

# ISU Cvičení 4

Tomáš Dvořák

Vysoké Učení Technické

Fakulta Informačních Technologii



26. 2. 2020

- Název operand1, operand2
  - Registr, konstanta, paměť
  - Registr, konstanta, paměť
  - MOV, ADD, ADDC, SUB, SBB, DIV, IDIV, MUL, IMUL, AND, OR, XOR, ...
- Některé instrukce mohou mít 1 (**INC**, **DEC**) nebo žádný operand (**CWB**)
- Dnes se podíváme na aritmetické a logické instrukce

- **ADD** OP1, OP2 P+R, R+K, R+R, P+K, R+K
  - $OP1 = OP1 + OP2$
- **SUB** OP1, OP2 P+R, R+K, R+R, P+K, R+K
  - $OP1 = OP1 - OP2$
- Instrukce ADD a SUB mažou a nastavují CF (bezznaménková aritmetika) a OF (znaménková aritmetika)
- **INC** OP1 P, R
  - $OP1 = OP1 + 1$
- **DEC** OP1 P, R
  - $OP1 = OP1 - 1$
- Instrukce INC a DEC neovlivňují nastavování CF
- **NEG** OP1 P, R
  - $OP1 = 0 - OP1$
  - If  $OP1 == 0$  then  $CF = 0$  else  $CF = 1$

- **XOR**/AND/OR **OP1**, **OP2**      **P+R**, **R+K**, **R+R**, **P+K**, **R+K**
  - **OP1** = **OP1** **xor**/and/or **OP2**
- Logické instrukce pracují na fixním počtu bitů, nenastavují některé flagy (třeba CF)

- **ADC** = add with carry **OP1**, **OP2** P+R, R+K, R+R, P+K, R+K
  - **OP1** = **OP1** + **OP2** + **CF**
- **SBB** = subtraction with borrow **OP1**, **OP2** P+R, R+K, R+R, P+K, R+K
  - **OP1** = **OP1** - **OP2** - **CF**
- Instrukce ADC a SBB používáme, pokud chceme počítat s číslem, které přesahuje velikost registrů. Takové číslo se pak uloží do více registrů
- Příklad: Uvažujeme 64 bitové číslo uložené v registrech **EDX:EAX** (**00000001h**):(**00000000h**) a od tohoto čísla odečteme číslo 1

```
mov EDX, dword 1      ; nejvyšší bity
mov EAX, dword 0      ; nejvyšší bity
sub EAX, dword 1      ; odečtení 1 od spodních bitů - nastaví se CF
sbb EDX, dword 1      ; odečtení CF od horních bitů
```

- **MUL** **OP1** **P, R**
  - $AH:AL = AL \times OP1$  (Je-li OP1 8 bitový)
  - $DX:AX = AX \times OP1$  (Je-li OP1 16 bitový)
  - $EDX:EAX = EAX \times OP1$  (Je-li OP1 32 bitový)
  - Bezznaménková aritmetika
- IMUL (použití stejně jako MUL)
  - Znaménková aritmetika
- Produkt násobení vždy v předem určených registrech

- **DIV OP1** **P, R**
  - **AL = AX ÷ OP1** (Je-li OP1 8 bitový)
    - AH = AX % OP1
  - **AX = DX:AX ÷ OP1** (Je-li OP1 16 bitový)
    - DX = DX:AX % OP1
  - **EAX = EDX:EAX ÷ OP1** (Je-li OP1 32 bitový)
    - EDX = EDX:EAX % OP1
  - Beznaménková aritmetika
- IDIV (použití stejně jako DIV)
  - Znaménková aritmetika
- Produkt dělení vždy v předem určených registrech
- Současně s dělením se počítá zbytek po dělení (modulo, %)

- Při práci se znaménkovou aritmetikou je občas nutné rozšířit velikost hodnoty
- **CBW**
  - $AH:AL$  = rošíření  $AL$
- **CWD**
  - $DX:AX$  = rošíření  $AX$
- **CDQ**
  - $EDX:EAX$  = rošíření  $EAX$
- Rozšíření hodnoty se provádí nakopírováním nejvyššího bitu na horní bity výsledku



## Convert Byte to Word (AL->AH:AL)

```
section .data
    num db -101    ; 1001 1011b = 9Bh

section .text
    mov al, [num]
    cbw           ; AL = 9Bh,  AH = FFh
                  ; AX = FF9Bh
```

## Convert Word to Doubleword (AX->DX:AX)

```
section .data
    num dw -101    ; FF9Bh

section .text
    mov ax, [num]
    cbw           ; AX = FF9Bh
                  ; DX:AX = FFFFh:FF9Bh
```

## Convert Doubleword to Quadword (EAX->EDX:EAX)

```
section .data
    num dq -101    ; FFFFFFFF9Bh

section .text
    mov eax, [num]
    cbw           ; EAX = FFFFFFFF9Bh
    ; EDX:EAX = FFFFFFFFh:FFFFFFF9Bh
```

K rozšíření **bezznaménkového** čísla těmito instrukcemi je nutné **vynulovat** AH (resp. DX nebo EDX , v závislosti na použité instrukci).

```
mov al, [num]
cbw
xor ah, ah
```

Děkuji za pozornost

- Ve znaménkové aritmetice vypočítejte a vypište 16 bitovou hodnotu polynomu
$$P(x) = 3x^3 - 7x^2 + x$$
- Hodnotu proměnné  $x$  načtěte z klávesnice jako 8 bitové číslo

- Ve znaménkové aritmetice vypočítejte a vypište 64 bitovou hodnotu polynomu
$$P(x) = x^2 + 35x$$
- Hodnotu proměnné  $x$  načtěte z klávesnice jako 32 bitové číslo

- V bezznaménkové aritmetice vypočítejte a vypište 32 bitovou hodnotu polynom
$$P(a,b,c) = ((a + b - 4) \times c) \% a$$
- Hodnotu proměnných  $a$ ,  $b$  a  $c$  načtěte z klávesnice jako 32 bitová čísla

- V bezznaménkové aritmetice vypočítejte obsah pravoúhlého trojúhelníku. Z klávesnice načtěte délku odvěsen jako dvě 16 bitová čísla a vypište výsledek

- Ve znaménkové aritmetice vypočítejte absolutní hodnotu z čísla zadaného z klávesnice jako 32 bitové číslo