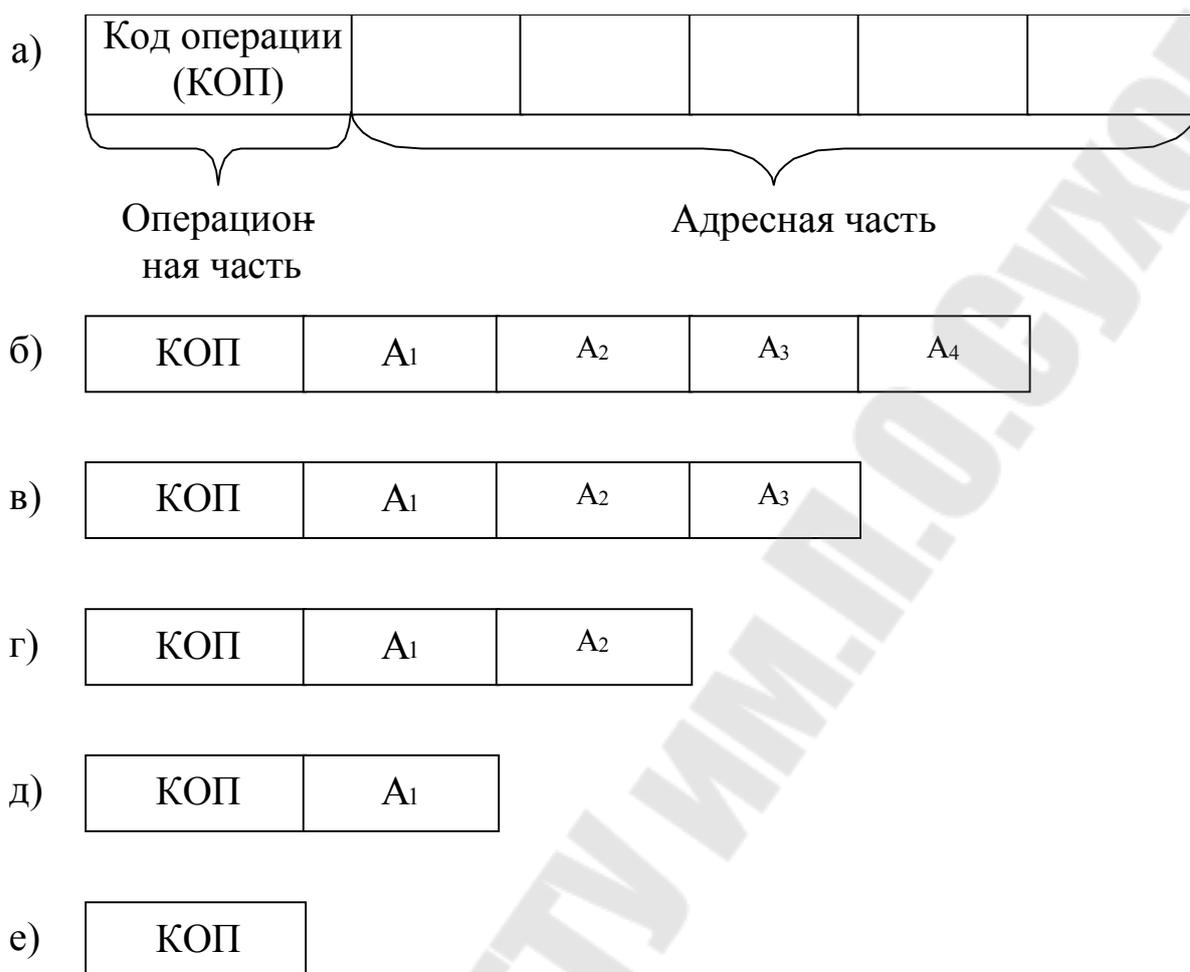


Рисунок 9 - Структуры команд: а) обобщенная, б) четырехадресная, в) трехадресная, г) двухадресная, д) одноадресная, е) безадресная

Адресный поиск предполагает, что искомый операнд извлекается из ячейки, номер которой формируется на основе информации в адресном поле команды. Ниже мы будем рассматривать только реализацию адресного принципа поиска операнда.

Следует различать понятия исполнительного адреса и адресного кода.

Адресный код – это информация об адресе операнда, содержащегося в команде.

Исполнительный адрес – это номер ячейки памяти, к которой фактически производится обращение.

В современных ЭВМ адресный код, как правило, не совпадает с исполнительным адресом. Таким образом, способ адресации можно

определить как способ формирования исполнительного адреса операнда A_i по адресному коду команды A_K .

В системах команд современных ЭВМ часто предусматривается возможность использования нескольких способов адресации операндов для одной и той же операции. Для указания способа адресации в некоторых системах команд выделяется специальное поле в команде - «метод» (указатель адресации). В этом случае любая операция может выполняться с любым способом адресации, что значительно упрощает программирование.

Адресуемые в командах операнды хранятся в основной памяти (ОП) и регистровой памяти (РП), рис. 10.

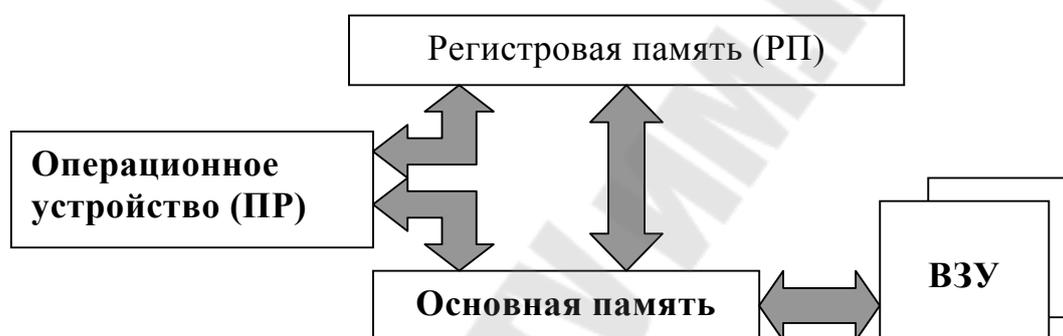


Рисунок 10 - Памяти для хранения адресуемых операндов

1.2. Классификация способов адресации по наличию адресной информации в команде

По наличию адресной информации в команде различают явную и неявную адресацию.

При явной адресации операнда в команде есть поле адреса этого операнда, в котором задается адресный код A_K . Большинство методов адресации являются явными.

При неявной адресации адресное поле в команде отсутствует, адрес операнда подразумевается кодом операции.

Метод неявной адресации операндов используется во всех процессорах. Основное его назначение - уменьшение длины команды за счет исключения части адресов. При этом методе код операции точно задает адрес операнда. Например, из команды исключается адрес приемника результата. При этом подразумевается, что результат в этой команде помещается на место второго операнда

1.3. Классификация способов адресации по кратности обращения в память

Широко используются следующие методы адресации операнда с различной кратностью обращения (R) в память:

1. Непосредственная ($R = 0$).
2. Прямая ($R = 1$).
3. Косвенная ($R > 2$).

Непосредственная адресация операнда. При этом способе операнд располагается в адресном поле команды. Обращение к регистровой памяти (РП) или оперативной памяти (ОП) не производится. Таким образом, уменьшается время выполнения операции, сокращается объем памяти. Непосредственная адресация удобна для задания констант, длина которых меньше или равна длине адресного поля команды.

Прямая адресация операндов. При этом способе (рис. 11) адресации обращение за операндом в РП или ОП производится по адресу коду в поле команды, т.е. исполнительный адрес операнда совпадает с адресным кодом команды ($A_{И} = A_{К}$).

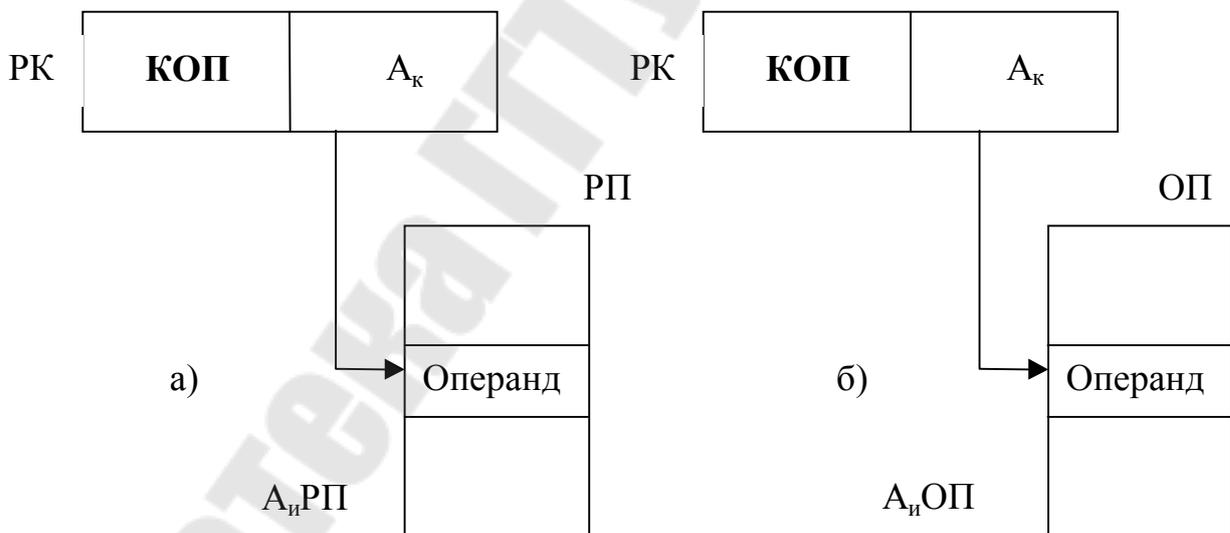


Рисунок 11 - Схема прямой адресаций

Обеспечивая простоту программирования, этот метод имеет существенные недостатки, так как для адресации к ячейкам памяти большой емкости (число адресов M велико) требуется «длинное» адресное поле в команде. Прямая адресация используется широко в со-

четании с другими способами адресации. В частности, вся адресация к «малой» регистровой памяти ведется только с помощью прямой адресации.

Косвенная адресация операндов. При этом способе адресный код команды указывает адрес ячейки памяти, в которой находится не сам операнд, а лишь адрес операнда, называемый указателем операнда. Адресация к операнду через цепочку указателей (косвенных адресов) называется косвенной.

Адрес указателя, задаваемый программой, остается неизменным, а косвенный адрес может изменяться в процессе выполнения программы. Косвенная адресация, таким образом, обеспечивает переадресацию данных, т.е. упрощает обработку массивов и списковых структур данных, упрощает передачу параметров подпрограммам, но не обеспечивает перемещаемость программ в памяти (рис. 12).

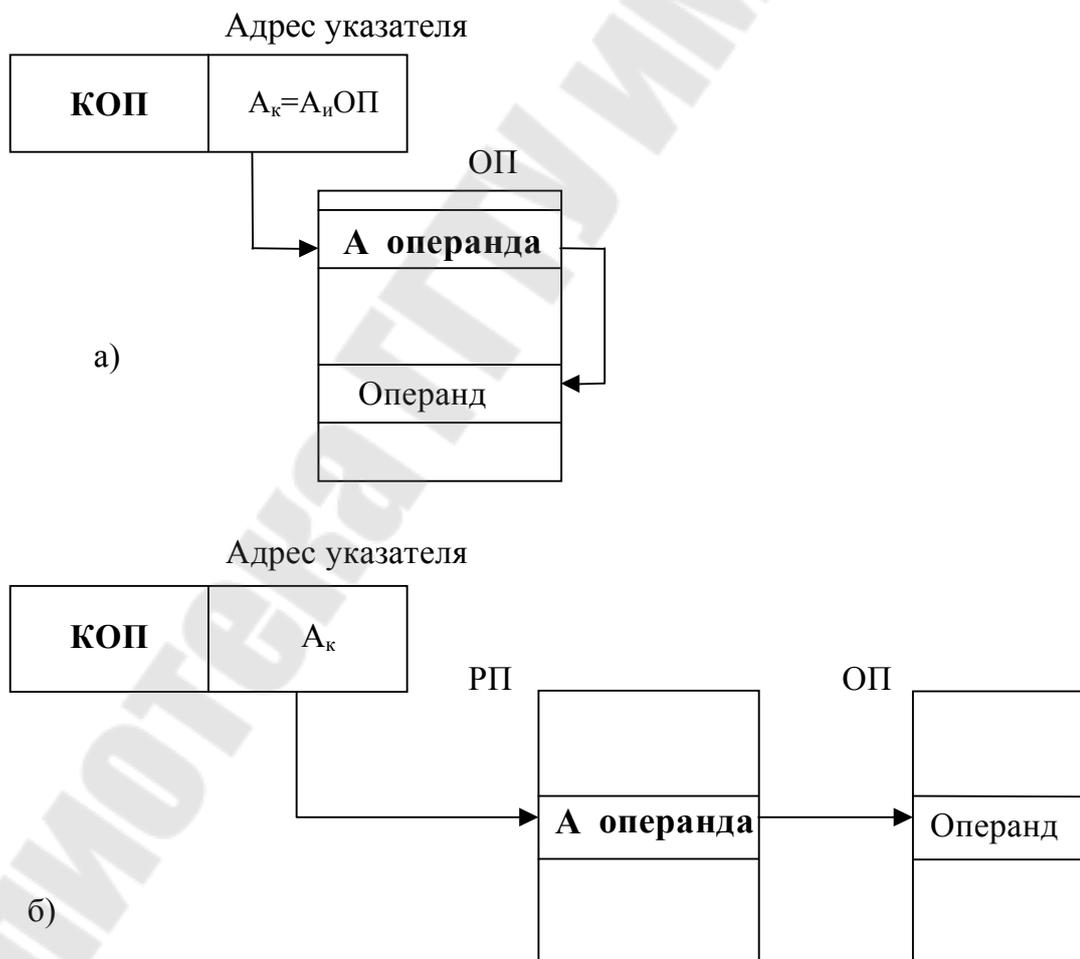


Рисунок 12 - Косвенная адресация

1.4. Классификация по способу формирования исполнительных адресов ячеек памяти

Способы формирования адресов ячеек памяти ($A_{И}$) можно разделить на абсолютные и относительные.

Абсолютные способы формирования предполагают, что двоичный код адреса ячейки памяти - $A_{И}$ может быть извлечен целиком либо из адресного поля команды (в случае прямой адресации), или из какой-либо другой ячейки (в случае косвенной адресации), никаких преобразований кода адреса не производится.

Относительные способы формирования $A_{И}$ предполагают, что двоичный код адреса ячейки памяти образуется из нескольких составляющих: B - код базы, I - код индекса, C - код смещения, используемых в сочетаниях (B и C), (I и C), (B , I и C). При относительной адресации применяются два способа вычисления адреса $A_{И}$:

- суммирование кодов составляющих адреса
($A_{И} = B + C$; $A_{И} = I + C$; $A_{И} = B + I + C$);
- совмещение (конкатенация) кодов составляющих адреса
(например $A_{И} = B/C$).

Базирование способом суммирования. В команде адресный код $A_{К}$ разделяется на две составляющие: $A_{Б}$ - адрес регистра в регистровой памяти, в котором хранится база B (базовый адрес); C - код смещения относительно базового адреса (рис. 3.5).

С помощью метода относительной адресации удается получить так называемый перемещаемый программный модуль, который одинаково выполняется процессором независимо от адресов, в которых он расположен. Начальный адрес программного модуля (база) загружается, при входе в модуль, в базовый регистр. Все остальные адреса программного модуля формируются через смещение относительно начального адреса (базы) модуля. Таким образом, одна и та же программа может работать с данными, расположенными в любой области памяти, без перемещения данных и без изменения текста программы только за счет изменения содержания всего одного базового регистра. Однако время выполнения каждой операции при этом возрастает.

Базирование способом совмещения составляющих. Для увеличения емкости адресной ОП без увеличения длины адресного поля команды можно использовать для формирования исполнительного адреса совмещение (конкатенацию) кодов базы и смещения (рис. 13).

Однако в данном случае начальные адреса массивов не могут быть реализованы произвольно, а должны иметь в младших разрядах n нулей, где n – длина поля смещения.

Индексная адресация. Для работы программ с массивами, требующими однотипных операций над элементами массива, удобно использовать индексную адресацию. Схема индексной адресации аналогична базированию путем суммирования (см. рис. 14).

В этом случае адрес i -го операнда в массиве определяется как сумма начального адреса массива (задаваемого полем смещения C) и индекса I , записанного в одном из регистров РП, называемом теперь индексным регистром. Адрес индексного регистра задается в команде полем адреса индекса — $A_{ин}$ (аналогично A_B).

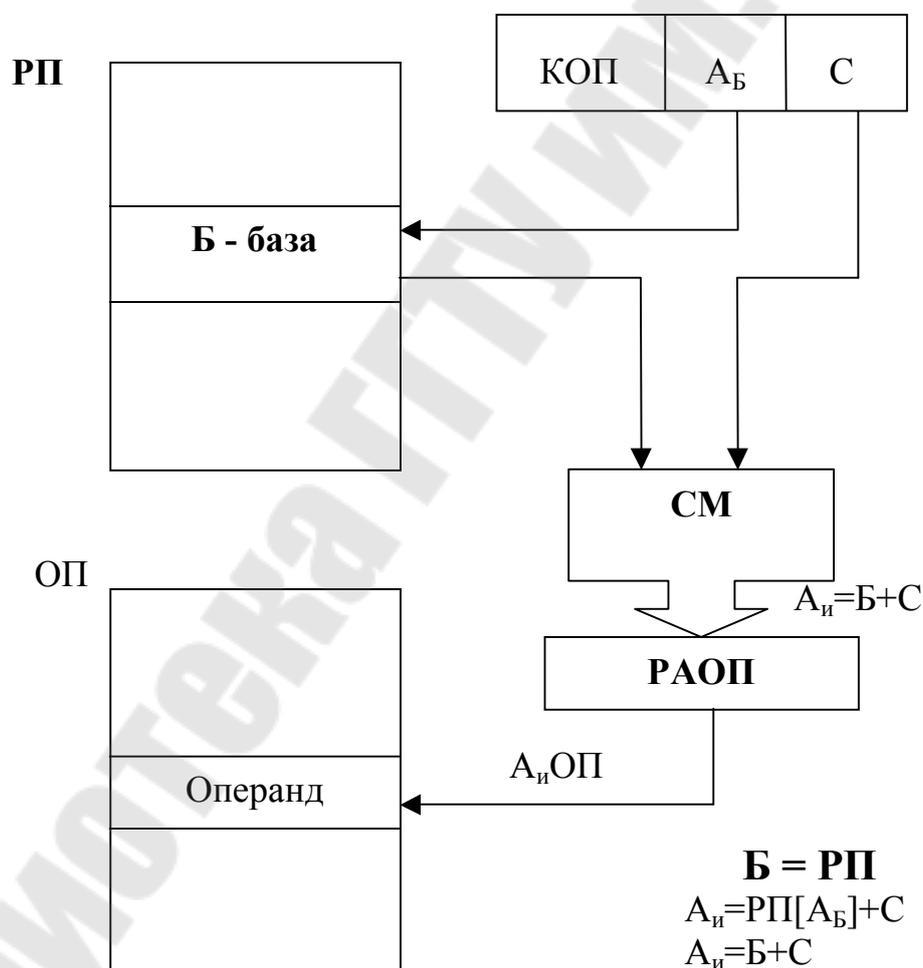


Рисунок 13 - Схема формирования относительного адреса способом суммирования кодов базы и смещения, где **СМ** – сумматор, **РАОП** – ре-

гистр адреса ОП, Б – база (базовый адрес), С – смещение, A_B – адрес регистра базы

В каждом i -м цикле содержимое индексного регистра изменяется на величину постоянную (часто равную 1). Использование индексной адресации значительно упрощает программирование циклических алгоритмов.

Для эффективной работы при относительной адресации применяется комбинированная индексация с базированием, при которой адрес операнда вычисляется как сумма трех величин (рис. 15):

$$A_{И} = Б + И + С.$$

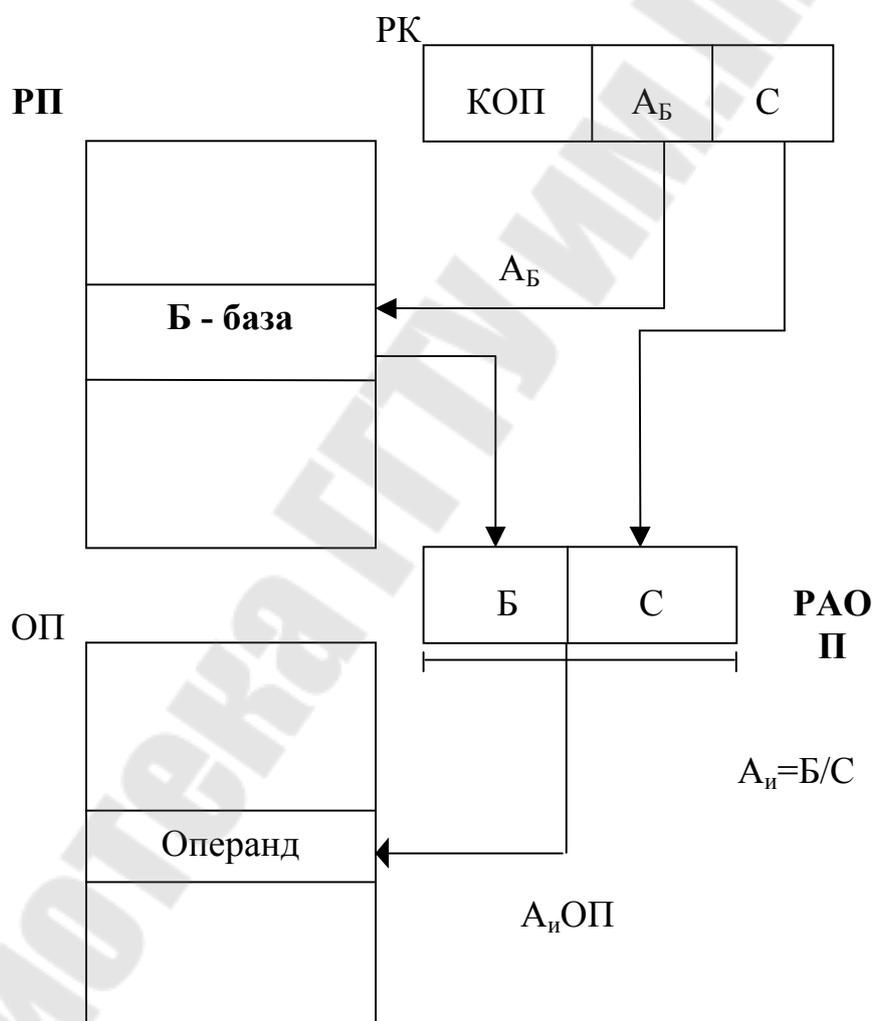


Рисунок 14 - Схема формирования относительного адреса способом совмещения кодов базы и смещения.

Стековая память (стек) является эффективным элементом современных ЭВМ, реализует неявное задание адреса операнда. Хотя

Данные без знака

На рис. 17 приведены 3 формата данных без знака с фиксированной точкой.

Диапазон представления целых чисел: от -2^{64} до 2^{64} .

Данные в формате с плавающей точкой

Формат включает три поля: знака, мантиссы и порядка (рис. 18). Поле мантиссы содержит значащие биты числа, а поле порядка содержит степень 2 и определяет масштабирующий множитель для мантиссы. Поддерживаются блоком FPU.

Диапазон представления:

нормализованных чисел с одинарной точностью: от $\pm 2.9 \cdot 10^{-38}$ до $\pm 1.7 \cdot 10^{38}$;

нормализованных чисел с двойной точностью: от $\pm 2,23 \cdot 10^{-308}$ до $\pm 1,79 \cdot 10^{308}$;

нормализованных чисел с расширенной точностью: от $\pm 3,37 \cdot 10^{4932}$ до $\pm 1,18 \cdot 10^{4932}$.

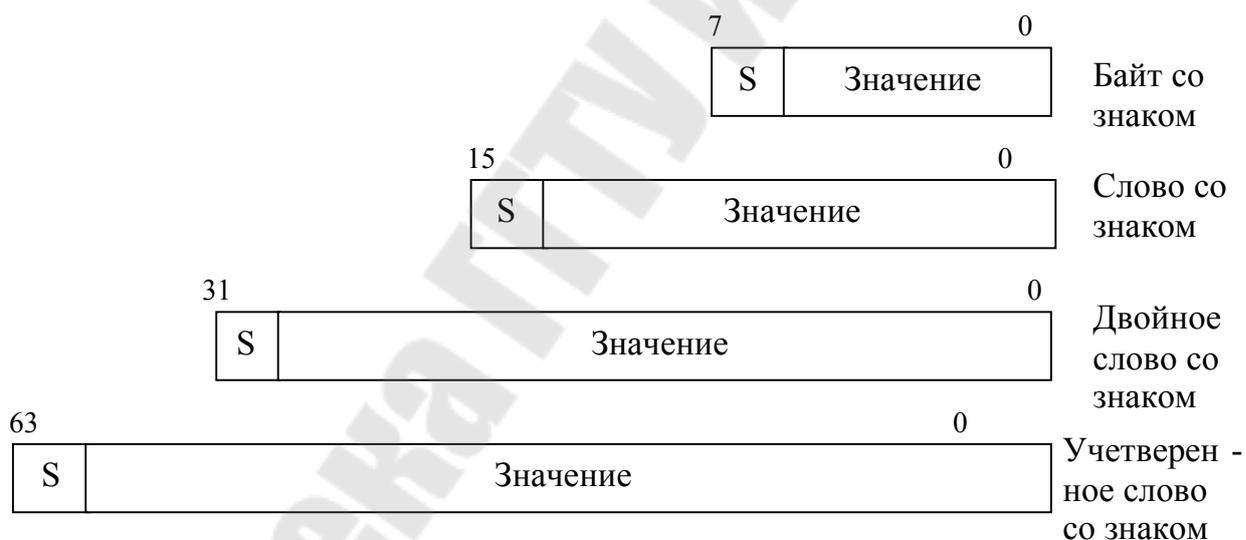


Рисунок 16 - Данные со знаком

Двоично-десятичные данные

Неупакованное BCD – одна цифра (биты 0÷3 – цифра, остальные - игнорируются). *Упакованное BCD* – две цифры (по 4 бита на цифру). 80-разрядное упакованное BCD в блоке FPU (биты 0÷71 – цифры, остальное игнорируется) см. рис. 19.



Рисунок 17 - Данные без знака



Рисунок 18 - Данные с плавающей точкой

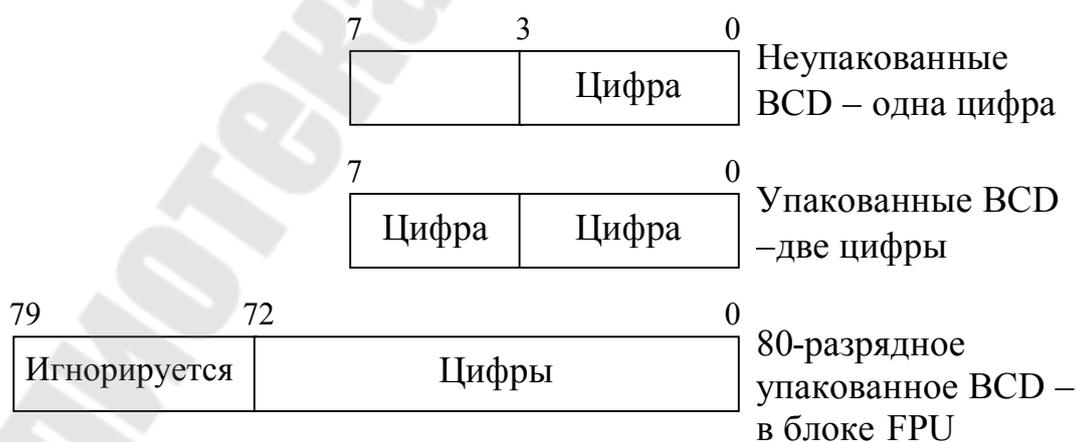


Рисунок 19 – Форматы двоично-десятичных данных

Данные типа строка

Строка представляет собой непрерывную последовательность бит, байт, слов или двойных слов (рис. 4.5).

Строка бит может быть длиной до 1 Гбита, а длина остальных строк может составлять от 1 байта до 4 Гбайтов. Поддерживается АЛУ.

Символьные данные

Поддерживаются строки символов в коде ASCII и арифметические операции (сложение, умножение) над ними (рис. 21). На каждый символ ASCII отводится по восемь бит. Поддержка осуществляется блоком АЛУ.

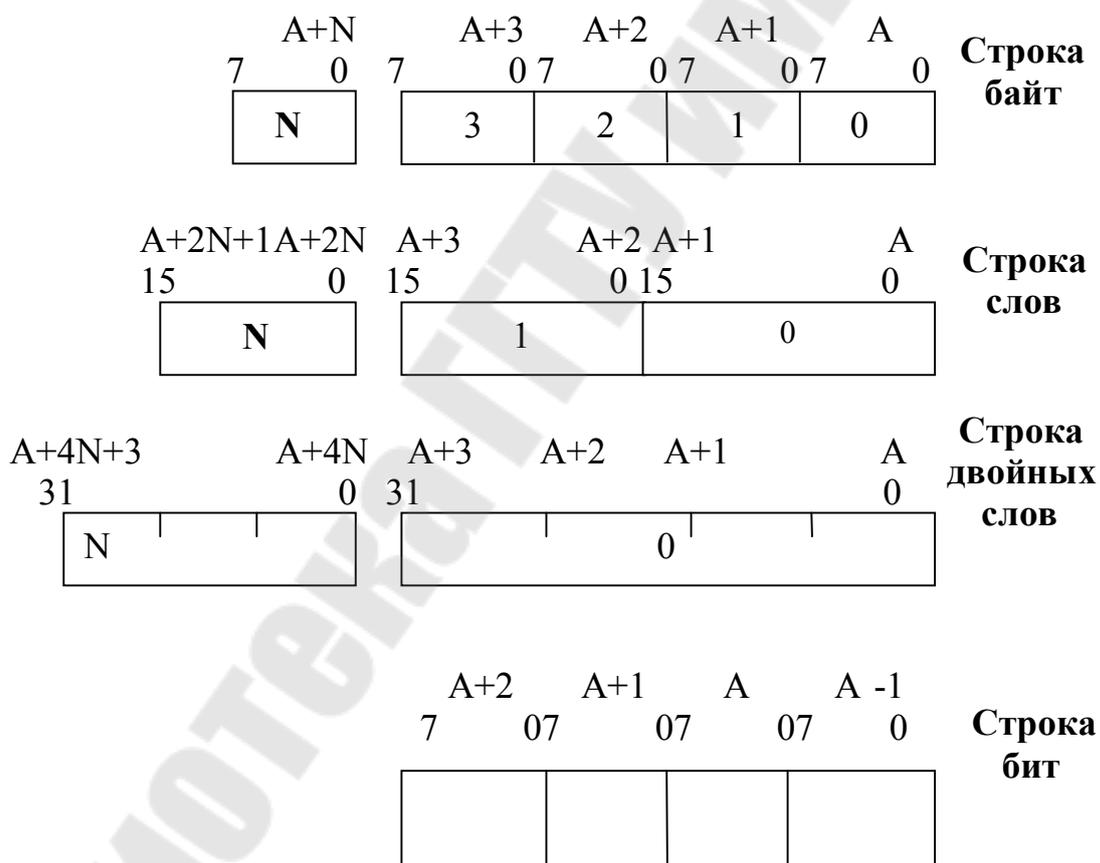


Рисунок 20 – Данные типа строка

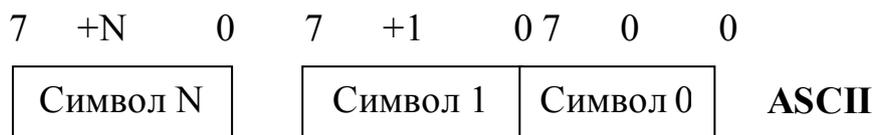


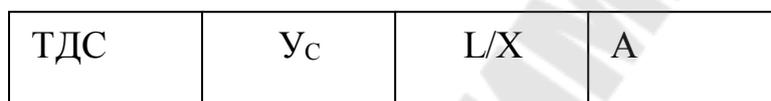
Рисунок 21 - Символьные данные

Данные типа указатель

Указатель содержит величину, которая определяет адрес фрагмента данных. Поддерживается два типа указателей, приведенных на рис. 22.



а)



б)

Рисунок 22 - Структура описания данных: а) с теговой организацией памяти; б) дескриптор данных

длинный указатель (дальний) – смещение (32 разряда) + селектор (16 разрядов);

короткий указатель (ближний) – смещение (32 разряда).

Дальний указатель применяется в том случае, когда программа передает управление в другой сегмент памяти. Такой указатель с помощью селектора определяет новый сегмент и 32-битное смещение внутри этого сегмента. Ближний указатель – это 32-битное смещение, то есть расстояние в байтах от базы того сегмента, в котором находится нужный операнд.

Теги и дескрипторы. Самоопределяемые данные

Одним из эффективных средств совершенствования архитектуры современных ЭВМ является теговая организация памяти, при которой каждое хранящееся в памяти (или регистре) слово снабжается указателем - тегом (рис. 23,а).

Тег определяет тип данных - целое двоичное число, число с плавающей точкой, десятичное число, адрес, строка символов, дескрип-

тор и т.д. В поле тега обычно указывается не только тип, но и длина (формат) и некоторые другие его параметры. Теги формируются компилятором.

Наличие тегов придает хранящимся в машине данным свойство самоопределяемости, вносящее принципиальные особенности в архитектуру и функционирование ЭВМ.

В обычных ЭВМ, соответствующих классической модели фон Неймана, тип данных - операндов и их формат задаются кодом операции команды, а в ряде случаев размер (формат) определяется следующими полями команды.

Теговая организация памяти, напротив, позволяет достигнуть инвариантности команд относительно типов и форматов операндов, что приводит к значительному сокращению набора команды машины.

Другие преимущества теговой организации памяти ЭВМ:

- упрощается и делается более регулярной структура процессора;
- облегчается работа программиста, в том числе при отладке программ;

- упрощаются компиляторы и сокращаются временные затраты на компиляцию (так как отпадает необходимость выбора типа команды в зависимости от типов данных);

- облегчается обнаружение ошибок, связанных с некорректным заданием типа данных (например, при попытке сложить адрес с числом с плавающей точкой);

- теговая организация памяти способствует реализации принципа независимости программ от данных;

- использование тегов приводит к экономии памяти, так как в программах обычных машин имеется большая информационная избыточность на задание типов и размеров операндов при их использовании несколькими командами.

Недостатки теговой организации памяти - некоторое замедление работы процессора из-за того, что установление соответствия типа команды типу данных, в обычных ЭВМ выполняемое на этапе компиляции, при использовании тегов переносится на этап выполнения программы.

В архитектуре некоторых ЭВМ используются дескрипторы - служебные слова, содержащие описание массивов данных и команд, причем дескрипторы могут употребляться как в машинах с теговой организацией памяти, так и без тегов.

Дескриптор содержит сведения о размере массива данных, его местоположении (в ОП или внешней памяти), адресе начала массива, типе данных, режиме защиты данных (например, запрет записи в ячейки массива) и некоторых других параметрах данных. Отметим, что задание в дескрипторе размера массива позволяет контролировать выход за границу массива при индексации его элементов. На рис. 23 в качестве примера представлен один из видов дескрипторов - дескриптор данных.

Дескриптор содержит специфический тег - ТДс, указывающий, что данное слово является дескриптором определенного вида; Ук - группа указателей; А - адрес начала массива данных; L - длина массива; X - индекс.

Использование в архитектуре ЭВМ дескрипторов подразумевает, что обращение к информации в памяти производится через дескрипторы, которые при этом можно рассматривать как дальнейшее развитие аппарата косвенной адресации.

Адресация информации в памяти может осуществляться с помощью цепочки дескрипторов, при этом реализуется многоступенчатая косвенная адресация. То есть адресная часть дескриптора содержит адрес начала массива других дескрипторов, в адресной части каждого из которых содержится в свою очередь, адрес начала массива дескрипторов данных.

3. ПРОЦЕССОРЫ

В информационных технологиях различают процессоры:

- центральные;
- специализированные;
- ввода/вывода;
- передачи данных;
- коммуникационные.

Центральный процессор (ЦП) – это основное устройство ЭВМ, осуществляющее обработку данных и выполняющее функции управления системой (инициирование ввода/вывода, управление доступом к основной памяти, обработку сигналов, поступающих от различных внешних устройств и от внутренних устройств ЭВМ и др.).

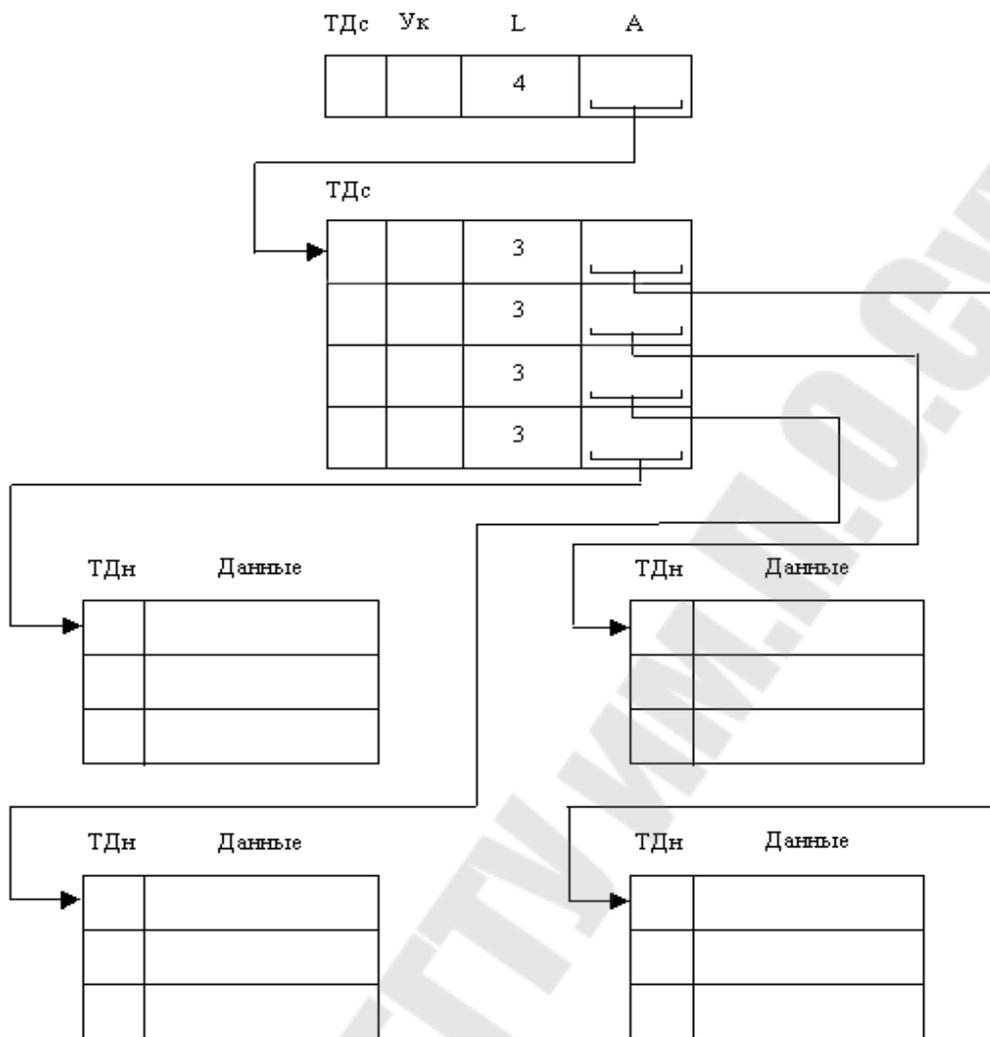


Рисунок 23 - Описание двумерного массива данных древовидной структурой дескрипторов: ТДс и ТДн – теги дескрипторов и данных.

3.1. Логическая структура ЦП

Организация центрального процессора (ЦП) определяется архитектурой и принципами работы ЭВМ (состав и форматы команд, представление чисел, способы адресации, общая организация машины и её основные элементы), а также технико-экономическими показателями.

Логическую структуру ЦП представляет ряд функциональных средств (см. рис. 24):

- средства обработки;
- средства управления системой и программами;
- локальная память;

- средства управления вводом/выводом и памятью;
- системные средства.

Средства обработки обеспечивают выполнение операций с числами с фиксированной точкой, с числами с плавающей точкой, с десятичными данными и с полями переменной длины.

Локальная память состоит из регистров общего назначения, регистров с плавающей точкой, а также управляющих регистров.

Средства управления памятью подразделяются на средства управления доступом к ОП, средства предварительной выборки команд и данных, буферную память и средства защиты памяти.

Средства управления вводом/выводом обеспечивают приоритетный доступ программ к периферийным устройствам через каналы ввода/вывода (или контроллеры).

К **системным средствам** относятся средства службы времени: часы астрономического времени, таймер, коммутатор и т. д.

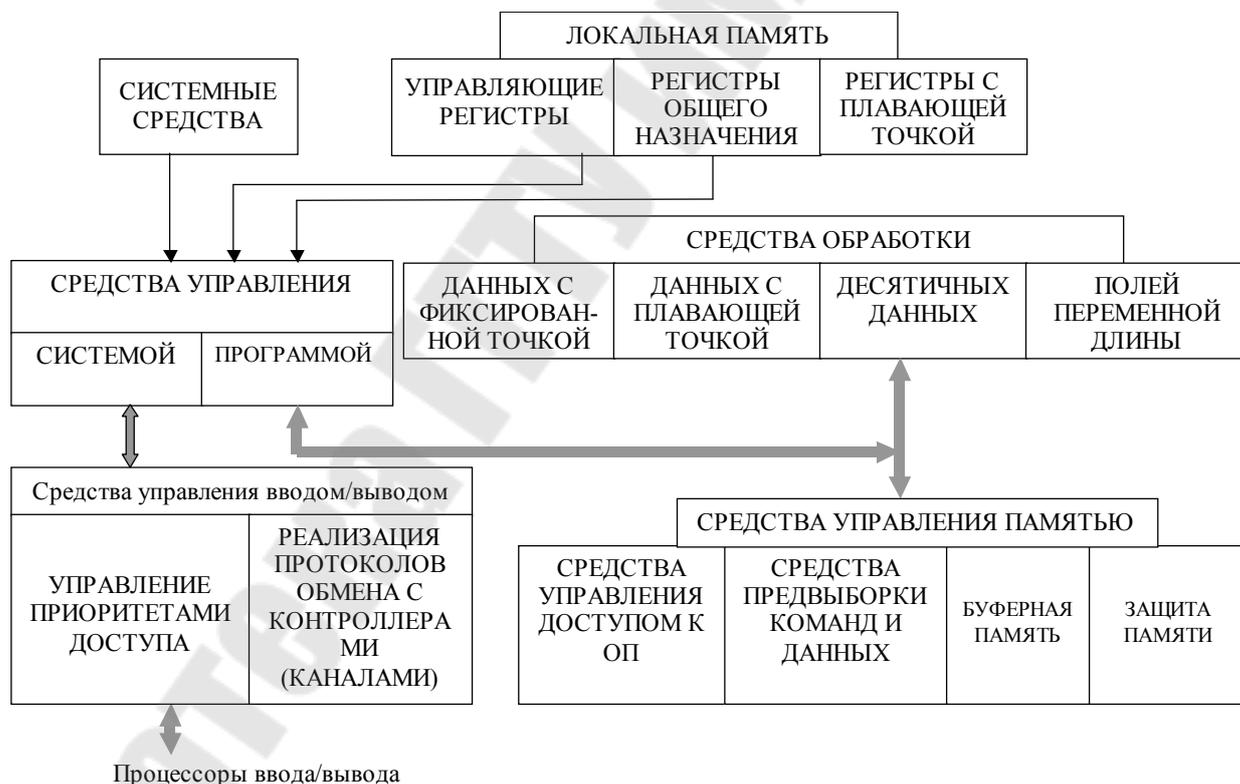


Рисунок 24 - Логическая структура ЦП

Существует обязательный минимальный (стандартный) набор функциональных средств для каждого типа центрального процессора. Он включает в себя:

- регистры общего назначения;
- средства выполнения стандартного набора операций;

с) средства управления вычислительным процессом.

Конкретная реализация ЦП может различаться составом средств, способом их реализации, техническими параметрами.

3.2. Структурная схема процессора

Структурная схема ЦП изображена на рис. 25.

Все функциональные средства по своей структуре разбиваются на следующие устройства:

- Центральное устройство управления;
- Арифметико-логическое устройство;
- Устройство управления памятью;
- Сверхоперативное запоминающее устройство;
- Устройство предварительной выборки команд и данных;
- Интерфейс магистрали.

Центральное устройство управления (ЦУУ) включает дешифратор команд, блок управления и блок прерываний.

Дешифратор команд дешифрирует (декодирует) команды, которые поступают из блока предварительной выборки.

Блок управления (БУ) формирует последовательности управляющих сигналов, которые поступают на все блоки процессора, обеспечивающие выполнение текущей команды и переход к выполнению следующей.

Блок прерывания обеспечивает реакцию ЭВМ на запросы прерываний от различных источников (устройств) внутри и вне ЦП.

Арифметико-логическое устройство (АЛУ) выполняет все арифметические и логические операции ЭВМ. В состав устройства входят:

- сумматоры,
- буферные и рабочие регистры,
- специализированные аппаратные средства (блок ускоренного умножения),
- собственный блок управления (иногда).

Во многих современных процессорах операции с плавающей точкой выполняются в отдельном блоке, который имеет собственные регистры данных, регистры управления и работает параллельно с блоком операций с фиксированной точкой.

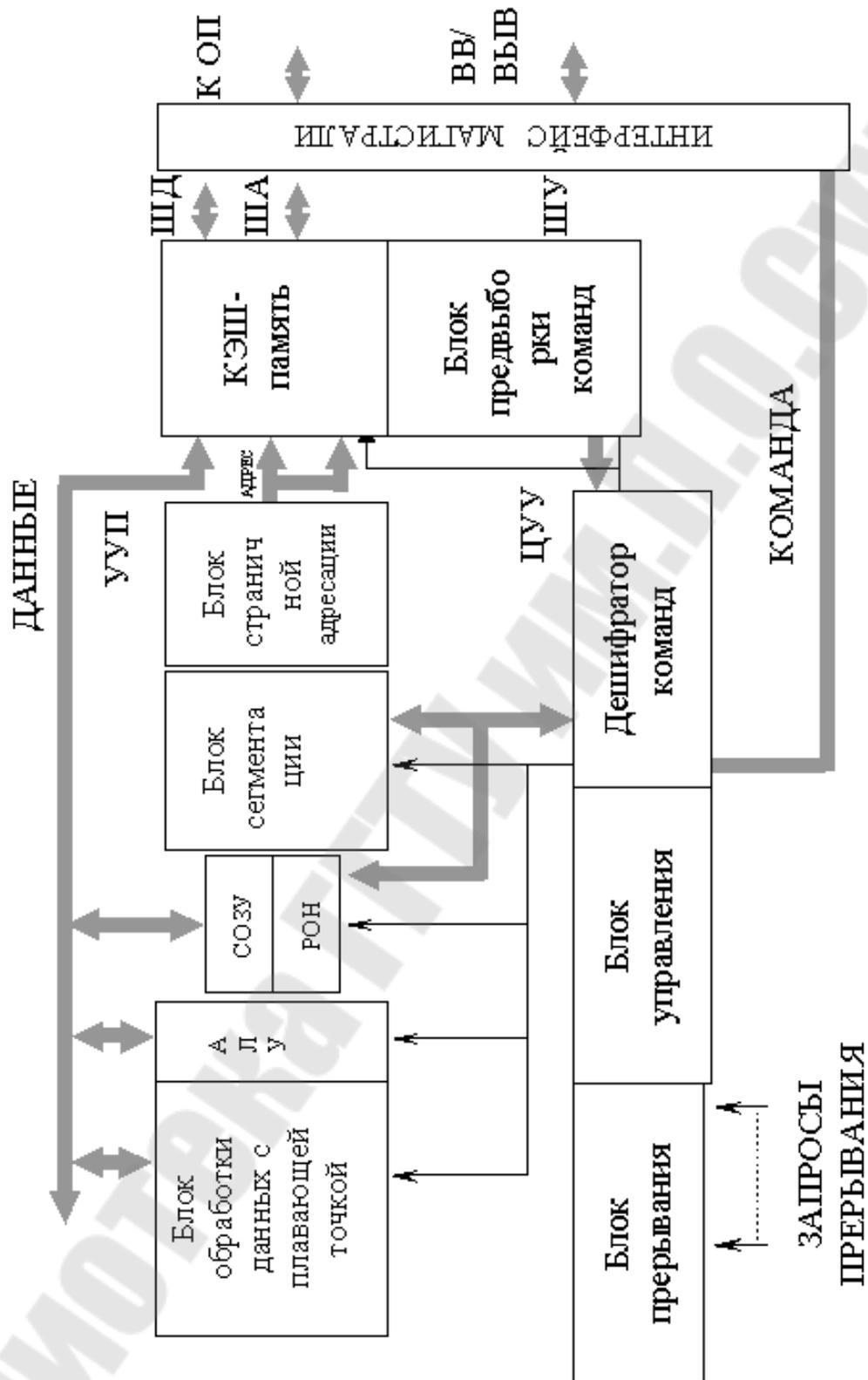


Рисунок 25 - Структурная схема ЦП

Сверхоперативное запоминающее устройство (СОЗУ) – (регистровый файл) содержит регистры общего назначения, в которых хранятся данные и адреса.

Устройство предвыборки команд и данных включает блок предвыборки команд и внутреннюю кэш-память процессора (кэш первого уровня).

Блок предвыборки команд осуществляет формирование очереди команд, причем выборка из памяти осуществляется в промежутках между магистральными циклами команд.

Во внутренней кэш-памяти осуществляется буферизация часто используемых команд и данных. Благодаря этому существенно повышается производительность процессора, сокращается число обращений к ОП.

Устройство управления памятью (диспетчер памяти) предназначено для сопряжения ЦП и подсистемы ввода/вывода с ОП. Оно состоит из блока сегментации и блока страничной адресации, осуществляющих двухступенчатое формирование физического адреса ячейки памяти: сначала в пределах сегмента, а затем в пределах страницы.

Наличие двух этих блоков, их параллельное функционирование обеспечивают максимальную гибкость проектируемой системы.

Сегментация полезна для организации памяти локальных модулей и является инструментом программиста, в то время как страницы позволяют системному программисту эффективно использовать физическую память ЭВМ.

Интерфейс магистрали реализует протоколы обмена (связь по определенным правилам) ЦП с памятью, каналами (контроллерами) ввода/вывода и другими активными устройствами системы ЭВМ. Обмен осуществляется с помощью шин данных, адреса и управления.

В современных суперскалярных процессорах может использоваться от 2 до 6 параллельно работающих исполнительных устройств. Это могут быть:

- несколько целочисленных устройств;
- устройство плавающей точки (блок FPU);
- устройство выполнения переходов;
- устройство загрузки/записи.

Устройство выполнения переходов обрабатывает команды условных переходов. Если условия перехода доступны, то решение о направлении перехода принимается немедленно, в противном случае

выполнение последующих команд продолжается по предположению (спекулятивно).

Пересылки данных между кэш-памятью данных, с одной стороны, и регистрами общего назначения и регистрами плавающей точки, с другой, обрабатываются устройством загрузки/записи.

4. РЕГИСТРОВЫЕ СТРУКТУРЫ ЦЕНТРАЛЬНОГО ПРОЦЕССОРА

Набор регистров и их структуры рассмотрим на примере процессоров Intel с CISC-архитектурой. Можно выделить следующие группы регистров:

1. Основные функциональные регистры (используются при выполнении прикладных программ) :

- регистры общего назначения (РОН);
- указатель команд;
- регистр флагов;
- регистры сегментов.

2. Регистры процессора (FPU) обработки чисел с плавающей точкой (используются при выполнении прикладных программ):

- регистры данных;
- регистр тегов;
- регистр состояния;
- регистр указателей команд и данных FPU;
- регистр управления FPU.

3. Системные регистры (используются при выполнении системных программ):

- регистры управления микропроцессора;
- регистры системных адресов.

4. Регистры отладки и тестирования (используются при отладке и тестировании).

Все 16-разрядные регистры микропроцессоров 8086, 80186, 80286 входят в состав набора 32-разрядных регистров для обеспечения обратной совместимости на аппаратном и программном уровне.

4.1. Основные функциональные регистры

Содержимое этих регистров определяется текущей задачей, т.е. в эти регистры автоматически загружается новое значение при переключении задач.

Регистры общего назначения. Восемь 32-разрядных регистров предназначены для хранения данных и адресов. Они поддерживают работу с данными разрядностью 1, 8, 16, 32 и 64 бита, битовыми полями длиной от 1 до 32 бит и адресами размером 16 и 32 бита. Младшие 16 разрядов этих регистров (рис. 26) доступны отдельно при использовании соответствующего имени, например регистр EAX (имя AX для 16 разрядов).

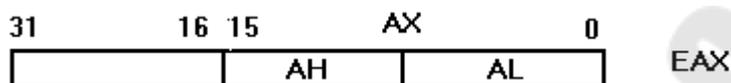


Рисунок 26 - Структура регистра общего назначения EAX

При операциях с байтами можно отдельно обращаться к младшему байту (разряды 0 - 7) и старшему байту (8 - 15) по именам AL и AH. Доступ к отдельным байтам обеспечивает дополнительную гибкость при операциях с данными.

Регистры сегментов и дескрипторы сегментов. Шесть 16-разрядных сегментных регистров (CS, SS, DS, ES, FS, GS) содержат значения селекторов сегментов, указывающих на текущие адресуемые сегменты памяти. С каждым из них связан программно-недоступный регистр дескриптора сегмента (рис. 27).

В защищенном режиме каждый сегмент может иметь размер от 1 байта до 4 Гбайт, в режиме реальных адресов максимальный размер сегмента составляет 64 Кбайта.

Селектор в CS обеспечивает обращение к текущему сегменту команд, селектор в SS — к текущему сегменту стека, селекторы в DS, ES, FS, GS — к текущим сегментам данных. Каждый регистр дескриптора содержит 32-разрядный размер сегмента и другие необходимые атрибуты.

Когда в регистр сегмента загружается новое значение селектора, содержимое соответствующего регистра дескриптора автоматически корректируется. В реальном режиме базовый адрес сегмента получается путем сдвига значения селектора на 4 разряда влево (20 разрядов), максимальный размер и атрибуты сегмента в реальном режиме имеют фиксированные значения.

		Базовый адрес	Размер сегмента	Другие атрибуты		
15	0					
Селектор	CS					
Селектор	SS					
Селектор	US					
Селектор	RS					
Селектор	PS					
Селектор	GS					

Рисунок 27 - Регистры сегментов и соответствующие регистры дескрипторов

Указатель команд. Указатель команд (рис. 28) представляет собой 32-разрядный регистр с именем EIP, содержимое которого используется в качестве смещения при определении адреса следующей выполняемой команды. Смещение задается относительно базового адреса сегмента команд CS. Младшие 16 бит (0 — 15) содержат 16-разрядный указатель команд с именем IP, который используется при 16-разрядной адресации.



Рисунок 28 - Структура регистра указателя команд

Указатель команд непосредственно программисту недоступен. Его содержимое изменяется при выполнении команд передачи управления и прерываний.

Регистр флагов является 32-разрядным, имеет имя EFLAGS. Его разряды содержат признаки результата выполнения команды, управляют обработкой прерываний, последовательностью вызываемых задач, вводом/выводом и рядом других процедур.

4.2. Регистры процессора обработки чисел с плавающей точкой

Набор регистров, входящих в блок (FPU), изображен на рис. 29



Рисунок 29- Регистры блока FPU

Регистр тегов FPU содержит 16-разрядное слово, включающее восемь двухбитовых тегов. Каждый тег (признак) характеризует содержимое одного из регистров данных.

Тег определяет, является ли регистр пустым (незаполненным) - код 11 или в него введено конечное число — 00 (достоверное значение), или нуль -01, неопределенное значение (бесконечность) — 10 (нет числа и неподдерживаемый формат). Слово тегов позволяет оптимизировать функционирование FPU посредством идентификации пустых и непустых регистров данных, проверить содержимое регистра без сложного декодирования хранящихся в нем данных.

5. ОБЩИЕ ПРИНЦИПЫ ОРГАНИЗАЦИИ СИСТЕМЫ ПЕРЫВАНИЯ ПРОГРАММ

Во время работы ЭВМ внутри нее самой или во внешней среде (в объекте, управляемом ЭВМ) могут возникнуть события, требующие немедленных ответных действий.

Прерывание программы – это способность ЭВМ при возникновении определенных ситуаций, требующих немедленной реакции ЭВМ, прекратить выполнение текущей программы и передать управление программе, реализующей реакцию ЭВМ на возникшую ситуацию.

Устройства, требующие вмешательства ЭВМ называются **источниками прерываний**.

Каждое событие, требующее прерывание, сопровождается сигналом, который называется **запросом прерывания**.

Программу, затребованную запросом прерывания, называют **прерывающей программой**.

Реакция ЭВМ состоит в том, что процессор формирует управляющий сигнал, требуемый прекращения выполнения текущей программы, а затем формирует управляющий сигнал перехода к обработке другой программы, соответствующей происходящему событию. По завершении этой программы ЭВМ возвращается к выполнению текущей программы.

Так происходит и при реализации **мультипрограммного режима работы**, когда обязательно наличие системы приоритетных прерываний.

Мультипрограммный режим основан на том, что выполнение огромного большинства программ включает в себя операции, связанные с приостановкой работы ЦП (операции по вводу исходных данных, выводу результатов, загрузка программы с ВЗУ и тому подобное). Паузы же в работе ЦП используются для решения других задач. Чем больше одновременно обрабатывается программ (выше **коэффициент мультипрограммирования**), тем больше вероятность того, что в любой момент времени либо микропроцессор, либо то или иное устройство будет задействовано, вследствие чего производительность возрастает.

Запросы прерывания могут возникать внутри самой ЭВМ или во внешней среде. К первым относятся: запросы при переполнении разрядной сетки, при попытке деления на 0, при выходе из установленной для программы области памяти, затребование периферийным устройством операции ввода/вывода, завершение операции ввода/вывода устройством или возникновение особой ситуации при этой операции. Запросы во внешней среде возникают от других ЭВМ, от аварийных и некоторых других датчиков управления объектами и другое. Таким образом, запросы прерывания генерируются несколькими развивающимися параллельно во времени процессами, которые в некоторые моменты времени требуют вмешательства процессора. К этим параллельным процессам в частности, относят процесс выполнения текущей программы, процесс контроля правильности работы

ЭВМ, операции ввода/вывода, технологические процессы в управляемом машиной объекте и другие.

Возможность прерывания программ позволяет эффективно использовать производительность процессора при наличии нескольких, протекающих параллельно во времени процессов, требующих в произвольные моменты времени вмешательства центрального процессора. В первую очередь это относится к организации параллельной во времени работы процессора и пункта управления машины, а также к использованию ЭВМ для управления в реальном времени технологического процесса. В некоторых ЭВМ, наряду или вместо прерывания с переключением управления на другую программу, используется так называемая приостановка, когда по соответствующему запросу приостанавливается выполнение программы и с помощью аппаратных средств выполняется некоторая процедура без изменения содержания счетчика команды, а по ее окончании продолжается выполнение приостановленной программы. Чтобы ЭВМ могла реализовывать прерывания программ с достаточно высоким быстродействием, не требуя при этом больших усилий от программиста, необходимо придать соответствующие аппаратные и программные средства, совокупность которых получила название система прерывания программ. В качестве аппаратных средств используется блок прерывания (контроллер прерывания). **Программные средства** – специальные программы, каждая из которых соответствует определенному прерыванию.

Основные функции системы прерывания:

- запоминание состояния прерываемой программы и переход к прерывающей программе;
- восстановление состояния прерванной программы и возврат к ней.

При наличии нескольких источников запросов прерывания, между ними должны быть установлены приоритетные соотношения. Приоритет определяет, какой из поступивших запросов подлежит обработке в первую очередь, и устанавливает, имеет ли данный запрос (прерывающая программ) право прерывать ту или иную программу.

Для оценки эффективности систем прерываний могут быть использованы следующие характеристики: Общее число запросов прерывания (входов в систему прерываний) и время реакции (время между появлением запроса прерывания и моментом прерывания текущей программы).

Приведем упрощенную диаграмму процесса (см. рис. 30).

Для одного и того же запроса задержки в исполнении прерывающей программы зависят от того, сколько программ со старшим приоритетом ждут обслуживания, поэтому время реакции определяют для запроса с наивысшим приоритетом. Время реакции зависит от того, в какой момент допустимо прерывание. Как правило, в современных ЭВМ прерывание допускается после окончания текущей команды. В этом случае время реакции определяется в основном длительностью выполняемой команды. Оно может оказаться недопустимо большим для ЭВМ, работающих в реальном масштабе времени.

Дело в том, что в таких машинах часто допускается выполнение прерывания после любого такта команды (микрокоманды). Однако при этом возрастает количество информации, подлежащей запоминанию и восстановлению при переключении программ.

Имеются ситуации, в которых желательно немедленное прерывание. Если аппаратура контроля обнаружила ошибку, то целесообразно сразу же прервать операцию, пока ошибка не оказала влияние на следующие такты работы программы.



Рисунок 30 - Упрощенная временная диаграмма процесса прерывания

Затраты времени на переключение программ (издержки прерывания) равны суммарному расходу времени на заполнение и восстановление состояния программы.

Глубина прерывания – максимальное число программ, которые могут прерывать друг друга. Если после перехода к прерывающей программе и до ее окончания прием запросов прекращается, то говорят, что система имеет глубину n , равную 1. Глубина n , равна N , если

допускается последовательное прерывание до N программ. Глубина прерывания обычно совпадает с числом уровней приоритетов в системе прерываний.

Вот так выглядят временные диаграммы для прерывающих программ в системах с различной глубиной прерывания (предполагаем, что приоритет каждого последующего запроса выше предыдущего).

Следует отметить, что система с большей глубиной прерывания обеспечивает более быструю реакцию на срочные запросы. Если запрос на прерывание от определенного источника прерывания окажется не обслуженным к моменту прихода нового запроса от того же источника, то возникает так называемое насыщение системы прерываний. В этом случае предыдущий запрос будет утерян.

Число уровней прерывания (число классов прерывания). В ЭВМ число различных запросов (причин) прерывания может достигать нескольких десятков или сотен. В таких случаях часть запросов разделяют на отдельные классы или уровни. Совокупность запросов, инициирующих одну и ту же прерывающую программу, образует **класс или уровень прерывания**. Разделение запросов на классы прерывания представлено на рис. 32.

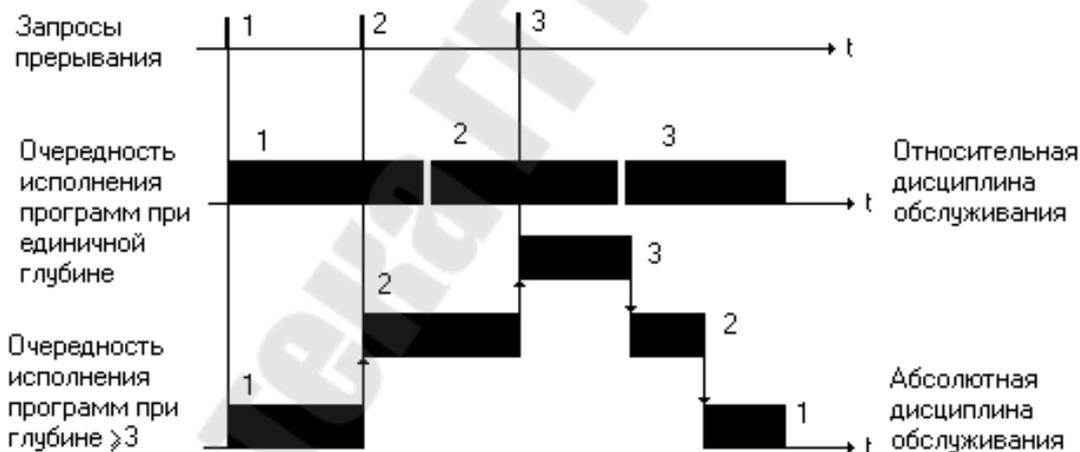


Рисунок 31 - Процессы прерывания с различной глубиной прерывания и дисциплиной обслуживания.

Запросы всех источников прерывания поступают на РгЗП, устанавливая соответствующие его разряды в единицу, которая указывает на наличие запроса прерывания определенного источника. Запросы

классов прерывания **ЗПК** формируются схемами ИЛИ, объединяющих разряды **РгЗП**, относящихся к соответствующим уровням прерывания. Еще одна схема ИЛИ формирует общий сигнал прерывания **ОСП**, поступающий в устройство управления процессора. После принятия запроса прерывания на исполнение и передачу управления прерывающей программе соответствующий триггер **РгЗП** сбрасывается. Следует отметить, что объединение запросов в классы прерывания позволяет уменьшить объем аппаратуры, но приводит к замедлению работы системы прерываний.

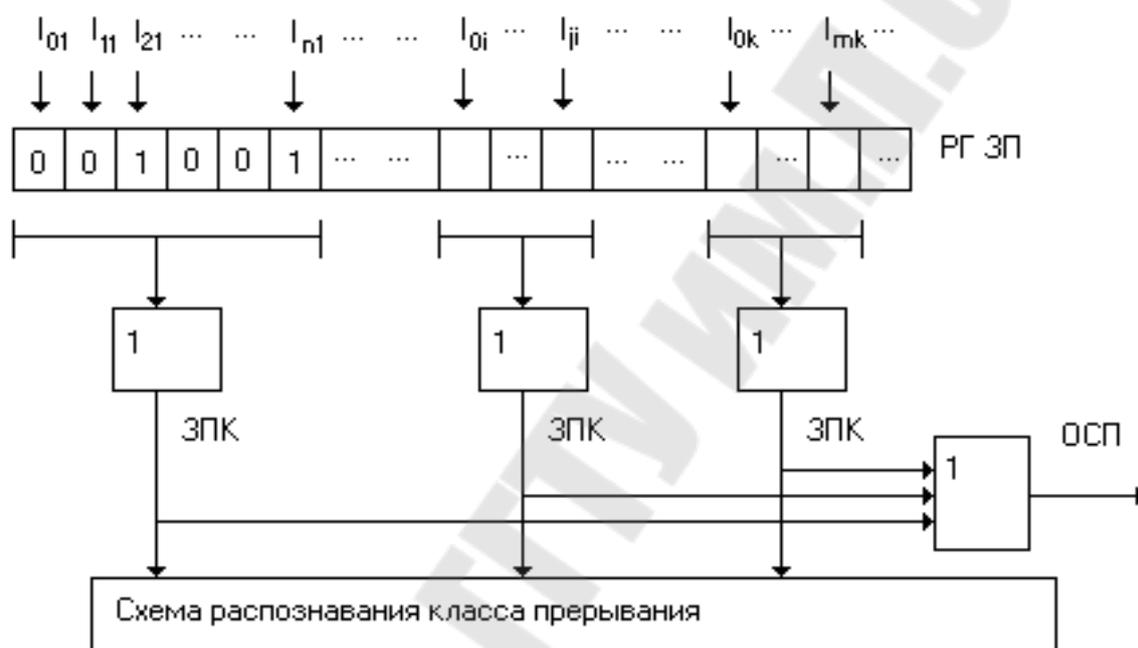


Рисунок 32 - Разделение запросов на классы прерывания

Относительная степень важности программ, их частота повторения, относительная степень срочности в ходе вычислительного процесса могут меняться, требуя установления новых приоритетных соотношений. Поэтому во многих случаях приоритет между прерывающими программами не может быть зафиксирован раз и навсегда. Необходимо иметь возможность изменять по мере необходимости приоритетные соотношения программным путем.

В ЭВМ широко применяются два способа программно-управляемого приоритета прерывающих программ:

- использование порога прерывания;
- использование маски прерывания.

Использование порога прерывания позволяет в ходе вычислительного процесса программным путем изменить уровень приоритета процессора (а значит, и обрабатываемой в данный момент на процессоре программы) относительно приоритетов запросов источников прерывания, другими словами, задать порог прерывания, то есть минимальный уровень приоритета запроса прерывания, которому разрешается прерывать программу, выполняемую на процессоре. Порог прерывания задается командой программы, устанавливая в регистре код порога прерывания. Специальная схема выделяет наиболее приоритетный запрос, сравнивает его приоритет с порогом прерывания, и если он оказывается выше порога, вырабатывает общий сигнал прерывания, и начинается процедура прерывания.

Маска прерывания представляет собой двоичный код, разряды которого поставлены в соответствие запросам или классам (уровням) прерываний. Маска загружается командой программы в регистр маски (см. рис. 33)



Рисунок 33 - Маскирование прерываний

Состояние 1 в разряде PGM разрешает, а состояние 0 запрещает (маскирует) прерывание текущей программы от соответствующего запроса. Таким образом, программа, изменяя маску в PGM, может устанавливать произвольные соотношения между программами, без переконмутации линий, по которым поступают запросы прерываний. Каждая прерывающая программа может установить свою маску. При формировании маски 1 устанавливается в разряды, соответствующие запросам (прерывающим программам) с более высоким, чем у данной

программы, приоритетом. Схемы И выделяют поступившие незамаскированные запросы прерываний, из которых специальная схема выделяет наиболее приоритетный и формирует код его номера. С замаскированными запросами, в зависимости от причин прерываний поступают двояким образом: или игнорируется, или запоминается.

6. МАТЕРИНСКАЯ ПЛАТА И ЕЕ ОСНОВНЫЕ ИНТЕРФЕЙСЫ

Материнская плата – это один из важнейших элементов ЭВМ, определяющий ее облик и обеспечивающий взаимодействие всех подключаемых к материнской плате устройств.

На материнской плате размещаются все основные элементы ЭВМ, такие как:

- набор системной логики или чипсет – основной компонент материнской платы, определяющий какой тип процессора, тип ОЗУ, тип системной шины можно использовать;

- слот для установки процессора. Определяет, какой именно тип процессоров можно подсоединить к материнской плате. Соответственно для каждого типа процессора необходимо использовать свой слот для установки.

- центральный процессор – основное устройство ЭВМ, выполняющее математические, логические операции и операции управления всеми остальными элементами ЭВМ, о нем более подробно упоминалось выше;

- контроллер ОЗУ (оперативно запоминающее устройство). Раньше контроллер ОЗУ встраивали в чипсет, но сейчас большинство процессоров имеют встроенный контроллер ОЗУ, что позволяет увеличить общую производительность и разгрузить чипсет.

- ОЗУ – набор микросхем для временного хранения данных. В современных материнских платах имеется возможность подключения одновременно нескольких микросхем ОЗУ, обычно четырех или более.

- ППЗУ (БИОС), содержащие программное обеспечение, осуществляющее тестирование основных компонентов ЭВМ и настройку материнской платы. И память CMOS хранящая настройки работы BIOS. Часто устанавливают несколько микросхем памяти CMOS для возможности быстрого восстановления работоспособности ЭВМ в экстренном случае, например, неудачной попытки разгона;

- аккумулятор или батарейка, питающая память CMOS;
- контроллеры каналов ввода-вывода: USB, COM, LPT, ATA, SATA, SCSI, FireWire, Ethernet и др. Какие именно каналы ввода-вывода будут поддерживаться, определяется типом используемой материнской платы. В случае необходимости, дополнительные контроллеры ввода-вывода можно устанавливать в виде плат расширения;
- кварцевый генератор, вырабатывающий сигналы, по которым синхронизируется работа всех элементов ЭВМ;
- таймеры;
- контроллер прерываний. Сигналы прерываний от различных устройств поступают не напрямую в процессор, а в контроллер прерываний, который устанавливает сигнал прерывания с соответствующим приоритетом в активное состояние;
- разъемы для установки плат расширения: видеокарт, звуковой карты и т.д.;
- регуляторы напряжения, преобразующие исходное напряжение в требуемое для питания компонентов установленных на материнской плате;
- средства мониторинга, измеряющие скорость вращения вентиляторов, температуру основных элементов ЭВМ, питающее напряжение и т.д.;
- звуковая карта. Практически все материнские платы содержат встроенные звуковые карты, позволяющие получить приличное качество звука. При необходимости можно установить дополнительную дискретную звуковую карту, обеспечивающую лучшее звучание, но в большинстве случаев это не требуется;
- встроенный динамик. Главным образом используется для диагностики работоспособности системы. Так по длительности и последовательности звуковых сигналов при включении ЭВМ можно определить большинство неисправностей аппаратуры;
- шины – проводники для обмена сигналами между компонентами ЭВМ.

Основу материнской платы составляет печатная плата. На печатной плате располагаются сигнальные линии, часто называемые сигнальными дорожками, соединяющими между собой все элементы материнской платы.

Обычно современные материнские платы имеют шесть слоев: три сигнальных слоя, слой заземления и две пластины питания.

Однако количество слоев питания и сигнальных слоев может варьироваться, в зависимости от особенностей материнских плат.

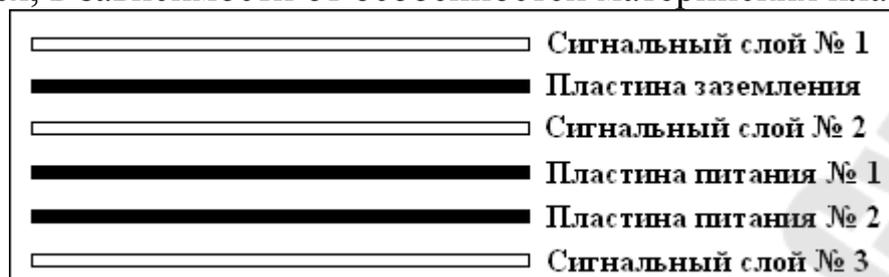


Рисунок 34 – Расположение слоев в печатной плате материнской платы

На печатной плате располагаются все компоненты материнской платы и разъемы для подключения плат расширения и периферийных устройств. На рис. 35 изображена структурная схема типовой материнской платы

Чипсет или набор системной логики – это основной набор микросхем материнской платы, обеспечивающий совместное функционирование центрального процессора, ОЗУ, видеокарты, контроллеров периферийных устройств и других компонентов, подключаемых к материнской плате. Именно он определяет основные параметры материнской платы: тип поддерживаемого процессора, объем, канальность и тип ОЗУ, частоту и тип системной шины и шины памяти, наборы контроллеров периферийных устройств и так далее.

Как правило, современные наборы системной логики строятся на базе двух компонентов, представляющих собой отдельные чипсеты, связанные друг с другом высокоскоростной шиной.

Однако последнее время появилась тенденция объединения северного и южного моста в единый компонент, так как контроллер памяти все чаще встраивают непосредственно в процессор, тем самым разгружая северный мост, и появляются все более быстрые и быстрые каналы связи с периферийными устройствами и платами расширения. А также развивается технология производства интегральных схем, позволяющая делать их более миниатюрными, дешевыми и потребляющими меньше энергии.

Объединение северного и южного моста в один чипсет позволяет поднять производительность системы, за счет уменьшения времени взаимодействия с периферийными устройствами и внутренними компонентами, ранее подключаемыми к южному мосту, но значительно усложняет конструкцию чипсета, делает его более сложным для модернизации и несколько увеличивает стоимость материнской платы.

Но пока что большинство материнских плат делают на основе чипсета разделенного на два компонента. Называются эти компоненты Северный и Южный мост.

Названия Северный и Южный - исторические. Они означают расположение компонентов чипсета относительно шины PCI: Северный находится выше, а Южный - ниже. Почему мост? Это название дали чипсетам по выполняемым ими функциям: они служат для связи различных шин и интерфейсов.

Причины разделения чипсета на две части следующие:

1. Различия скоростных режимов работы.

Северный мост работает с самыми быстрыми и требующими большой пропускной способности шины компонентами. К числу таких компонентов относится видеокарта и память. Однако сегодня большинство процессоров имеют встроенный контроллер памяти, а многие и встроенную графическую систему, хотя и сильно уступающую дискретным видеокартам, но все же часто применяемую в бюджетных персональных компьютерах, ноутбуках и нетбуках. Поэтому, с каждым годом нагрузки на северный мост снижаются, что уменьшает необходимость разделения чипсета на две части.

2. Более частое обновление стандартов периферии, чем основных частей ЭВМ.

Стандарты шин связи с памятью, видеокартой и процессором изменяются гораздо реже, чем стандарты связи с платами расширения и периферийными устройствами. Что позволяет, в случае изменения интерфейса связи с периферийными устройствами или разработки нового канала связи, не изменять весь чипсет, а заменить только южный мост. К тому же северный мост работает с более быстрыми устройствами и устроен сложнее, чем южный мост, так как от его работы во многом зависит общая производительность системы. Поэтому его изменение - дорогая и сложная работа. Но, несмотря на это, наблюдается тенденция объединения северного и южного моста в одну интегральную схему.

6.1. Основные функции Северного моста

Северный мост, как следует из его названия, выполняет функции контроля и направления потока данных из 4-х шин:

1. Шины связи с процессором или системной шиной.
2. Шины связи с памятью.

3. Шины связи с графическим адаптером.

4. Шины связи с южным мостом.

В соответствии с выполняемыми функциями и устроен северный мост. Он состоит из интерфейса системной шины, интерфейса шины связи с южным мостом, контроллера памяти, интерфейса шины связи с графической картой.

На данный момент большинство процессоров имеют встроенный контроллер памяти, так что функцию контроллера памяти можно считать для северного моста устаревшей.

В бюджетных ЭВМ иногда в северный мост встраивают графическую систему. Однако на данный момент более распространенную практику имеет установка графической системы непосредственно в процессор, так что эту функцию северного моста тоже будем считать устаревшей.

Таким образом, основная задача чипсета - грамотно и быстро распределять все запросы от процессора, видеокарты и южного моста, расставлять приоритеты и создавать, если это необходимо, очередность. Причем он должен быть настолько сбалансирован, чтобы как можно сильнее сократить простои при попытке доступа компонентов ЭВМ к тем или иным ресурсам.

Рассмотрим более подробно существующие интерфейсы связи с процессором, графическим адаптером и южным мостом.

На данный момент существуют следующие интерфейсы связи процессора с северным мостом: FSB, DMI, HyperTransport, QPI.

FSB (Front Site Bus) - системная шина, используемая для связи центрального процессора с северным мостом в 1990-х и 2000-х годах. FSB разработана компанией Intel и впервые использовалась в компьютерах на базе процессоров Pentium.

Частота работы шины FSB является одним из важнейших параметров работы ЭВМ и во многом определяет производительность всей системы. Обычно она - в несколько раз меньше частоты работы процессора.

Частоты, на которых работают центральный процессор и системная шина, имеют общую опорную частоту и в упрощенном виде рассчитываются, как $V_{п} = V_{о} * k$, где $V_{п}$ - частота работы процессора, $V_{о}$ - опорная частота, k - множитель. Обычно в современных системах опорная частота равняется частоте шины FSB.

Большинство материнских плат позволяют вручную увеличивать частоту системной шины или множитель, изменяя настройки в BIOS.

В старых материнских платах подобные настройки изменялись с помощью перестановки перемычек. Увеличение частоты системной шины или множителя увеличивает производительность ЭВМ. Однако в большинстве современных процессоров средней ценовой категории множитель заблокирован, и единственный способ поднять производительность вычислительной системы – это увеличить частоту системной шины.

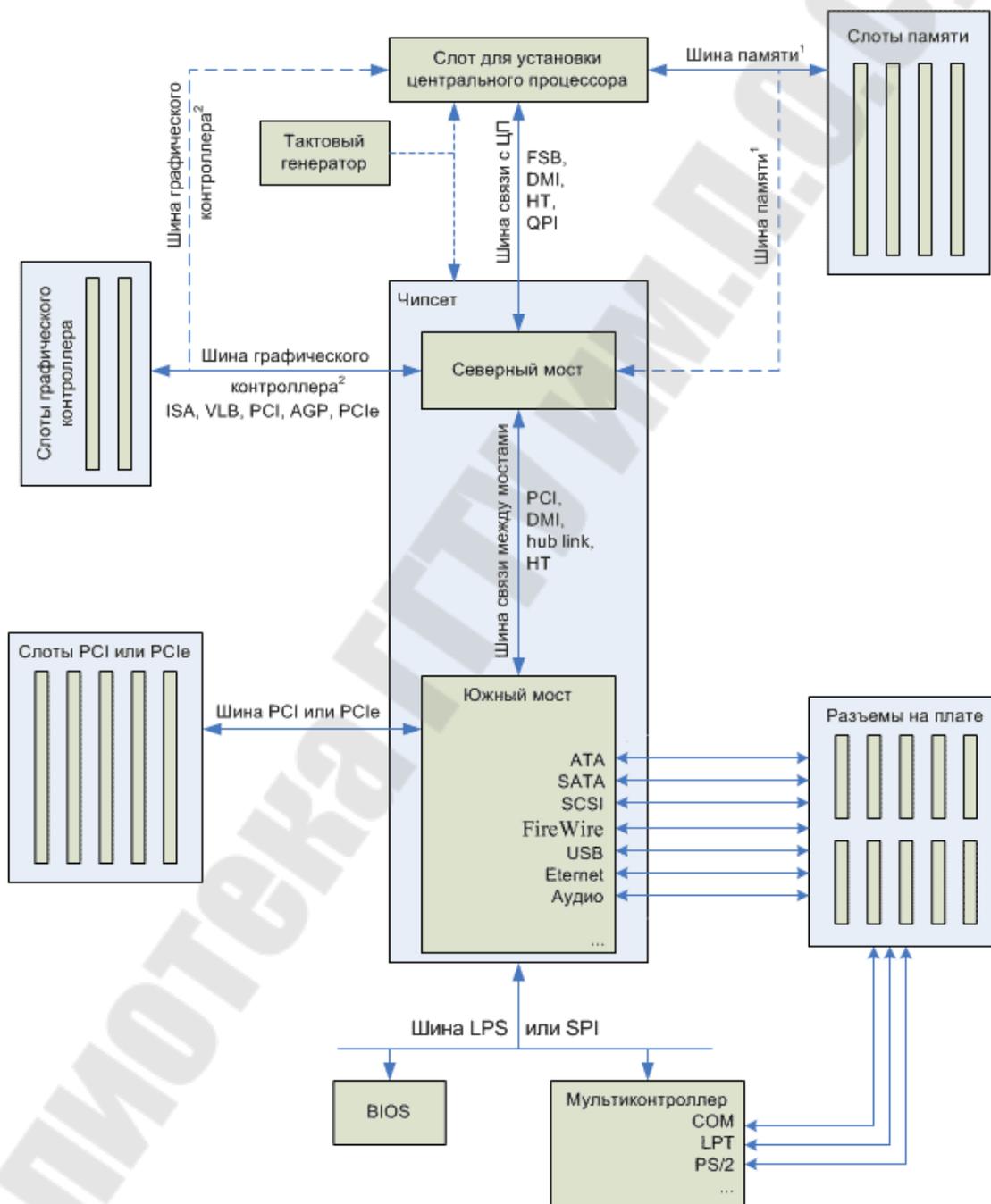


Рисунок 35 – Структурная схема типовой материнской платы

Частота системной шины FSB постепенно возрастала с 50 МГц, для процессоров класса Intel Pentium и AMD K5 в начале 1990-х годов, до 400 МГц, для процессоров класса Xeon и Core 2 в конце 2000-х. При этом пропускная способность возрастала с 400 Мбит/с до 12800 Мбит/с.

Шина FSB использовалась в процессорах типа Atom, Celeron, Pentium, Core 2, и Xeon вплоть до 2008 года. На данный момент эта шина вытеснена системными шинами DMI, QPI и Hyper Transport.

HyperTransport – универсальная высокоскоростная шина типа точка-точка с низкой латентностью, используемая для связи процессора с северным мостом. Шина HyperTransport - двунаправленная, то есть для обмена в каждую сторону выделена своя линия связи. К тому же она работает по технологии DDR (Double Data Rate), передавая данные, как по фронту, так и по спаду тактового импульса.

Технология разработана консорциумом HyperTransport Technology во главе с компанией AMD. Стоит отметить, что стандарт HyperTransport - открытый, что позволяет использовать его в своих устройствах различным компаниям.

Первая версия HyperTransport была представлена в 2001 году, и позволяла производить обмен со скоростью 800 МТр/с (800 Мега Транзакций в секунду или 838860800 обменов в секунду) с максимальной пропускной способностью - 12.8 ГБайт/с. Но уже в 2004 году была выпущена новая модификация шины HyperTransport (v.2.0), обеспечивающая 1.4 ГТр/с с максимальной пропускной способностью - 22.4 ГБайт/с, что почти в 14 раз превышало возможности шины FSB.

18 августа 2008 года была выпущена модификация 3.1, работающая со скоростью 3.2 ГТр/с, с пропускной способностью - 51.6 Гбайт/с. На данный момент это - самая быстрая версия шины HyperTransport.

Технология HyperTransport - очень гибкая, и позволяет варьировать, как частоты шины, так и ее разрядность. Это позволяет использовать ее не только для связи процессора с северным мостом и ОЗУ, но и в медленных устройствах. При этом возможность уменьшения разрядности и частоты ведет к экономии энергии.

Минимальная тактовая частота шины – 200 МГц, при этом данных будут передаваться со скоростью - 400 МТр/с, из-за технологии DDR, а минимальная разрядность - 2 бита. При минимальных параметрах максимальная пропускная способность составит 100 Мбайт/с. Все следующие поддерживаемые частоты и разрядности - кратны мини-

мальной тактовой частоте и разрядности вплоть до скорости - 3.2 ГТр/с, и разрядности - 32 бита, для ревизии HyperTransport v 3.1.

DMI (Direct Media Interface) – последовательная шина типа точка-точка, используемая для связи процессора с чипсетом и для связи южного моста чипсета с северным. Разработана компанией Intel в 2004 году.

Для связи процессора с чипсетом обычно используется 4 канала DMI, обеспечивающих максимальную пропускную способность до 10 Гбайт/с, для ревизии DMI 1.0, и 20 Гбайт/с, для ревизии DMI 2.0, представленной в 2011 году. В бюджетных мобильных системах может использоваться шина с двумя каналами DMI, что в два раза снижает пропускную способность по сравнению с 4-х канальным вариантом.

Часто в процессоры, использующие связь с чипсетом по шине DMI, встраивают, наряду с контроллером памяти, контроллер шины PCI Express, обеспечивающий взаимодействие с видеокартой. В этом случае надобность в северном мосте отпадает, и чипсет выполняет только функции взаимодействия с платами расширения и периферийными устройствами. При такой архитектуре материнской платы не требуется высокоскоростного канала для взаимодействия с процессором, и пропускной способности шины DMI хватает с избытком.

QPI (QuickPath Interconnect) – последовательная шина типа точка-точка, используемая для связи процессоров между собой и с чипсетом. Представлена компанией Intel в 2008 году и используется в HiEnd процессорах типа Xeon, Itanium и Core i7.

Шина QPI - двунаправленная, то есть для обмена в каждую сторону предусмотрен свой канал, каждый из которых состоит из 20 линий связи. Следовательно, каждый канал – 20-разрядный, из которых на полезную нагрузку приходится только 16 разрядов. Работает шина QPI со скоростью - 4.8 и 6.4 ГТр/с, при этом максимальная пропускная способность составляет 19,2 и 25,6 ГБайт/с соответственно.

Вначале для связи с графическим процессором использовали общую шину ISA, VLB, а затем PCI, но очень быстро пропускной способности этих шин перестало хватать для работы с графикой, тем более после распространения трехмерной графики, требующей огромных мощностей для расчета и высокой пропускной способности шины для передачи текстур и параметров изображения.

На замену общим шинам пришла специализированная шина AGP, оптимизированная для работы с графическим контроллером.

AGP (Accelerated Graphics Port) – специализированная 32-разрядная шина для работы с графическим адаптером, разработанная в 1997 году компанией Intel.

Шина AGP работала на тактовой частоте - 66 МГц, и поддерживала два режима работы: с памятью DMA (Direct Memory Access) и памятью DME (Direct in Memory Execute).

В режиме DMA основной памятью считалась память, встроенная в видеоадаптер, а в режиме DME – память видеокарты, которые вместе с основной памятью находились в едином адресном пространстве, и видеоадаптер мог обращаться, как к встроенной памяти, так и к основной памяти компьютера.

Наличие режима DME позволяло уменьшить объем встраиваемой в видеоадаптер памяти и тем самым уменьшить его стоимость. Режим работы с памятью DME получил название AGP-текстурирование.

Однако очень скоро пропускной способности шины AGP перестало хватать для работы в режиме DME, и производители стали увеличивать объемы встраиваемой памяти. Вскоре и увеличение встраиваемой памяти перестало помогать и пропускной способности шины AGP стало категорически не хватать.

В 2002 году вышла ревизия 3.0 шины AGP. Опорная частота шины по-прежнему осталась неизменной, однако дополнительный тактовый импульс, запускающийся синхронно с опорной частотой, составлял уже 266 МГц. При этом за 1 такт опорной частоты передавалось уже 8 блоков, а максимальная скорость составила 2.1 Гбайт/с.

Но, несмотря на все улучшения шины AGP, видеоадаптеры развивались быстрее и требовали более производительной шины. Так на смену шине AGP пришла шина PCI express.

PCI express – последовательная двунаправленная шина типа точка-точка, разработанная в 2002 некоммерческой группой PCI-SIG, в состав которой входили такие компании, как Intel, Microsoft, IBM, AMD, Sun Microsystems и другие.

Основная задача, стоящая перед шиной PCI express, – это замена графической шины AGP и параллельной универсальной шины PCI.

Ревизия шины PCI express 1.0 работает на тактовой частоте 2.5 ГГц, при этом пропускная суммарная способность одного канала составляет 400 Мбайт/с, так как на каждые переданные 8 бит данных приходится 2 служебных бита и шина двунаправленная, то есть обмен в обе стороны идет одновременно. В шине обычно используется несколько каналов: 1, 2, 4, 8, 16 или 32, в зависимости от требуемой

пропускной способности. Таким образом, шины на базе PCI express в общем случае представляют собой набор самостоятельных последовательных каналов передачи данных.

Так при использовании шины PCI express для связи с видеокартами обычно используется 16-ти канальная шина, а для связи с платами расширения – одноканальная шина.

Теоретическая максимальная суммарная пропускная способность 32-х канальной шины составляет 12.8 Гбайт/с. При этом, в отличие от шины PCI, делившей пропускную способность между всеми подключенными устройствами, шина PCI express построена по принципу топологии типа «звезда» и каждому подключаемому устройству в единоличное владение отдается вся пропускная способность шины.

В ревизии PCI express 2.0, представленной 15 января 2007 года, пропускная способность шины была увеличена в 2 раза. Для одного канала шины суммарная пропускная способность составила 800 Мбайт/с, а для 32-х канальной шины – 25.6 Гбайт/с.

В ревизии PCI express 3.0, представленной в ноябре 2010 года, пропускную способность шины еще в 2 раза увеличили, причем максимальное количество транзакций увеличилось с 5 до 8 млрд, а максимальная пропускная способность увеличилась в 2 раза, благодаря изменению принципа кодирования информации, при котором на каждые 129 бит данных приходится всего 2 служебных бита, что в 13 раз меньше, чем в ревизиях 1.0 и 2.0. Таким образом, для одного канала шины суммарная пропускная способность стала 1.6 Гбайт/с, а для 32-х канальной шины – 51.2 Гбайт/с.

Однако PCI express 3.0 только выходит на рынок и первые материнские платы с поддержкой этой шины начали появляться в конце 2011 года, а массовый выпуск устройств с поддержкой шины PCI express 3.0 запланирован на 2012 год.

Стоит отметить, что на данный момент пропускной способности PCI express 2.0 вполне хватает для нормального функционирования видеоадаптеров и переход на PCI express 3.0 не даст существенного прироста производительности в связке процессор – видеокарта.

В ближайшем будущем планируется выпуск ревизии PCI express 4.0, в котором скорость будет увеличена еще в 2 раза.

В последнее время наметилась тенденция встраивания интерфейса PCI express непосредственно в процессор. Обычно в таких процессорах также встроен контроллер памяти. В результате, надобность в серверном мосте отпадает, и чипсет строят на основе одной интеграль-

ной схемы, основная задача которой – обеспечение взаимодействия с платами расширения и периферийными устройствами.

6.2. Интерфейсы связи с южным мостом

Довольно долгое время для связи северного моста с южным использовалась шина PCI.

PCI (Peripheral component interconnect) – шина для подключения плат расширения к материнской плате, разработанная в 1992 году компанией Intel. Также долгое время использовалась для связи северного моста с южным. Однако по мере повышения производительности плат расширения ее пропускной способности стало не хватать. Она была вытеснена более производительными шинами вначале из задач связи северного и южного моста, а в последние годы и для связи с платами расширения стали использовать более быструю шину – PCI express.

Основные технические характеристики шины PCI, следующие:

Ревизия	1.0	2.0	2.1	2.2	2.3
Дата релиза	1992 г.	1993 г.	1995 г.	1998 г.	2002 г.
Разрядность	32	32	32/64	32/64	32/64
Частота	33 МГц	33 МГц	33/66 МГц	33/66 МГц	33/66 МГц
Пропускная способность	132 МБайт/с	132 МБайт/с	132/264/528 МБайт/с	132/264/528 МБайт/с	132/264/528 МБайт/с
Сигнальное напряжение	5 В	5 В	5/3.3 В	5/3.3 В	5/3.3 В
Горячая замена	нет	нет	нет	есть	есть

Существуют и другие ревизии шин PCI, например, для использования в ноутбуках и других портативных устройствах, или переход-

ные варианты между основными ревизиями, но так как на данный момент интерфейс PCI практически вытеснен более скоростными шинами, то не буду подробно описывать характеристики всех ревизий.

В результате, со временем пропускной способности шины стало не хватать, и для связи между северным и южным мостом стали использовать такие шины, как: hub link, DMI, HyperTransport, а шина PCI еще ненадолго осталась в качестве связи с платами расширения.

Первой на замену PCI пришла шина hub link.

Шина hublink – 8-битная шина типа точка-точка, разработанная компанией Intel. Шина работает на частоте – 66 МГц, и передает 4 байта за такт, что позволяет получить максимальную пропускную способность – 266 Мбайт/сек.

Ввод шины hublink изменил архитектуру материнской платы и разгрузил шину PCI. Шина PCI стала использоваться только для связи с периферийными устройствами и платами расширения, а шина hublink использовалась только для связи с северным мостом.

Пропускная способность шины hublink была сравнима с пропускной способностью шины PCI, но из-за того, что ей не приходилось делить канал с другими устройствами, а шина PCI разгружалась, то пропускной способности было вполне достаточно. Но вычислительная техника не стоит на месте, и шина hublink на данный момент практически не используется, из-за недостаточного быстродействия. Она была вытеснена такими шинами, как DMI и HyperTransport.

Были и другие интерфейсы для связи северного моста с южным, но большинство из них уже безнадежно устарели или редко используются.

6.3. Основные функции Южного моста

Южный мост отвечает за организацию взаимодействия с медленными компонентами ЭВМ: платами расширения, периферийными устройствами, устройствами ввода-вывода, каналами межмашинного обмена и так далее.

То есть, Южный мост ретранслирует данные и запросы от подключенных к нему устройств в северный мост, который передает их в процессор или ОЗУ, и принимает от северного моста команды процессора и данные из ОЗУ, и ретранслирует их в подключенные к нему устройства.

В состав южного моста входят:

- контроллер шины связи с северным мостом (PCI, hublink, DMI, HyperTransport и т.д.);
- контроллер шины связи с платами расширения (PCI, PCIE и т.д.);
- контроллер линий связи с периферийными устройствами и другими ЭВМ (USB, FireWire, Ethernet и т.д.);
- контроллер шины связи с жесткими дисками (ATA, SATA, SCSI и т.д.);
- контроллер шины связи с медленными устройствами (шины ISA, LPC, SPI и т.д.).

Рассмотрим более подробно интерфейсы связи, используемые южным мостом, и встроенные в него контроллеры периферийных устройств.

Интерфейсы связи северного моста с южным мы уже рассматривали. Поэтому сразу перейдем к интерфейсам связи с платами расширения.

На данный момент основными интерфейсами для обмена с платами расширения являются PCI и PCIexpress. Однако интерфейс PCI активно вытесняется, и в ближайшие несколько лет практически уйдет историю, и будет использоваться только в некоторых специализированных ЭВМ.

6.4. Интерфейсы связи с периферийными устройствами, устройствами ввода - вывода и другими ЭВМ

Существует большое разнообразие интерфейсов для связи с периферийными устройствами и другими ЭВМ, наиболее распространенные из них встраиваются в материнскую плату, но также можно добавлять любой из интерфейсов с помощью плат расширения, подключаемых к материнской плате через шину PCI или PCIexpress.

Приведу краткое описание и характеристики наиболее популярных интерфейсов.

USB (Universal Serial Bus) – универсальный последовательный канал передачи данных для подключения к ЭВМ среднескоростных и низкоскоростных периферийных устройств.

Шина строго ориентирована и состоит из контроллера канала и подключаемых к нему нескольких оконечных устройств. Обычно контроллеры канала USB встроены в южный мост материнской пла-

ты. В современных материнских платах могут размещаться до 12 контроллеров канала USB с двумя портами каждый.

Соединение между собой двух контроллеров канала или двух конечных устройств невозможно, поэтому напрямую соединить два компьютера или два периферийных устройства между собой по USB-каналу нельзя.

Однако для связи двух контроллеров канала между собой можно использовать дополнительные устройства. Например, эмулятор Ethernet адаптера. Два компьютера подключаются к нему по USB каналу, и оба видят оконечное устройство. Ethernet адаптер ретранслирует данные, получаемые от одного компьютера к другому, эмулируя сетевой протокол Ethernet. Однако при этом необходимо устанавливать специфические драйвера эмулятора Ethernet адаптера на каждый подключаемый компьютер.

Интерфейс USB имеет встроенные линии питания, благодаря чему позволяет использовать устройства без собственного источника питания или одновременно с обменом данными подзаряжать аккумуляторы конечных устройств, например телефонов.

Однако, если между контроллером канала и оконечным устройством используется размножитель (USB-hub), то он должен обладать дополнительным внешним питанием, чтобы обеспечить все подключаемые к нему устройства питанием, требуемым по стандарту интерфейса USB. Если использовать USB-hub без дополнительного источника питания, то, при подключении нескольких устройств без собственных источников питания, они, скорее всего, работать не будут.

USB поддерживает «горячее» подключение оконечных устройств. Это возможно, из-за более длинного заземляющего контакта, чем сигнальные контакты. Поэтому, при подключении оконечного устройства, вначале замыкаются контакты заземления, и разность потенциала компьютера и оконечного устройства выравнивается. Следовательно, дальнейшее соединение сигнальных проводников не приводит к скачку напряжения.

На данный момент существует три основные ревизии интерфейса USB (1.0, 2.0 и 3.0). Причем они совместимы снизу-вверх, то есть устройства, предназначенные для ревизии 1.0, будут работать с интерфейсом ревизии 2.0, соответственно, устройства, предназначенные для USB 2.0, будут работать с USB 3.0, однако устройства для USB 3.0, скорее всего не будут работать с интерфейсом USB 2.0.

Рассмотрим основные характеристики интерфейса, в зависимости от ревизии.

USB 1.0 – первая версия интерфейса USB, выпущенная в ноябре 1995 года. В 1998 году ревизия была доработана, устранены ошибки и недочеты. Полученная ревизия USB 1.1 первой получила массовое распространение.

Технические характеристики ревизий 1.0 и 1.1 следующие:

- скорость передачи данных – до 12 Мбит/с (режим Full-Speed) или 1,5 Мбит/с (режим Low-Speed);
- синхронная передача данных (по запросу);
- полудуплексный обмен (одновременно передача возможна только в одном направлении);
- максимальная длина кабеля – 5 метров, для режима Low-Speed, и 3 метра, для режима Full-Speed;
- максимальное количество подключённых устройств к одному контроллеру (включая размножители) – 127;
- возможно подключение устройств, работающих в режимах с различной пропускной способностью, к одному контроллеру USB;
- напряжение питания для периферийных устройств – 5 В;
- максимальная сила тока – 500 мА;
- кабель состоит из четырех линий связи (две линии – для приема и передачи данных, и две линии – для питания периферийных устройств) и заземляющей оплетки.

USB 2.0 – ревизия, вышедшая в апреле 2000 года. Основное отличие от предыдущей версии – повышение максимальной скорости передачи данных до 480 Мбит/с. На практике, из-за больших задержек между запросом на передачу данных и началом передачи, скорости в 480 Мбит/с достичь не удастся.

Технические характеристики ревизии 2.0 следующие:

- скорость передачи данных – до 480 Мбит/с (Hi-speed), до 12 Мбит/с (режим Full-Speed) или до 1,5 Мбит/с (режим Low-Speed);
- синхронная передача данных (по запросу);
- полудуплексный обмен (одновременно передача возможна только в одном направлении);
- максимальная длина кабеля – 5 метров;
- максимальное количество подключённых устройств к одному контроллеру (включая размножители) – 127;
- возможно подключение устройств, работающих в режимах с различной пропускной способностью, к одному контроллеру USB;

- напряжение питания для периферийных устройств – 5 В;
- максимальная сила тока – 500 мА;
- кабель состоит из четырех линий связи (две линии – для приема и передачи данных, и две линии – для питания периферийных устройств) и заземляющей оплетки.

USB 3.0 – ревизия, вышедшая в ноябре 2008 года. В новой ревизии на порядок была увеличена скорость, до 4800 Мбит/с, и почти в два раза – сила тока, до 900 мА. При этом сильно изменился внешний вид разъемов и кабелей, но совместимость снизу-вверх осталась. Т.е. устройства, работающие с USB 2.0, смогут подключаться к разъему 3.0, и будут работать.

Технические характеристики ревизии 3.0 следующие:

- скорость передачи данных – до 4800 Мбит/с (режим SuperSpeed), до 480 Мбит/с (режим Hi-speed), до 12 Мбит/с (режим Full-Speed) или до 1,5 Мбит/с (режим Low-Speed);
- двухшинная архитектура (шина Low-Speed/Full-Speed/High-Speed и отдельно шина SuperSpeed);
- асинхронная передача данных;
- дуплексный обмен в режиме SuperSpeed (одновременно возможна передача и прием данных) и симплексный в остальных режимах.
- максимальная длина кабеля – 3 метра;
- максимальное количество подключённых устройств к одному контроллеру (включая разнотипные) – 127;
- напряжение питания для периферийных устройств – 5 В;
- максимальная сила тока – 900 мА;
- улучшенная система управления питанием, позволяющая экономить энергию при бездействии оконечных устройств;
- кабель состоит из восьми линий связи. Четыре линии связи такие же, как и в USB 2.0. Дополнительные две линии связи – для приема данных, и две – для передачи в режиме SuperSpeed, и две – заземляющие оплетки: одна – для кабелей передачи данных в режиме Low-Speed/Full-Speed/High-Speed, и одна – для кабелей, используемых в режиме SuperSpeed.

IEEE 1394 (Institute of Electrical and Electronic Engineers) – стандарт последовательной высокоскоростной шины, принятый в 1995 году. Различные компании называют шины, разработанные по этому стандарту, по-разному. У Apple – FireWire, у Sony – i.LINK, у Yamaha – mLAN, у Texas Instruments – Lynx, у Creative – SB1394, и так далее.

Из-за этого часто возникает путаница, но, несмотря на разные названия, это одна и та же шина, работающая по одному стандарту.

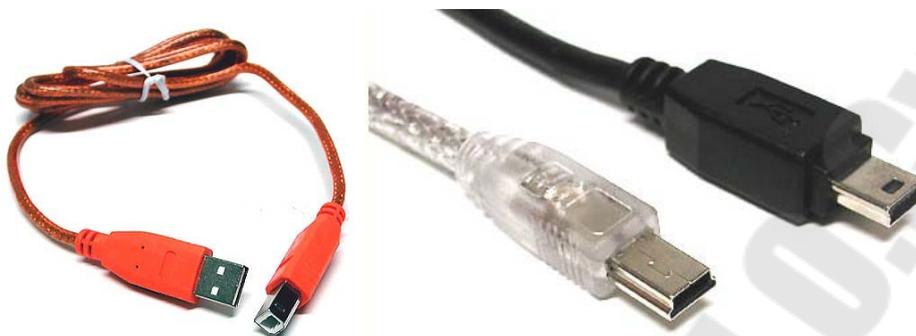


Рисунок 36 – Интерфейс USB (тип А, тип В, мини USB)

Эта шина предназначена для подключения высокоскоростных периферийных устройств, таких как внешние жесткие диски, цифровые видеокамеры, музыкальные синтезаторы и так далее.

Основные технические характеристики шины следующие:

- максимальная скорость передачи данных изменяется от 400 Мбит/с, у ревизии IEEE 1394, до 3.2 Гбит/с, у ревизии IEEE 1394b;
- максимальная длина связи между двумя устройствами изменяется от 4.5 метров, у ревизии IEEE 1394, до 100 метров, у ревизии IEEE 1394b и старше;
- максимальное количество устройств, последовательно подключаемых к одному контроллеру, – 64, в том числе и IEEE-концентраторы. При этом все подключаемые устройства делят между собой пропускную способность шины. К каждому IEEE-концентратору можно подключить еще 16 устройств. Вместо подключения устройства можно подключить шинную перемычку, через которую можно будет подключить еще 63 устройства. Всего можно подключить до 1023 шинных перемычек, что позволит организовать сеть из 64 449 устройств. Больше устройств подключить нельзя, так как в стандарте IEEE 1394 каждое устройство имеет 16-разрядный адрес;
- возможность объединения в сеть нескольких компьютеров;
- горячее подключение и отключение устройств;
- возможность использования устройств, питающихся от шины и не имеющих собственного источника питания. При этом максимальная сила тока – до 1.5 Ампер, а напряжение – от 8 до 40 Вольт.

Ethernet – стандарт построения компьютерных сетей на базе технологии пакетной передачи данных, разработанный в 1973 году Робертом Меткалфом из корпорации Xerox PARC.



Рисунок 37 – Интерфейс IEEE 1394

Стандарт определяет виды электрических сигналов и правила проводных соединений, описывает форматы кадров и протоколы передачи данных.

Существуют десятки разных ревизий стандарта, но наиболее распространенными на сегодняшний день является группа стандартов: Fast Ethernet и Gigabit Ethernet.

Fast Ethernet обеспечивает передачу данных со скоростью до 100 Мбит/с. И дальность передачи данных в одном сегменте сети без повторителей – от 100 метров (группа стандартов 100BASE-T, использующая для передачи данных витую пару) до 10 километров (группа стандартов 100BASE-FX, использующая для передачи данных одномодовое оптоволокно).

Gigabit Ethernet обеспечивает передачу данных со скоростью до 1 Гбит/с. И дальность передачи данных в одном сегменте сети без повторителей – от 100 метров (группа стандартов 1000BASE-T, использующая для передачи данных четыре витых пары) до 100 километров (группа стандартов 1000BASE-LH, использующая для передачи данных одномодовое оптоволокно).

Для передачи больших объемов информации существуют стандарты десяти, сорока и ста гигабитного Ethernet, работающего на базе оптоволоконных линий связи.

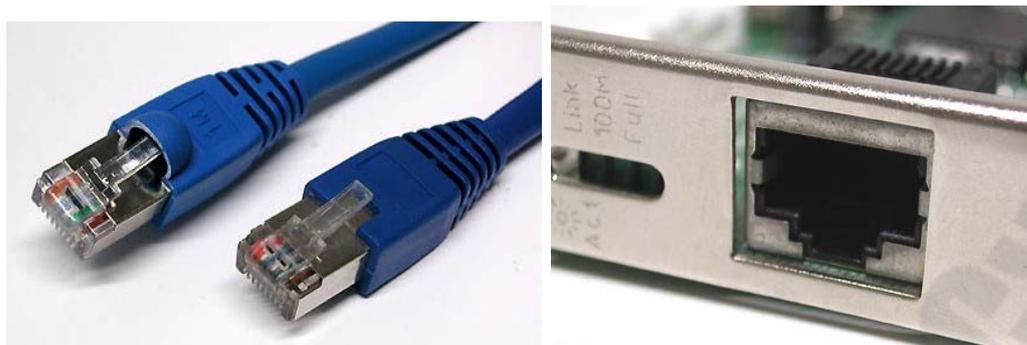


Рисунок 38 – Элементы интерфейса Ethernet

Wi-Fi – беспроводная линия связи, созданная в 1991 году в Нидерландской компанией NCR Corporation/AT&T. WiFi основывается на стандарте IEEE 802.11. и используется, как для связи с периферийными устройствами, так и для организации локальных сетей.

Wi-Fi позволяет соединять два компьютера или компьютер и периферийное устройство напрямую по технологии точка-точка, либо организовывать сеть с использованием точки доступа, к которой одновременно могут подключаться несколько устройств.

Максимальная скорость передачи данных зависит от используемой ревизии стандарта IEEE 802.11, но на практике будет значительно ниже заявленных параметров, из-за накладных расходов, наличия препятствий на пути распространения сигнала, расстояния между источником сигнала и приемником и других факторов. На практике средняя пропускная способность в лучшем случае будет в 2-3 раза меньше заявленной максимальной пропускной способности. Существует множество других интерфейсов для связи с периферийными устройствами и организации локальных сетей. Однако они редко встраиваются в материнскую плату и обычно используются в виде плат расширения. Поэтому эти интерфейсы, наравне с описанными выше, будем рассматривать в статье посвященной межмашинному взаимодействию, а сейчас перейдем к описанию интерфейсов связи южного моста с жесткими дисками. В зависимости от ревизии стандарта пропускная способность Wi-Fi следующая:

Ревизия стандарта	Тактовая частота	Заявленная максимальная мощность	Средняя скорость передачи данных на практике	Дальность связи в помещении/открытой местности
-------------------	------------------	----------------------------------	--	--

802.11a	5 ГГц	54 Мбит/с	18.4 Мбит/с	35/120 м
802.11b	2.4 ГГц	11 Мбит/с	3.2 Мбит/с	38/140 м
802.11g	2.4 ГГц	54 Мбит/с	15.2 Мбит/с	38/140 м
802.11n	2.4 или 5 ГГц	600 Мбит/с	59.2 Мбит/с	70/250 м

6.5. Интерфейсы шин связи южного моста с жесткими дисками.

Первоначально для связи с жесткими дисками использовался интерфейс АТА, но позже он был вытеснен более удобными и современными интерфейсами SATA и SCSI. Приведем краткий обзор этих интерфейсов.

АТА (Advanced Technology Attachment) или ПАТА (Parallel ATA) – параллельный интерфейс связи, разработанный в 1986 году компанией Western Digital. В то время он назывался IDE (Integrated Drive Electronics), но позже был переименован в АТА, а с появлением в 2003 году интерфейса SATA, ПАТА был переименован в РАТА.

Использование интерфейса РАТА подразумевает, что контроллер жесткого диска располагается не на материнской плате или в виде платы расширения, а встроен в сам жесткий диск. На материнской плате, а именно в южном мосте, располагается только контроллер канала РАТА.

Для подключения жестких дисков с интерфейсом РАТА обычно используется 40-проводный шлейф. С введением режима РАТА/66 появилась его 80-проводная версия. Максимальная длина шлейфа – 46 см. К одному шлейфу можно подключить и два устройства, при этом одно из них обязательно должно быть ведущим, а другое – ведомым.

Существует несколько ревизий интерфейса РАТА, отличающиеся скоростью передачи данных, режимами работы и другими особенностями. Ниже приведены основные ревизии интерфейса РАТА.

Ревизия интерфейса	Теоретическая максимальная пропускная способность	Год выхода окончательной версии ревизии
РАТА	8.3 Мбайт/с	1994

Ultra PATA	16.6 Мбайт/с	1996
PATA/33	33.3 Мбайт/с	1998
PATA/66	66.7 Мбайт/с	2000
PATA/100	100 Мбайт/с	2002
PATA/133	133 Мбайт/с	2003

На практике пропускная способность шины гораздо ниже заявленной теоретической пропускной способности, из-за накладных расходов на организацию протокола обмена и других задержек. К тому же, если к шине подключено два жестких диска, то пропускная способность будет делиться между ними.

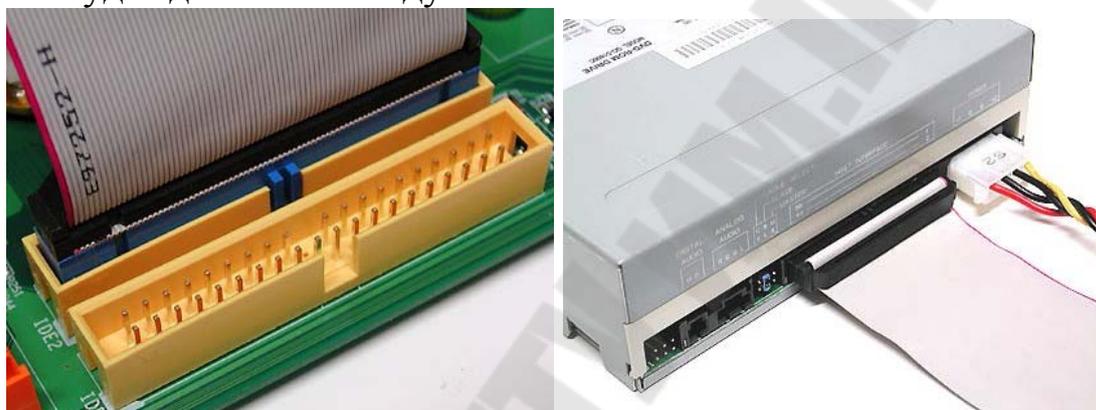


Рисунок 39 – Элементы интерфейса PATA

В 2003 году на замену интерфейса PATA пришел интерфейс SATA. **SATA (Serial ATA)** – последовательный интерфейс связи южного моста с жесткими дисками, разработанный в 2003 году.

При использовании интерфейса SATA каждый накопитель подключается своим кабелем. Причем кабель значительно уже и удобнее кабеля, используемого в интерфейсе PATA, и имеет максимальную длину до 1 метра. Отдельным кабелем на жесткий диск подается питание.

И даже, несмотря на то, что общее количество кабелей увеличивается, по сравнению с интерфейсом PATA, так как каждый накопитель подключается двумя кабелями, свободного места внутри системного блока становится значительно больше. Это приводит к улучшению КПД системы охлаждения, упрощает доступ к различным элементам компьютера, да и выглядит изнутри системный блок более презентабельно.

На данный момент существует три основных ревизии интерфейса SATA. В таблице ниже приведены основные параметры ревизий.

Ревизия интерфейса	Максимальная теоретическая/полезная пропускная способность	Тактовая частота	Год выхода
SATA 1.0	1.5/1.2 Гбит/с	1.5 ГГц	2003
SATA 2.0	3/2.4 Гбит/с	3 ГГц	2005
SATA 3.0	6/4.8 Гбит/с	6 ГГц	2008

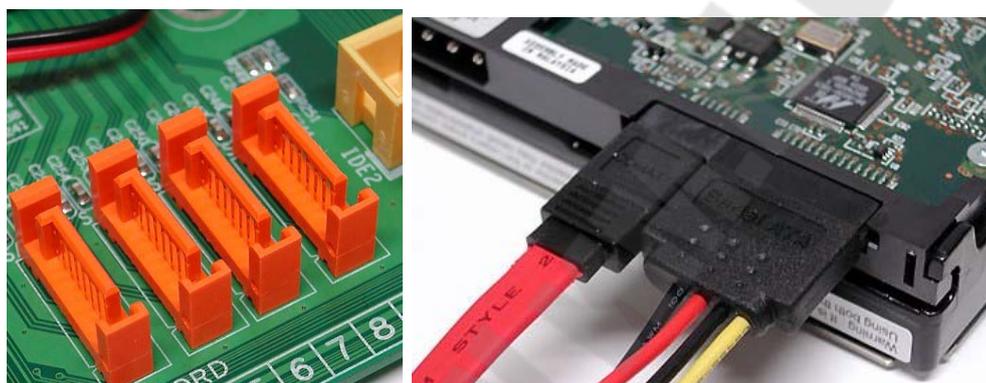


Рисунок 40 – Элементы интерфейса SATA

SCSI (Small Computer System Interface) – универсальная шина для подключения высокоскоростных устройств, таких как: жесткие диски, приводы DVD и Blue-Ray, сканеры, принтеры и так далее. Шина обладает высокой пропускной способностью, но сложно устроенная и дорогостоящая. Поэтому в основном применяется в серверах и промышленных вычислительных системах.

Первая ревизия интерфейса была представлена в 1986 году. На данный момент существует около 10 ревизий шины. В таблице ниже приведены основные параметры наиболее популярных ревизий.

Ревизия интерфейса	Разрядность	Частота передачи данных	Макс. пропускная способность	Длина кабеля (м)	Макс. кол-во устройств	Год выхода
SCSI-1	8 бит	5 МГц	40 МБит/с	6	8	1986
SCSI-2	8 бит	10 МГц	80 МБит/с	3	8	1989
SCSI-3	8 бит	20 МГц	160 МБит/с	3	8	1992
Ultra-2 SCSI	8 бит	40 МГц	320 МБит/с	12	8	1997

Ultra-3 SCSI	16 бит	80 МГц	1.25 ГБит/с	12	16	1999
Ultra-320 SCSI	16 бит	160 МГц	2.5 ГБит/с	12	16	2001
Ultra-640 SCSI	16 бит	320 МГц	5 ГБит/с	12	16	2003

Увеличение пропускной способности параллельного интерфейса сопряжено с рядом трудностей и, в первую очередь, это защита от электромагнитных помех. А каждая линия связи является источником электромагнитных помех. Чем больше линий связи будет в параллельной шине, тем больше они будут создавать помех друг для друга. Чем выше частота передачи данных, тем больше электромагнитных помех, и тем сильнее они оказывают влияние на передачу данных.

Кроме этой проблемы есть менее существенные, такие как:

- сложность и высокая цена производства параллельной шины;
- проблемы в синхронной передаче данных по всем линиям шины;
- сложность устройства и высокая цена контроллеров шины;
- сложность организации полнодуплексного устройства;
- сложность обеспечения каждого устройства своей шиной и т.д.

В результате, проще отказаться от параллельного интерфейса в пользу последовательного с большей тактовой частотой. При необходимости можно использовать несколько последовательных линий связи, располагающихся дальше друг от друга и защищенных экранирующей оплеткой. Так поступили при переходе от параллельной шины PCI к последовательной PCI express, от PATA к SATA. По тому же пути развития пошла и шина SCSI. Так в 2004 году появился интерфейс SAS.

SAS (Serial Attached SCSI) – последовательная шина типа точка-точка, заменившая параллельную шину SCSI. Для обмена по шине SAS используется командная модель SCSI, но пропускная способность увеличена до 6 Гбит/с (ревизия SAS 2, вышедшая в 2010 году).

В 2012-2013 году планировался выпуск ревизии SAS 3, обладающей пропускной способностью – 12 Гбит/с, однако устройства, поддерживающие эту ревизию, в массовом порядке начнут появляться не раньше 2014 - 2015 года.

Также не стоит забывать, что шина SCSI была общая, позволяющая подключать до 16 устройств, и все устройства делили между собой пропускную способность шины. А шина SAS использует топологию

точка-точка. А, следовательно, каждое устройство подключается своей линией связи и получает всю пропускную способность шины.

Контроллер SCSI и SAS встраивается в материнскую плату редко, так как они достаточно дорогостоящие. Обычно они подключаются, как платы расширения к шине PCI или PCI express.

6.6. Интерфейсы связи с медленными компонентами материнской платы

Для связи с медленными компонентами материнских плат, например, с пользовательским ПЗУ или контроллерами низкоскоростных интерфейсов, используются специализированные шины, такие как: ISA, MCA, LPS и другие.

Шина ISA (Industry Standard Architecture) – 16-разрядная шина, разработанная в 1981 году. ISA работала на тактовой частоте 8 МГц, и обладала пропускной способностью до 8 Мбайт/с. Шина давно устарела и на практике не используется.

Альтернативой шине ISA была шина MCA (Micro Channel Architecture), разработанная в 1987 году компанией Intel. Эта шина была 32-х разрядная с частотой передачи данных – 10 МГц, и пропускной способностью – до 40 Мбит/с. Поддерживала технологию Plug and Play. Однако закрытость шины и жесткая лицензионная политика компании IBM сделали ее непопулярной. На данный момент шина на практике не используется.

Настоящей заменой для ISA стала шина LPC (Low Pin Count), разработанная компанией Intel в 1998 году и используемая по сей день. Работает шина на тактовой частоте – 33,3 МГц, что обеспечивает пропускную способность в 16,67 МБит/с.

Пропускная способность шины совсем небольшая, но для связи с медленными компонентами материнской платы вполне достаточная. С помощью этой шины к южному мосту подключается многофункциональный контроллер (Super I/O), в состав которого входят контроллеры медленных интерфейсов связи и периферийных устройств:

- параллельного интерфейса;
- последовательного интерфейса;
- инфракрасного порта;
- интерфейса PS/2;
- накопителя на гибком магнитном диске и других устройств.

Также Шина LPC обеспечивает доступ к BIOS'у (базовой системе ввода вывода).

6.7. BIOS (Basic Input-Output System)

BIOS (Basic Input-Output System - базовая система ввода-вывода) – это программа, прошитая в постоянное запоминающее устройство (ПЗУ). В нашем случае ПЗУ встроено в материнскую плату, однако своя версия BIOS присутствует почти во всех элементах ЭВМ (в видеокарте, в сетевой карте, дисковых контроллерах и т.д.), да и вообще почти во всем электронном оборудовании (и в принтере, и в видеокамере, и в модеме, и т.д.).

BIOS материнской платы отвечает за проверку работоспособности контроллеров, встроенных в материнскую плату, и большинства устройств, подключенных к ней (процессора, памяти, видеокарты, жестких дисков и т.д.). Происходит проверка при включении питания компьютера в программе Power-On Self Test (POST).

Далее BIOS производит инициализацию контроллеров, встроенных в материнскую плату, и некоторых подключенных к ним устройств, и устанавливает их базовые параметры работы, например, частоту работы системной шины, процессора, контроллера ОЗУ, параметры работы жестких дисков, контроллеров встроенных в материнскую плату и т.д.

Если проверяемые контроллеры и аппаратура исправны и настроены, то BIOS передает управление операционной системе.

Пользователи могут управлять большинством параметров работы BIOS и даже обновлять его.

Обновление BIOS требуется очень редко, если, например, разработчиками обнаружена и устранена принципиальная ошибка в программе инициализации какого-либо из устройств, либо если требуется поддержка нового устройства (например, новой модели процессора). Но, в большинстве случаев, выход нового типа процессора или памяти требует кардинального «абгрейда» компьютера. Скажем за это производителям электроники «спасибо».

Для настройки параметров BIOS предусмотрено специально меню, войти в которое можно, нажав сочетание клавиш, указанное на экране монитора во время проведения тестов POST. Обычно для входа в меню настройки BIOS требуется нажать клавишу DEL.

В этом меню можно установить системное время, параметры работы дисководов и жестких дисков, увеличить (или уменьшить) тактовую частоту процессора, памяти и системной шины, шин связи и на-

строить другие параметры работы компьютера. Однако тут стоит быть крайне осторожным, так как неправильно установленные параметры могут привести к ошибкам в работе или даже вывести компьютер из строя.

Все настройки BIOS хранятся в энергозависимой памяти CMOS, работающей от батарейки или аккумулятора, установленного на материнской плате. Если батарейка или аккумулятор разрядились, то компьютер может не включиться или работать с ошибками. Например, будет установлено неверное системное время или параметры работы некоторых устройств.

7. УСТРОЙСТВА ВВОДА/ВЫВОДА ИНФОРМАЦИИ В ИНФОРМАЦИОННЫХ СИСТЕМАХ

Человек взаимодействует с информационными системами главным образом через устройства ввода-вывода (input-output devices). Прогресс в области информационных технологий достигается не только благодаря возрастающей скорости процессоров и емкости запоминающих устройств, но также за счет совершенствования устройств ввода и вывода данных. Устройства ввода-вывода называются также периферийными устройствами (peripheral devices).

7.1. Устройства ввода

Клавиатура (keyboard) – традиционное устройство ввода данных в компьютер. Клавиатурами оснащены как персональные компьютеры, так и терминалы мэйнфреймов. Клавиатура современного компьютера содержит обычно 101 или 102 клавиши (может быть и больше для мультимедийных клавиатур или специализированных), разделенные на 4 блока: - алфавитно-цифровой блок – содержит клавиши латинского и национального алфавитов, а также клавиши цифр и специальных символов;

- блок управляющих клавиш;
- блок расширенной цифровой клавиатуры;
- блок навигации.

Мышь (трекбол) – устройство «графического» управления. Манипулятор «мышь» (в обиходе просто «мышь» или «мышка») — одно из указательных устройств ввода (англ. pointing device), обеспечивающих интерфейс пользователя с компьютером.

Мышь воспринимает своё перемещение в рабочей плоскости (обычно — на участке поверхности стола) и передаёт эту информацию компьютеру. Программа, работающая на компьютере, в ответ на перемещение мыши производит на экране действие, отвечающее направлению и расстоянию этого перемещения.

В дополнение к детектору перемещения мышь имеет от одной до трех (или более) кнопок, а также дополнительные элементы управления (колёса прокрутки, потенциометры, джойстики, трекболы, клавиши и т. п.), действие которых обычно связывается с текущим положением курсора (или составляющих специфического интерфейса).

Наибольшее распространение получили оптико-механические мыши, оптические и лазерные.

Мышь можно подключить одним из четырех способов:

1. При помощи специального адаптера в виде платы расширения (устанавливается в материнскую плату)
2. При помощи стандартного последовательного порта (COM1 или COM2 в зависимости от размера гнезда на кабеле мыши — 9 или 25 контактный) — устарел
3. При помощи порта PS/2 (6-ти контактный круглый разъем, зеленого цвета)
4. При помощи порта USB (прямоугольный).

Сенсорные экраны (touch screens) предназначены для тех, кто не может пользоваться обычной клавиатурой. Пользователь может ввести символ или команду прикосновением пальца к определенной области экрана. По технологии изготовления различают резистивные и емкостные. Сенсорные экраны используются в основном на мобильных устройствах, на складах продукции, в ресторанах, супермаркетах.

Устройства автоматизированного ввода информации считывают информацию с носителя, где она уже имеется. Примерами таких систем могут служить кассовые терминалы, сканеры штрих-кодов и другие системы оптического распознавания символов. Одно из преимуществ устройств автоматизированного ввода данных состоит в том, что при их использовании исключаются некоторые ошибки, неизбежные при вводе информации с клавиатуры. Сканер штрих-кодов делает менее чем одну ошибку на 10000 операций, в то время как обученный наборщик ошибается один раз при вводе каждых 1000 строк.

Основные виды устройств автоматизированного ввода информации – системы распознавания магнитных знаков, системы оптического распознавания символов, системы ввода информации на базе светового пера, сканеры, системы распознавания речи, сенсорные датчики и устройства видеозахвата.

Системы распознавания магнитных знаков (Magnetic Ink Character Recognition, MICR) используются в основном в банковской сфере. В нижней части обычного банковского чека находится код, нанесенный специальными магнитными чернилами. В коде содержится номер банка, номер расчетного счета и номер чека. Система считывает информацию, преобразовывает ее в цифровую форму и передает в банк для обработки.

Системы оптического распознавания символов (Optical Character Recognition, OCR) преобразуют специальным образом нанесенную на носитель информацию в цифровую форму. Наиболее широко используемые устройства этого типа – сканеры штрих-кодов (bar-code scanners), которые применяются в кассовых терминалах магазинов. Эти системы используются также в больницах, библиотеках, на военных объектах, складах продукции и в компаниях по перевозке грузов. В дополнение к данным, идентифицирующим предмет, на который нанесен штрих-код, последний может содержать информацию о време-

ни, дате и физическом положении предмета; таким образом, можно, например, отслеживать передвижение груза.

Ручные устройства распознавания информации, такие как перьевые планшеты, особенно полезны для людей, работающих в сферах сбыта продукции и сервиса – такие работники избегают "общения" с клавиатурой. Устройства перьевого ввода обычно содержат плоский экран и световое перо, похожее на шариковую ручку. Перьевые планшеты преобразуют буквы и цифры, написанные пользователем на экране, в цифровую форму, и передают эти данные в компьютер для обработки. Например, United Parcel Service (UPS), известнейшая в мире компания по доставке грузов, заменила обычные планшеты с листками бумаги, использовавшиеся водителями, на портативные перьевые планшеты. Эти устройства используются для подтверждения заказов, и передачи другой информации, необходимой для погрузки и доставки грузов. К недостаткам систем данного вида следует отнести недостаточную точность распознавания информации, написанной от руки.

Сканеры (scanners) преобразуют в цифровую форму графическую информацию (рисунки, чертежи и пр.) и большие объемы текстовой информации. Системы распознавания речи (voice input devices) преобразуют в цифровую форму произносимые пользователем слова. Существует два режима работы подобных устройств. В режиме управления (command mode) вы произносите команды (такие как "открыть документ", "запустить программу" и т.д.), которые выполняются компьютером. В режиме диктовки (dictation mode) можно надиктовывать компьютеру любой текст. К сожалению, точность распознавания речи таких систем оставляет желать лучшего. Человеческий голос имеет множество оттенков, на точность распознавания может повлиять интонация, громкость речь, окружающий шум, даже банальный насморк. Тем не менее, работа над совершенствованием этих устройств ввода информации продолжается и, несомненно, у них большое будущее. Некоторые отделения Почтовой службы США используют системы распознавания речи для повышения эффективности труда работников, занятых упаковкой и сортировкой почтовых грузов. Вместо того чтобы вводить ZIP-код, работник произносит его, в то время как его руки заняты упаковкой.

Сенсорные датчики (sensors) – это устройства для ввода в компьютер пространственной информации. Например, корпорация General Motors использует сенсоры в своих легковых автомобилях для пере-

дачи в бортовой компьютер машины данных об окружающем пространстве и маршруте. Сенсорные датчики также нашли применение в системах виртуальной реальности, игровых приставках и симуляторах.

Устройства видеозахвата (video capture devices) представляют собой небольшие цифровые видеокамеры, соединенные с компьютером. Устройства видеозахвата применяются в основном в системах видеоконференций, которые получают все большее распространение.

Благодаря развитию локальных сетей и Интернет, появилась возможность организовывать *видеоконференцсвязь*, находясь в любой точке планеты.

7.2. Устройства вывода

7.2.1. Мониторы и проекторы

Первое что нужно решить, для каких именно целей, в большей мере, будет использоваться монитор. Здесь не обойтись без поверхностного ознакомления с существующими видами матриц жидкокристаллических мониторов. Существует как минимум три основных вида жидкокристаллических матриц мониторов.

Матрица — это массив пикселей, пропускающих и фильтрующих свет. Это основная часть ЖК-монитора и она определяет 90% его качества. Современные ЖК-мониторы оснащаются тремя различными типами матриц, каждый тип вне зависимости от конкретной модели имеет одинаковые достоинства и недостатки по отношению друг к другу, от конкретной модели зависит только выраженность этих качеств и недостатков.

1) TN — самый старый и дешевый в производстве тип матриц, для него характерно минимальное время отклика, относительно плохая цветоподача, маленькие углы обзора с заметным искажением цветов при изменении угла наблюдения (особенно по вертикали — «эффект негатива»), невысокая контрастность, серый «чёрный» цвет. Хорошо подходит для динамичных игр, если, конечно, цветоподача конкретной модели находится на приемлемом для виртуальных развлечений уровне.

2) VA (MVA, PVA и прочие названия с -VA) — время отклика пикселя большее, чем на TN, но при этом достаточно хорошая цветоподача, большие углы обзора без существенного искажения цветов при изменении угла наблюдения, высокая контрастность, по цене до-

роже TN. Можно сказать, золотая середина, подходит для всего и имеет относительно невысокую цену.

3) S-IPS — большее время реакции матрицы, чем на VA и, соответственно, TN, но при этом отличная цветоподача, почти идеальные углы обзора (практически без видимых искажений цветов при уменьшении угла наблюдения), хороший контраст. Наилучшим образом подходит для всего, где не важен быстрый отклик пикселя.

Разделение мониторов на эти три вида условно, поскольку у каждой модели свои параметры от которых и следует отталкиваться при выборе монитора.

Основные технические показатели монитора:

1. Типы матриц — технология, по которой изготовлен ЖК-дисплей; основные – TN (TN+film), IPS, MVA/PVA.

2. Время отклика (время реакции матрицы) — минимальное время, необходимое пикселю для изменения своей яркости, чем оно меньше, тем лучше. Определяется в миллисекундах (мс).

3. Разрешение — горизонтальный и вертикальный размеры, выраженные в пикселях. В отличие от ЭЛТ-мониторов, ЖК имеют одно фиксированное разрешение, остальные достигаются интерполяцией.

4. Размер точки (размер пикселя) — расстояние между центрами соседних пикселей. Непосредственно связан с физическим разрешением.

5. Соотношение сторон экрана (пропорциональный формат) — отношение ширины к высоте (5:4, 4:3, 3:2 (15÷10), 8:5 (16÷10), 5:3 (15÷9), 16:9 и др.)

6. Контрастность — отношение яркостей самой светлой и самой тёмной точек при заданной яркости подсветки. В некоторых мониторах используется адаптивный уровень подсветки с использованием дополнительных ламп, приведённая для них цифра контрастности (так называемая динамическая) не относится к статическому изображению.

7. Яркость — количество света, излучаемое дисплеем, обычно измеряется в канделах на квадратный метр.

8. Угол обзора — это максимальный угол, с которого зритель способен различить четкое изображение на экране ЖК-монитора.

9. Диагональ монитора (размер) — это длина диагонали по внешним углам экрана. Определяется в дюймах — 1 дюйм = 2,54 см.

В мультимедийном проекторе проекционная лампа, ЖК-матрица и оптическая система конструктивно размещаются в одном кор-

пусе, что делает их похожими на диапроекторы, предназначенные для просмотра слайдов или диафильмов. По принципу действия мультимедийный проектор не отличается от оверхед-проектора: изображение создается с помощью мощной проекционной лампы и встроенного в проектор электронно-оптического модулятора, управляемого сигналом видеоадаптера ПК, а затем посредством оптической системы проецируется на внешний экран. Основным отличием в мультимедийных проекторах является конструкция модулятора и способы построения и переноса изображения на экран. В зависимости от конструкции модулятора проекторы бывают следующих типов: TFT-проекторы; полисиликоновые проекторы и DMD/DLP-проекторы.

В зависимости от способа освещения модулятора мультимедийные проекторы подразделяют на проекторы просветного и отражательного типов.

Основным элементом установки является миниатюрная ЖК-матрица, выполненная по технологии TFT, как и ЖК-экран плоскопанельного цветного монитора. Равномерное освещение поверхности ЖК-матрицы достигается за счет применения системы линз, называемой конденсором.

В настоящее время наиболее используемой в конструкциях ЖК-проекторов отражательного типа является технология DMD/DLP, разработанная фирмой Texas Instruments.

В *DMD/DLP-проекторах отражательного типа* излучение источника света модулируется изображением при отражении от матрицы. В DMD/DLP-проекторах в качестве отражающей поверхности используется матрица, состоящая из множества электронно - управляемых микрозеркал, размер каждого из которых около 1 мкм. Каждое микрозеркало имеет возможность отражать падающий на него свет либо в объектив, либо в поглотитель, что определяется уровнем поданного на него электрического сигнала. При попадании света в объектив образуется яркий пиксел экрана, а в поглотитель - темный. Такие матрицы обозначаются аббревиатурой DMD (Digital Micromirror Device - цифровой микрозеркальный прибор), а технология, на которой основан их принцип действия, — DLP (*Digital Light Processing* — цифровая обработка света).

Как правило, в одной DMD-матрице содержится около 848 x 600 = 508 800 микрозеркал, что превосходит SVGA-разрешение (800x600 = 480 000 пикселей).

По сравнению с ЖК-технологиями технология DLP обладает следующими преимуществами: практически полным отсутствием зернистости изображения, высокой яркостью и равномерностью ее распределения. К недостаткам одноматричных DMD-проекторов следует отнести заметное мелькание кадров

7.2.2. Принтеры

Принтеры предназначены для вывода информации на твердые носители, большей частью на бумагу. Существует большое количество разнообразных моделей принтеров, которые различаются по принципу действия, интерфейсу, производительности и функциональным возможностям. По принципу действия различают: матричные, струйные и лазерные принтеры.

Виды принтеров:

- Лазерные принтеры;
- Струйные принтеры;
- Матричные принтеры;
- МФУ (Многофункциональное устройство);
- Светодиодные принтеры;
- 3D – принтеры;
- Другие принтеры;
- Плоттеры.

В лазерных принтерах используется электрографический принцип создания изображения. Процесс печати включает в себя создание невидимого рельефа электростатического потенциала в слое проводника с последующей его визуализацией. Визуализация осуществляется с помощью частиц сухого порошка-тонера, наносимого на бумагу.

Тонер представляет из себя смешанную смесь частиц железа (металло-порошка) и частиц пластмассы.

Наиболее важными частями лазерного принтера являются:

- полупроводниковый барабан (Представляет собой металлический цилиндр, покрытый тонкой светочувствительной плёнкой. Покрытие может заряжаться как «+» так и в «-». Когда луч света попадает на поверхность барабана, проводимость освещённого участка повышается и заряд с него исчезает, образуя не заряженную золу. Для формирования на поверхности барабана равномерного статического заряда используется тонкий зарядный провод или сетка);

- лазер;

- картридж (барабан, контейнер с тонером, отстойник, направлятель бумаги (ракель)).

- прецизионная оптико-механическая система, перемещающая луч;

- собственный мощный процессор(или контроллер) и оперативная память;

Лазерный принтер является достаточно сложным устройством. Выдача изображения на печаль, взаимодействие с компьютером, выполнение настроек в режиме диалога с пользователем, проведение самотестирования и др. действия выполняются в принтере под управлением спец. контроллера.

Контроллер выполняет функции:

- принимает от компьютера изображение;

- запоминает изображение в ОП принтера;

- тестирует внутренние узлы принтера;

- обеспечивает работу с панелью управления принтером;

- посылает управляющие сигналы всем функциональным блокам принтера.

Принцип печати лазерного принтера. С помощью зарядного провода на поверхность светочувствительного барабана равномерно наносится статический заряд. Полупроводниковый лазер низкой мощности генерирует узкий направленный луч, который, отражаясь от вращающегося шестигранного зеркала, пробегает вдоль поверхности барабана. При освещении лучом лазера управляемого специальным контроллером, поверхность барабана разряжается в точках соответствующих строке изображения, формируя ее невидимый образ на поверхности барабана. Когда луч лазера попадает на предварительно заряженный барабан, заряд «стекает» с освещённой поверхности. Таким образом, освещаемые и не освещаемые лазером участки барабана имеют разный заряд. Формируется на поверхности барабана скрытое невидимое человеку изображение. Затем происходит проявление изображения, т. е. превращение скрытого электронного изображения в видимое. Заряженные частицы тонера притягиваются только к тем местам барабана, которые имеют противоположный заряд по отношению к тонеру. Валиком подаётся бумага, проходит через блок термического закрепления тонера (fuser) , где тонер под воздействием температуры и давления плавится на поверхности бумаги (т. е. делается оттиск порошка с барабана на бумагу).

Остатки порошка скапливаются в отстойнике - устройстве очистки лазерного принтера.

Когда изображение с поверхности барабана переносится на бумагу, не все частички тонера прилипают к бумаге, и небольшое количество тонера остаётся на барабане. Прежде чем барабан будет заряжен для печати следующей страницы, специальный чистящий узел удаляет остатки тонера с поверхности барабана, чтобы не испачкать им следующую страницу.

Качество печати лазерных принтеров зависит, в основном, от разрешающей способности принтера, т. е. количества точек, печатаемых на каждый дюйм:

- 300 точек на дюйм - большинство деловых документов получаются вполне удовлетворительными, но мелкие буквы и полутоновые изображения печатаются плохо;

- 600 точек на дюйм – качество достаточно для деловых документов и даже (хотя и с натяжкой) для издательских нужд – при чёрно-белой полиграфии и невысоких требований компьютер печати полутоновых изображений;

- 1200 точек на дюйм – такие принтеры печатают документы с высоким качеством, и даже полутоновые изображения получаются вполне удовлетворительными;

- 1800 точек на дюйм и более – такие принтеры используются для издательских нужд при подготовке особо высококачественных изданий.

- Большинство принтеров оборудовано автоподатчиком бумаги. В одних принтерах бумага загружается сверху (вертикальная подача), в других кладется в специальный лоток внизу (горизонтальная подача)

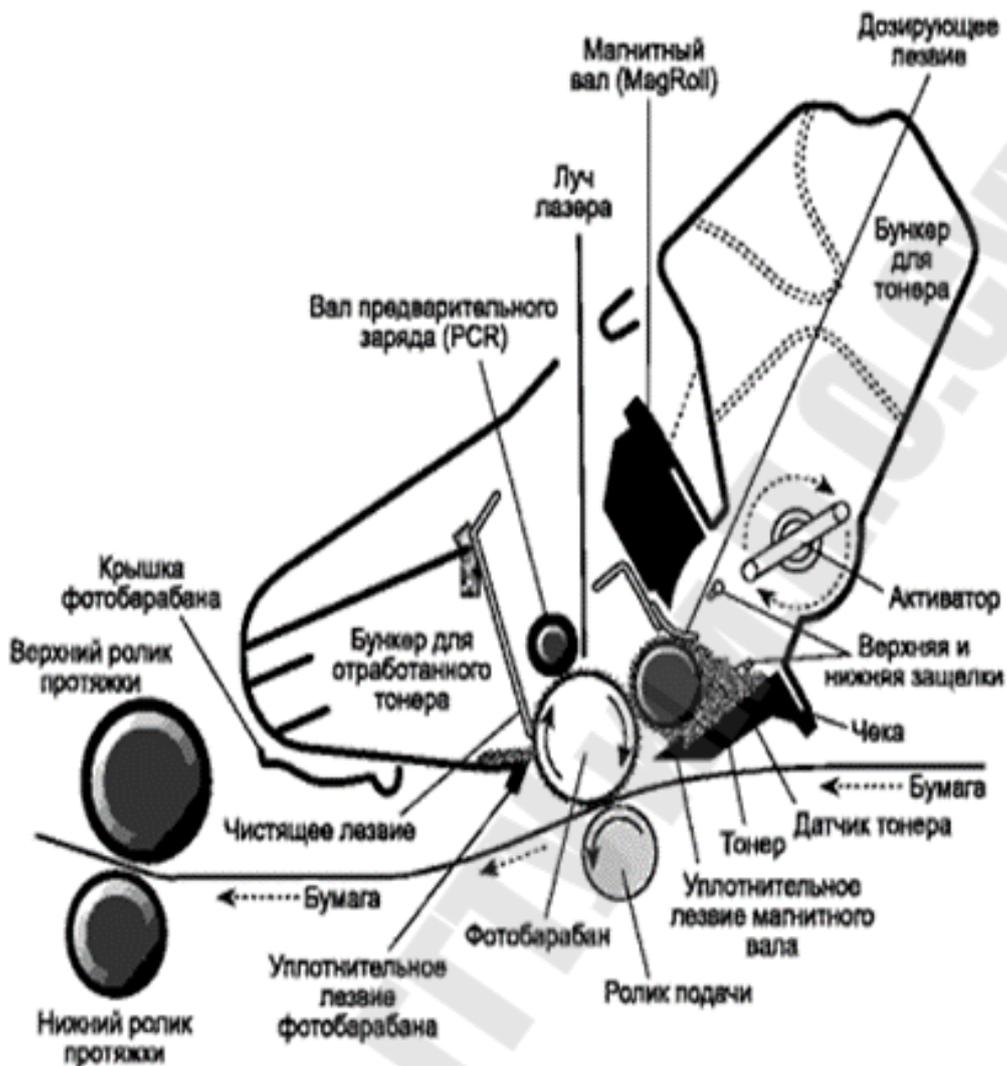


Рисунок 41 – Принцип работы лазерного принтера

Программирование работой принтера. Принтер может иметь один или оба из встроенных набор команд(языков управления принтером):

PCL (Printer Control Language) – имеет полезные встроенные эффекты и позволяет загружать масштабируемые шрифты в формате TrueType, избавляя компьютер от необходимости переводить эти шрифты в растровую форму при посылке на принтер. Работает с побитными шрифтами и графикой.

PostScript понимают язык описания страниц PostScript, разработанный фирмой Adobe. Это мощный язык, позволяющий описывать изображения в объектно-ориентированной форме. Использует математическое описание формы букв и графических изображений, а кон-

кретное расположение точек на отпечатке рассчитывается в принтере, работает с любым разрешением. Применяются в издательском деле, т.к. приготовленный для них документ может без изменений выведен на любой фотонаборный автомат. Позволяют увеличивать и уменьшать и изображение, напечатать зеркальное и негативное изображение.

Достоинства лазерных принтеров:

- Печать лазерного принтера имеет высокую устойчивость к воздействию внешних негативных факторов (вода и светостойкость) и долговечность;

- Высокая производительность, высокая интенсивность печати экономит время и деньги, это идеальный офисный принтер;

- Низкое соотношение шум\скорость, т.е. документы печатаются тихо и без лишних проблем;

- Картриджа хватает на 2000-5000 тысячи копий;

- Самая низкая удельная стоимость печатного листа (измеряется в центах за страницу) достигается низкой удельной стоимостью материалов и комплектующих;

- Не предъявляют высоких требований к качеству бумаги, однако неплотная и рыхлая бумага в них часто застревает и негативно влияет на систему подачи бумаги (требуется более частая профилактика и обслуживание).

Недостатки лазерных принтеров:

- при печати выделяется озон с характерным запахом, в воздух попадают вредные частицы порошка и с точки зрения техники безопасности это вредно для здоровья человека;

- при сжигании распечатанных документов из тонера образуется вредная на организмы двуокись углерода;

- высокое электропотребление;

- относительно высокая стоимость принтера;

- большие габариты (по сравнению со струйным);

- некоторые не поддерживают печать из ДОС (актуально для устаревших программных приложений);

- большая стоимость цветных лазерных принтеров не дает возможности для простого потребителя получать цветную печать.

Цветные лазерные принтеры значительно дороже, т.к. содержит в себе 4 обычных, действующих согласованно. Обеспечивают неплохую разрешающую способность от 600-1200 точек на дюйм, насыщенные яркие краски и приемлемую цветопередачу.

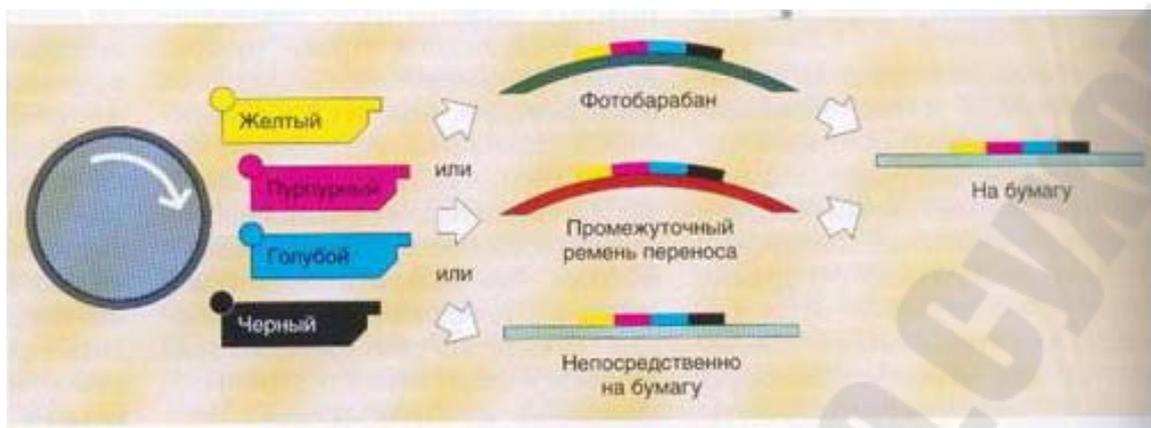


Рисунок 42 – Принципы переноса информации в цветных лазерных принтерах

Методу струйной печати уже почти сто лет. Лорд Рейли, лауреат нобелевской премии по физике, сделал свои фундаментальные открытия в области распада струй жидкости и формирования капель еще в прошлом веке, датой рождения технологии струйной печати можно считать только 1948 год. Именно тогда шведская фирма Siemens Elema подала патентную заявку на устройство, работающее как гальванометр, но оборудованное не измерительной стрелкой, а распылителем, с помощью которого регистрировались результаты измерений.

Струйная технология печати, абстрагируясь от деталей, состоит в том, что изображение наносится на бумагу путем "выстреливания" (под давлением) красителя из крохотного сопла. Одно или несколько сопел устанавливаются на печатающей головке, которая аналогично точечно-матричным принтерам в процессе работы устройства перемещается относительно бумаги.

Различают два основных типа струйных принтеров:

1. с непрерывной подачей красителя;
2. с капельным микродозатором.

В устройствах первого типа формируется непрерывный поток из маленьких капель, которые заряжаются и, пролетая через электрическое поле, отклоняются в вертикальной плоскости пропорционально их заряду. Вспомним, что горизонтальное отклонение обеспечивается перемещением печатающей головки. Капли, которые не должны делать точку на бумаге, отклоняются в специальный желоб, по которому краска возвращается в резервуар для последующего использования. Отклонение капель может быть бинарным, при котором капля

попадает либо в определенную точку по вертикали на бумаге, либо в желоб возврата. Такой принцип используется для печатающих головок с несколькими вертикально расположенными соплами. Имеются и устройства с мультиотклонением, используемым при недостаточном количестве сопел, в частности, когда печатающая головка имеет единственное сопло.

Принтеры второго типа (с капельным микро-дозатором) содержат матрицу или столбец вертикально расположенных сопел, и принцип формирования изображений в них аналогичен точечно-матричным печатающим устройствам. При горизонтальном движении печатающей головки из сопел в нужные моменты времени "выстреливаются" капли, которые попадают на бумагу. В этом случае отпадает необходимость отклонять поток капель.

Принтеры с непрерывной подачей красителя, по сравнению с устройствами с капельным микро-дозатором, имеют большее быстродействие, но и являются более сложными.

Конструктивно струйные принтеры отличаются технологиями дозирования красителя при печати и вариантом размещения головки с соплами (дюзами). Сейчас в струйных принтерах применяют две технологии дозирования красителя: *пьезоэлектрическую* и *термоструйную* ("пузырьковую"). Параметром, характеризующим возможности технологий дозирования красителя, выступает минимальный объем формируемой капли. От этого во многом зависит разрешающая способность печатающего устройства по горизонтали. В настоящее время рекорд по минимальному объему красителя принадлежит пьезоэлектрической технологии.

Качество печати струйных принтеров в основном определяется разрешением и цветовым охватом.

Ныне разрешение механизмов печати струйных принтеров достигло такого уровня (1200-4800 dpi), что в дальнейшем его повышении нет никакого смысла. В отличие от электрографической печати, где полутоновое (цветное) изображение формируется элементами растра, определяемыми линиатурой, *при струйной печати допускается наложение точек друг на друга с целью получения заданного цвета*. Тем самым, растр в обычном понимании на струйном отпечатке отсутствует, и скорее следует сравнивать его с "зерном" обычной фотографии. Снимок, распечатанный с разрешением 2880 dpi на специальной бумаге, без лупы невозможно отличить от фотоотпечатка.

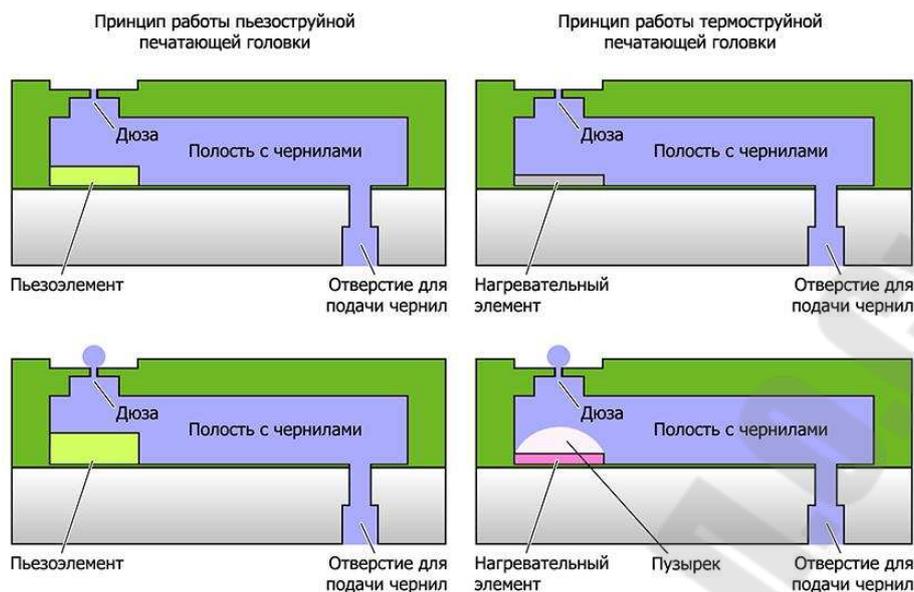


Рисунок 43 – Принципы формирования капли в струйных принтерах

Гораздо труднее на струйном принтере воспроизвести цветовой охват фотографии. На цветопередачу влияет несколько факторов: одними из главных являются характеристики используемых чернил. Хотя во всех принтерах используют чернила с цветовым спектром *СМУК*, абсолютно точно воспроизвести палитру *СМУ* невозможно, поскольку сами чернила полупрозрачны и смешиваются как с друг другом, так и с бумагой.

Особую трудность представляет воспроизведение тонов с низкой оптической плотностью. Поэтому в последних моделях так называемых *фотопринтеров* (то есть предназначенных для печати изображений фотографического качества) в дополнение к основной палитре стали применять другие цвета: *Light Magenta*, *Light Cyan*, *Gray*. Такой подход позволил существенно улучшить цветовой охват струйного отпечатка и довести его практически до уровня фотографии. Сегодня даже специалист на глаз не сможет отличить высококачественный струйный отпечаток от фотографии.

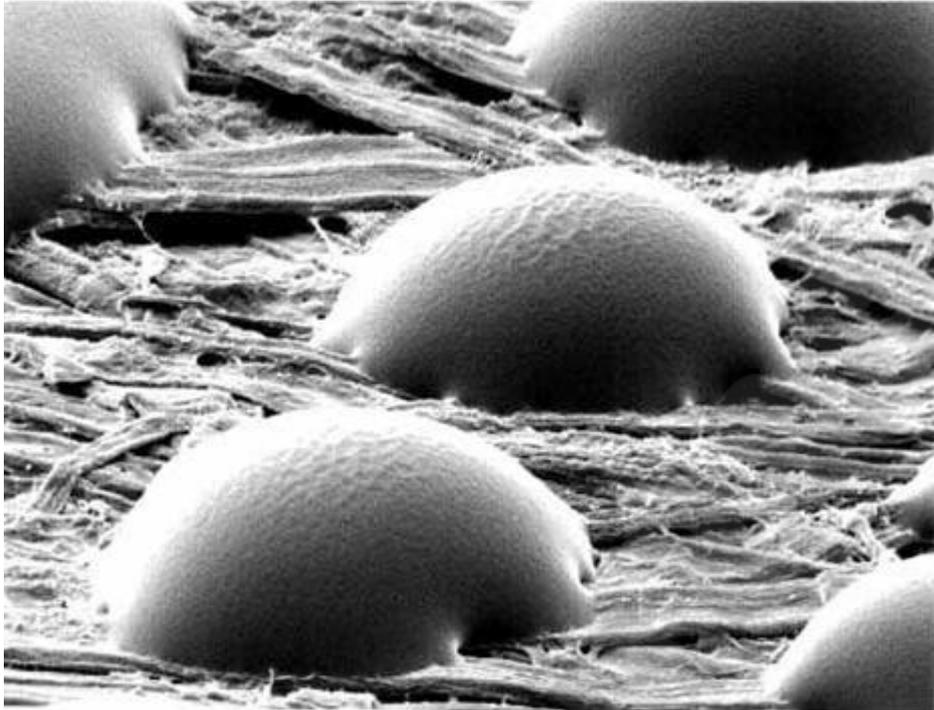


Рисунок 44 – Капли краски на бумаге под микроскопом

Зародившись во второй половине XX века, матричная технология долгие годы не знала настоящих соперников, способных потеснить ее на завоеванных позициях. На многие десятилетия эта технология стала фактическим стандартом для систем печати и, пройдя множество перевоплощений, безраздельно господствовала на рынке вплоть до 80-х гг. Серьезнейший удар - молниеносно и чрезвычайно эффективно - нанесла ей компания Hewlett-Packard выпустившая в 1984 г. первый принтер, основанный на новой и, как показало время, весьма перспективной лазерной технологии печати. Затем последовали более доступные для массового потребителя струйные аппараты, и эра матричной печати стремительно покатилась к закату.

На сегодняшний день, действительно, матричные принтеры в качестве массового продукта, нацеленного на персонального домашнего или офисного потребителя, малые и средние предприятия или рабочие группы, абсолютно утратили свою привлекательность. Причиной тому стало техническое превосходство иных технологий печати, обеспечивающих заметно более высокое качество отображения текста и графики; лучшая эргономика новых устройств, как минимум работающих гораздо тише своих матричных предшественников; и не в последнюю очередь маркетинговые программы производителей, направленные на переориентацию пользователей на новые решения.

Однако технология матричной печати все же сохранила за собой весьма значительную долю рынка систем печати. Несмотря на все достижения альтернативных решений, матричная печать остается вне конкуренции в нескольких специфических областях и, вероятно, сохранит там свои позиции еще очень долго - трудно найти что-либо близкое по производительности, надежности, простоте обслуживания и дешевизне использования, когда речь идет об очень больших объемах печати, а уж если необходимо получить несколько полностью идентичных копий документа одновременно, заменить матричный принтер просто нечем.

За сорок лет существования матричной технологии печати в производстве основанных на ней принтеров отметились великое множество фирм по всему миру - от локальных миниатюрных компаний до гигантских транснациональных корпораций. Однако жесткая конкуренция и сужение рыночной ниши заставили многих из них свернуть производство; другие же, следуя инстинкту самосохранения, объединили усилия и остаются в бизнесе до сих пор. На текущий момент производством оборудования для различных вариантов матричной печати занимаются пять широко известных компаний, в той или иной степени связывающих свое будущее с этой технологией, - Epson, Lexmark International, Oki Electric, Printronix и TallyGenicom.

Печатающая головка принтера содержит вертикальный ряд тонких металлических стержней (иглолок), 9 шт. - называется 9-точечным. Каждая иглолка управляется собственным электромагнитом. Когда на электромагнит подается импульс тока, управляемая им иглолка выталкивается и ударяет по красящей ленте. Управляющий сигнал может одновременно подаваться на любую комбинацию электромагнитов. При печати импульсы тока нагревают головку, чтобы не допустить перегрева головки используется металлический теплоотводный радиатор. Головка перемещается специальным прецизионным механическим приводом. Головка принтера движется вдоль печатаемой строки, а стержни в нужный момент ударяют по бумаге через красящую ленту. Получаются точки. После удара головка смещается в сторону для печати следующей колонки точек. Возникающее при этом множество точек создает рисунок символа, таким образом, на бумаге формируется изображение.

Бывают принтеры 24 и 48 – игольчатые, они обеспечивают более качественную печать. Улучшить качество печати можно с помощью печати в несколько проходов (2 или 4).

В матричных принтерах используются красящая лента на синтетической основе, заключенная в пластмассовую кассету. Лента склеена в кольцо и продвигается только в одном направлении. Кассета с лентой устанавливается неподвижно, однако в некоторых моделях принтеров кассета надевается непосредственно на печатающую головку и перемещается вместе с ней. Такая кассета компактнее, однако, в ней помещается ленты меньше и ее ширина.

Достоинства матричных принтеров:

- Прочность и надежность принтера неприхотливость и долговечность;

- Возможность печатать на многослойных бланках (увеличение производительность, распечатка одновременно до 6 копий на листах проложенных через копировальную бумагу) ;

- Дешевые расходные материалы (лента и бумага).

Недостатки матричных принтеров:

- Шумит;

- Низкое качество печати;

- Обычно не поддерживают цветную печать.

Многофункциональные устройства сочетают в себе функции копира, сканера, принтера и факса. Чаще всего в набор входит или копир, принтер и сканер или копир, принтер и факс. В результате МФУ идеально подходят для маленьких офисов, где постоянно не хватает места на рабочем столе, в том числе и для "домашних офисов".

Существуют МФУ на базе лазерной технологии принтеров (для быстрой печати текста) и с технологией струйного принтера (для печати фотографий).

Цены на МФУ всегда в полтора-два раза ниже, чем, если покупать все три устройства с аналогичными характеристиками по отдельности.

Светодиодные принтеры

Очень похожи на лазерные принтеры. Поэтому их иногда путают. Отличие состоит только в том, что на фотобарабане изображение рисует не луч лазера (как в лазерных принтерах), а система светодиодов (как в копировальных аппаратах).

Большинство светодиодных принтеров монохромны, т.е. черно-белые, как и лазерные принтеры.

Так же как и лазерные принтеры, светодиодные принтеры бывают цветными. Так же как и лазерные принтеры, светодиодные принтеры бывают сетевыми для работы больших рабочих групп.

Основной бренд-производитель, распространенный в России: Оки. Менее распространены Lexmark и Xerox.

Расходным материалом является тонер.

Бывают режущие и печатающие плоттеры. Подавляющее большинство печатающих плоттеров - это струйные плоттеры, т.е. основанные на струйной технологии. Упрощенно струйные плоттеры можно представлять себе просто как большой напольный (т.е. стоящий на полу) струйный принтер для широкоформатной печати на больших листах и на рулонах. Их форматы бывают А2, А1, А0 и более.

Струйные плоттеры применяют в основном для печати больших плакатов. Поэтому они как правило имеют высокую скорость печати и большой ресурс по сравнению с простыми струйными принтерами. Но качество печати у струйных плоттеров ниже, чем у обычных струйных принтеров, т.к. большие плакаты предназначены для просмотра с дальнего расстояния.

Т.к. большие плакаты часто висят на улице под дождем и под прямыми солнечными лучами с ультрафиолетом, то часто в струйных плоттерах применяют специальные водостойкие и светостойкие краски.

8. ОСОБЕННОСТИ ПОСТРОЕНИЯ СЕТЕЙ

При создании вычислительных сетей разработчикам приходится решать множество самых разных задач, связанных с кодированием и синхронизацией электрических (оптических) сигналов, выбором конфигурации физических и логических связей, разработкой схем адресации устройств, созданием различных способов коммутации, мультиплексированием и демультимплексированием потоков данных, совместным использованием передающей среды. В данной лекции мы сформулируем все эти задачи, причем в той последовательности, в которой они возникали в процессе развития и совершенствования сетевых технологий.

Начнем с наиболее простого случая непосредственного соединения двух устройств физическим каналом, такое соединение называется связью "точка-точка" (point-to-point).

Частным случаем связи "точка-точка" является соединение компьютера с периферийным устройством. Для обмена данными компьютер и периферийное устройство (ПУ) оснащены внешними интер-

фейсами или портами (рис. 45). В данном случае к понятию "интерфейс" относятся:

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

Со стороны компьютера логикой передачи сигналов на внешний интерфейс управляют:

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

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

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

Таким образом, по каналу, связывающему внешние интерфейсы, передается следующая информация:

- данные, поступающие от контроллера на ПУ, например байты текста, который нужно распечатать на бумаге;
- команды управления, которые контроллер передает на устройство управления ПУ; в ответ на них оно выполняет специальные действия, например переводит головку диска на соответствующую дорожку или же выталкивает из принтера лист бумаги;
- данные, возвращаемые устройством управления ПУ в ответ на запрос от контроллера, например данные о готовности к выполнению операции.

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

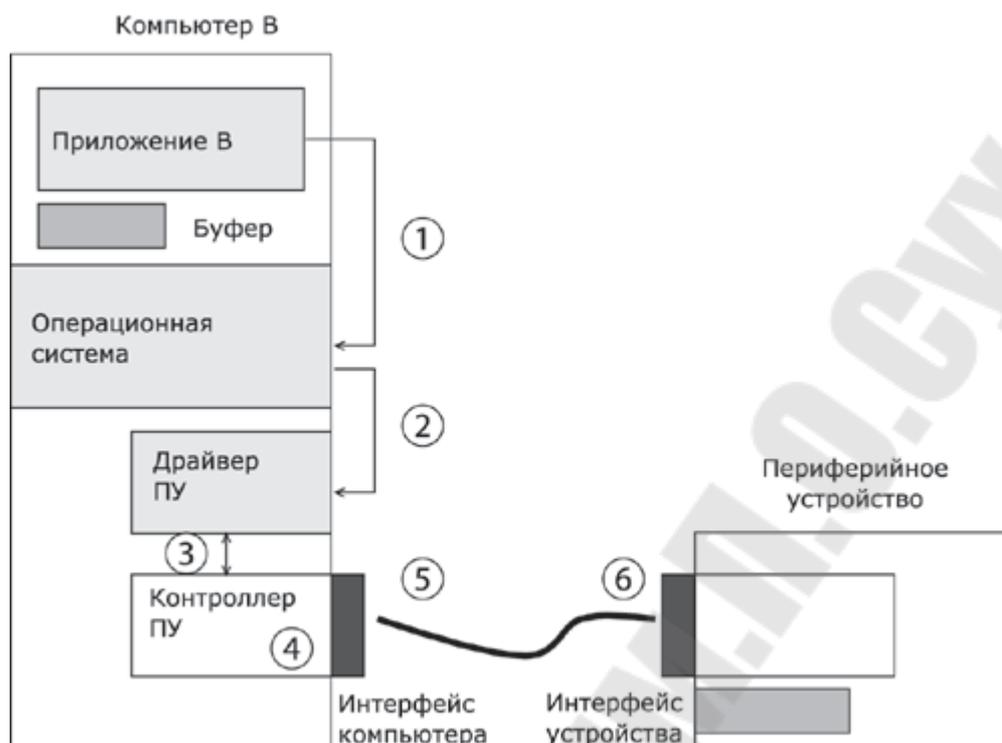


Рисунок 45 – Связь компьютера с периферийным устройством

1. Приложение обращается с запросом на выполнение операции печати к операционной системе. В запросе указываются: адрес данных в оперативной памяти, идентифицирующая информация принтера и операция, которую требуется выполнить.

2. Получив запрос, операционная система анализирует его, решает, может ли он быть выполнен, и если решение положительное, то запускает соответствующий драйвер, передавая ему в качестве параметров адрес выводимых данных. Дальнейшие действия, относящиеся к операции ввода-вывода, со стороны компьютера реализуются совместно драйвером и контроллером принтера.

3. Драйвер передает команды и данные контроллеру, который помещает их в свой внутренний буфер. Пусть, например, драйвер загружает значение некоторого байта в буфер контроллера ПУ.

4. Контроллер перемещает данные из внутреннего буфера во внешний порт.

5. Контроллер начинает последовательно передавать биты в линию связи, представляя каждый бит соответствующим электрическим сигналом. Чтобы сообщить устройству управления принтера о том, что начинается передача байта, перед передачей первого бита данных контроллер формирует стартовый сигнал специфической формы, а

после передачи последнего информационного бита — стоповый сигнал. Эти сигналы синхронизируют передачу байта. Кроме информационных бит, контроллер может передавать бит контроля четности для повышения достоверности обмена.

6. Устройство управления принтера, обнаружив на соответствующей линии стартовый бит, выполняет подготовительные действия и начинает принимать информационные биты, формируя из них байт в своем приемном буфере. Если передача сопровождается битом четности, то выполняется проверка корректности передачи: при правильно выполненной передаче в соответствующем регистре устройства управления принтера устанавливается признак завершения приема информации. Наконец, принятый байт обрабатывается принтером — выполняется соответствующая команда или печатается символ.

Обязанности между драйвером и контроллером могут распределяться по-разному, но чаще всего контроллер поддерживает набор простых команд, служащих для управления периферийным устройством, а на драйвер обычно возлагаются наиболее сложные функции реализации обмена. Например, контроллер принтера может поддерживать такие элементарные команды, как "Печать символа", "Перевод строки", "Возврат каретки" и т. п.

Драйвер же принтера с помощью этих команд реализует печать строк символов, разделение документа на страницы и другие более высокоуровневые операции (например, подсчет контрольной суммы последовательности передаваемых байтов, анализ состояния периферийного устройства, проверка правильности выполнения команды). Драйвер, задавая ту или иную последовательность команд, определяет тем самым логику работы периферийного устройства. Для одного и того же контроллера можно разработать различные драйверы, которые с помощью одного и того же набора доступных команд будут реализовывать разные алгоритмы управления одним и тем же ПУ.

Возможно распределение функций между драйвером и контроллером (ПУ).

Функции, выполняемые драйвером:

- ведение очередей запросов;
- буферизация данных;
- подсчет контрольной суммы последовательности байтов;
- анализ состояния ПУ;

- загрузка очередного байта данных (или команды) в регистр контроллера;
- считывание байта данных или байта состояния ПУ из регистра контроллера.

Функции, выполняемые контроллером:

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

А теперь предположим, что пользователь другого компьютера хотел бы распечатать текст. Сложность состоит в том, что к его компьютеру не подсоединен принтер, и требуется воспользоваться тем принтером, который связан с другим компьютером (рис.46).



Рисунок 46 - Взаимодействие двух компьютеров

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

Итак, механизм обмена байтами между двумя компьютерами определен. Теперь нужно договориться о правилах обмена сообщениями между приложениями А и В. Приложение В должно "уметь" расшиф-

ровать получаемую от приложения А информацию. Для этого программисты, разработывавшие приложения А и В, строго оговаривают форматы сообщений, которыми будут обмениваться приложения, и их семантику. Например, они могут договориться о том, что любое выполнение удаленной операции печати начинается с передачи сообщения, запрашивающего информацию о готовности приложения В; что в начале сообщения идет число, определяющее длину данных, предназначенных для печати; что признаком срочного завершения печати является определенная кодовая комбинация и т.п.

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

На стороне же компьютера В (на месте приложения В) должна работать другая специализированная программа — сервер, постоянно ожидающий прихода запросов на удаленный доступ к принтеру (или файлам, расположенным на диске) этого компьютера. Сервер, приняв запрос из сети, обращается к локальному ПУ, возможно, с участием локальной ОС.

Очень удобной и полезной функцией клиентской программы является способность отличить запрос к удаленному файлу от запроса к локальному файлу. Если клиентская программа умеет это делать, она сама распознает и перенаправляет (redirect) запрос к удаленной машине. Отсюда и название, часто используемое для клиентской части — редиректор. Иногда функции распознавания выделяются в особый программный модуль, в этом случае редиректором называют не всю клиентскую часть, а только этот модуль.

Программные клиент и сервер выполняют системные функции по обслуживанию запросов всех приложений компьютера А на удаленный доступ к файлам компьютера В. Чтобы приложения компьютера В могли пользоваться файлами компьютера А, описанную схему

нужно симметрично дополнить клиентом для компьютера В и сервером для компьютера А. Схема взаимодействия клиента и сервера с приложениями и локальной операционной системой приведена на рис 47.

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

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

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

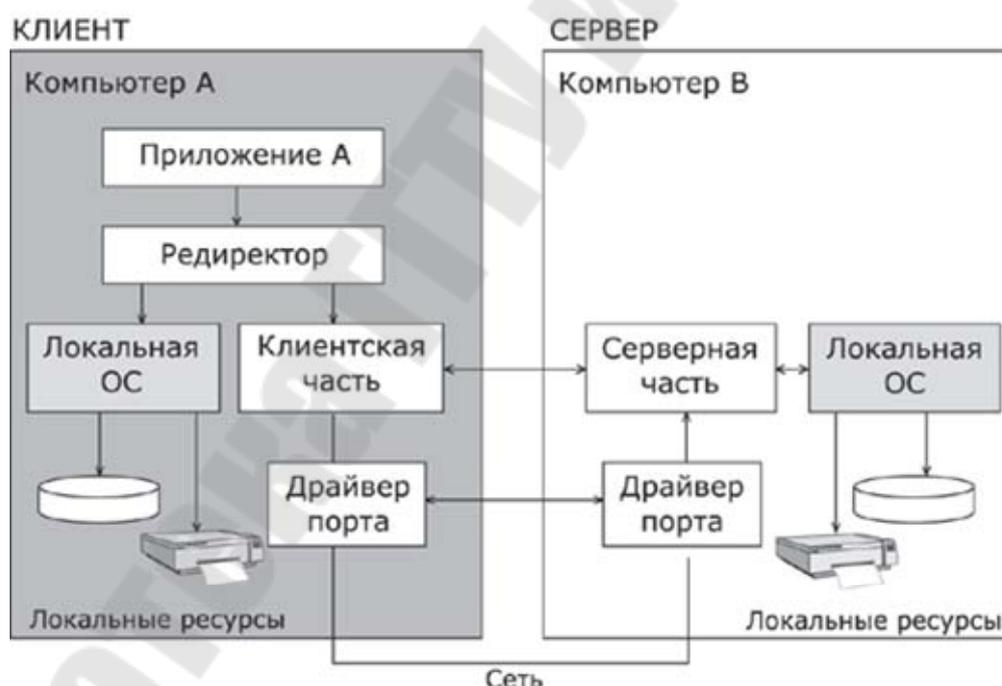


Рисунок 47 - Взаимодействие программных компонентов при связи двух компьютеров

- кодирование и модуляцию данных;
- взаимную синхронизацию передатчика одного компьютера с приемником другого;

- подсчет контрольной суммы и передача ее по линиям связи после каждого байта или после некоторого блока байтов.

В вычислительной технике для представления данных используется двоичный код. Представление данных в виде электрических или оптических сигналов называется кодированием. Существуют различные способы кодирования двоичных цифр 1 и 0, например потенциальный способ, при котором единице соответствует один уровень напряжения, а нулю — другой, или импульсный способ, когда для представления цифр используются импульсы различной или одной полярности.

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

В вычислительных сетях применяют как потенциальное, так и импульсное кодирование дискретных данных, а также специфический способ представления данных, который никогда не используется внутри компьютера, — модуляцию (Рис. 48). При модуляции дискретная информация представляется синусоидальным сигналом той частоты, которую хорошо передает имеющаяся линия связи.

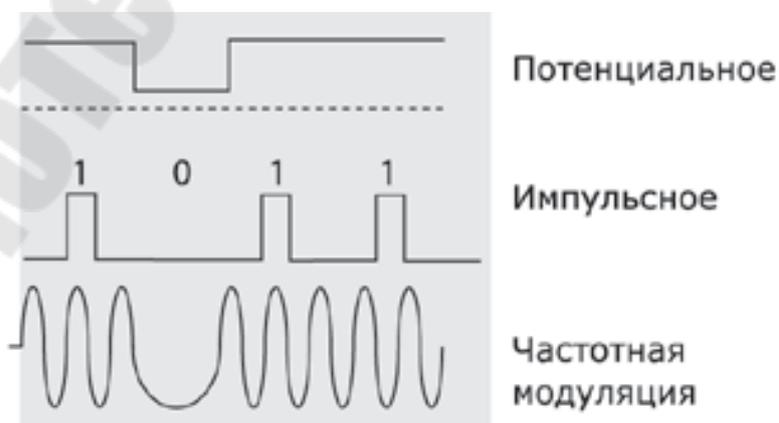


Рисунок 48 – Модуляция данных

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

На способ передачи сигналов влияет и количество проводов в линиях связи между компьютерами. Чтобы снизить стоимость линий связи в сетях, разработчики стараются сократить количество проводов и из-за этого используют не параллельную передачу всех бит одного байта или даже нескольких байт, как это делается внутри компьютера, а последовательную, побитную передачу, требующую всего одной пары проводов.

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

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

В каждый сетевой интерфейс, будь то порт маршрутизатора, концентратора или коммутатора, встроены средства, в той или иной мере решающие задачу надежного обмена двоичными сигналами, пред-

ставленными соответствующими электромагнитными сигналами. Некоторые сетевые устройства, такие как модемы и сетевые адаптеры, специализируются на физической передаче данных. Модемы выполняют в глобальных сетях модуляцию и демодуляцию дискретных сигналов, синхронизируют передачу электромагнитных сигналов по линиям связи, проверяют правильность передачи по контрольной сумме и могут выполнять некоторые другие операции. Сетевые адаптеры рассчитаны, как правило, на работу с определенной передающей средой — коаксиальным кабелем, витой парой, оптоволокном и т. п. Каждый тип передающей среды обладает определенными электрическими характеристиками, влияющими на способ использования данной среды, и определяет скорость передачи сигналов, способ их кодирования и некоторые другие параметры.

Концепция топологии локальной вычислительной сети в виде *звезды* пришла из области больших ЭВМ, в которой головная машина получает и обрабатывает все данные с периферийных устройств как активный узел обработки данных. Этот принцип применяется в системах передачи данных, например, в электронной почте RELCOM. Вся информация между двумя периферийными рабочими местами проходит через центральный узел локальной вычислительной сети.

Пропускная способность локальной вычислительной сети определяется вычислительной мощностью узла и гарантируется для каждой рабочей станции. Коллизий (столкновений) данных не возникает.

Кабельное соединение довольно простое, так как каждая рабочая станция связана с узлом. Затраты на прокладку кабелей высокие, особенно когда центральный узел географически расположен не в центре топологии локальной вычислительной сети.

При расширении локальных вычислительных сетей не могут быть использованы ранее выполненные кабельные связи: к новому рабочему месту необходимо прокладывать отдельный кабель из центра локальной вычислительной сети.

Топология в виде звезды является наиболее быстродействующей из всех топологий локальных вычислительных сетей, поскольку передача данных между рабочими станциями проходит через центральный узел (при его хорошей производительности) по отдельным линиям, используемым только этими рабочими станциями. Частота запросов передачи информации от одной станции к другой невысокая, по сравнению с достигаемой в других топологиях.

Производительность локальной вычислительной сети в первую очередь зависит от мощности центрального файлового сервера. Он может быть узким местом локальной вычислительной сети. В случае выхода из строя центрального узла нарушается работа всей локальной вычислительной сети.

Центральный узел управления - файловый сервер может реализовать оптимальный механизм защиты против несанкционированного доступа к информации. Вся локальная вычислительная сеть может управляться из ее центра.

Кольцевая топология локальной вычислительной сети подразумевает соединение рабочих станций друг с другой по кругу, т.е. рабочая станция 1 с рабочей станцией 2, рабочая станция 3 с рабочей станцией 4 и т.д. Последняя рабочая станция связана с первой. Коммуникационная связь замыкается в кольцо.

Прокладка кабелей от одной рабочей станции до другой может быть довольно сложной и дорогостоящей, особенно если географически рабочие станции расположены далеко от кольца (например, в линию).

Сообщения циркулируют регулярно по кругу. Рабочая станция посылает по определенному конечному адресу информацию, предварительно получив из кольца запрос. Пересылка сообщений является очень эффективной, так как большинство сообщений можно отправлять “в дорогу” по кабельной системе одно за другим. Очень просто можно сделать кольцевой запрос на все станции. Продолжительность передачи информации увеличивается пропорционально количеству рабочих станций, входящих в локальную вычислительную сеть.

Основная проблема при кольцевой топологии заключается в том, что каждая рабочая станция должна активно участвовать в пересылке информации, и в случае выхода из строя хотя бы одной из них вся локальная вычислительная сеть парализуется. Неисправности в кабельных соединениях локализуются легко.

Подключение новой рабочей станции требует кратко срочного выключения локальной вычислительной сети, так как во время установки кольцо должно быть разомкнуто. Ограничения на протяженность локальной вычислительной сети не существует, так как оно, в конечном счете, определяется исключительно расстоянием между двумя рабочими станциями.

Структура логической кольцевой цепи локальной вычислительной сети. Специальной формой кольцевой топологии является

логическая кольцевая локальная вычислительная сеть. Физически она монтируется как соединение звездных топологий. Отдельные звезды включаются с помощью специальных коммутаторов (англ. Hub -концентратор), которые по-русски также иногда называют “хаб”. В зависимости от числа рабочих станций и длины кабеля между рабочими станциями применяют активные или пассивные концентраторы. Активные концентраторы дополнительно содержат усилитель для подключения от 4 до 16 рабочих станций. Пассивный концентратор является исключительно разветвительным устройством (максимум на три рабочие станции). Управление отдельной рабочей станцией в логической кольцевой локальной вычислительной сети происходит так же, как и в обычной кольцевой локальной вычислительной сети. Каждой рабочей станции присваивается соответствующий ей адрес, по которому передается управление (от старшего к младшему и от самого младшего к самому старшему). Разрыв соединения происходит только для нижерасположенного (ближайшего) узла локальной вычислительной сети, так что лишь в редких случаях может нарушаться работа всей локальной вычислительной сети.

При *шинной* топологии среда передачи информации представляется в форме коммуникационного пути, доступного для всех рабочих станций, к которому они все должны быть подключены. Все рабочие станции могут непосредственно вступать в контакт с любой рабочей станцией, имеющейся в локальной вычислительной сети.

Рабочие станции в любое время, без прерывания работы всей локальной вычислительной сети, могут быть подключены к ней или отключены. Функционирование локальной вычислительной сети не зависит от состояния отдельной рабочей станции.

9. ВЫЧИСЛИТЕЛЬНЫЕ СИСТЕМЫ

Вычислительные системы (ВС) и сети представляют собой некий комплекс аппаратных (аппаратную платформу) и программных (системное ПО) средств, позволяющих обеспечить выполнение информационных, вычислительных или управленческих процедур в самых разных областях современной жизни.

Цели создания ВС:

- достижение сверхвысокой производительности;
- увеличение эффективности использования аппаратных средств системы;

- повышение надежности и живучести функционирования средств вычислительной техники.

Вычислительные системы относятся к категории сложных систем. При рассмотрении вопросов, связанных с их построением и функционированием, выделяют элементы системы и подсистемы как составные части структуры ВС.

Элемент системы — объект системы, не подлежащий дальнейшему расчленению на части при данном ее рассмотрении. Предметом изучения является не внутренняя структура элемента, а такие его свойства, которые определяют взаимодействие этого элемента с другими элементами системы или влияют на свойства системы в целом.

Подсистема — часть системы, представляющая собой совокупность некоторых ее элементов, выделенных по определенному функциональному признаку, и отличающаяся подчиненностью по своей цели функционирования единой цели функционирования всей системы.

Структура ВС — это организация системы из отдельных элементов с их взаимосвязями, которые определяются распределением функций, выполняемых ВС. В зависимости от уровня детализации при данном рассмотрении структуры ВС в качестве ее элементов могут пониматься отдельные модули системы вплоть до ЭВМ в целом.

Комплекс программных средств регулярного применения, предназначенный для придания ВС определенных свойств, повышения эффективности ее использования, облегчения эксплуатации и снижения трудоемкости подготовительной работы при решении задач, образует **системное программное обеспечение (СПО) ВС**.

В состав СПО входят:

- программы, управляющие работой ВС (операционная система ВС);

- системы автоматизации программирования;

- пакеты прикладных программ общего пользования и расширяющие возможности операционной системы;

- комплекс программ технического обслуживания ВС.

- Основные принципы построения ВС заключаются в следующем:

- обеспечение работы в различных режимах;

- модульность структуры технических и программных средств;

- унификация и типизация технических и программных средств;

- согласованность пропускных способностей отдельных функциональных частей системы;

- иерархия в организации управления процессом функционирования;

- способность системы к самоорганизации, самонастройке, адаптивности к изменению условий функционирования.

Система называется самоорганизующейся, если в ней на основании оценки воздействия внешней среды путем последовательного изменения своих свойств заложены возможности перехода к некоторому устойчивому состоянию, когда воздействия внешней среды сказываются в допустимых пределах.

В ВС возможность к самонастройке заложена преимущественно в структуре и функциях управляющей программы ОС. Примеры самонастройки: автоматическое перераспределение ресурсов системы с увеличением числа активных абонентов; повышение уровня мультипрограммной работы, т. е. увеличение числа одновременно обслуживаемых запросов абонентов, если ставится задача обеспечения минимального времени ожидания обслуживания, и др.

Это достигается за счет:

- развитой сети периферийного оборудования;

- использования современных средств автоматизации программирования;

- обеспечения независимости работы абонентов при подготовке своих программ;

- обеспечения доступа абонентов к базам данных и знаний, стандартным программам, пакетам прикладных программ, имеющимся в структуре информационного и программного обеспечения ВС;

- обеспечения гарантированной защиты индивидуальных программ и информационных массивов абонентов от несанкционированного доступа.

Широкое распространение информационно-вычислительных систем в самых разных областях современной жизни: промышленности, финансах, образовании и культуре, - делает актуальным вопрос о создании некоторых стандартов выполнения ИВС. Наиболее комплексно эти вопросы ставятся в научном направлении, которое определяет концепцию "открытых систем".

Открытая система для ИВС - это среда для прикладных программ, базирующаяся на стандартных интерфейсах и обеспечивающей мобильность прикладных систем, персонала и взаимодействие (интероперабельность) систем. Таким образом важнейшими свойствами открытой ИВС будут:

- мобильность прикладных программ, т.е. возможность переноса программ с одной аппаратной платформы на другую с минимальными доработками или даже без них;

- мобильность персонала, т.е. возможность подготовки персонала для работы на ИВС с минимальными временными и трудовыми затратами;

- четкие условия взаимодействия частей ИВС и сетей с использованием открытых (общедоступных) спецификаций.

Ключевым моментом в обеспечении свойств открытых ИВС - использование открытых спецификаций, т.е. спецификаций (которые поддерживаются открытым, гласным согласительным процессом, направленным на постоянную адаптацию новым технологиям).

Кластерные структуры или просто кластеры являются самым дешевым способом наращивания производительности уже установленных компьютеров.

Кластер представляет собой набор из нескольких ЭВМ, соединенных через некоторую инфраструктуру. В качестве такой инфраструктуры может выступать обычная компьютерная сеть, однако из соображений повышения производительности желательно иметь высокоскоростные соединения (FDDI, ATM, HiPPi и т.п.).

Кластеры могут быть образованы как из различных компьютеров (**гетерогенные кластеры**), так и из одинаковых (**гомогенные кластеры**).

Кластеры являются примером слабо связанных систем, поскольку модули памяти распределены по всей системе и закреплены за отдельными процессорами. Добавление компьютеров в кластер позволяет увеличивать пропускную способность памяти и наращивать производительность.

В кластерных системах для организации взаимодействия между процессами, выполняющимися на разных компьютерах при решении одной задачи, применяются различные модели обмена сообщениями (PVM, MPI и другие). Однако задача распараллеливания вычислений в таких системах с распределенной между отдельными компьютерами памятью в рамках этих моделей является более сложной, чем в моделях с общим полем памяти (как, например, в SMP-серверах).

10. СИСТЕМЫ ЗАЩИТЫ ИНФОРМАЦИИ

Создание всеобщего информационного пространства, массовое применение персональных компьютеров и внедрение компьютерных систем породили необходимость решения комплексной проблемы за-

щиты информации. В интегрированных и локальных системах обработки данных (СОД) с использованием разнообразных технических средств, включая компьютерные, под защитой информации принято понимать использование различных средств и методов, принятие мер и осуществление мероприятий с целью системного обеспечения надежности передаваемой, хранимой и обрабатываемой информации. Защитить информацию - это значит:

- обеспечить физическую целостность информации, т.е. не допустить искажений или уничтожения элементов информации;
- не допустить подмены (модификации) элементов информации при сохранении ее целостности;
- не допустить несанкционированного получения информации лицами или процессами, не имеющими на это соответствующих полномочий;
- быть уверенным в том, что передаваемые (продаваемые) владельцем информации ресурсы будут использоваться только в соответствии с обговоренными сторонами условиями.

Практика функционирования СОД показывает, что существует достаточно много способов несанкционированного доступа к информации:

- просмотр;
- копирование и подмена данных;
- ввод ложных программ и сообщений в результате подключения к каналам связи;
- чтение остатков информации на ее носителях;
- прием сигналов электромагнитного излучения и волнового характера;
- использование специальных программных и аппаратных "заглушек" и т.п.

Система защиты информации - это совокупность организационных, административных и технологических мер, программно-технических средств, правовых и морально-этических норм, направленных на противодействие угрозам нарушителей с целью сведения до минимума возможного ущерба пользователям владельцам системы.

Учитывая важность, масштабность и сложность решения проблемы сохранности и безопасности информации, рекомендуется разрабатывать архитектуру безопасности в несколько этапов:

- анализ возможных угроз;
- разработка системы защиты;

- реализация системы защиты;
- сопровождение системы защиты.

Программные средства и методы защиты активнее и шире других применяются для защиты информации в персональных компьютерах и компьютерных сетях, реализуя такие функции защиты, как разграничение и контроль доступа к ресурсам; регистрация и анализ протекающих процессов, событий, пользователей; предотвращение возможных разрушительных воздействий на ресурсы; криптографическая защита информации; идентификация и аутентификация пользователей и процессов и др.

В настоящее время наибольший удельный вес в этой группе мер в системах обработки экономической информации составляют специальные пакеты программ или отдельные программы, включаемые в состав программного обеспечения с целью реализации задач по защите информации. Технологические средства защиты информации - это комплекс мероприятий, органично встраиваемых в технологические процессы преобразования данных. Среди них:

- создание архивных копий носителей;
- ручное или автоматическое сохранение обрабатываемых файлов во внешней памяти компьютера;
- регистрация пользователей компьютерных средств в журналах;
- автоматическая регистрация доступа пользователей к тем или иным ресурсам;
- разработка специальных инструкций по выполнению всех технологических процедур и др.

В компьютерных системах сосредоточивается информация, право на пользование которой принадлежит определенным лицам или группам лиц, действующим в порядке личной инициативы или в соответствии с должностными обязанностями. Чтобы обеспечить безопасность информационных ресурсов, устранить возможность несанкционированного доступа, усилить контроль санкционированного доступа к конфиденциальной либо к подлежащей засекречиванию информации, внедряются различные системы опознавания, установления подлинности объекта (субъекта) и разграничения доступа. В основу построения таких систем закладывается принцип допуска и выполнения только таких обращений к информации, в которых присутствуют соответствующие признаки разрешенных полномочий. Ключевыми понятиями в этой системе являются "идентификация" и "аутентификация". **Идентификация** - это присвоение какому-либо объекту или

субъекту уникального имени или образа. **Аутентификация** - это установление подлинности, т.е. проверка, является ли объект (субъект) действительно тем, за кого он себя выдает.

Конечная цель процедур идентификации и аутентификации объекта (субъекта) - допуск его к информации ограниченного пользования в случае положительной проверки либо отказ в допуске в случае отрицательного исхода проверки.

Объектами идентификации и аутентификации могут быть: люди (пользователи, операторы и др.); технические средства (мониторы, рабочие станции, абонентские пункты); документы (ручные, распечатки и др.); магнитные носители информации; информация на экране монитора, табло и др.

Один из наиболее распространенных методов аутентификации - присвоение лицу или другому имени пароля и хранение его значения в вычислительной системе. **Пароль** - это совокупность символов, определяющая объект (субъект).

Для идентификации пользователей могут применяться сложные в плане технической реализации системы, обеспечивающие установление подлинности пользователя на основе анализа его индивидуальных параметров: отпечатков пальцев, рисунка линий руки, радужной оболочки глаз, тембра голоса и др. Но пока эти приемы носят скорее рекламный, чем практический характер.

Массовое использование ПК в сетевом режиме, включая выход в глобальную сеть Интернет, породило проблему заражения их компьютерными вирусами. **Компьютерным вирусом** принято называть специально написанную, обычно небольшую по размерам программу, способную самопроизвольно присоединяться к другим программам (т.е. заражать их), создавать свои копии (не обязательно полностью совпадающие с оригиналом) и внедрять их в файлы, системные области компьютера и в другие объединенные с ним компьютеры с целью нарушения нормальной работы программ, порчи файлов и каталогов, создания различных помех при работе на компьютере. Способ функционирования большинства вирусов - это такое изменение системных файлов компьютера, чтобы вирус начинал свою деятельность при каждой загрузке. Некоторые вирусы инфицируют файлы загрузки системы, другие специализируются на EXE-, COM- и других программных файлах. Всякий раз, когда пользователь копирует файлы на гибкий диск или посылает инфицированные файлы по модему, переданная копия вируса пытается установить себя на новый диск.

В зависимости от среды обитания вирусы классифицируются на загрузочные, файловые, системные, сетевые, файлово-загрузочные. Массовое распространение компьютерных вирусов вызвало разработку антивирусных программ, позволяющих обнаруживать и уничтожать вирусы, "лечить" зараженные ресурсы.

К настоящему времени зарубежными и отечественными фирмами и специалистами разработано большое количество антивирусных программ. Многие из них, получившие широкое признание, постоянно пополняются новыми средствами для борьбы с вирусами и сопровождаются разработчиками.

Список литературы

1. Головкин Б.А. Параллельные вычислительные системы. -М.: Наука.- 1980.
2. Ковалев А.В., Литвинов Д.А., Самовендюк Н.В. Лабораторный практикум по курсу «Организация и функционирование ЭВМ» для студентов специальности 1-40 01 02 "Информационные системы и технологии (по направлениям)" дневной формы обучения, Гомель: ГГТУ им. П.О. Сухого.- 2009г., - 45с.
3. Каган Б.М. Электронные вычислительные машины и системы.- М.:Энергия.-1991.
4. Краковяк С. Основы организации и функционирования ОС ЭВМ.- М.:Мир.-1988.
5. Ковалев А.В., Литвинов Д.А. Курс лекций по дисциплине «Организация и функционирование ЭВМ» для студентов специальности 1-40 01 02 "Информационные системы и технологии (по направлениям), Гомель: ГГТУ им. П.О. Сухого.- 2010г., - 56с.
6. Беляев С.Н., Козырева Г.М. и др. Вычислительные машины, комплексы и сети: Учебник / под ред. А.П. Пятибратова. – М.: Финансы и статистика, 1991.
7. Пятибратов А.П., Гудыко Л.П., Кириченко А.А. Вычислительные системы, сети и телекоммуникации. Учебник / Под ред. А.П. Пятибратова. - М.: Финансы и статистика, 1998.
8. Водяхо А.И., Горнец Н.Н., Пузанков Д.В. Высокопроизводительные системы обработки данных: Учеб. пособие для вузов. - М.: Высш. шк., 1997.
9. Гуров В.В. Синтез комбинационных схем в примерах М.: МИФИ, 2001г.
10. Цилькер Б.Я., Орлов С.А. Организация ЭВМ и систем. – СПб.: Питер, 2006. – 668 с.
11. Могилёв А.В. и др. Информатика – М.: Академия, 2004. – 848 с.
12. Молчанов А. Ю. Системное программное обеспечение – СПб.: Питер, 2003. – 396 с.
13. Новиков Ф.А. Дискретная математика для программистов – СПб.: Питер, 2000. – 304 с.
14. Компьютерные системы и сети. Учеб. пособие /В.П. Косарев и др. - М.: Финансы и статистика, 1999.
15. ГукМ. Аппаратные интерфейсы ПК. Энциклопедия. — СПб.: Питер, 2002. — 528 с.: ил.

16. Несвижский В. Программирование аппаратных средств в Windows. – 2 изд. СПб.: 2008 г. – 528 с.: ил.
17. Несвижский В. Программирование аппаратных средств в Windows. – СПб.: 2004 г. – 880 с.: ил.
18. Таненбаум Э.С. Архитектура компьютера. 5-е изд. Питер. 2006г.- 848с.
19. Крук Б.И. и др. Телекоммуникационные системы и сети.- М.:Горячая линия-Телеком, 2004. – 647 с.: ил.
20. Технология Bluetooth, Архипкин В.Я., Архипкин А.В., Москва, 2002.
21. Overview of Bluetooth Technology, Hongfeng Wang, 2001
22. Красковский Д., Богданов В. Персональный компьютер. Издательство: КомпьютерПресс, 2001 г.
23. Шпунт Я.Б. Выбор принтера для дома и офиса. Серия: Просто о ложном. Издательство: НТ Пресс, 2005 г.
24. Электрофотографический процесс, оборудование и материалы, принтеры, микрография: Ч.1. Дравин А.Б. -Б.м.. - 1991
25. Аванесов В.А. Работа с принтером hp designjet серии 500: учеб. Пособие В.А. Аванесов, Е.М. Москалева. - Ухта: УГТУ. - 2005.
26. Гинзбург А. Периферийные устройства. Принтеры, сканеры, цифровые камеры Милчев М., Солоницын Ю.. - СПб. и др.: Питер. – 2001
27. Головков, А. В. Диагностирование, техническое обслуживание и ремонт матричных принтеров для IBM PC-XT/AT [Текст] : монография / А.В. Головков. - М. : [б. и.], 1993.
28. «Лазерные принтеры. Взгляд на принтер изнутри. Технология лазерной печати» О. Колесниченко, М. Шарыгин, И. Шишигин, «ВНУ – Санкт –Петербург», Санкт – Петербург 1997 г.
29. Манпиль, Л. И. Устройства регистрации графической информации ЭВМ и систем [Текст] : учеб. пособие для вузов по спец. "Вычислит. машины, комплексы, системы и сети" и "Конструирование и технология электрон. вычислит. средств" / Л.И. Манпиль. - М. : Энергоатомиздат, 1995.

Ковалев Алексей Викторович

**ТЕХНИЧЕСКИЕ СРЕДСТВА
ИНФОРМАЦИОННЫХ СИСТЕМ**

**Курс лекций
по одноименной дисциплине
для слушателей специальности
1-40 01 73 «Программное обеспечение
информационных систем»
заочной формы обучения**

Подписано к размещению в электронную библиотеку
ГГТУ им. П. О. Сухого в качестве электронного
учебно-методического комплекса 18.03.15.

Рег. № 12Е.

<http://www.gstu.by>