

# Лекция 10. Арифметические команды

---

## Сложение

обычное - **ADD**

сложение с переносом - **ADC**

инкрементация (+1) - **INC**

*Команда*

*процессором:*

**add bx, 4**

**adc byte ptr ds:[8], 2**

**inc ax**

*Выполнение команды*

**; bx ← bx + 4**

**; ds:[8] ← ds:[8] + 2 + CF**

**; ax ← ax + 1**

---

## Вычитание:

обычное - **SUB**,

вычитание с заемом - **SBB**,

декрементация (-1) – **DEC**.

*Команда*

**sub bx, 4**

**sbb ds:[8], cl**

**dec ax**

*Выполнение команды процессором:*

;  $bx \leftarrow bx - 4$

;  $ds:[8] \leftarrow ds:[8] - cl - CF$

;  $ax \leftarrow ax - 1$

---

## Сравнение: CMP

*Команда*  
cmp ax, 7

*Выполнение команды процессором:*  
; ax - 7

При выполнении команды сравнения CMP процессор вычисляет **разность между первым и вторым операндом**, которая никуда не записывается.

Результат команды являются **установленные арифметические флаги**, по которым можно судить о соотношении этих величин.

---

## Изменение знака числа: NEG

*Команда*  
neg edx

*Выполнение команды процессором:*  
; edx ← 0 - edx

Изменение знака числового операнда получается  
вычитанием его из нуля.

---

## Умножение:

беззнаковых – MUL, знаковых – IMUL

Команды имеют **один явный операнд**. Второй - по умолчанию.

Произведение размещается процессором в определенных регистрах (см. описание команды в системе команд)

### Важно!

- форматы операндов должны быть одинаковые
- формат произведения получится в два раза длиннее

Пример. Получить произведение 258 на 7

---

- ❑ Выбор формата операндов – слова.
- ❑ Один из операндов по умолчанию надо разместить в AX
  - `mov cx, 258` ; 1-й операнд в cx
  - `mov ax, 7` ; 2-й операнд в ax
  - `mul cx` ; dx-ax ← ax \* cx

Полученное 4-байтное произведение процессор «разложит» по регистрам dx - старшее слово и ax - младшее слово

Ручная отладка:  $258 * 7 = 1806_{10} = 70e_{16}$

Ожидаем 4-х байтное произведение `00 00 07 0e` (в hex):

в DX=`00 00`, в AX - `070e`

---

---

## Деление (целочисленное):

беззнаковых – **DIV**, знаковых – **IDIV**

Команды имеют **один явно задаваемый операнд - делитель**.  
Делимое берется процессором по умолчанию из определенных регистров (см. систему команд).

### Важно !:

- Формат делимого должен быть в 2 раза длиннее формата делителя;

Пример. Разделить 25 на 4

---

Если делитель - 4 разместить в формате байта, то делимое -25 должно быть 2-х байтным и по умолчанию размещаться в AX

```
mov ax, 25      ; ax - делимое  
mov dl, 4       ; dl - делитель  
div dl          ; ax /dl, в al ←частное, в ah ← остаток
```

Ручная отладка:  $25/4 = 6$  (частное) и 1 (остаток).

Ожидаем обнаружить их в регистрах: al=06h, ah= 01h