

## Лекция 2. Многоуровневая организация ЭВМ

---

Участники вычислительного процесса



**Язык человеческого  
общения (образы)**



**Транслятор**

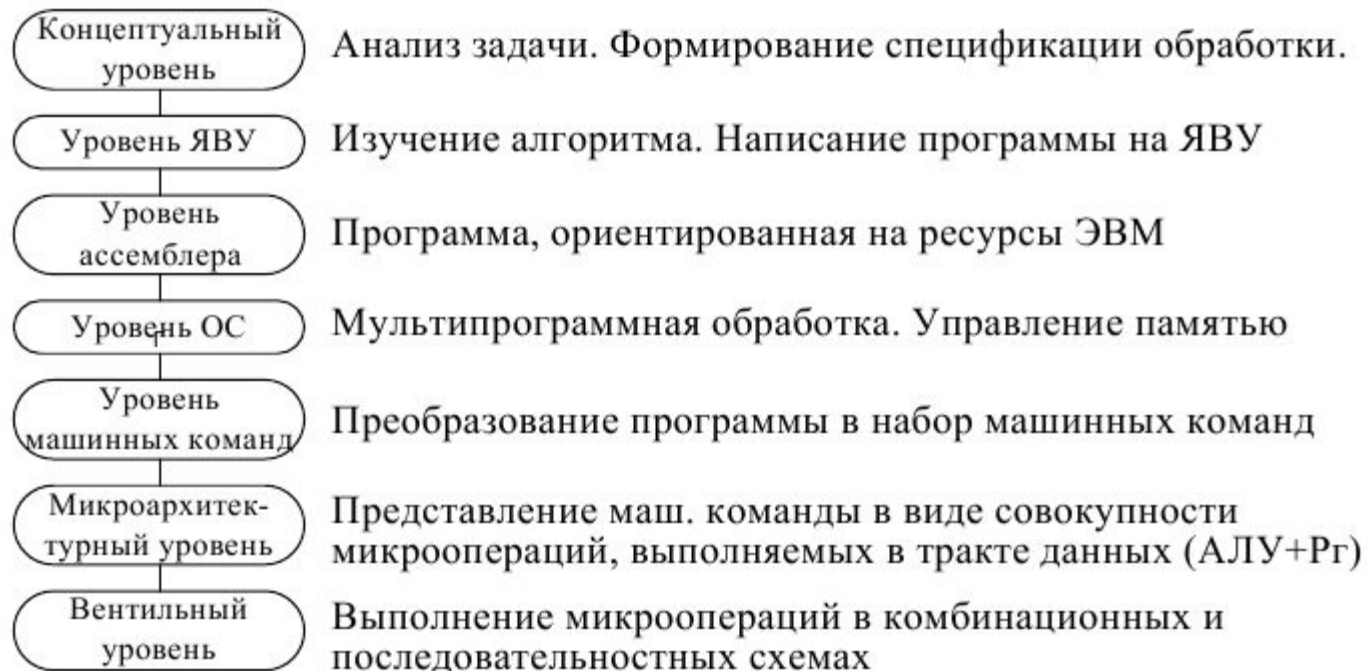


**Машинный язык  
(0010...101010)**

# Лекция 2. Многоуровневая организация ЭВМ

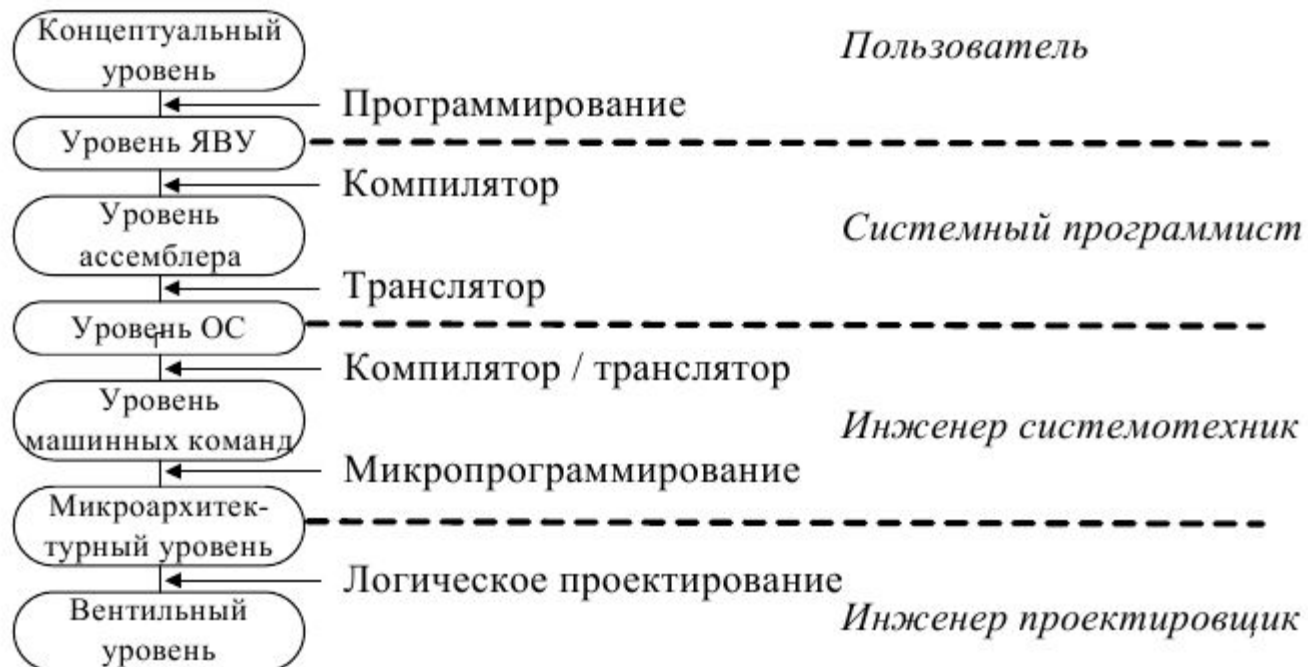
---

## Многоступенчатая обработка



# Лекция 2. Многоуровневая организация ЭВМ

## Многоступенчатая обработка



# Лекция 2. Многоуровневая организация ЭВМ

---

## Структурная организация и архитектура ЭВМ

### *Структура*

Характеристики, определяющие детали аппаратной реализации:

- количество функциональных блоков (БОД),
- принципы построения УУ,
- набор сигналов управления,
- технология функционирования памяти.

### *Архитектура*

Характеристики системы, доступные извне (со стороны программы):

- набор машинных команд,
- формат машинных команд,
- формат данных,
- способы адресации и т.д.

## Лекция 2. Многоуровневая организация ЭВМ

---

### **Концепция ВМ с хранимой в памяти программой: принципы фон-Неймана (John von Neumann)**

- *Принцип двоичного кодирования*: вся информация (команды и данные) кодируются двоичными цифрами 0 и 1 и имеют свой формат,
- *Принцип программного управления*: программа описывает выполняемые действия и представлена последовательностью управляющих слов – команд,

## Лекция 2. Многоуровневая организация ЭВМ

---

### **Концепция VM с хранимой в памяти программой: принципы фон-Неймана (John von Neumann)**

- *Принцип однородности памяти:* команды и данные хранят в одной памяти и внешне они неразличимы (принстонская и гарвардская архитектуры),
- *Принцип адресности:* память состоит из ячеек, в которых хранят единицы информации (слова). Для доступа к информации используют адрес – номер соответствующей

# Лекция 2. Многоуровневая организация ЭВМ

---

## **Структура и функции компьютера**

*Структура* – набор компонентов системы и способ объединения их в единое целое,

*Функции* – операции, выполняемые каждым компонентом в процессе реализации команды.

## **Рассмотрение сложных систем**

*Анализ* – с точки зрения описания,

*Синтез* – с точки зрения проектирования.

## Лекция 2. Многоуровневая организация ЭВМ

---

### Подходы анализа и синтеза

*Восходящий (снизу вверх)* – рассматривают функции элементов самого низкого уровня, поднимаясь затем по уровням иерархии (*p-n*-переход, транзистор, БЛЭ, сумматор, АЛУ, процессор, ЭВМ)

*Нисходящий (сверху вниз)* – рассматривают систему в целом, выделяя в ней независимые подсистемы (ЭВМ, процессор, ...).



# Лекция 2. Многоуровневая организация ЭВМ

---

## Функции компьютера (верхний уровень абстракции)

- обработка данных,
- хранение данных,
- перемещение данных,
- управление процессами.



## Лекция 2. Многоуровневая организация ЭВМ

---

### Структура компьютера (верхний уровень абстракции)

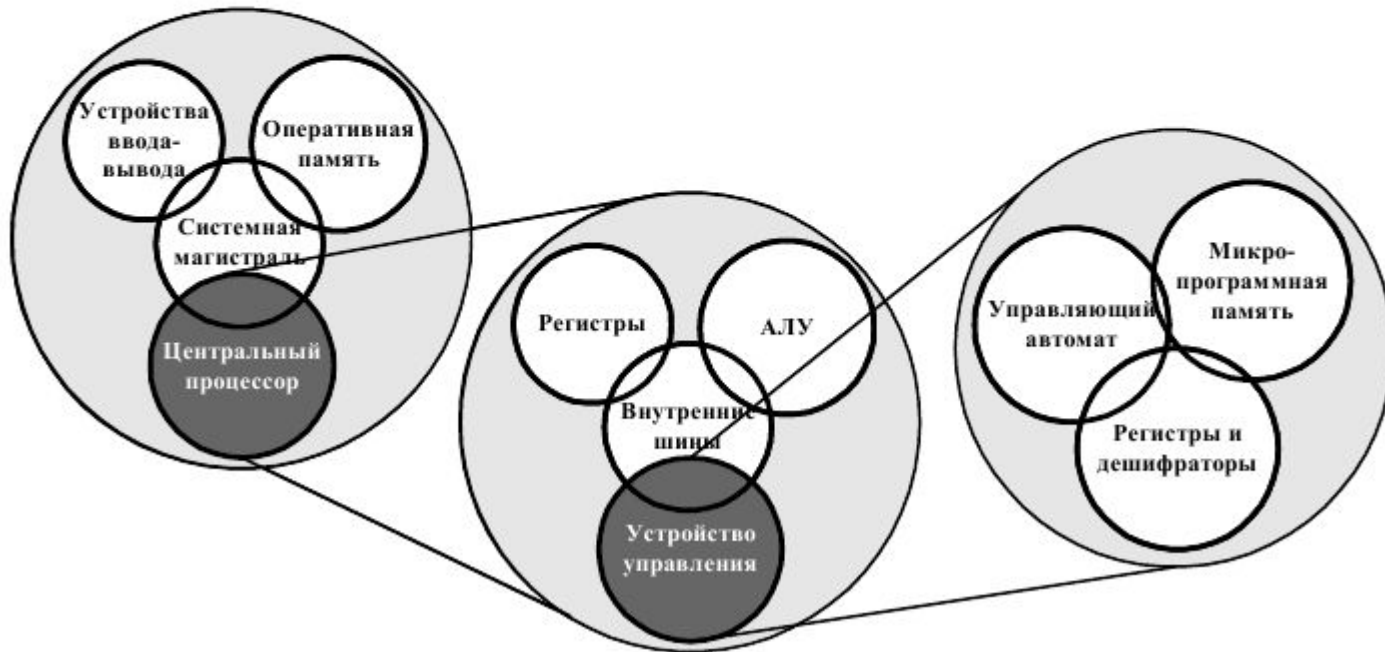
- центральный процессор (обработка данных),
- оперативная память (хранение данных),
- системная магистраль (перемещение данных),
- устройство управления (управление процессами).



# Лекция 2. Многоуровневая организация ЭВМ

---

## Иерархическая структура компьютера

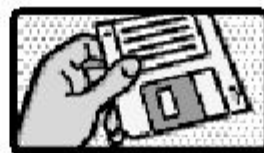


# Лекция 3. Структура и функции центрального процессора

---

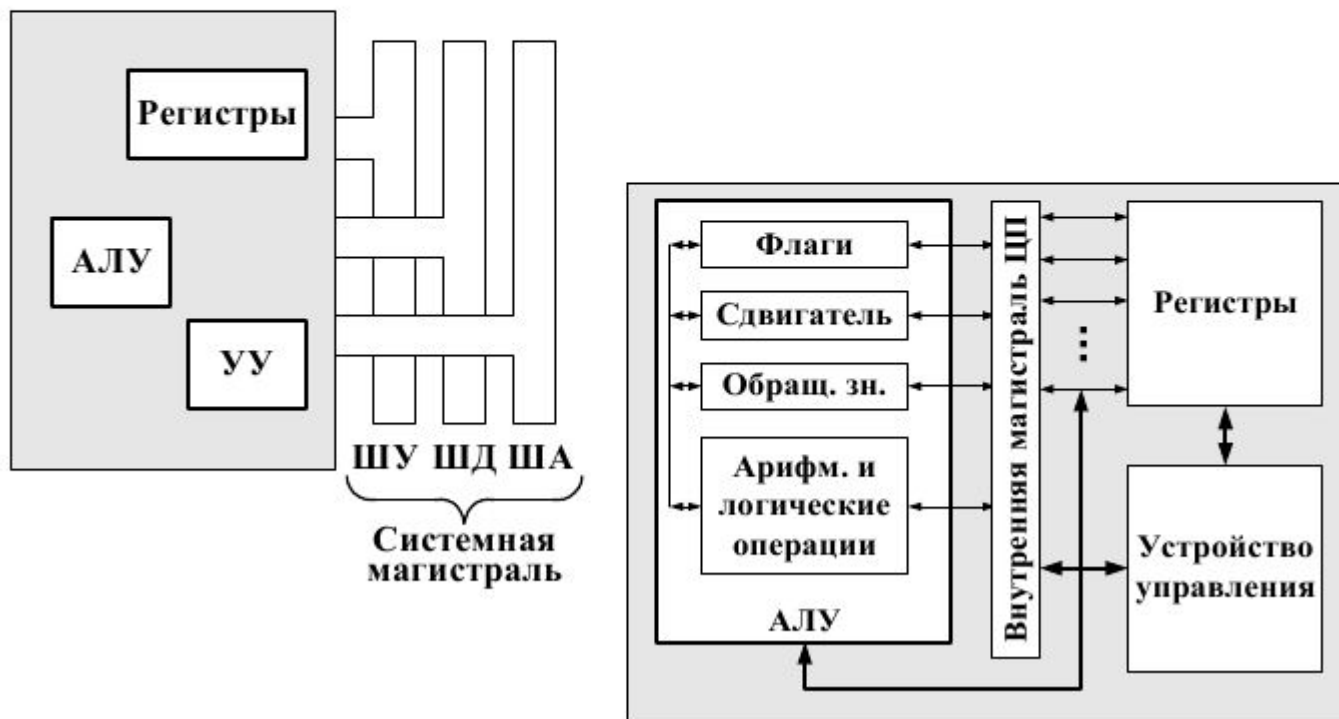
## Операции, выполняемые при обработке машинной команды

- Извлечение команды,
- Интерпретация команды,
- Извлечение данных,
- Обработка данных,
- Запись данных.



# Лекция 3. Структура и функции центрального процессора

## Внутренняя структура процессора



# Лекция 3. Структура и функции центрального процессора

---

## Организация регистров

Регистры процессора представляют память самого верхнего уровня. Их разделяют по функциям на *программно доступные*, которые минимизируют число обращений к медленной оперативной памяти и обеспечивают хранение промежуточных результатов, *управления и состояния*, которые используют для управления функционированием процессора (используют в программах ОС).

# Лекция 3. Структура и функции центрального процессора

## Организация регистров

### *Программно доступные*

- РОНЫ,
  - Регистры данных,
  - Регистры адреса,
  - Регистры кодов условий
- и др.

Универсальные. Программист использует по своему усмотрению

используют только для хранения операндов и результатов. Они не могут быть использованы при вычислении исполнительного адреса.

Используют для формирования адресов.

- **Указатель сегмента**, в котором хранится базовый адрес сегмента памяти.
- **Индексные регистры**, используют в режимах адресации с индексацией.
- **Указатель стека** содержит текущее значение указателя вершины стека в памяти.

Содержат однобитовые коды условий – флаги.

# Лекция 3. Структура и функции центрального процессора

---

## Организация регистров

### *Проблемы при проектировании программно доступных регистров:*

- Функциональное назначение (универсальные ↔ специализированные)
- Количество ПДР (8, 16, 32 ...),
- Разрядность,
- Сохранение ПДР в памяти при вызове подпрограмм, и др.



# Лекция 3. Структура и функции центрального процессора

---

## Организация регистров

### *Управления и состояния*

- Счетчик команд (СчК, PC – program counter),
- Регистр команд (РгК, IR – instruction register),
- Регистр адреса в памяти (РгАОП,  
MAR – memory address register),
- Регистр данных памяти (РгДОП, MBR – memory buffer register),
- Слово состояния программы (ССП, PSW – program status word).

# Лекция 3. Структура и функции центрального процессора

---

## Организация регистров IA-32

### 1. Основные функциональные регистры

- Регистры общего назначения,
- Указатель команд (EIP),
- Регистр флагов (EFLAGS),
- Регистры сегментов.

### 2. Регистры блока с ПТ

- Регистры данных R7-R0 (ST7 - ST0),
- Регистр тэгов (TW),
- Регистр состояния (FPSR),
- Регистр управления (FPCR),
- регистры-указатели команды (FIP) и операнда (FDP).

# Лекция 3. Структура и функции центрального процессора

---

## Организация регистров IA-32

### 3. Регистры обработки пакетов чисел с ПТ (SSE)

- Регистры пакетов данных (XMM7-XMM0),
- Регистр управления состоянием (MXCSR).

### 4. Системные регистры

- Регистры управления режимом CR4-CR0,
- Регистры системных адресов (GDTR, LDTR, IDTR, TR),
- Регистры отладки.

### 5. Служебные регистры

# Лекция 3. Структура и функции центрального процессора

---

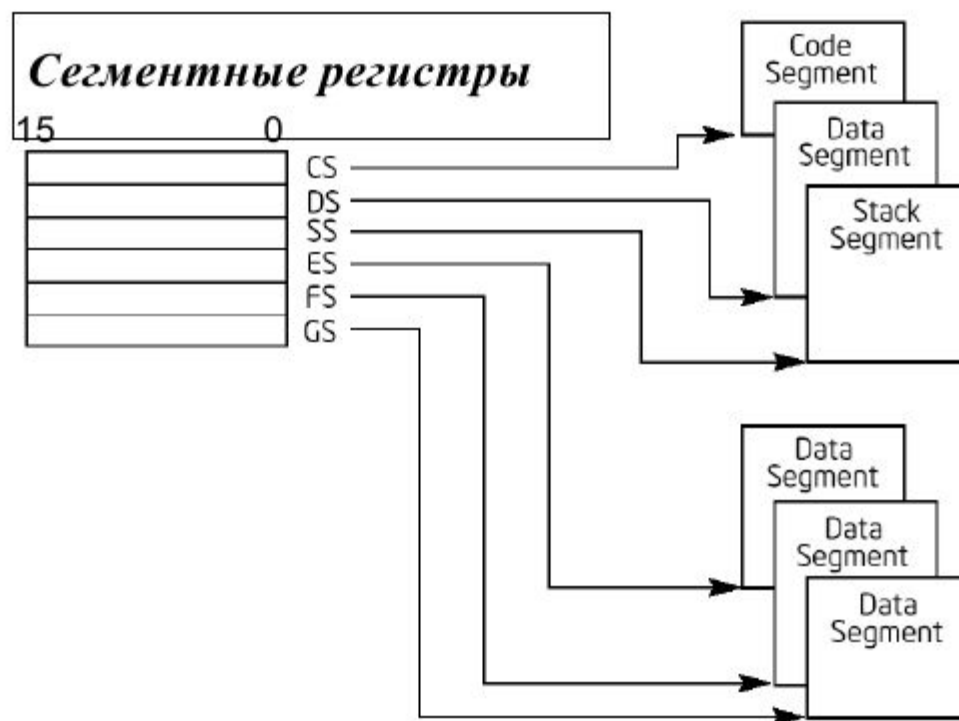
## Организация регистров IA-32

### *Регистры общего назначения*

31	16	15	8	7	0	16-bit	32-bit
	AH		AL			AX	EAX
	BH		BL			BX	EBX
	CH		CL			CX	ECX
	DH		DL			DX	EDX
			BP				EBP
			SI				ESI
			DI				EDI
			SP				ESP

# Лекция 3. Структура и функции центрального процессора

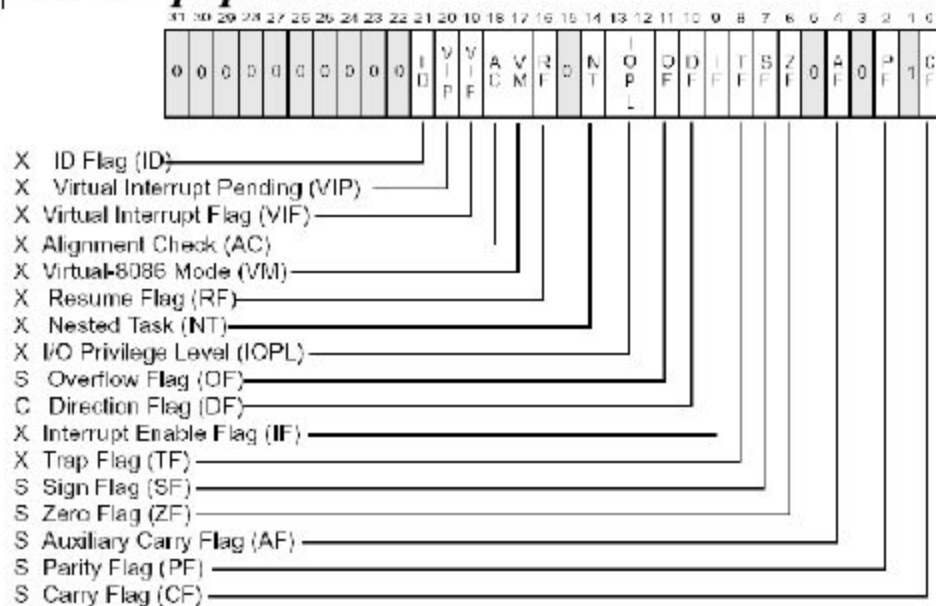
## Организация регистров IA-32



# Лекция 3. Структура и функции центрального процессора

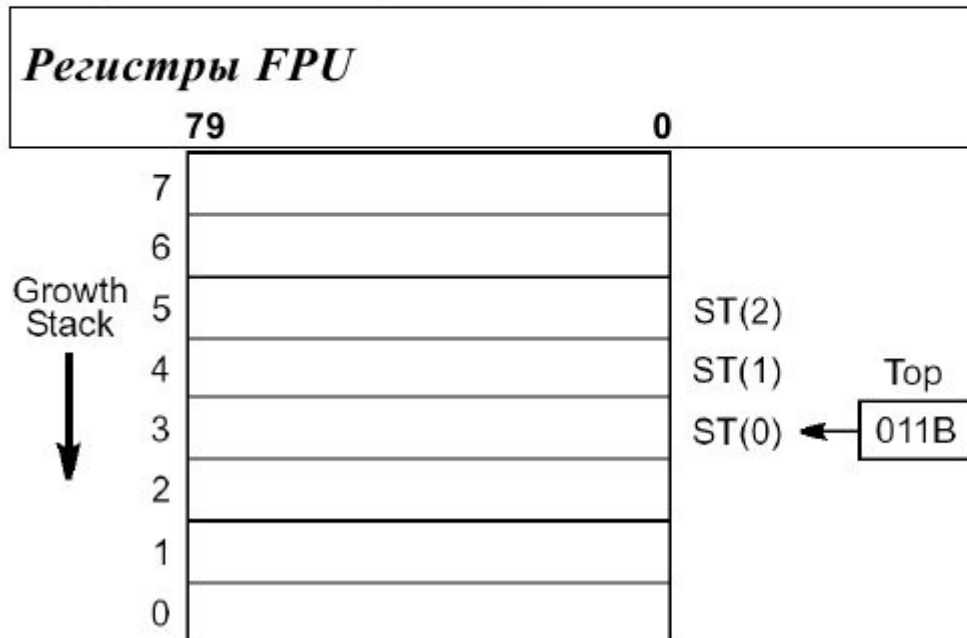
## Организация регистров IA-32

### Регистр флагов



# Лекция 3. Структура и функции центрального процессора

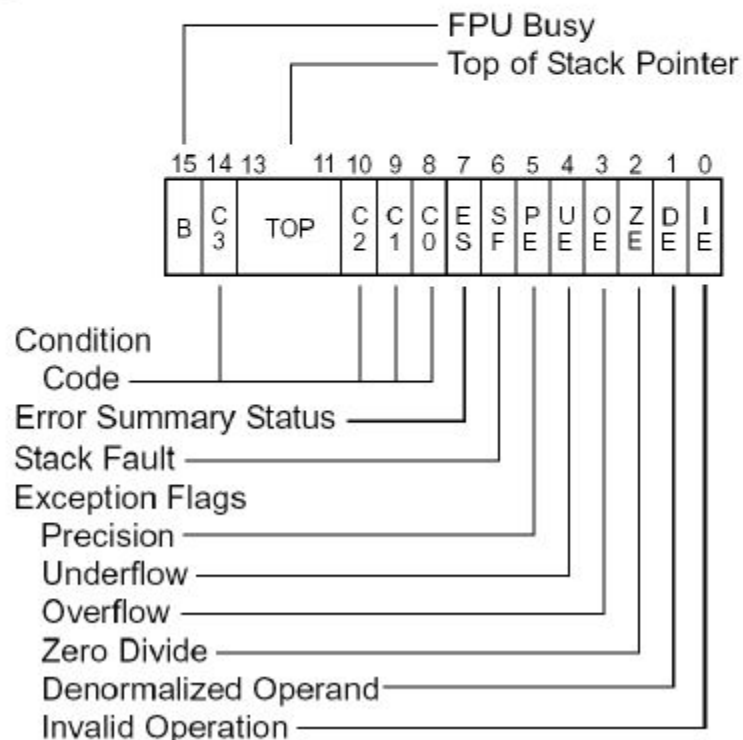
## Организация регистров IA-32



# Лекция 3. Структура и функции центрального процессора

## Организация регистров IA-32

*Регистры FPU  
(регистр состояния)*



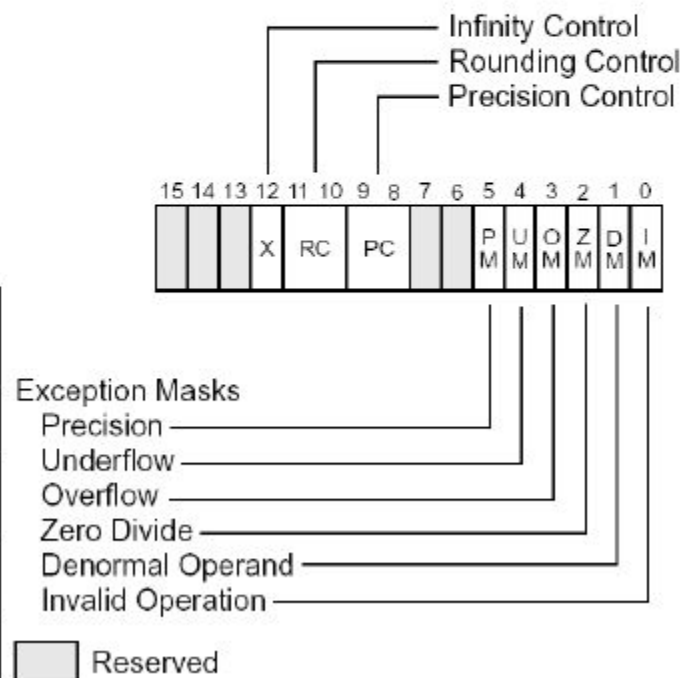


# Лекция 3. Структура и функции центрального процессора

## Организация регистров IA-32

**Регистры FPU**  
(регистр управления)

RC	Метод округления	PC	Точность представления
00	К ближайшему числу	00	Одинарная
01	К $-\infty$	01	Не используется
10	К $+\infty$	10	Двойная
11	К нулю	11	Расширенная

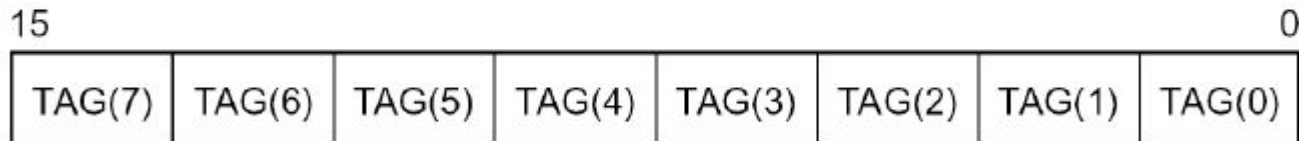


# Лекция 3. Структура и функции центрального процессора

---

## Организация регистров IA-32

*Регистры FPU  
(регистр тэгов)*



### TAG Values

- 00 — Valid
- 01 — Zero
- 10 — Special: invalid (NaN, unsupported), infinity, or denormal
- 11 — Empty

## Лекция 3. Структура и функции центрального процессора

---

### Арифметико-логическое устройство (АЛУ)

Минимальный набор операций, аппаратно реализуемых в АЛУ, должен включать:

- *основные логические операции* (И, ИЛИ, НЕ, Исключающее ИЛИ);
- *сдвиги* (логические и арифметические);
- *сложение и вычитание* чисел в формате с фиксированной точкой (ФТ).

На их основе программно можно реализовать остальные арифметические и логические операции над числами с ФТ и плавающей точкой (ПТ).

# Лекция 3. Структура и функции центрального процессора

## Арифметико-логическое устройство (АЛУ)

АЛУ как комплекс специализированных блоков обработки данных (БОД):

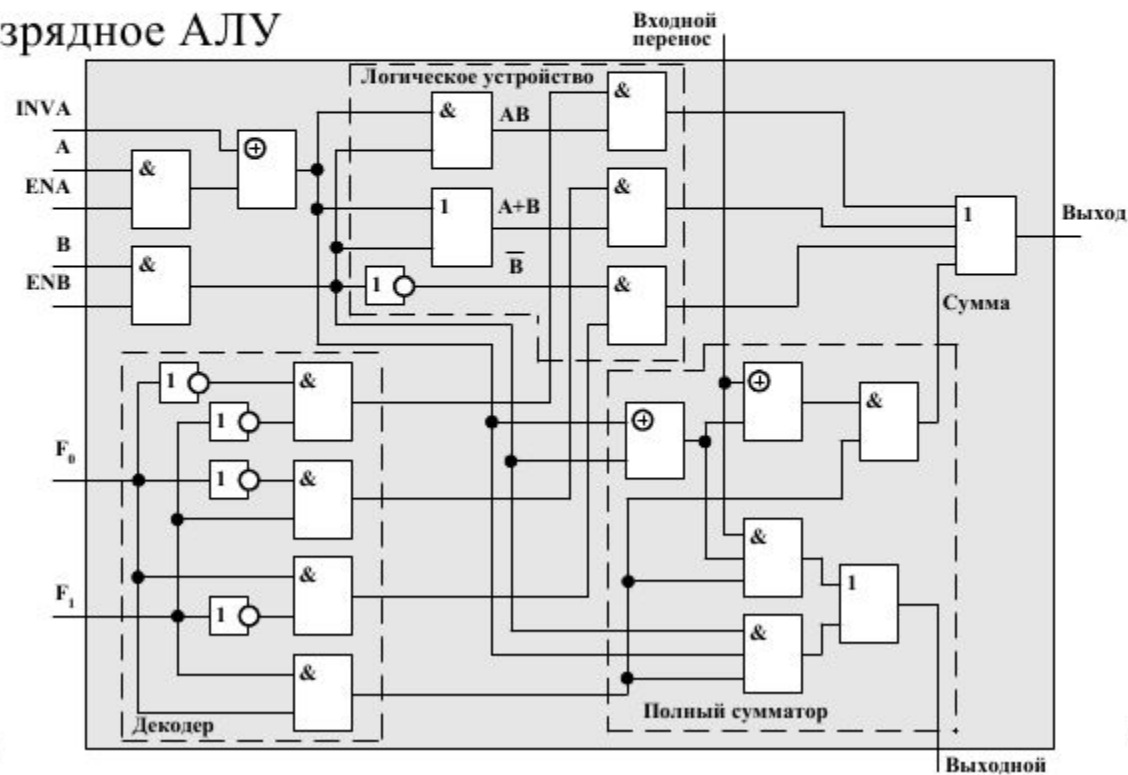
- БОД целочисленной арифметики;
- БОД логических операций и сдвигов;
- БОД арифметики с ПТ и т.д.

Сложение и вычитание с ФТ, сдвиги, логические операции	Умножение и деление с ФТ	Арифметические операции с ПТ	Элементарные функции и матричные операции
Аппаратура		Программа	
Аппаратура		Программа	
Аппаратура		Программа	

# Лекция 3. Структура и функции центрального процессора

## Арифметико-логическое устройство (АЛУ)

### Одноразрядное АЛУ



# Лекция 3. Структура и функции центрального процессора

---

## Арифметико-логическое устройство (АЛУ)

### Структура АЛУ



*Жесткая*

*Магистральная*

Комбинационные схемы жестко  
распределены между всеми  
регистрами

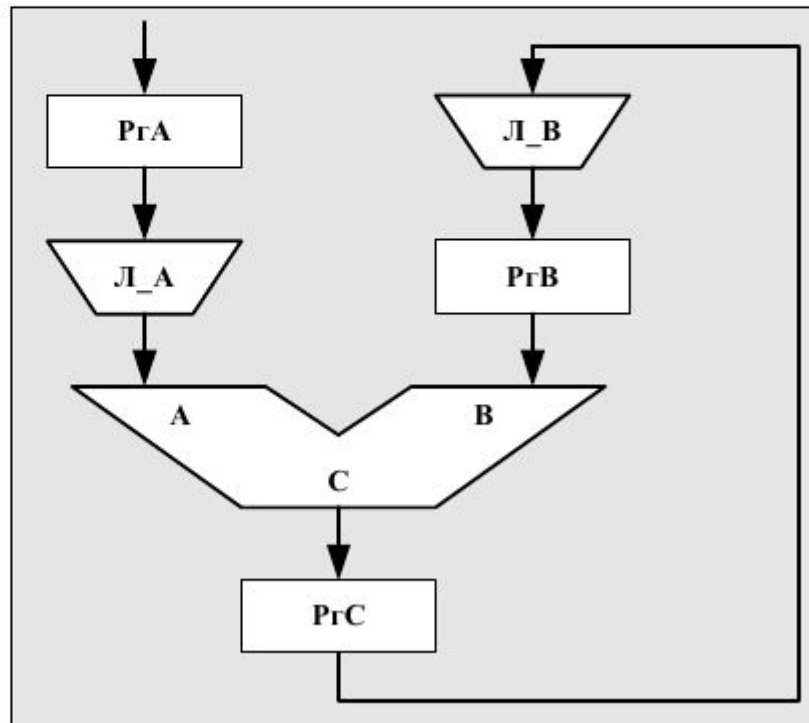
БОД и регистры  
взаимодействуют между собой  
с помощью магистралей (шин)

# Лекция 3. Структура и функции центрального процессора

## АЛУ с жесткой структурой

РгА через логическую схему Л\_А соединен со входом А.

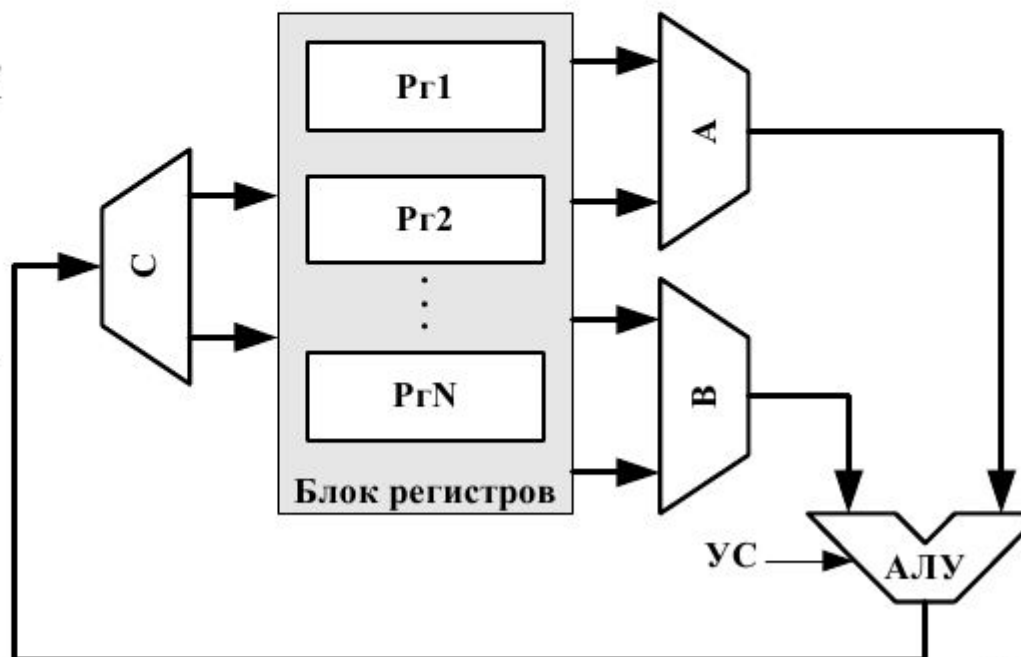
Л\_В обеспечивает коммутацию выходного регистра РгС с регистром РгВ.



# Лекция 3. Структура и функции центрального процессора

## АЛУ с магистральной структурой

Мультиплексоры *A* и *B* обеспечивают выбор любого регистра  $Pr_i$  в качестве операнда. Демультимплексор *C* обеспечивает запись результата в любой регистр.





# Лекция 3. Структура и функции центрального процессора

---

## Арифметико-логическое устройство (АЛУ)

### Затраты времени на выполнение операции



*Жесткая структура*

*Магистральная структура*

$$T = t_{\text{в}} + t_{\text{оп}} + t_{\text{с}}$$

$$T = t_{\text{в}} + t_{\text{оп}} + t_{\text{с}} + t_{\text{MUX}} + t_{\text{DMX}}$$

$t_{\text{в}}$  – длительность выборки операнда из регистра,

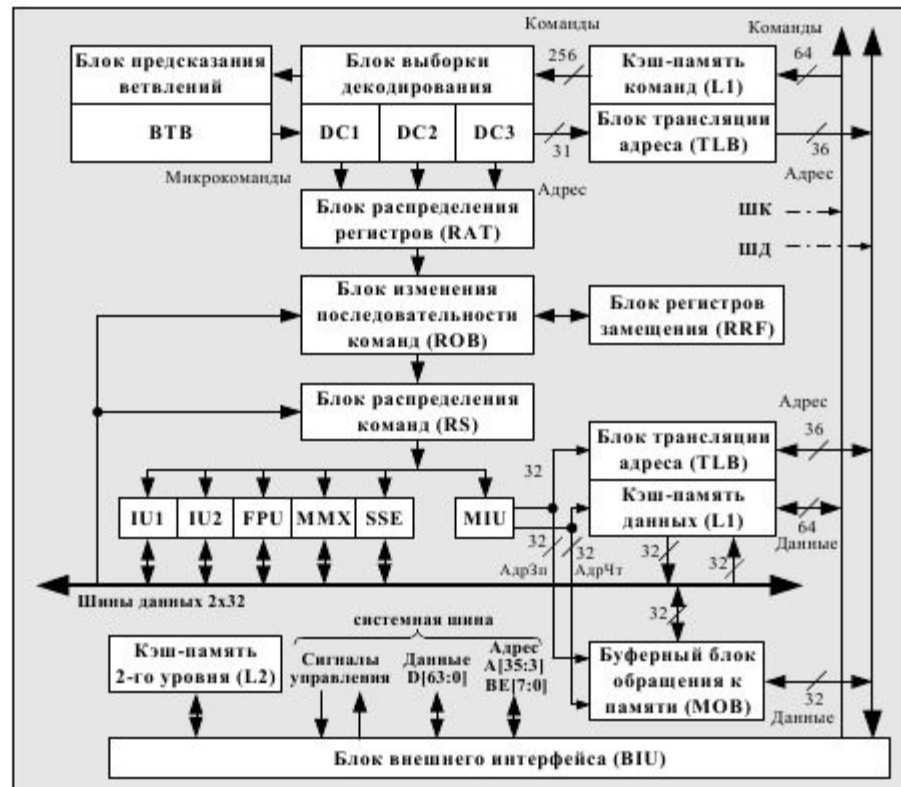
$t_{\text{оп}}$  – длительность операции,  $t_{\text{с}}$  – длительность сохранения

результата в регистре,  $t_{\text{MUX}}$  – задержка мультиплексора,

$t_{\text{DMX}}$  – задержка демультимплексора.

# Лекция 3. Структура и функции центрального процессора

## Структура процессора Intel P6



## Лекция 3. Структура и функции центрального процессора

---

### Структура процессора Intel P6

**Блок внешнего интерфейса (BIU):** реализует протокол обмена процессора с системной шиной.

**Внутренняя кэш-память команд и данных:** 16 кбайт каждая служит для размещения наиболее часто используемых команд и данных.

**Блок выборки декодирования** выбирает из кэш-памяти команд 32 байта командного кода, производит выделение и декодирование команды. Простые команды (выполняются за 1 микрооперацию) декодируют DC1 и DC2. Сложные команды (требуют несколько микроопераций) декодирует DC3.

# Лекция 3. Структура и функции центрального процессора

---

## Структура процессора Intel P6

**Блок предсказания ветвления** формирует адрес следующей выбираемой команды при ветвлении программы (условные переходы).

**Блок распределения регистров (RAT)** производит выделение регистров, необходимых для выполнения декодированных команд. Для логических регистров с ФТ или ПТ выделяет один из 40 физических регистров из **блока регистров замещения**.

**Блок изменения последовательности команд (ROB)** обеспечивает эффективное выполнение потока декодированных команд.

# Лекция 3. Структура и функции центрального процессора

---

## Структура процессора Intel P6

**Блок распределения (RS)** направляет микрокоманды в соответствующие исполнительные устройства (БОД).

Обеспечивает выполнение трех команд одновременно.

**Исполнительные блоки IU1 и IU2** производят обработку целых чисел (БФТ – блоки с фиксированной точкой).

**Блок FPU** выполняет операции над числами с плавающей точкой (БПТ – блок с плавающей точкой).

**Блок MMX** реализует одновременную обработку нескольких упакованных символов.

# Лекция 3. Структура и функции центрального процессора

---

## Структура процессора Intel P6

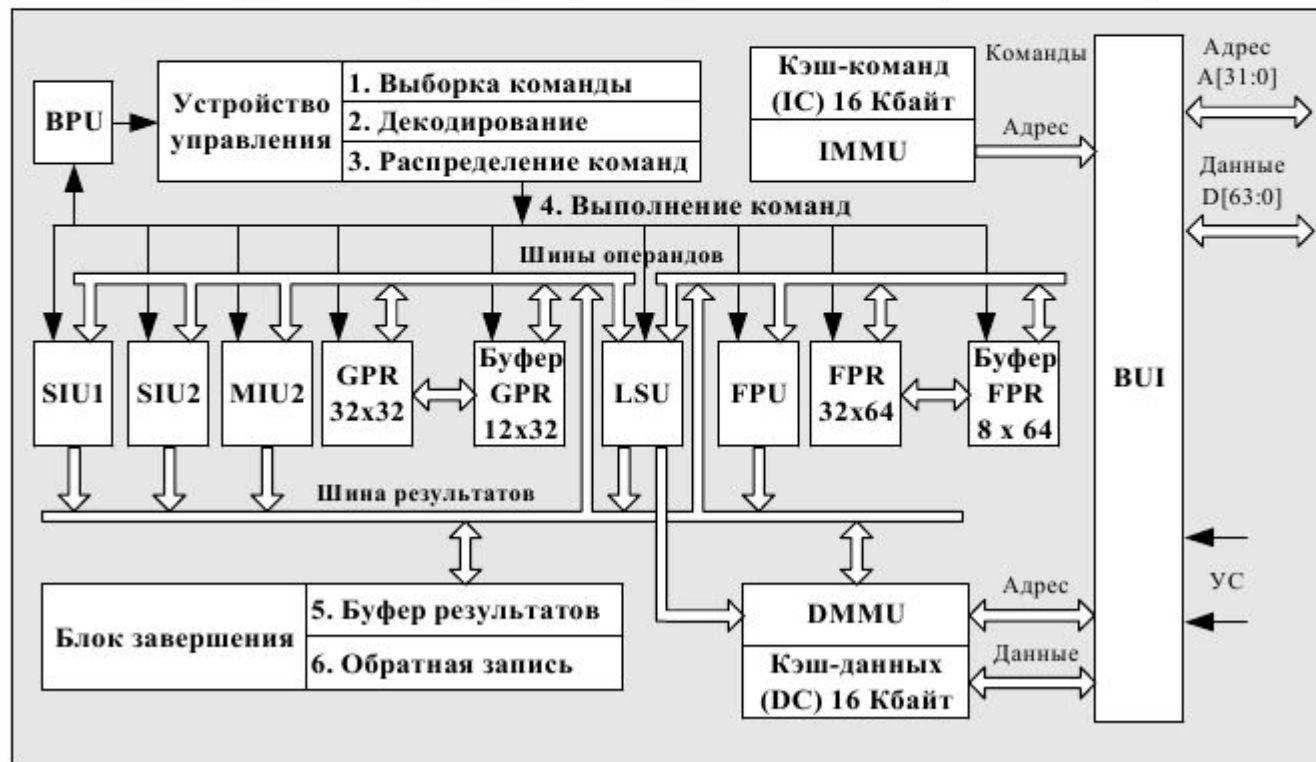
**Блок SSE** выполняет операции над потоком чисел с ПТ.

**Блок интерфейса с памятью (MIU)** вычисляет адреса операндов, выбираемых из памяти, и реализует интерфейс с кэш-памятью данных или внешней памятью.

**Буферный блок обращения к памяти (MOB)** обеспечивает спекулятивную выборку операнда из кэш-памяти второго уровня или внешней памяти при чтении и промежуточное хранение 32 байтов при записи.

# Лекция 3. Структура и функции центрального процессора

## Структура микропроцессора MPC604 (Motorola)



## Лекция 3. Структура и функции центрального процессора

---

### Структура микропроцессора MPC604 (Motorola)

**BPU** – блок обработки ветвлений.

**SIU1, SIU2** – устройства выполнения простых одноцикловых целочисленных операций.

**MIU** – устройство выполнения сложных многоцикловых целочисленных операций.

**FIU** – устройство обработки чисел в формате с плавающей точкой.

**LSU** – блок выборки данных из памяти и загрузки в регистры данных GPR и FPR.



# Лекция 3. Структура и функции центрального процессора

---

## Цикл обработки машинной команды

Процесс обработки отдельной команды принято называть **циклом обработки**. Основные операции:

- Вычисление адреса команды ( $СчК = СчК + const$ );
- Извлечение команды ( $РгК = ОП[СчК]$  );
- Расшифровка кода операции;
- Вычисление адреса операнда;
- Извлечение операнда;
- Обработка данных;
- Сохранение результата;
- Обработка прерываний.



# Лекция 3. Структура и функции центрального процессора

## Цикл обработки машинной команды



# Лекция 3. Структура и функции центрального процессора

## Цикл обработки машинной команды

Поток данных на фазе извлечения машинной команды

