ОНЛАЙН-ОБРАЗОВАНИЕ

Меня хорошо слышно && видно?

Напишите в чат, если есть проблемы!

Ставьте + если все хорошо

OTUS

Арифметические, логические команды

Команды условного и безусловного перехода

Пакулов Артур

Команды перехода

Безусловный — **JMP** label/r

Условный — **JXX** label/r

Прямой переход - JXX label

Косвенный переход - JXX ax

Есть «n» ➔ переход нужно выполнить при флаг == 0

Нет «n» ➔ переход нужно выполнить при флаг == 1

Команды перехода

Безусловный — **JMP** label/r

Условный — **JXX** label/r

Мнемокод	*Содержательное условие для перехода после CMP op1,op2*	*Состояние флагов для перехода*
	для любых чисел:	
JE	op1=op2	ZF=1
JNE	op1<>op2	ZF=0
	для чисел со знаком:	
JL/JNGE	op1<op2	SF<>OF
JLE/JNG	op1<=op2	SF<>OF или ZF=1
JG/JNLE	op1>op2	SF=OF и ZF=0
JGE/JNL	op1>=op2	SF=OF
	для чисел без знака:	
JB/JNAE	op1<op2	CF=1
JBE/JNA	op1<=op2	CF=1 или ZF=1
JA/JNBE	op1>op2	CF=0 и ZF=0
JAE/JNB	op1>=op2	CF=0

Команды изменения флагов

```
CMP  op1, op2    op1 - op2

Test op1, op2    op1 & op2
```

Результат не записывается!

Изменяются флаги!

Команды изменения флагов

```
CMP  op1, op2    { op1 - op2
Test op1, op2    { op1 & op2
```

Результат не записывается!

Изменяются флаги!

Мнемокод	*Условие перехода*	*Мнемокод*	*Условие перехода*
JZ	ZF=1	JNZ	ZF=0
JS	SF=1	JNS	SF=0
JC	CF=1	JNC	CF=0
JO	OF=1	JNO	OF=0
JP	PF=1	JNP	PF=0

Команды изменения флагов

```
CMP  op1, op2    op1 - op2
Test op1, op2    op1 & op2
```

Результат не записывается!

Изменяются флаги!

Мнемокод	*Условие перехода*	*Мнемокод*	*Условие перехода*
JZ	ZF=1	JNZ	ZF=0
JS	SF=1	JNS	SF=0
JC	CF=1	JNC	CF=0
JO	OF=1	JNO	OF=0
JP	PF=1	JNP	PF=0

Существуют эквивалентные команды переходов

jz/je, jne/jnz, jb/jc,jnb/jnc

Команды перехода

```
LOOP   LABEL
```

Cx – счётчик повторов

Jcxz after – проверка регистра cx

Досрочный выход из цикла:

Loope/loopz – по счётчику и пока == 0

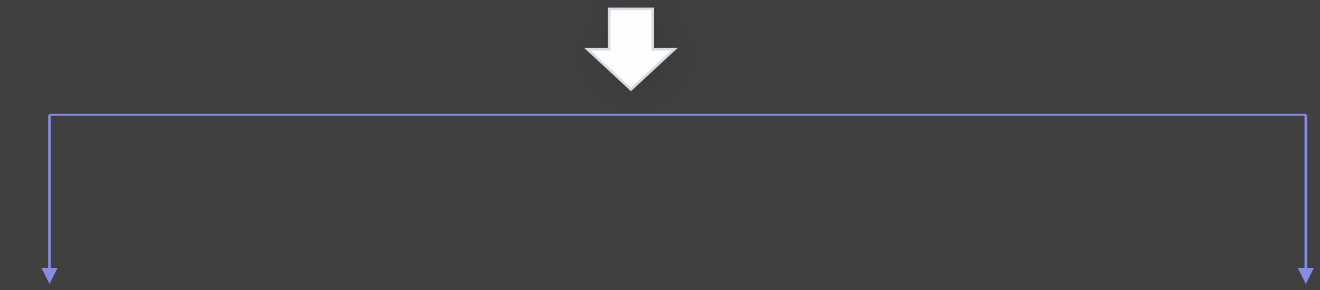
Внимание к деталям!

> **Для багхантеров**

```
mov al, 0x80

mov bl, 0x20

cmp al, bl
```



JA L

Будет ли переход?

JG L

Будет ли переход?

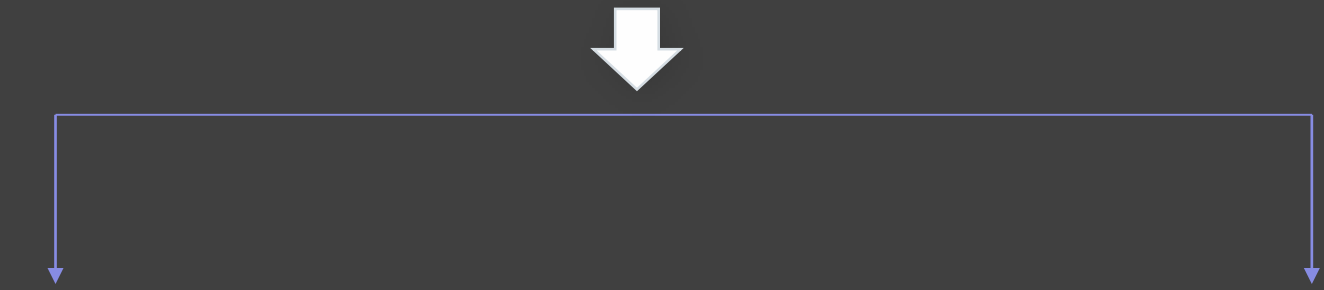
Внимание к деталям!

> **Для багхантеров**

```
mov al, 0x80

mov bl, 0x20

cmp al, bl
```



JA L

Будет переход!

JG L

Не будет ли перехода!

Представление данных

- Целые числа без знака
- Целые числа со знаком
- Вещественные числа
- Символы

Целые числа со знаком

N = 2^k

Пример:

1 байт = 2^8 бит ➔ . . . . . = [0..255]

Под знак отводится самый старший бит

. . . . . .

➔Остаётся 2^7 бит = -[0..127], +[0..127]

В итоге: [-128, 127]

Арифметические операции

Различают беззнаковые целые и знаковые целые числы

Описание	Беззнаковые целые	Знаковые целые
Сложение	ADD	
Вычитание	SUB	
Умножение	MUL	IMUL
Деление	DIV	IDIV

Арифметические операции

Беззнаковые целые. ADD

Сложение по модулю K

250 + 10 = 260 (100000100b)

CF

$$\text{сумма}(x,y)=(x+y) \bmod 2^k=\begin{cases} x+y, & \text{если } x+y< 2^k, \ CF=0 \\ x+y-2^k, & \text{если } x+y>=2^k, \ CF=1 \end{cases}$$

Арифметические операции

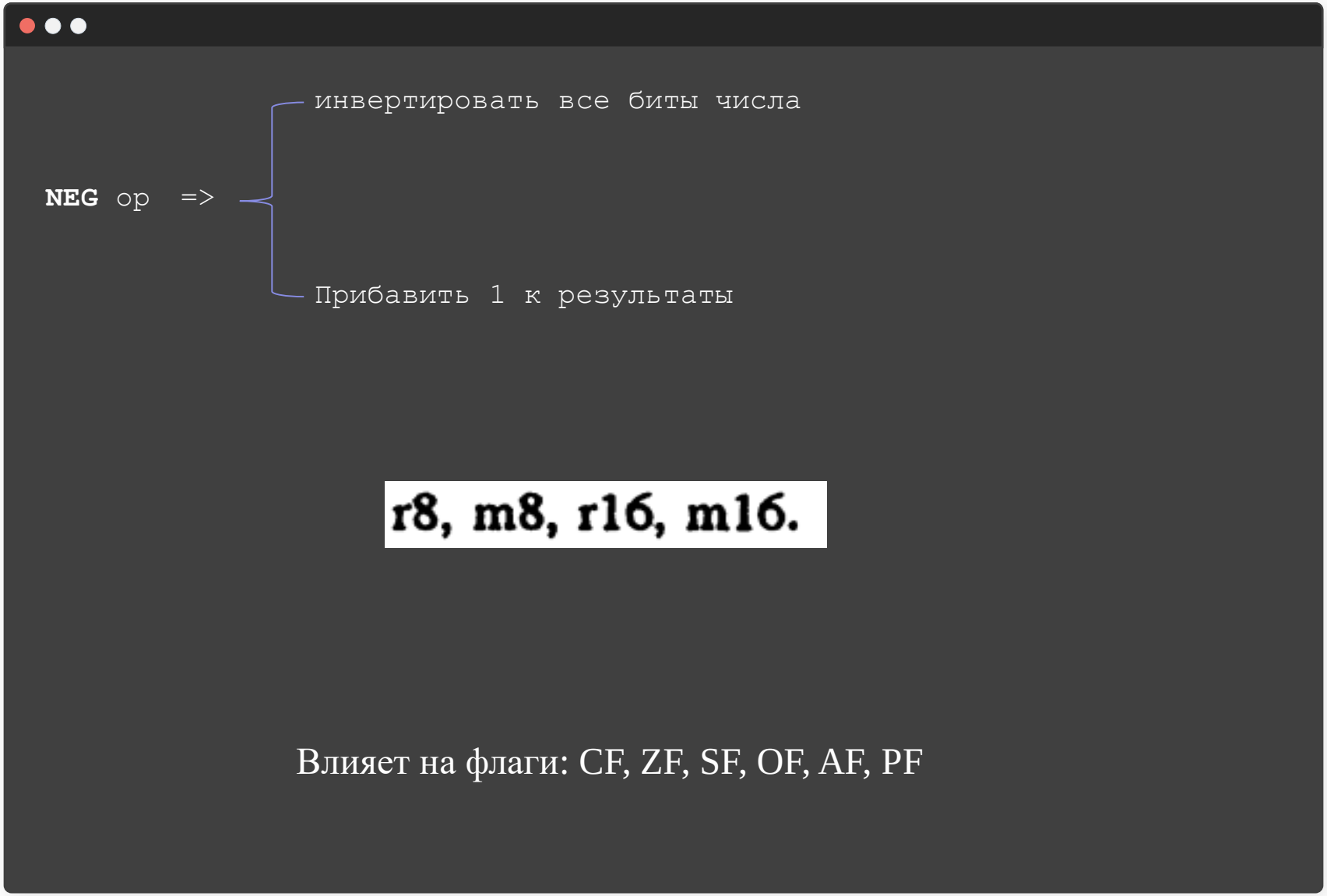
```
INC op      =>  op = op + 1

DEC op      =>  op = op - 1
```

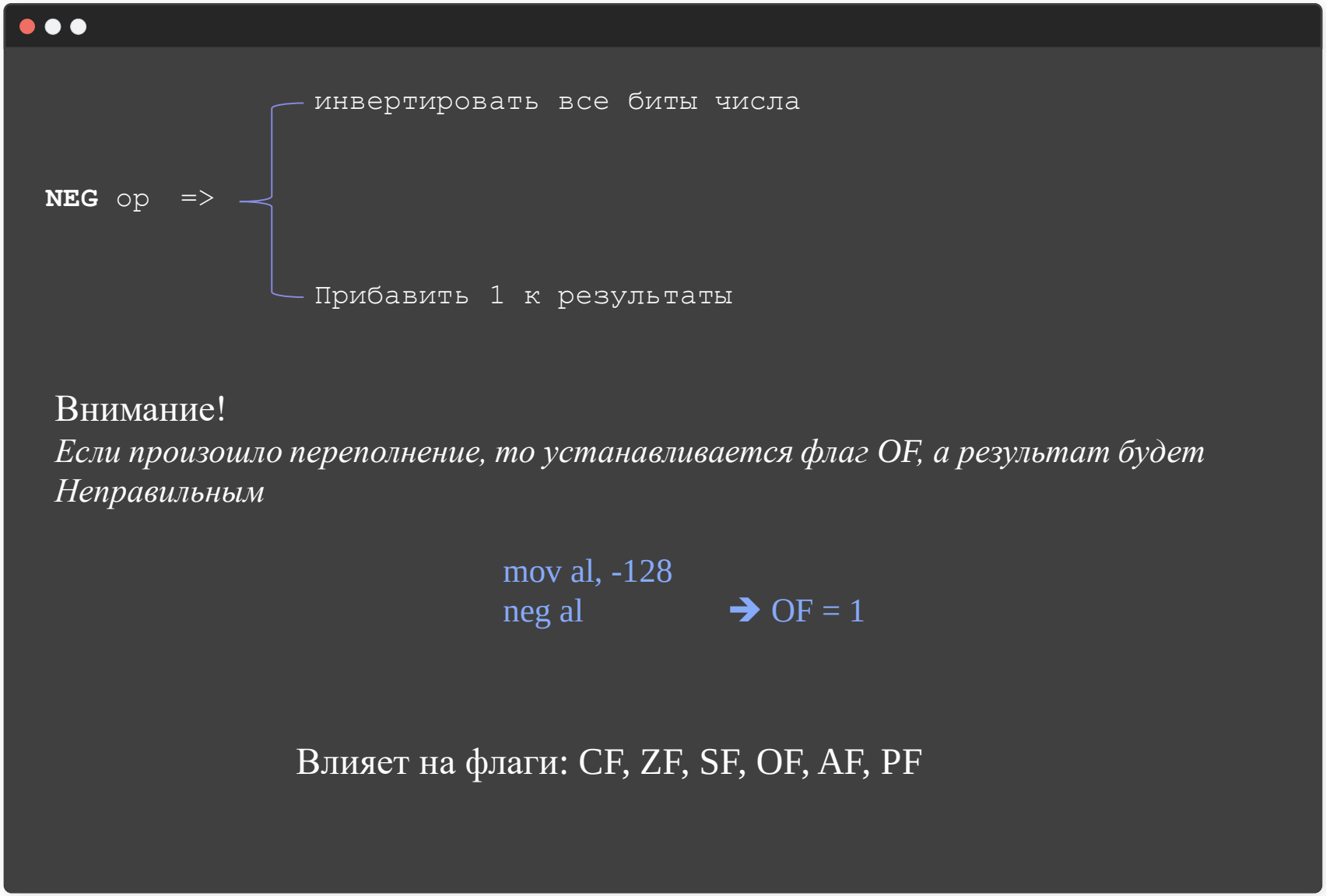
r8, m8, r16, m16.

Влияет на флаги: OF,SF,ZF,AF и PF (нет CF)

Изменение знака



Изменение знака



Внимание!

Если произошло переполнение, то устанавливается флаг OF, а результат будет Неправильным

```
mov al, -128
neg al          ➔ OF = 1
```

Влияет на флаги: CF, ZF, SF, OF, AF, PF

Сложение с учётом переноса

```
ADC op1, op2    =>  op1 + op2 + CF
```

op1	*op2*	
r8	i8, r8, m8	сложение/вычитание байтов
m8	i8, r8	
r16	i16, r16, m16	сложение/вычитание слов
m16	i16, r16	

Влияет на флаги: CF, ZF, SF, OF, AF, PF

Сложение с учётом переноса

```
Пример: 1234 + 500

;1234 = 00000100 11010010 al:bl

;500  = 00000001 11110100 cl:dl

;res  = 1734 = 0x06c6 ax:bx

mov al, 00000100b

mov bl, 11010010b

mov cl, 00000001b

mov dl, 11110100b
```

Вычитание с учётом заёма

```
SBB op1, op2    =>  op1 - op2 - CF
```

op1	*op2*	
r8	i8, r8, m8	сложение/вычитание байтов
m8	i8, r8	
r16	i16, r16, m16	сложение/вычитание слов
m16	i16, r16	

Влияет на флаги: CF, ZF, SF, OF, AF, PF

Команда умножения

```
MUL   op2          Умножение целых без знака

IMUL  op2          Умножение целых со знаком
```

Первый множитель в Al/Ax

Второй множитель в op2

Результат произведения записывается в AX/DX:AX

Влияет на флаги: CF, ZF, SF, OF, AF, PF

CF = OF = 0 *результат произведение уменьшается в размер типа*

CF = OF = 1 *результат произведение не уменьшается в размер типа*

Команда деления

```
DIV  op          Деление целых без знака

IDIV op          Деление целых со знаком
```

Делимое находится в AX/DX:AX

Op – byte/word

```
деление слова на байт:
AH:=AX mod op, AL:=AX div op                (op: r8, m8)
деление двойного слова на слово:
DX:=(DX,AX) mod op, AX:=(DX,AX) div op  (op: r16, m16)
```

В старшую часть попадает остаток

В младшую часть - целое

Команда деления

```
DIV  op          Деление целых без знака

IDIV op          Деление целых со знаком
```

- ✓ Деление на 0
- ✓ Результат не помещается в ячейку

```
MOV AX,600
MOV BH,2
DIV BH  ;
```

В старшую часть попадает остаток

В младшую часть - целое

Расширение слова до двойного слова/ байта до слова

Беззнаковые

Знаковые

Обнуление старшей части

CWD word ➔ DWORD

CBW byte ➔ WORD

Флаги не меняются

Логические операции

```
And, Or, Xor, Not
```

Операция	Синтаксис
&	and op1, op2
|	or op1, op2
^	xor op1, op2
!	not op

Операции сдвига

```
Shr, Shl
```

Операция	Синтаксис
<<	Shl op1, op2
>>	Shr op1, op2

Операции сдвига

Быстрое возведение в степень основания 2

```
Mov ax, x
```

➔ x * 2^y

```
Shl ax, y
```

OTUS

Вопросы???

ДЗ

Написать кейген к Otus_Crackme_01.exe

[fb42bfad815a9563b9f6fdd362b47f70]

CRACKME.EXE
[66f573036f8b99863d75743eff84f15d]

Пакулов Артур

A.Pakulov.Otus@Gmail.com

Спасибо
за внимание!