**|13.01.20| Архитектура компьютерных систем. |1 Пара|**

**Состав ПК:**   
ПК строится из конкретных наборов устройств.  
В блоке компьютера расположены:  
1. Материнская плата в разъёмы который вставляются контроллеры устр.  
 2.Жесткие и магнитные диски, дисководы и другие накопители…  
 3.Блок питания.  
**Архитектура по принцип Apple:**  
 1. Изготовление узлов и сборку должна осуществлять одна фирма.  
 2. Настройка ПК и замена его узлов, должны заниматься только профессионалы.

**IBM** строятся на базе принципа открытой архитектуры.

Компьютер составлен из отдельных узлов, и пользователю предоставляется широкие возможности изменять состав компьютера, изменяя одни узлы другими.

Производством узлов и сборкой самих ПК занимаются фирмы разных стран.

Под архитектурой принято понимать совокупность всех программных доступных аппаратных средств процессов.

**Понятие архитектуры является КОМПЛЕКСНЫМ и включает в себя:**

1. Структурную схему компьютера.
2. Средства и способы доступы к элементам структурной схемы.
3. Организация и разрядность интерфейса.
4. Набор регистров.
5. Организация и способы адресации памяти.
6. Способы представлении и форматы данных.
7. Набор машинных команд.
8. Форматы машинных команд.
9. Обработка прерываний.

**Общие архитектурные свойства и принципы. (Архитектура фон Неймана).**

1. Принцип хранение программ.
2. Принцип микро программирования.
3. Линейная пространство памяти.
4. Последовательное выполнение программ.
5. Безразличие к целевому назначению данных.

**Индивидуальные архитектурные принципы. (Intel I486 и Pentium).**

1. Супер скалярное архитектура.

I486 – появился конвейер, при котором исполнение команды разбивается на несколько этапов.

I486 5 этапов:

1. Выборка команды из кэш памяти и оперативной памяти.
2. Декодирование команды.
3. Генерация адреса. (Определение операндов в памяти)
4. Выполнение операций с помощью АЛУ. (Арифметико-логическое устройство)
5. Запись результатов.

Один конвейер скалярное (I486) более одного конвейера это супер скалярное (семейство Pentium).

1. Раздельное кэширование кода и данных.
2. Предсказание правильного адреса перехода.
3. Улучшенный блок вычисления с плавающей точкой.
4. Расширенная 64-битовая шина данных.

**|13.01.20| Архитектура компьютерных систем. |2 Пара|**

Характеристика ЭВМ определяющие её структуры:

1. Технические и эксплуатационные характеристики ЭВМ. (Быстродействие, производительность, указатель надежности, достоверности, точности, емкость оперативной памяти, габаритные размерные, стойкость технических и программных средств, особенности эксплуатации)
2. Характеристики и состав функциональны модулей, базовая конфигурация ЭВМ; Возможности расширения состава технических и программных средств, возможность изменения структуры
3. Состав программного обеспечения ЭВМ и сервисных услуг. (Операционная система или среда, пакеты прикладных программ, и средства автоматизации программирования)

Быстродействие это число команд, выполняемых ЭВМ за одну секунду.

Сравнение по быстродействию различных типов ЭВМ, не обеспечивает достоверных оценок. Очень часто вместо характеристики быстродействия используют связанную с ней характеристику производительность.

Производительность это объем работ, осуществляемых ЭВМ в единицу времени.

Применяются также относительные характеристики производительности. Фирма Intel для оценки процессоров предложила тест, получивший название индекс iCOMP (Intel Comparative Microprocessor Performance). При его определении учитываются четыре главных аспекта производительности: работа с целыми числами, с плавающей запятой, графикой и видео. Данные имеют 16- и 32-разрядной представление. Каждый из восьми параметров при вычислении участвует со своим весовым коэффициентом, определяемым по усредненному соотношению между этими операциями в реальных задачах. По индексу iCOMP ПМ Pentium 100 имеет значение 810, а Pentium 133-1000.

Емкость запоминающих устройств. Емкость памяти измеряется количеством структурных единиц информации, которое может одновременно находится в памяти. Этот показатель позволяет определить, какой набор программ и данных может быть одновременно размещен в памяти.

Наименьшей структурной единицей информации является бит- одна двоичная цифра. Как правило, емкость памяти оценивается в более крупных единицах измерения - байтах (байт равен восьми битам). Следующими единицами измерения служат 1 Кбайт = 210 = 1024 байта, 1 Мбайт = 210 Кбайта = 220 байта, 1 Гбайт =210 Мбайта = 220 Кбайта = 230 байта.

Емкость оперативной памяти (ОЗУ) и емкость внешней памяти (ВЗУ) характеризуются отдельно. Этот показатель очень важен для определения, какие программные пакеты и их приложения могут одновременно обрабатываться в машине.

Надежность это способность ЭВМ при определенных условиях выполнять требуемые функции в течение заданного периода времени (стандарт ISO (Международная организация стандартов) 2382/14-78).

Высокая надежность ЭВМ закладывается в процессе ее производства. Применение сверхбольшие интегральные схемы (СБИС) резко сокращают число используемых интегральных схем, а значит, и число их соединений друг с другом. Модульный принцип построения позволяет легко проверять и контролировать работу всех устройств, проводить диагностику и устранение неисправностей.

Точность это возможность различать почти равные значения (стандарт ISO - 2382/2-76).

Точность получения результатов обработки в основном определяется разрядностью ЭВМ, а также используемыми структурными единицами представления информации (байтом, словом, двойным словом).

Достоверность это свойство информации быть правильно воспринятой.

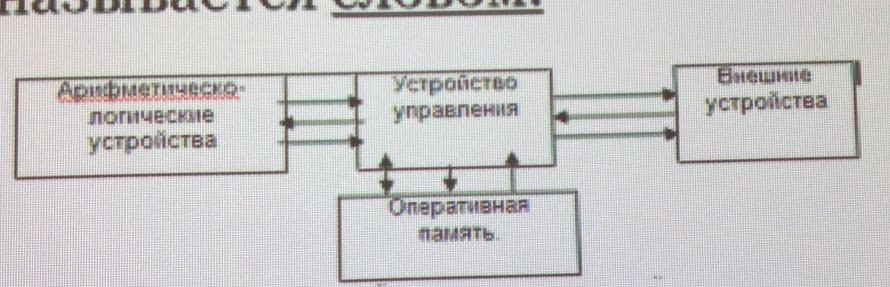
Достоверность характеризуется вероятностью получения безошибочных результатов. Заданный уровень достоверности обеспечивается аппаратурно-программными средствами контроля самой ЭВМ. Возможны методы контроля достоверности путем решения эталонных задач и повторных расчетов. В особо ответственных случаях проводятся контрольные решения на других ЭВМ и сравнение результатов.

**Классификация средств.**

1. Разделяется на аналоговые и цифровые.
2. Разделяется на 4 класса ПК:
   1. МАЙНФРАЙМ.
   2. Машины РС-6000.
   3. Средние ЭВМ. (АС/400)
   4. Компьютер на платформе Intel.

**Общие принципы построения современных ЭВМ.**

1. Основной принцип построения всех современных ЭВМ – это программное управление. (Представление алгоритма, решению любой задачи в виде программы вычисления)
2. Алгоритм – это конкретный порядок действий приводящий к конкретному результату за конечное время.
3. Программа – это упорядоченная последовательность команд подлежащая обработке.
4. Принципы программного управления может быть осуществлён различными способами: стандартом для построения практически всех ЭВМ стал способом, описанный Фон-Нейманом в 1945г. построений ещё первых образцов ЭВМ. Суть его заключается в следующем: все вычисления, предписанные алгоритмом решения задач должны бать представлены в виде программы, состоящие из последовательности управляющих слов команд. Каждая команда содержит указание на конкретную выполняемую операцию места нахождения (адреса) операндов и ряд служебных признаков. Операнды — это переменные значения, которых участвуют в операциях преобразования данных, списков (массив) всех переменных (входных данных промежуточных значений и результатов вычислений) является ещё одним неотъемлемым вычислением другой программы. Для доступа к программам, командам и операциям используют их адреса. В качестве адресов выступают номера ячеек памяти ЭВМ предназначенных для хранения объектов. Информация (командная и данные: числовые, текстовая, графическая и т.п.) копируется двоичными цифрами о и 1, поэтому различные типы информации, размещенные в памяти ЭВМ практически не различимы, идентификация их возможна только при выполнении программ согласно её логике по контексту. Последовательность битов в формате имеющая определённый смысл называется — полем. Например: каждой команде программы различают поле кода, операция поля адресов, операндов приблизительно к числовой информации выделяют знаковые разряды поля значащих разрядов чисел старшие и младшие разряды. Последовательность, состоящая из определённого принятого для данной ЭВМ числа байтов, называется словом.



**|13.01.20| Архитектура компьютерных систем. |3 Пара|**

**Системы исчисления.**

В ЭВМ применяется 2ичная система исчисления. Для преобразования числовой, текстовой, графической, звуковой необходимо применить кодирования.

Кодирование – это преобразования данных одного типа, через данные другого типа…

В ЭВМ применяется система 2ичная кодирования. Целые числа кодируются 2ичным кодом, путём деления числа на 2.

**Не числовая информация кодируется следующим образом:**

Все возможные значения кодирований информации, нумеруются и эти номера кодируются с помощью двоичного кода…

Для представления текстовой информации, используется таблица нумерации символов (таблица кодировки).

Для кодирования графических данных применяется, растровая. Координаты и их свойства описываются с помощью целых чисел, ЧБ графические объекты могут быть описаны комбинации точек с 256 градацию серого цвета. То есть для кодирования яркоскти любой точки, достаточно 8 разрядного 2ичного числа!

Режим представления цветной графики в системе RGB с использованием 24 разрядов, называется полноцветным. Для полноцветного режима в системе CMYK необходимо 32 разряда.

**Системы исчисления** – подразделяется на позиционные и непозиционные.   
**Позиционные** – подразделяются на однородные и смешанные.

**Непозиционные системы** – каждая цифра имеет величину не зависящую от её позиции.  
**Позиционные системы** – значение каждой цифры зависит от её позиции в числе.

**Однородные системы** – для всех разрядов (позиций) числа, набор допустимых символов одинаков.  
**Смешанные системы** – в каждом разряде числа, набор допустимых символов может отличаться от наборов других символов.

8ричная система исчисления применяется в цифровой технике, имеет основание 8, и использует для записи числа от 0 до 7. Для перевода в 10чную систему, необходимо умножить н 8 в степени N - номер разряда.

**16ричная система исчисления** – имеет основание 16 и использует для записи числа.

*То есть для перевода в 8речную систему, сначала мы преобразуем 16ричное число в 2ичную, а потом разбив на группы по 3 разряда, что бы преобразовать число в 2ичное необходимо каждую цифру, представить в виде 4 разрядного 2ичного числа, после этого разделяем число на группу по 3 цифры с права налево, после этого переводим в 2ичную группу в 8ричную систему, умножив каждый разряд на 2 в степени N – номер разряда*.

Преобразование в десятичную систему счисления

Имеется число a1a2a3 в системе счисления с основанием b. Для перевода в 10-ю систему необходимо каждый разряд числа умножить на bn, где n — номер разряда. Таким образом, (a1a2a3)b = (a1\*b2 + a2\*b1 + a3\*b0)10.

Пример: 1012 = 1\*22 + 0\*21 + 1\*20 = 4+0+1 = 510

Преобразование из десятичной системы счисления в другие

Целая часть:

1. Последовательно делим целую часть десятичного числа на основание системы, в которую переводим, пока десятичное число не станет равно нулю.

2. Полученные при делении остатки являются цифрами искомого числа. Число в новой системе записывают, начиная с последнего остатка.

Дробная часть:

1. Дробную часть десятичного числа умножаем на основание системы, в которую требуется перевести. Отделяем целую часть. Продолжаем умножать дробную часть на основание новой системы, пока она не станет равной 0.

2. Число в новой системе составляют целые части результатов умножения в порядке, соответствующем их получению.

Пример: переведем 1510 в восьмеричную:

15\8 = 1, остаток 7

1\8 = 0, остаток 1

Записав все остатки снизу вверх, получаем итоговое число 17. Следовательно, 1510 = 178.

Преобразование из двоичной в восьмеричную и шестнадцатеричную системы

Для перевода в восьмеричную — разбиваем двоичное число на группы по 3 цифры справа налево, а недостающие крайние разряды заполняем ведущими нулями. Далее преобразуем каждую группу, умножая последовательно разряды на 2n, где n — номер разряда.

В качестве примера возьмем число 10012: 10012 = 001 001 = (0\*22 + 0\*21 + 1\*20) (0\*22 + 0\*21 + 1\*20) = (0+0+1) (0+0+1) = 118

Для перевода в шестнадцатеричную — разбиваем двоичное число на группы по 4 цифры справа налево, затем — аналогично преобразованию из 2-й в 8-ю.

Преобразование из восьмеричной и шестнадцатеричной систем в двоичную

Перевод из восьмеричной в двоичную — преобразуем каждый разряд восьмеричного числа в двоичное 3-х разрядное число делением на 2 (более подробно о делении см. выше пункт “Преобразование из десятичной системы счисления в другие”), недостающие крайние разряды заполним ведущими нулями.

Для примера рассмотрим число 458: 45 = (100) (101) = 1001012

Перевод из 16-ой в 2-ю — преобразуем каждый разряд шестнадцатеричного числа в двоичное 4-х разрядное число делением на 2, недостающие крайние разряды заполняем ведущими нулями.

Преобразование дробной части любой системы счисления в десятичную

Преобразование осуществляется также, как и для целых частей, за исключением того, что цифры числа умножаются на основание в степени “-n”, где n начинается от 1.

Пример: 101,0112 = (1\*22 + 0\*21 + 1\*20), (0\*2-1 + 1\*2-2 + 1\*2-3) = (5), (0 + 0,25 + 0,125) = 5,37510

Преобразование дробной части двоичной системы в 8- и 16-ую

Перевод дробной части осуществляется также, как и для целых частей числа, за тем лишь исключением, что разбивка на группы по 3 и 4 цифры идёт вправо от десятичной запятой, недостающие разряды дополняются нулями справа.

Пример: 1001,012 = 001 001, 010 = (0\*22 + 0\*21 + 1\*20) (0\*22 + 0\*21 + 1\*20), (0\*22 + 1\*21 + 0\*20) = (0+0+1) (0+0+1), (0+2+0) = 11,28

Преобразование дробной части десятичной системы в любую другую

Для перевода дробной части числа в другие системы счисления нужно обратить целую часть в ноль и начать умножение получившегося числа на основание системы, в которую нужно перевести. Если в результате умножения будут снова появляться целые части, их нужно повторно обращать в ноль, предварительно запомнив (записав) значение получившейся целой части. Операция заканчивается, когда дробная часть полностью обратится в нуль.

Для примера переведем 10,62510 в двоичную систему:

0,625\*2 = 1,25

0,250\*2 = 0,5

0,5\*2 = 1,0

Записав все остатки сверху вниз, получаем 10,62510 = (1010), (101) = 1010,1012.