

Organisasi Sistem Komputer

Bab 4. Operasi Aritmatika dan Perbandingan

4.1 Instruksi Penjumlahan

4.2 Instruksi Perkalian

4.3 Instruksi Pembagian



Pembahasan:

- Instruksi DIV dan IDIV



Pembagian

- Terdapat dua instruksi untuk pembagian:
 - Pembagian bilangan tidak bertanda: `DIV`
 - Pembagian bilangan bertanda: `IDIV`



Instruksi DIV dan IDIV

- Sintaks:

DIV *src*

IDIV *src*

Terbagi	<i>src</i> (Pembagi)	Hasil	Sisa
AX	reg/mem8	AL	AH
DX:AX	reg/mem16	AX	DX
EDX:EAX	reg/mem32	EAX	EDX

membagi AX dengan pembagi register 8-bit atau memori 8 bit dan hasil bagi disimpan dalam AL dan sisa bagi disimpan dalam AH

membagi DX:AX dengan pembagi register 16 bit atau memori 16 bit dan hasil bagi disimpan dalam AX dan sisa bagi disimpan dalam DX

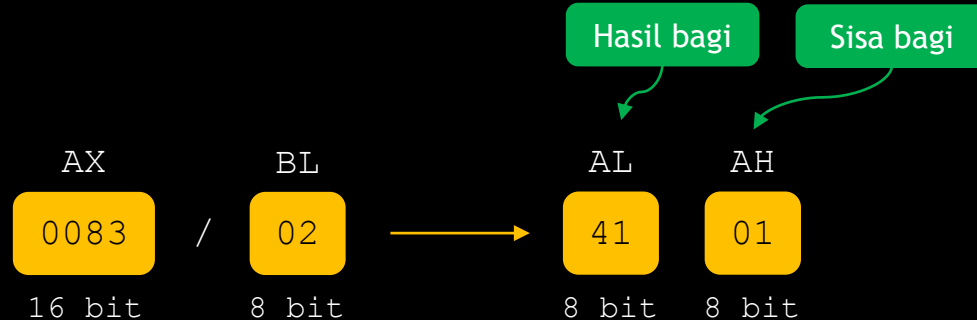
membagi EDX:EAX dengan pembagi register 32 bit atau memori 32 bit dan hasil bagi disimpan dalam EAX dan sisa disimpan dalam EDX

Pembagian Tidak Bertanda (DIV)

- Contoh #1: Pembagian tidak bertanda 8-bit

```
mov    ax, 0083h    ; terbagi
mov    bl, 2         ; pembagi
div     bl            ; al = 41h, ah = 01h
```

Terbagi	src (Pembagi)	Hasil	Sisa
AX	reg/mem8	AL	AH
DX:AX	reg/mem16	AX	DX
EDX:EAX	reg/mem32	EAX	EDX

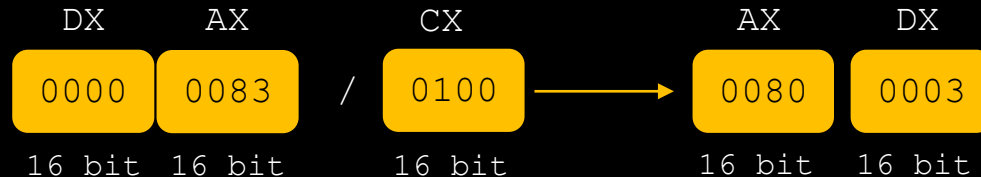


Pembagian Tidak Bertanda (DIV)

- Contoh #2: Pembagian tidak bertanda 16-bit

```
mov    dx, 0          ; bersihkan register dx
mov    ax, 8003h       ; terbagi
mov    cx, 100h        ; pembagi 16 bit
div    cx              ; AX = 0080h, DX = 0003h
```

Terbagi	src (Pembagi)	Hasil	Sisa
AX	reg/mem8	AL	AH
DX:AX	reg/mem16	AX	DX
EDX:EAX	reg/mem32	EAX	EDX



Pembagian Tidak Bertanda (DIV)

- Contoh #3: Pembagian tidak bertanda 32-bit

```
segment .data
    dividend    DQ      0000000800300020h
    divisor     DW      00000100h
segment .text
    mov     edx, DWORD [dividend + 4]
    mov     eax, DWORD [dividend]
    div     [divisor]
```



Pembagian Bertanda (IDIV)

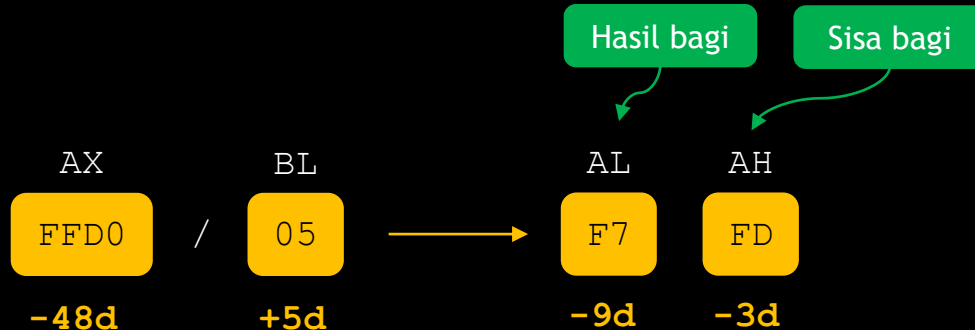
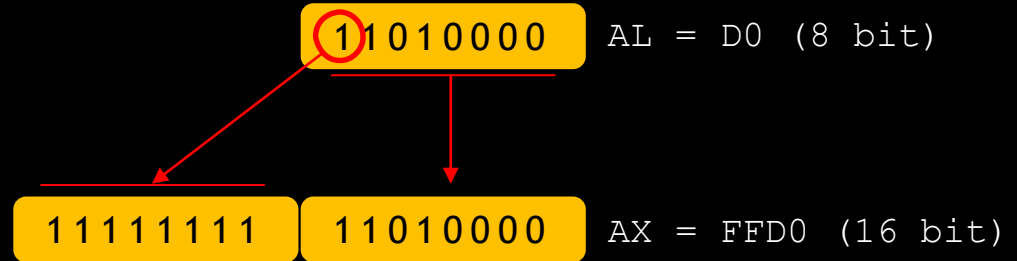
- Pembagian bertanda hampir sama dengan pembagian tidak bertanda, dengan satu perbedaan penting: terbagi (dividend) harus di-ekstensi dengan bit tanda sebelum dibagi
- Instruksi untuk mengekstensi tanda:
 - CBW – *convert byte to word*, mengekstensi bit tanda dari AL ke AH
 - CWD – *convert word to double word*, mengekstensi bit tanda dari AX ke DX
 - CDQ – *convert double word to quad word*, mengkonversi bit tanda dari EAX ke EDX

Pembagian Bertanda (IDIV)

- Contoh #1: Pembagian bertanda 8-bit

```
segment .data
    byteVal    DB    -48
segment .text
    mov     al, [byteVal]
    cbw
    mov     bl, +5
    idiv    bl
```

Instruksi CBW



Instruksi IDIV

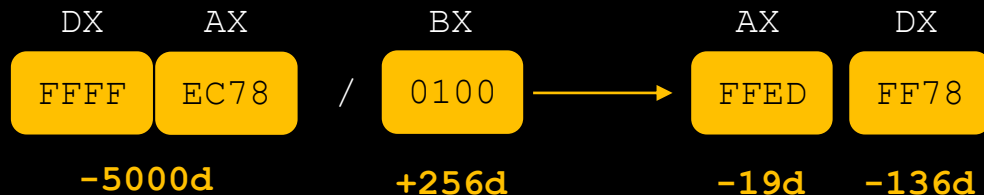
- Contoh #2: Pembagian bertanda 16-bit

```
Segment .data
```

```
wordVal    DW    -5000
```

```
Segment .text
```

```
mov     ax, [wordVal]    ; terbagi
cwd                      ; ekstensi tanda AX ke DX
mov     bx, +256          ; pembagi
idiv    bx                ; DX:AX / BX -> Hasil bagi AX = -19
                        ; sisa bagi DX = -136
```



Instruksi IDIV

- Contoh #3: Pembagian bertanda 32-bit

```
segment .data
```

```
    dwordVal    DD    +50000
```

```
segment .text
```

```
    mov     eax, dwordVal    ; terbagi
```

```
    cdq                      ; ekstensi tanda EAX ke EDX
```

```
    mov     ebx, -256        ; pembagi
```

```
    idiv    ebx              ; EDX:EAX / EBX -> Hasil bagi AEX = -195
```

```
                          ; sisa bagi EDX = +80
```



Overflow Pada Pembagian

- CPU tidak mengubah flag status setelah `DIV` dan `IDIV`
- Program berhenti jika instruksi pembagian menghasilkan hasil bagi yang tidak dapat dimuat di operand destination (kondisi overflow pembagian terjadi)
- Untuk mencegah:
 - Gunakan pembagi 32 bit dan terbagi 64 bit untuk mengurangi kemungkinan terjadi kondisi overflow
 - Test pembagi terhadap 0, untuk mencegah pembagian dengan 0