

# **BLM312 Mikroişlemciler**

## **Addressing Modes - Supporting Material**

## High level code:

```
int i = 20;
```

## Assembly data types:

```
X DB 12H ; define byte = 8 bits (0001 0010)
```

```
Y DW 1212H ; define word = 16 bits (0001 0010 0001 0010)
```

```
K DD 12121212H ; define double word = 32 bits
```

```
Z DQ 1212121212121212H ; define quad word = 64 bits
```

# Data Addressing Modes

1. Register
2. Immediate
3. Direct Data Addressing
4. Register Indirect Addressing
5. Base-Plus-Index Addressing
6. Register Relative Addressing
7. Base Relative-Plus-Index Addressing

# Register Addressing Mode

Her iki operandın da (kaynak ve hedef) register olduğu anlamına gelir.

Örnek: **MOV AX,CX** ;*copy CX into AX*

## Dikkat edilecek hususlar:

- Tüm operandlar aynı boyutta olmalıdır.
- CS registerine yazma yapılamaz (ancak okunabilir)
- Bir segment registerinden başka segment registerine kopyalama yapılamaz

# Register Addressing Mode

## Examples:

- **Mov AX, CL** ; *Illegal instruction (registers' mixed sizes, AX→16 bits, CL → 8 bits)*
- **Mov DS, AX** ; *Legal (registers' same sizes)*
- **Mov CS, BX** ; *illegal (CS register can not a destination)*
- **Mov BX, CS** ; *legal*
- **Mov DS, SS** ; *illegal (segment-to-segment)*

# Immediate Addressing Mode

Kaynağın (source) sabit bir veri olduğu anlamına gelir.

## Dikkat edilecek hususlar:

- Immediate veriler hedef operand olamaz
- Onaltılık (hexal) veriler bir harfle başlıyorsa, derleyici verilerin 0 ile başlamasını ister.

## Examples:

- MOV CL, 05H ; *CL=05H (05 is the immediate or constant data)*
- **MOV 04H, CH** ; *illegal ( 04H is a data not a storage)*
- MOV CX, 07H ; *Legal (CX = 0007H)*

# Immediate Addressing Mode

## Examples:

- MOV CX, 12ABH ; *Legal (CX = 12ABH)*
- MOV CX, 12ABCDH ; *Legal (CX=ABCDH)*
- MOV DH, 20 ; *Legal ( 20 is a decimal data)*
- MOV DH, 10001111B ; *Legal (B refers to binary) DH=10001111*
- **MOV AH,CDH** ; *Illegal (If hexadecimal data begin with a letter, the assembler requires the data starts with a 0)*
- MOV AH, 0CDH ; *Legal*
- **MOV CL, AH** ; *Do I mean AH is immediate data or register AH?*
  - MOV CL, 0AH ; *Legal (refers to immediate data addressing)*
  - MOV CL, AH ; *Legal (refers to register addressing)*

# Direct Data Addressing

Doğrudan adresleme, bir bellek konumu ile bir register arasında bir baytı veya kelimeyi taşır.

Example#1: **MOV [1234H], AX**

Notes:

- Byte → 8 bits
- Word → 16 bits
- Double words → 32 bits
- Quadword → 64 bits

Notes:

- 1. Brackets [ ] indicates memory
- 2. Destination is a memory location
- 3. Source AX
- 4. Instruction meaning: copy the contents of AX (word, 16 bits) into the memory locations.
- 5. 1234H is offset data
- 6. The corresponding segment register is DS.
- 7. Logical addresses of these memory locations are DS: 1234H, DS:1235H



# Register Indirect Addressing

Not: ofset, data değildir register'dir. Bu, registerin ofset adresini içerdiği anlamına gelir.

Example: **MOV [BX], CH** ; where **DS =1000, SS =5000H, BX =332AH**

**Kaç tane mantıksal ve fiziksel (*logical and physical*) adres gerekli?**

- Kaynak operand (CH) 8-bitlik olduğu için bir byte'lık veri transferi yapılacağından bir tane mantıksal ve bir tane fiziksel adres gerekir.

**Oluşturulan mantıksal ve fiziksel adresler nelerdir?**

- BX ofset adresini içerir. Artık BX, DS registeri ile birleştirilmiştir (bkz. Tablo 2-3)

TABLE 2-3 Default 16-bit segment and offset combinations.

Segment	Offset	Special Purpose
CS	IP	Instruction address
SS	SP or BP	Stack address
DS	BX, DI, SI, an 8- or 16-bit number	Data address
ES	DI for string instructions	String destination address

# Register Indirect Addressing

- Yani mantıksal (*logical*) adres: 1000H: 332AH  
Fiziksel (*physical*) adres:  $1000 \times 10H + 332AH = 1332AH$

MOV [BX], CH ; CH'yi BX ofsetli ile bir bellek konumu kopyalamak anlamına gelir. Parantez bellek demektir. Dolaylı adresleme modunu (***Register Indirect Addressing***) ifade eder.

# Register Indirect Addressing

Örnek: Aşağıdaki data segment bellek bölümünü göz önünde bulundurun:

Memory address (logical)	Memory address (physical)	Memory contents (Byte-based)
2000:3222H	23222H	ABH
<b>2000:3223H</b>	<b>23223H</b>	<b>12H</b>
<b>2000:3224H</b>	<b>23224H</b>	<b>BCH</b>
<b>2000:3225H</b>	<b>23225H</b>	<b>B5H</b>
<b>2000:3226H</b>	<b>23226H</b>	<b>ACH</b>
2000:3227H	23227H	22H
2000:3228H	23228H	44H

# Register Indirect Addressing

BX = 3223H verildiğinde, aşağıdaki komutun içerikleriyle birlikte etkilenen bölümleri ne olacaktır:

**MOV EAX, [BX]**

- Komut, bellekten EAX registerine kopyalama anlamına gelir. Etkilenen kısım (hedef), içeriği ACB5BC12H olan EAX registeridir.

Kaç tane bellek hücresi gereklidir?

- 4 (EAX 32-bit (4 bayt) olduğundan)

# Register Indirect Addressing

Gerekli mantıksal ve fiziksel adresler nelerdir?

- İlk ofset adresi BX registerinde saklanır.
- Mantıksal adresler:
  - 2000: 3223H, 2000: 3224H, 2000: 3225H ve 2000: 3226H
- Fiziksel adresler:
  - 23223H, 23224H, 23225H ve 23226H

Yani, little endian politikasına göre, **EAX = ACB5BC12H**.

Memory address (logical)	Memory address (physical)	Memory contents (Byte-based)
2000:3222H	23222H	ABH
2000:3223H	23223H	12H
2000:3224H	23224H	BCH
2000:3225H	23225H	B5H
2000:3226H	23226H	ACH
2000:3227H	23227H	22H
2000:3228H	23228H	44H

# Register Indirect Addressing

**Çok önemli not:** `MOV [DI], 10H` ; 10H'yi DI ile ofsetli bellek konumuna kopyalayın. Belirsiz (ambiguous) olduğu için bu geçersiz komuttur.

10H (bayt, 8 bit), 0010H (word, 16 bit), 00000010H (double words, 32 bit), 000000000000000010H (quadword, 64 bit). Bu sayıların tümü 10H anlamına gelir (baştaki sıfırlar göz ardı edilir)

Çözüm: bazı direktifler gerekli:

Byte ptr (10H), word ptr (0010H), dword ptr (00000010H), qword ptr (000000000000000010H).

`MOV byte ptr [DI],10H ; 10H`

`MOV word ptr [DI],10H ; 0010H`

`MOV dword ptr [DI],10H ; 00000010H`

`MOV qword ptr [DI],10H ; 000000000000000010H`

# Base-Plus-Index Addressing

Base registers = BX (base register) and BP (base pointer register)

Index registers = SI (source index) and DI (destination index)

Taban (base) ve indeks registerleri tarafından belirlenen bir bellek operandına sahip olduğumuz anlamına gelir.

**Example: MOV [BX+SI], AX ; where AX = 223AH, BX =2233H, SI=0003H, DS = 2000H, SS =4000H**

TABLE 2-3 Default 16-bit segment and offset combinations.

Segment	Offset	Special Purpose
CS	IP	Instruction address
SS	SP or BP	Stack address
DS	BX, DI, SI, an 8- or 16-bit number	Data address
ES	DI for string instructions	String destination address

İçerikleriyle birlikte etkilenen kısım nelerdir?

Etkilenen kısım → destination → memory → offset registers BX and SI → data segment

# Base-Plus-Index Addressing

Example. MOV [BX+SI], AX ; where AX = 223AH, BX =2233H, SI=0003H, DS = 2000H, SS =4000H

Etkilenen mantıksal adresler: 2000: (2233+0003) → 2000:2236H, 2000: 2237H

Etkilenen fiziksel adresler : DSx10H + BX + SI → 22236H, 22237H

Memory address (logical)	Memory address (physical)	Memory contents (Byte-based)
2000:2235H	22235H	ABH
<b>2000: 2236H</b>	22236H	<b>3AH</b>
<b>2000: 2237H</b>	22237H	<b>22H</b>
2000: 2238H	22238H	B5H
2000: 2239H	22239H	ACH
2000: 223AH	2223AH	22H
2000: 223BH	2223BH	44H



# Base-Plus-Index Addressing

Dikkat: **MOV [BX + BP], CL** ; DS=1000H, SS=2000H, BX=200AH, BP=0003H

Çözüm: Bu tamamen geçersiz komut - iki tane base register olamaz

Example: **MOV [BP + DI], BH** ; DS=2000H, SS=1000H, BP=0002H, DI=0008H, BX = 3ABCH (BH=3AH, BL=BCH).

Etkilenen kısım → destination → memory → offset registers BP and DI

Etkilenen mantıksal adres: 1000: (0002+0008) → 1000:000AH

Etkilenen fiziksel adres : SSx10H + BP + DI → 1000AH

TABLE 2-3 Default 16-bit segment and offset combinations.

Segment	Offset	Special Purpose
CS	IP	Instruction address
SS	SP or BP	Stack address
DS	BX, DI, SI, an 8- or 16-bit number	Data address
ES	DI for string instructions	String destination address

# Base-Plus-Index Addressing

Example: `MOV [BP + DI], BH` ; **DS=2000H, SS=1000H, BP=0002H, DI=0008H, BX = 3ABCH (BH=3AH, BL=BCH).**

Memory address (logical)	Memory address (physical)	Memory contents (Byte-based)
2000:000AH	2000AH	ABH
2000: 000BH	2000BH	3AH
2000: 000CH	2000CH	22H
2000: 000DH	2000DH	B5H
.	.	.
.	.	.
.	.	.
<b>1000:000AH</b>	<b>1000AH</b>	<b>3AH</b>
1000: 000BH	1000AH	22H
1000: 000CH	1000AH	44H
1000: 000DH	1000AH	22H

# Register Relative Addressing

Anlamı: bir register (***base*** veya ***index***) ve bir göreceli (***displacement*** veya ***constant***) veriyle ofsetlenen bir bellek erişimi vardır.

Example: MOV [SI + 0020H] , ECX ; where DS = 5000H, SS= 6000H, SI= 0006H, ECX= 226677AAH.

Etkilenen kısım destination → memory → offset register SI and relative data (0020H, 16 bits). Aşağıdaki tabloya göre seçilen segment DS'dir.

TABLE 2-3 Default 16-bit segment and offset combinations.

Segment	Offset	Special Purpose
CS	IP	Instruction address
SS	SP or BP	Stack address
DS	BX, DI, SI, an 8- or 16-bit number	Data address
ES	DI for string instructions	String destination address

# Register Relative Addressing

Example: MOV [SI + 0020H] , ECX ; where DS = 5000H, SS= 6000H, SI= 0006H, ECX= 226677AAH.

Etkilenen mantıksal adres: 5000: (0020+0006) → 5000:0026H, 5000:0027H, 5000:0028H, 5000:0029H

Etkilenen fiziksel adres : DSx10H + SI + 0020 → 50026H, 50027H, 50028H, 50028H

Memory address (logical)	Memory address (physical)	Memory contents (Byte-based)
2000:000AH	2000AH	ABH
2000: 000BH	2000BH	3AH
2000: 000CH	2000CH	22H
2000: 000DH	2000DH	B5H
.	.	.
.	.	.
.	.	.
5000:0026H	50026H	AAH
5000:0027H	50027H	77H
5000:0028H	50028H	66H
5000:0029H	50029H	22H

(ECX= 226677AAH)

# Base Relative-Plus-Index Addressing

Anlamı: Bir base register (BP veya BX), bir indeks registeri (SI veya DI) ve ayrıca göreceli (***displacement*** veya ***constant***) veriye sahip olmamız gerekir.

Ex. MOV DX, [BP+SI+ 0030H] ; where SS=1000H, DS=4000H, BP=0002H, SI=6.

Etkilenen kısım içeriği ile birlikte nedir?

Memory address (logical)	Memory address (physical)	Memory contents (Byte-based)
1000:0037H	10037H	ABH
1000: 0038H	10038H	3AH
1000: 0039H	10039H	22H
1000: 003AH	1003AH	B5H
.	.	.
.	.	.
.	.	.
4000:0037H	40037H	AAH
4000: 0038H	40038H	77H
4000: 0039H	40039H	66H
4000: 003AH	4003AH	22H

Analiz: Etkilenen kısım (hedef) bir register'dir (DX, 16 bit). Kaynak, bir bellektir. Kaç tane bellek konumuna erişmemiz gerekiyor? 2 konum (çünkü her bellek konumu bir bayt uzunluğunda ve hedef iki bayt uzunluğunda (16 bit)).

# Base Relative-Plus-Index Addressing

Şimdi, oluşturulan mantıksal adresler nelerdir?

TABLE 2-3 Default  
16-bit segment and  
offset combinations.

Segment	Offset	Special Purpose
CS	IP	Instruction address
SS	SP or BP	Stack address
DS	BX, DI, SI, an 8- or 16-bit number	Data address
ES	DI for string instructions	String destination address

Yukarıdaki tabloya göre, BP → yığın segmenti (SS) ve SI → veri segmenti (DS).  
Hangisini seçmeliyim? Cevap → yığın segmenti (SS)

**Mantıksal adresler: 1000H: 0038H, 1000H:0039H**

**Fiziksel adresler: 10038H and 10039H**

**Sorunun cevabı: DX= 223AH**