

# Система команд Intel

## Условные переходы

Институт Информационных Технологий

Челябинский Государственный Университет

2011г.

# Система команд

## Команда копирования данных MOV

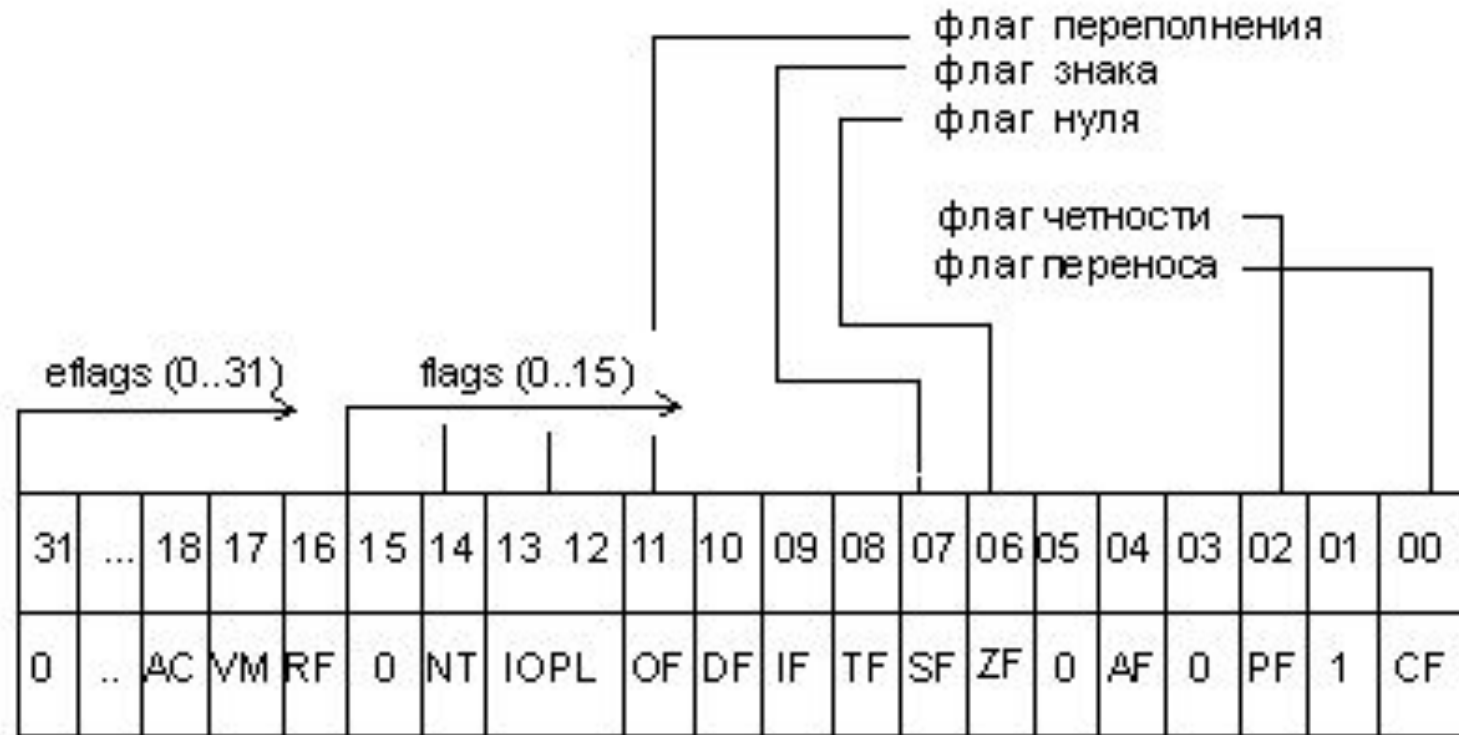
Опкод	Команда	1 операнд	2 операнд
88	MOV	r/m8	r8
89	MOV	r/m16/32	r16/32
8A	MOV	r8	r/m8
8B	MOV	r16/32	r/m16/32

## Команда сложения ADD

Опкод	Команда	1 операнд	2 операнд
00	ADD	r/m8	r8
01	ADD	r/m16/32	r16/32
02	ADD	r8	r/m8
03	ADD	r16/32	r/m16/32
04	ADD	AL	imm8
05	ADD	eAX	imm16/32

# Система команд

## Регистр флагов Intel

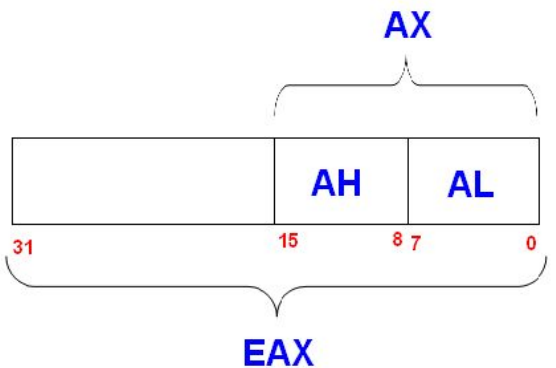


# Система команд

Пример:  
**mov ax,-10**  
**mov bx,-11**  
**add ax,bx**

**1111111111110110**  
**+**  
**1111111111110101**  
**= 11111111111101011**

**CF=1**    **Флаг переноса**  
**ZF=0**    **Флаг нуля**  
**SF=1**    **Флаг знака**  
**OF=0**    **Флаг переполнения**  
**PF=1**    **Флаг четности**

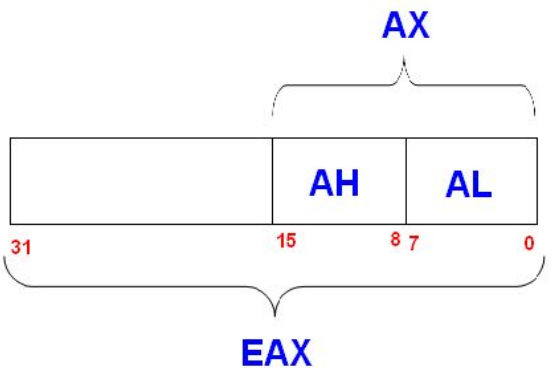


# Система команд

Пример:  
**mov ax,-10**  
**mov bx,11**  
**add ax,bx**

**1111111111110110**  
**+**  
**0000000000001011**  
**= 0000000000000001**

**CF=1**    **Флаг переноса**  
**ZF=0**    **Флаг нуля**  
**SF=0**    **Флаг знака**  
**OF=0**    **Флаг переполнения**  
**PF=0**    **Флаг четности**

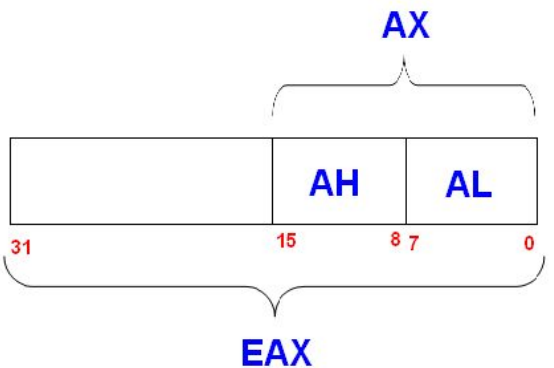


# Система команд

Пример:  
**mov al,127**  
**mov bl,1**  
**add al,bl**

**01111111**  
**+**  
**00000001**  
**= 10000000**

**CF=0**    **Флаг переноса**  
**ZF=0**    **Флаг нуля**  
**SF=1**    **Флаг знака**  
**OF=1**    **Флаг переполнения**  
**PF=0**    **Флаг четности**

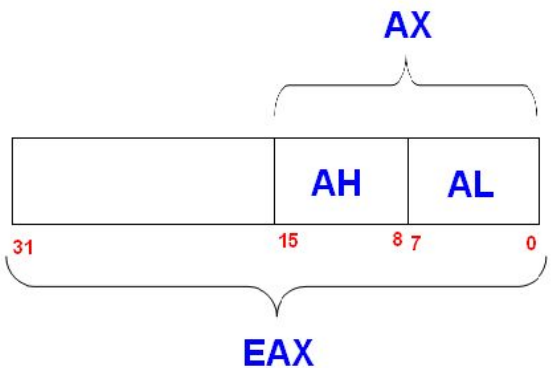


# Система команд

Пример:  
**mov al,255**  
**mov bl,1**  
**add al,bl**

**11111111**  
**+**  
**00000001**  
**= 00000000**

**CF=1** Флаг переноса  
**ZF=1** Флаг нуля  
**SF=0** Флаг знака  
**OF=0** Флаг переполнения  
**PF=1** Флаг четности



# Система команд

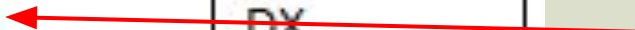
## Команды для работы с шиной

Опкод	Команда	1 операнд	2 операнд
E4	IN	AL	imm8
E5	IN	eAX	imm8
E6	OUT	imm8	AL
E7	OUT	imm8	eAX
EC	IN	AL	DX
ED	IN	eAX	DX
EE	OUT	DX	AL
EF	OUT	DX	eAX

Порт



Приемник  
данных



Источник  
данных



**IN** – читать данные с шины

**OUT** – отправить данные на шину



# Система команд



Отправить «00000101»  
на 8 порт

```
mov dx,8  
out dx,00000101b
```



Получить из 9 порта,  
записать в EAX

```
mov dx,9  
in eax,dx
```



# Система команд

## Команды перехода

Команда безусловного перехода

Адрес перехода

```
mov ax,bx  
jmp label3  
mov dx,cx  
add ax,dx  
label3: mov cx,ax
```

Команды будут  
пропущены

# Система команд

## Команды условного перехода

```
if (a>b)
{
...
}
```

Условный оператор в языке высокого уровня транслятор заменяет на по крайней мере 2 команды процессора:

- Команда сравнения
- Команда условного перехода

```
mov ax,a
cmp ax,b
jle label1
...
label1:
```

Процессор Intel не умеет работать с двумя операндами в памяти, поэтому один из них копируем в регистр

Сравниваем

Если **a<=b** переходим на адрес **label1**

# Система команд

## Команда `cmp`

38	<code>CMP</code>	<code>r/m8</code>	<code>r8</code>
39	<code>CMP</code>	<code>r/m16/32</code>	<code>r16/32</code>
3A	<code>CMP</code>	<code>r8</code>	<code>r/m8</code>
3B	<code>CMP</code>	<code>r16/32</code>	<code>r/m16/32</code>
3C	<code>CMP</code>	<code>AL</code>	<code>imm8</code>
3D	<code>CMP</code>	<code>eAX</code>	<code>imm16/32</code>

Алгоритм работы команды `cmp`:

1. **Вычесть из 1-го операнда 2-й**
2. **Соответствующим образом изменить регистр флагов**

Результат вычитания нигде не сохраняется

# Система команд

## Команды условного перехода

Сравнить

**A=00000011**

**B=00000001**

**00000011**

-

**00000001**

=00000010

CF=0

ZF=0

A>B

Сравнить

**A=00000011**

**B=00000011**

**00000011**

-

**00000011**

=00000000

CF=0

ZF=1

A=B

Сравнить

**A=00000001**

**B=00000011**

**00000001**

-

**00000011**


=11111110

CF=1

ZF=0

A<B

Занимаем  
у старшего  
разряда



# Система команд

## Команды условного перехода

Команда	Расшифровка	Со знаком/ беззнаковое	Условие перехода	Описание
JA	Jump if first operand is <b>Above</b> second operand	беззнаковое	if (CF = 0) and (ZF = 0) then jump	Переход, если первое больше второго. Числа без знака
JAE	Jump if first operand is <b>Above</b> or <b>Equal</b> to second operand	беззнаковое	if CF = 0 then jump	Переход, если первое больше или равно второму. Числа без знака
JB	Jump if first operand is <b>Below</b> second operand	беззнаковое	if CF = 1 then jump	Переход, если первое меньше второго. Числа без знака
JE	Jump if first operand is <b>Equal</b> to second operand	Со знаком и беззнаковое	if ZF = 1 then jump	Переход, если первое равно второму.

# Система команд

## Команды условного перехода

Команда	Расшифровка	Со знаком/ беззнаковое	Условие перехода	Описание
JG	<b>Jump</b> if first operand is <b>Greater</b> then second operand	Со знаком	if (ZF = 0) and (SF = OF) then jump	Переход, если первое больше второго. Числа со знаком
JGE	<b>Jump</b> if first operand is <b>Greater</b> or <b>Equal</b> to second operand	Со знаком	if SF = OF then jump	Переход, если первое больше или равно второму. Числа со знаком
JL	<b>Jump</b> if first operand is <b>Less</b> then second operand	Со знаком	if SF $\neq$ OF then jump	Переход, если первое меньше второго. Числа со знаком
JNE	<b>Jump</b> if first operand is <b>Not Equal</b> to second operand	Со знаком и беззнаковое	if ZF = 0 then jump	Переход, если первое не равно второму.

# Система команд

## Команды условного перехода

### Программа на C++

```
int a=5,b=3;  
if(a>b)  
{  
    b=a;  
}  
a++;
```

```
mov eax,a  
cmp eax,b  
jle label1  
mov b,eax  
label1:  
inc a;
```

Intel не умеет работать с двумя операндами в памяти

CF=0  
OF=0  
ZF=0

Переходим, если  $eax > b$ . Числа со знаком

b=a

a++