

# Chap 8

- Ax 16 bit  
AH 0000 0000 iken

32 bit EAX kaydı

Ax = alt 16 bit

EAX = tam 32 bit kayıt

üst kısım register olarak yok.

- 16 bitler pointer olarak kullanılamaz ama EAX, ECX, EDX pointer olabilir.

`mov AX, [ECX]` : ECX registerındaki adresi kullanarak

32 bitlerin pointer olarak kullanılması esnek bellek yönetimini sağlar.

- 386 386 ve üzeri işlemcilerde çalışır 8086/286 da çalışmaz.  
32 bit destekler (EAX, ECX...)

- 86 32 bit desteklemez varsayılan olarak tüm x86 işlemcilerde çalışır.

- `MOV RESULT, EAX`

Little Endiana göre saklanır

`RESULT`: D0-D7 (LSB)

`RESULT+1`: D8-D15

`RESULT+2`: D16-D23

`RESULT+3`: D24-D31 (MSB)

- 2 tane 8-byte sayıyı toplamak

`MOV SI, OFFSET DATA1` ; pointer operand 1

`MOV DI, OFFSET DATA2` ; pointer operand 2

`MOV BX, OFFSET DATA3` ; pointer SUM

`MOV CX, 2` ; 2 iterasyonda çözülür

1 byte 8 bit

AL, 15H

1 byte

EAX = 4 byte

BACK: `MOV EAX, DWORD PTR[SI]`

`ADC EAX, DWORD PTR[DI]`

`MOV DWORD PTR[BX], EAX`

⋮

`LOOP BACK`

- C/C++ evrensel ve taşınabilir

Assembly daha hızlı ama taşınabilir değil  
farklı bir CPU için farklı bir kod yazmak gerekir.

C/C++'ın taşınabilirliği + Assembly hızı

- 

Assembly kodu C programına yerleştirilebilir.  
"asm" kelimesi ile

Microsoft "\_asm" kullanır.

Inline assembly düşük seviyede daha hızlıdır.