

Organisasi Sistem Komputer

Bab 3. Perpindahan Data

3.1 Instruksi MOV

3.2 Instruksi MOVZX dan MOVSX



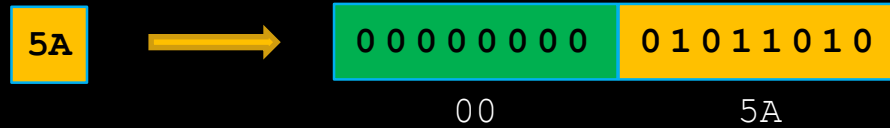
Pembahasan:

- Instruksi MOZX dan MOV SX

Membesarkan Ukuran Data

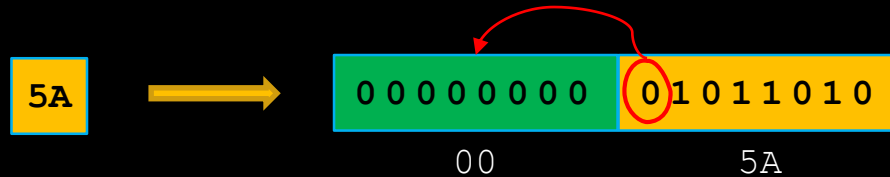
- Terkadang kita perlu untuk membesarkan ukuran data
- Untuk membesarkan ukuran bilangan tidak bertanda, kita menambahkan 0 ke sebelah kiri:

- Misal: 5A (90d)

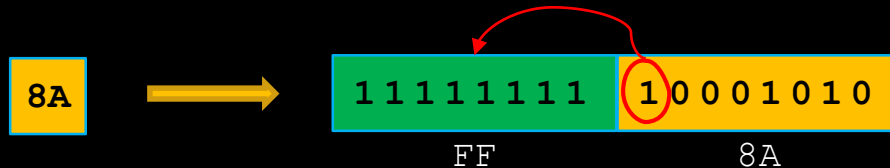


- Untuk membesarkan ukuran bilangan bertanda, kita mengekstensi bit tanda:

- Misal: 5A (+90d)



- Misal: 8A (-118d)



Instruksi MOZX dan MOSX

- Terdapat dua instruksi perpindahan data untuk membesarkan ukuran data:
 - MOVZX (*move with zero-extend*) untuk membesarkan ukuran bilangan tidak bertanda
 - MOVSX (*move with sign-extend*) untuk membesarkan ukuran bilangan bertanda
- Sintaks:

MOVZX *destination, source*

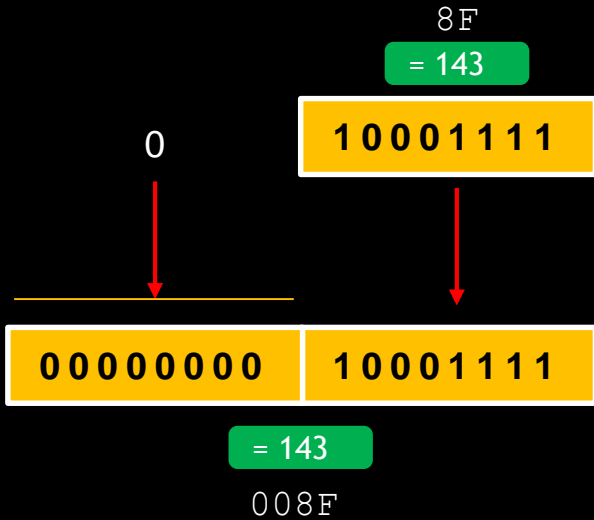
MOVSX *destination, source*

- Operand destination: register 16-bit/32-bit
- Operand source: register 8-bit/16 bit atau memori 1 byte/2 byte
- Operand destination tentunya harus lebih besar dari operand source

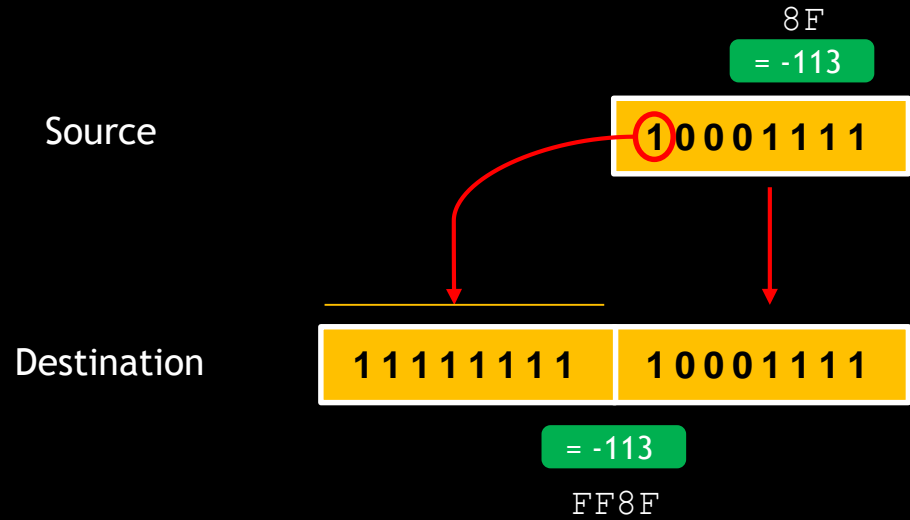


Instruksi MOVZX dan MOVSX

`MOVZX dst, src`



`MOVSX dst, src`



Menggunakan MOZX dan MOSX

- Membesarkan dari register 16-bit ke 32-bit
 - `movzx eax, ax ; ekstensi nol ax ke eax`
 - `movsx eax, ax ; ekstensi tanda ax ke eax`
- Membesarkan dari register 8-bit ke 32-bit:
 - `movzx eax, al ; ekstensi nol al ke eax`
 - `movsx eax, al ; ekstensi tanda al ke eax`
- Membesarkan dari register 8-bit ke 16-bit:
 - `movzx ax, al ; ekstensi nol al ke ax`
 - `movzx bx, al ; ekstensi nol ax ke ebx`
- Membesarkan nilai dari memori harus menggunakan size specifier:
 - `movzx ebx, [L] ; error "size not specified"`
 - `movzx ebx, byte [L] ; ekstensi nol nilai 1 byte pada; alamat L ke ebx`



Menggunakan MOZX dan MOSX

- Kita harus konsisten mengartikan data yang kita simpan:

```
mov    al, 0A7h    ; saya mengartikan ini sebagai nilai tidak  
                        ; bertanda, maka ini adalah decimal 167  
mov    bl, 0A7h    ; saya mengartikan ini sebagai nilai bertanda,  
                        ; maka ini adalah decimal -89
```

; gunakan instruksi yang sesuai dengan pengartian nilai kita

```
movzx   eax, al    ; ekstensi tidak bertanda ke nilai 4-byte  
movsx   ebx, bl    ; ekstensi bertanda ke nilai 4-byte
```



Latihan

- Misalkan kode berikut:

```
mov     al, 0B2h
movsx   eax, al
mov     bx, eax
movzx   ebx, bx
```

- Berapa nilai dalam eax setelah semua instruksi dieksekusi?
- Berapa nilai dalam ebx setelah semua instruksi dieksekusi?