

# Базовая архитектура IBM PC

## Представление данных

**Данные представляются в виде целых чисел в следующих форматах:**

**1 байт**

**2 байта = слово**

**4 байта = двойное слово**

**8 байтов = четверное слово**

**Кодирование данных:**

**дополнительный код**

# Базовая архитектура IBM PC

## Адресация

Разрядность шины адреса: 20 разрядов

Максимальный адрес:  $2^{20}$

$2^{20}$  байтов = 1 Мбайт

Адрес задаётся с помощью двух 16-разрядных значений – сегмента и смещения – и вычисляется по формуле

$$\langle \text{целевой адрес} \rangle = \langle \text{сегмент} \rangle * 16 + \text{смещение}$$

# **Базовая архитектура IBM PC**

## **Система команд**

- 1. Команды пересылки:**
  - а) между регистрами и памятью;**
  - б) между регистрами и устройствами ввода-вывода.**
- 2. Команды управления.**
- 3. Арифметические и логические команды.**
- 4. Команды манипулирования битами.**
- 5. Команды для обработки строк.**
- 6. Команды для поддержки механизма прерываний.**
- 7. Команды изменения состояния процессора.**

# **Базовая архитектура IBM PC**

## **Организация вычислительного процесса**

**Вычислительный процесс организован в полном соответствии с принципами фон Неймана.**

**Для ускорения введено понятие конвейера команд, из которого извлекается следующая команда.**

# Базовая архитектура IBM PC

## Регистры процессора: регистр флагов



**В регистре хранятся данные о состоянии процессора и результатах выполнения некоторых команд.**

**C – carry flag (флаг переноса) – выполнение операции привело к возникновению переноса**

**P – parity flag (флаг четности) – количество единиц в младшем байте результата чётно**

**A – auxiliary carry flag (флаг дополнительного переноса) – используется при операциях с двоично-десятичными числами**

# Базовая архитектура IBM PC

## Регистры процессора: регистр флагов

- Z – zero flag (флаг нуля)** – результатом операции был ноль
- S – sign flag (флаг знака)** – старший разряд результата имеет значение «1»
- T – trap flag (флаг трассировки)** – используется программами-отладчиками
- I – interrupt flag (флаг прерывания)** – процессор реагирует на прерывания
- D – direction flag (флаг направления)** – используется командами обработки строк
- O – overflow flag (флаг переполнения)** – устанавливается при переполнении (результат операции не помещается в регистре)

# Базовая архитектура IBM PC

## Регистры процессора: РОУы

<b>AX</b>	<b>AH</b>	<b>AL</b>
<b>BX</b>	<b>BH</b>	<b>BL</b>
<b>CX</b>	<b>CH</b>	<b>CL</b>
<b>DX</b>	<b>DH</b>	<b>DL</b>
<b>SI</b>		
<b>DI</b>		
<b>BP</b>		
<b>SP</b>		

# Базовая архитектура IBM PC

## Регистры процессора: РОУны

### Регистр AX (accumulator, аккумулятор)

**Это регистр-накопитель. Наиболее эффективно его использование в арифметических и логических операциях, а также в операциях пересылки, т.к. именно эти операции оптимизированы для использования регистра AX и, как правило, обладают более высоким быстродействием.**



# Базовая архитектура IBM PC

## Регистры процессора: РОНЫ

### Регистр ВХ (base, базовый регистр)

**В некоторых операциях этот регистр используется для реализации расширенной адресации.**

# Базовая архитектура IBM PC

## Регистры процессора: РОНЫ

### Регистр CX (counter, счётчик)

Обычно этот регистр используется как счётчик, указывающий количество выполнений команды или группы команд (циклические вычисления, сдвиги).

# Базовая архитектура IBM PC

## Регистры процессора: РОНЫ

### Регистр DX (data, регистр данных)

**Этот регистр используется в операциях умножения и деления, а также является единственным регистром, в котором может быть указан адрес порта в командах ввода-вывода.**

# Базовая архитектура IBM PC

## Регистры процессора: РОНЫ

### Регистры SI, DI

**Индексные регистры источника (SI, source index) и приёмника (DI, destination index), содержащие смещения относительно некоторого базового адреса. Обычно используются для выполнения операций над массивами данных.**

# Базовая архитектура IBM PC

## Регистры процессора: РОНЫ

### Регистр BP

**Базовый регистр, в котором содержится смещение относительно начала сегмента, в качестве которого по умолчанию предполагается сегмент стека. Обычно используется при организации вычислений в стековых структурах.**

# Базовая архитектура IBM PC

## Регистры процессора: РОНЫ

### Регистр SP (stack pointer, указатель стека)

**В SP содержится смещение относительно начала сегмента стека. При операциях со стеком система сама следит за изменениями содержимого SP в соответствии с выполняемыми операциями. В SP содержится адрес младшего байта данных, который был послан в стек последним.**

# Базовая архитектура IBM PC

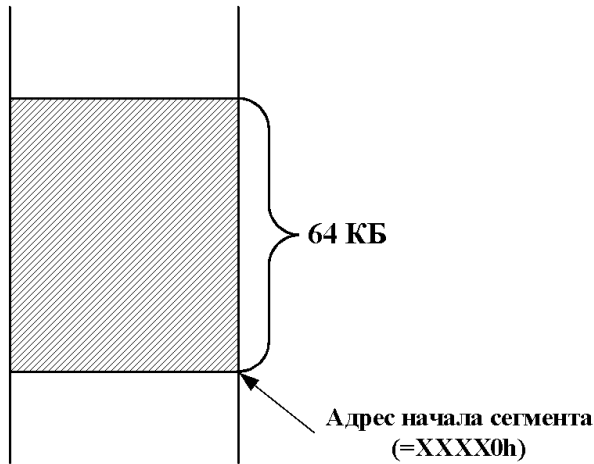
## Регистры процессора: РОНЫ

**Регистр IP (instruction pointer, счётчик команд)**

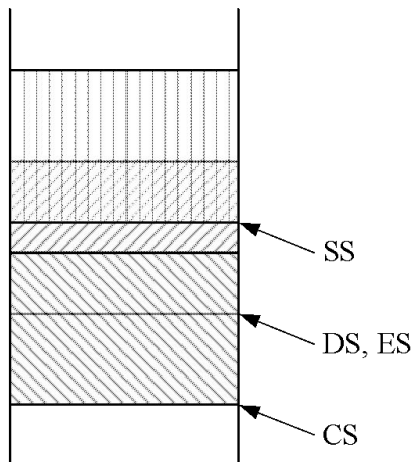
**Регистр содержит адрес команды, следующей за выполняемой в текущий момент, в сегменте памяти, который задаётся регистром CS.**

# Базовая архитектура IBM PC

## Сегменты



При работе с данными в пределах сегмента изменяется только смещение, адрес начала сегмента **не** меняется.

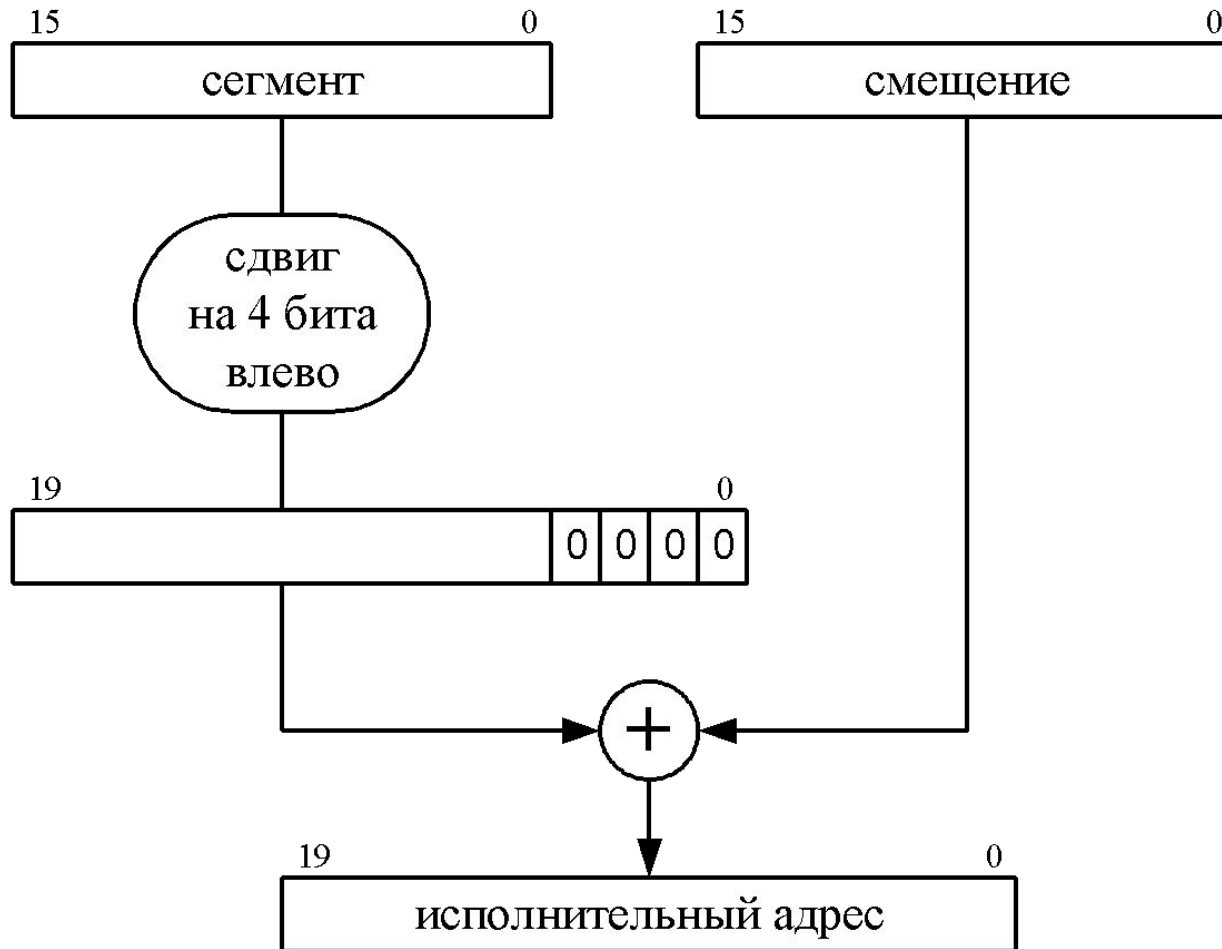


Сегменты в памяти могут перекрываться.



# Базовая архитектура IBM PC

## Сегменты: адресация



# Базовая архитектура IBM PC

## Регистры процессора: сегментные регистры

**CS (code segment)** – указывает на сегмент, в котором содержатся команды программы (начальный адрес сегмента кода). Адрес команды – CS:IP.

**DS (data segment)** – адресует начало сегмента данных.

**ES (extra segment)** – указывает на дополнительный сегмент данных; используется обычно при строковых операциях при формировании адреса приёмника данных.

**SS (stack segment)** – адресует сегмент стека.

# **Базовая архитектура IBM PC**

## **Система прерываний**

**Предусмотрены прерывания аппаратные и программные.**

**Всего в системе может быть до 255 прерываний.**

**Для реализации механизма прерываний выделен 1 КБайт оперативной памяти.**

# **Базовая архитектура IBM PC**

## **Организация ввода-вывода**

**Способы организации ввода-вывода:**

- 1) использование портов ввода-вывода;**
- 2) ввод-вывод с использованием ОП.**

**Для реализации первого способа в системе выделено 4 КБайта ОП.**

**Второй способ задействует также и порты ввода-вывода.**