

# Архитектура ВС (лекция 2)

И снова Мега-преподаватель  
еще более к.т.н., и полный доцент  
Яблоков Евгений Николаевич

# Измерение производительности ЭВМ

Производительность ЭВМ можно измерить проведением группы тестов.

Производительность измеряется для конкретного параметра ЭВМ. 2 наиболее популярных параметра для оценки:

- 1) IPS – Instructions Per Second (операций в секунду с целыми числами)
- 2) FLOPS – Floating-point Operations Per Second (операций в секунду числами с плавающей запятой)

# Измерение производительности ЭВМ (2)

- **Пиковая производительность** – максимум быстродействия компьютера при идеальных условиях. Этот максимум определяется как число операций, выполняемое в единицу времени всеми имеющимися в процессоре обрабатывающими логико-арифметическими устройствами.
- **Теоретическая производительность** – сравнительная единица измерения производительности компьютерных процессоров, означающая количество миллионов теоретических операций в секунду, которые способен выполнять тот или иной процессор.
- **Реальная производительность** – быстродействие компьютера в реальных условиях работы. Существенно ниже пиковой и теоретической производительности

# Измерение производительности ЭВМ (3)

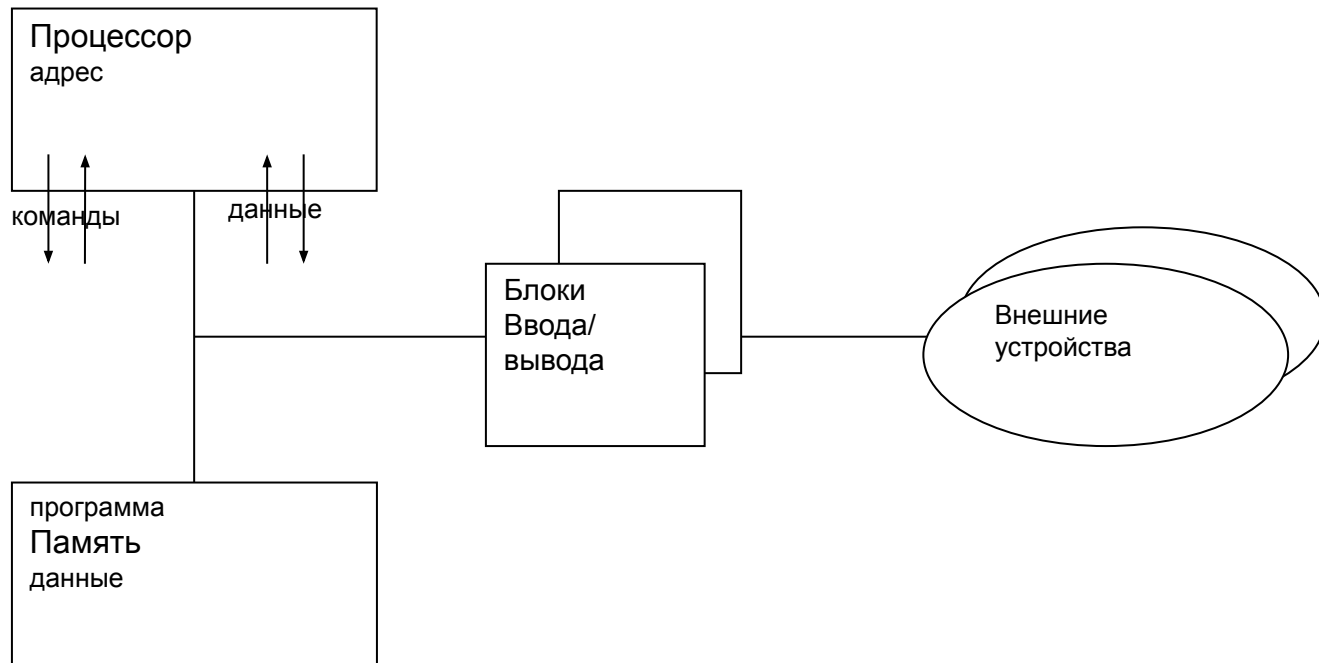
Существуют три проблемы, связанные с анализом результатов тестирования производительности:

- **1. Проблема выбора** – отделение показателей, которым можно доверять безоговорочно от тех, которые могут быть недостоверными
- **2. Проблема адекватности оценок** – выбор контрольно-оценочных тестов, наиболее точно характеризующих результат при обработке типовых задач пользователя
- **3. Проблема интерпретации** – правильное истолкование результатов тестирования

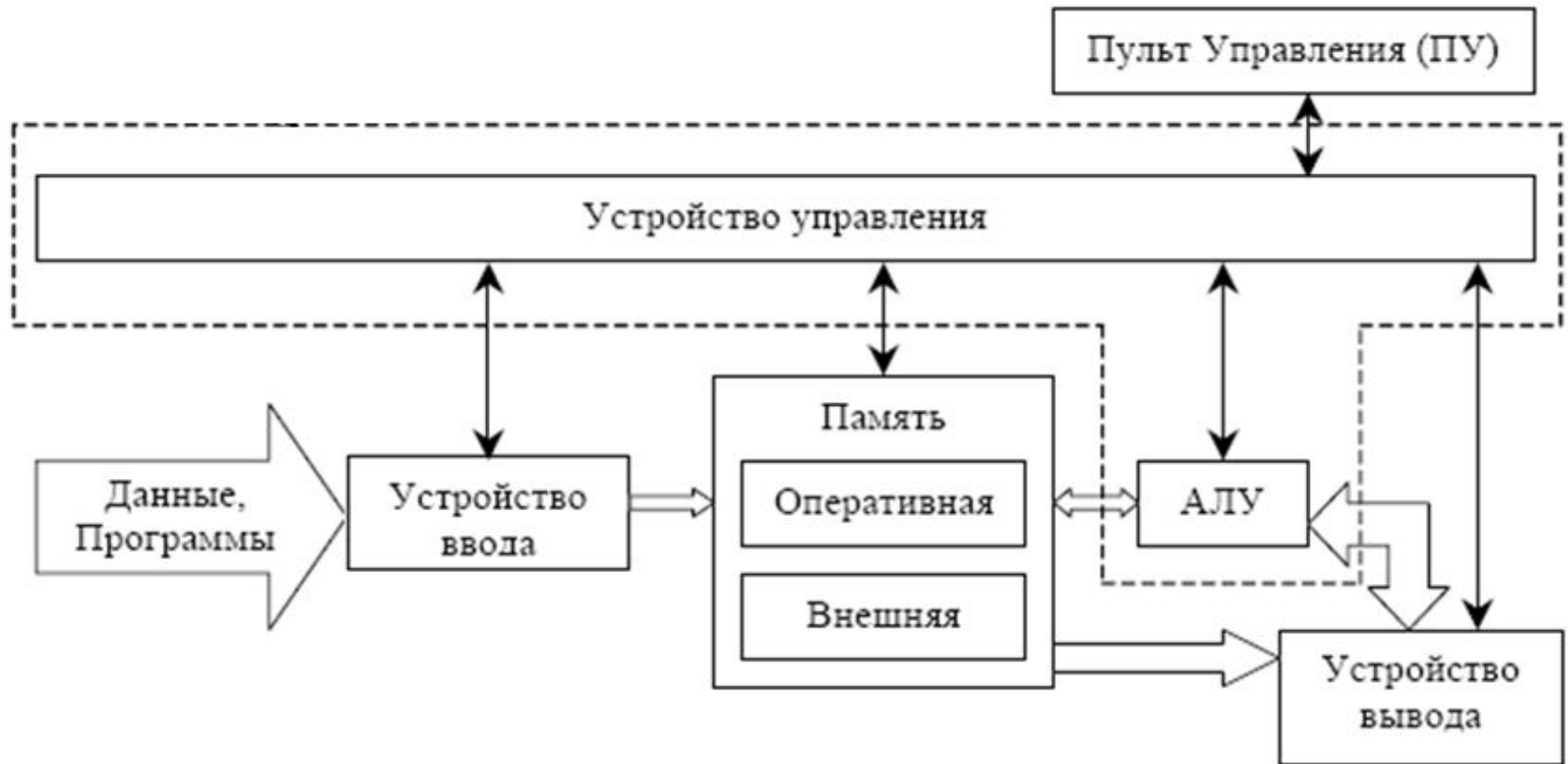
# Виды тестов

- **Тесты производителей.** Главная особенность данных тестов – ориентация на сравнение ограниченного множества однотипных моделей, часто относящихся к одному семейству. (Слоган - смотри, а этот красненький!)
- **Стандартные тесты.** Претендуют на роль полностью универсальных средств измерения. Пример – PC Mark. (Слоган - Мой айфон показал преимущество на 60 единиц над твоим самсунгом, мвахахахахаха! А еще у него мега камера!)
- **Пользовательские тесты.** Тесты предназначены для выбора компьютеров и ПО, наиболее подходящих для определенных прикладных задач. (Слоган - Да эта помойка даже dotV не тянет на максималках!)

# Общие принципы построения ЭВМ



# Общие принципы построения ЭВМ (2)



# Понятие алгоритма

**Алгоритм** – некоторая однозначно определенная последовательность действий, состоящая из формально заданных операций над исходными данными, приводящая к решению за конечное число шагов.

## Свойства алгоритма

- **Дискретность.** Действия выполняются по шагам, а сама информация дискретна)
- **Детерминированность.** Сколько бы раз один и тот же алгоритм не реализовывался для одних и тех же данных результат один и тот же.
- **Массовость.** Алгоритм «решает задачу» для различных исходных данных из допустимого множества и дает всегда «правильный» результат.

**Программа** – описание алгоритма на каком-либо языке.



# 6-ти уровневая система ЭВМ



# Принципы фон-Неймана

- Принцип произвольного доступа к основной памяти
- Принцип универсальности
- Принцип двоичного кодирования
- Принцип однородности памяти
- Принцип адресности
- **Принцип программного управления**

# Принцип программного управления

- 1. Принцип представления.** Любой алгоритм представляется в виде некоторой последовательности управляющих слов – команд.
- 2. Принцип условного перехода.** В процессе вычислений в зависимости от полученных промежуточных результатов возможен автоматический переход на тот или иной участок программы.
- 3. Принцип хранения.** Команды в ЭВМ представляются в такой же кодируемой форме, как и любые данные и хранятся в таком оперативном запоминающем устройстве (ОЗУ).
- 4. Принцип иерархии запоминающих устройств**

Информация в ЭВМ отличается не представлением, а способом ее использования!

НИКАК НЕЛЬЗЯ ПО ЯЧЕЙКЕ ПАМЯТИ ПОНЯТЬ ЧТО ЭТО ТАКОЕ – КОМАНДА, ИЛИ ДАННЫЕ, ИЛИ АДРЕС!

ТОЛЬКО КОМПЛЕКСНЫЙ АНАЛИЗ РЕГИСТРОВ ПРОЦЕССОРА ДАСТ ТАКУЮ ИНФОРМАЦИЮ!

# Кодирование данных

- **Текстовые данные**

Таблица ASCII или ANSI

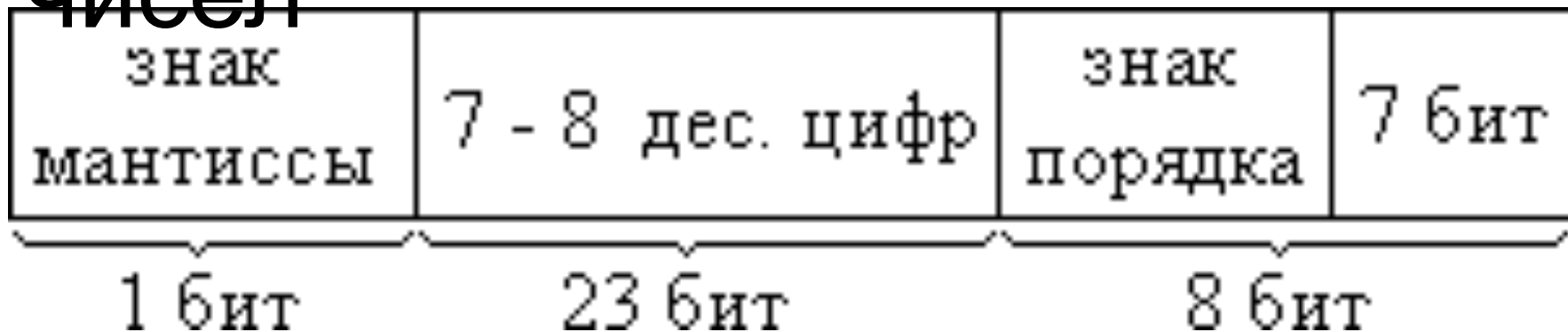
- **Целые числа**

В зависимости от разрядности данных и/или разрядности ОС

- **Кодирование вещественных чисел**

Вещественные числа в компьютере заменяются их кодами, которые образуют конечное дискретное множество. Каждый  $X_i$  од... оказывается представителем интервала  $\text{зн}[x_i; x_i + \Delta x]$  континуума

# Кодирование вещественных чисел



Возьмем простое число 25,324 в форме с плавающей точкой.

1) Переведем его в двоичную систему счисления с 24 значащими цифрами.  $25,324 = 11001,0101001011110001101$

2) Запишем в форме нормализованного двоичного числа с плавающей точкой:

$$0,110010101001011110001101 * 10^{101}$$

Здесь мантисса, основание системы счисления и порядок записаны в двоичной системе.

3) Вычислим машинный порядок (если нет знака порядка - нормализуем)

$$M_p2 = 101 + 1000000 = 100\ 0101$$

# Ошибки вещественных чисел

## • Ошибка округления

Математически

$$100000000 + 0.0000000001 = 100000000.0000000001$$

Для компьютера

$$100000000 + 0.0000000001 = 100000000$$

## Потеря значимости и переполнение

Математически

$$0.0000000001 / 100000000 = 0.00000000000000000001$$

Для компьютера

$$0.0000000001 / 100000000 = 0$$

# Вместо заключения

- ЭВМ функционирует на нескольких абстрактных уровнях, пользователю чаще всего доступен только верхний
- ЭВМ хранит данные и команды в памяти, которые извлекаются ей по мере необходимости
- Память ЭВМ, какой большой она бы не была ограничена физическими возможностями самой ЭВМ
- Двоичное кодирование было выбрано из-за простоты реализации на физическом уровне