

MİKROİŞLEMCİLER (BLM202)

HAFTA - 6

Dr. Bilgin YAZLIK, RTTP, PMP

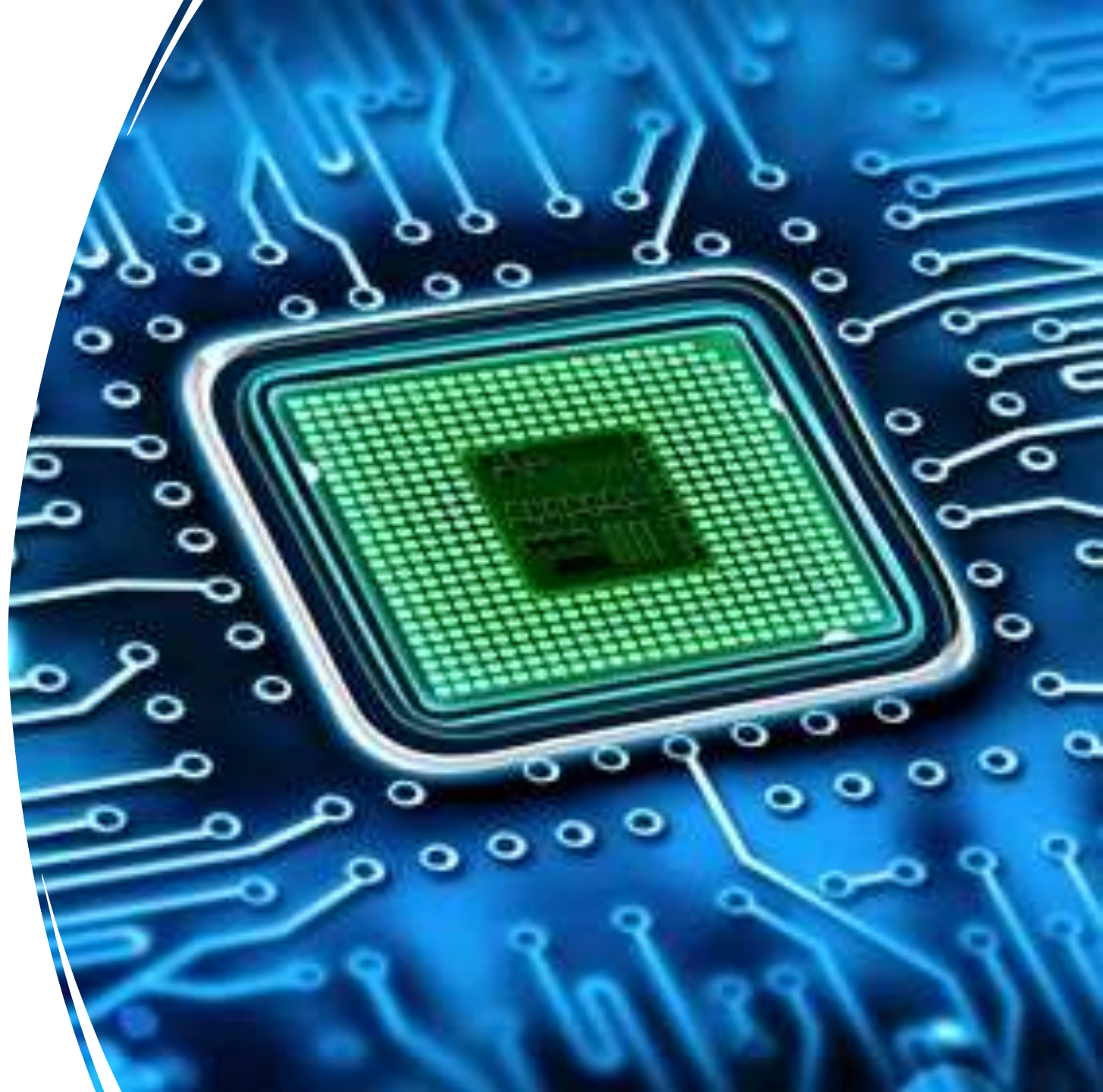


BİLGİSAYAR MÜHENDİSLİĞİ



6. HAFTA

- Veri Aktarım Komutları



VERİ AKTARIM KOMUTLARI / DERS SÜRESİNCE KULLANILACAK KISALTMALAR

acc	Akümülatör (EAX/AX/AL yazmaçlarından herhangi biri)
reg	8/16/32 bitlik herhangi bir yazmaç
regb	8 bitlik bir yazmaç
regw	16 bitlik bir yazmaç
reg32	32 bitlik bir yazmaç
sreg	herhangi bir kesim yazmacı
mem	herhangi bir bellek adresi
idata	8/16/32 bitlik herhangi bir değer (3,0Ah,012EFh,'A' gibi)
[..]	İşaret grubu yazmaç veya görelî konum değerini gösteren sayı ile erişilebilen bellek içeriği
disp8/disp16	8 bit(-128...0...127)/16 bit(-32768...0...32767)ile ifade edilebilecek büyüklükteki bir sayı kadar ifade
dest/src	Variş işleneni /Kaynak işleneni
opr/opr #	İşlenen/işlenen 1/işlenen 2/işlenen 3
italik yazı	değişken isimleri
0/1/x/?/--	İşlem sonucunda bayrak değeri clear(0),set(1), işleme göre değişmemiş (x), belirsiz (?), değişmemiş olabilir(-)
16p	16 bit korumalı kip

MOV (mov data)

- MOV reg,idata
- MOV mem,idata
- MOV reg,reg
- MOV mem,reg
- MOV sreg,reg
- MOV reg,sreg
- MOV mem,sreg

- İki işlenen ile kullanılan MOV komutu ikinci işlenendeki veriyi ilk işlenene aktarır. MOV komutunun bazı kısıtlamaları vardır. MOV komutunun her iki işleneni mem ve sreg olamayacağı gibi ilk işlenen sreg ise ikincisi idata olamaz.

- MOV dest,src
- Şeklinde ifade edilen komut src'nin değerini dest'e yerleştirir. Bayraklar üzerinde herhangi bir değişiklik yapmaz. MOV komutunda her iki işlenenin de aynı tipte olması gerekir. Aksi halde derleyici tip uyuşmazlığı hatası verir.

MOV ÖRNEK



MOV Mydata,AX ; Mydata isimli bellek alanına AX yazmacında bulunan değer yerleştirilmektedir. Mydata isimli bellek alanı yazmaç ile aynı tipte tanımlanmış olmalıdır.

Mydata dw 8;

Mov ax, mydata;

Mov mydata,ax;



MOV AL,Result; Result isimli bellek alanından, AL yazmacına veri aktarılmaktadır. Result isimli bellek alanının byte olarak tanımlanması gerekmektedir.



MOV Mydata,12 ; Mydata isimli bellek alanına 12 değeri yerleştirilmektedir. Kaynak işlenen olan 12 sayısı hem byte hem word bir büyük olarak ifade edilebileceği dikkate alındığında Mydata isimli değişkenin byte veya word olarak tanımlanmış olmasının işleyişe bir etkisi olmayacak, her iki durumda da atama işlemi yapılacaktır.

Mydata dw 8;

MOV Mydata, 12;

MOV ÖRNEK

- **MOV WORD PTR[1001B],120H;**
- Bellek alanına adres ile erişilmesi durumunda tip uyumsuzluğu söz konusu olabilir. Programcı bellek üzerinde işlemleri hangi boyutta (byte/word) yapmak istediğine bağlı olarak gerekli tip tanımlamaları yapmalıdır. Bunu için doğru tipte adresleme modu kullanılmalıdır.

- **MOV WORD PTR [adres], kaydedilecek veri;**

- 16 bit veri olarak kaydeder.

MOV [BX], 12 ;yanlış

MOV byte ptr [BX], 12 ;doğru

BYTE 8 BİT, WORD 16 BİT

12 SAYISI BYTE OLARAK: 0000 1100

12 SAYISI WORD OLARAK: 0000 0000 0000 1100

Örnek Program

- #make_COM#
- ; COM file is loaded at CS:0100h
- ORG 100h
- X DW 35;
- mov ax, 15 ; reg, idata
- MOV [1001H], 5 ; mem, idata
- MOV bx, ax ; reg, reg
- MOV cx, [1001H] ; reg, mem
- MOV [1002H], bx ; mem, reg
- MOV SI, ax ; sreg, reg
- MOV SI, [1002H] ; sreg, mem
- MOV cx, SI ; reg, sreg
- MOV [1003h], SI ; mem, reg
- HLT

MOVSX (mov with sign extension)

8086 EMULATÖRDE ÇALIŞMAZ

- MOVSX reg, reg
- MOVSX reg, mem
- Bu komut 8 bitlik değeri işareti ile birlikte 16 veya 32 bitlik alana, 16 bitlik değeri işareti ile birlikte 32 bitlik alana yerleştirir. İşaret, en anlamlı bitin yüksek anlamlı byte veya word ile tekrar edilmesi ile aktarılmaktadır. Genel olarak ilk işlenen ikinci işlenenden bir üst boyutta olmalıdır. Böylece ikinci işlenende bulunan değer daha üst büyüklüğe işaretli sayı olarak aktarılmış olur.

ÖRNEK

- **MOVSX EAX,AL** ; AL yazmacındaki 8 bitlik değeri 32 bite uzatarak EAX yazmacına aktarır. Bu işlem sonucunda AL yazmacının işaret biti (7.bit), 8-31. bitler boyunca tekrarlanır. AL=67H ise bu işlemden sonra EAX=00000067H olacaktır.
- **MOVSX EDI,WORD PTR[ESI]** ; ESI yazmacının belirlediği bellek gözündeki 16 bit değeri 32 bite uzatarak EDI yazmacına aktarır. 15. bit ile ifade edilen işaret 16-31. bitlerde tekrarlanacaktır.
- **MOVSX CX,DL**; DL yazmacındaki 8 bitlik değeri 16 bite uzatarak CX yazmacına yerleştirir. DL=80H ise sonuçta CX=0FF80H değerini alacaktır.

MOVZX (Move with zero extension)

8086 EMULATÖRDE ÇALIŞMAZ

- MOVZX reg,reg
- MOVZX reg,mem
- Bu komut 8 bitlik değeri 16 veya 32 bitlik alana 16 bitlik değeri 32 bitlik alana yerleştirir. İşaret uzantısı ise en anlamlı bitlerin 0 ile doldurulması ile elde edilir. Genel olarak ilk işlenen ikinci işlenenden bir üst boyutta olmak zorundadır. Böylece ikinci işlenende bulunan değer üst büyüklüğe işaretsiz sayı olarak aktarılmış olur.

ÖRNEK

- **MOVZX EAX,AL;** AL yazmacındaki 8 bitlik değeri 32 bitlik değere uzatarak EAX yazmacına yerleştirir. 8-31. Bitler 0'dır. AL=80H ise sonuçta EAX=00000080H olacaktır.
- **MOVZX EDI,WORD PTR[ESI]** ; ESI ile belirlenen bellek gözündeki değeri 32 bite uzatarak EDI yazmacına yerleştirir. 16-31. Bitler 0'dır. WORD PTR ifadesi ile bellekten 1 VWORD alınacağı belirlenmektedir.
- **MOVZX CX,DL;** DL yazmacının 8 bitlik değeri 16 bite uzatılarak CX yazmacına yerleştirilir. 8-15. Bitler 0'dır.
- **MOVZX BX, BL;** BL yazmacındaki değeri işaret değeri 0 olacak şekilde BX yazmacına yayar. Ancak MOV BH,0 işlemi buna nazaran daha kısa sürede tamamlanacaktır.

LEA (load effective adres)

- LEA regw, mem
- İşlem sonucunda regw de oluşan değer mem ile tanımlı bellek alanının, kesiminin başından itibaren kaç byte olduğunu (görelî konum) belirlemektedir. Elde edilecek adres 16 bit uzunluğunda olacaktır. Bu nedenle işlem sonucunun oluşacağı kısım mutlaka word tipinde bir yazmaç olmalıdır. Aynı işlem OFFSET komutu ile de gerçekleştirilebilmektedir.

ÖRNEK

- **LEA SI,Mydata** ; Mydata isimli bellek alanının, tanımlı olduğu offset adres değeri SI yazmacında oluşacaktır. Yandaki bellek haritasında verilen değerlere göre SI=1002H olacaktır.
- Değişkenin offset (bağıl konum) adresini elde etmek için kullanılır.
- Örneğin büyük bir veriyi hafızaya aktardığımızda bu verinin hafızanın neresinde konumlandığı bilgisini temin eder.

	7	0	
1004H	34H		Mydata
1003H	4FH		
1002H	0ABH		
1001H	53H		
1000H	2CH		

ÖRNEK KOD

- #make_COM#
- ; COM file is loaded at CS:0100h
- ORG 100h;
- X DW 35;
- LEA BX,X;

LDS (load data segment register)

- LDS regw,mem
- regw ile belirlenen yazmaç ile DS yazmacına, mem ile belirlenen bellek bölgesinde bulunan değerler yüklenir.
- Yapılan işlem: $\text{regw} = [\text{mem}]$
- Bu komut bir seferde DS:SI ikilisinin değerlerinin bellek alanından alınmasını sağlamaktadır. Genel olarak SI yazmacı ile kullanılmasının arkasında, bellek erişimlerinde DS ve SI yazmaçlarının birlikte kullanılması yatar. Benzer şekilde DI ve BX yazmaçları da bu komut ile birlikte kullanılabilir. Mem değeri doğrudan bir bellek adresi olarak verilebildiği gibi bir değişken ile de verilebilir. Bu durumda kullanılacak olan değişken 32 bit (doubleword) olacak şekilde tanımlanmış olması gereklidir.

ÖRNEK

- MOV [4326], 1520H;
- MOV [4328], 2021H;
- LDS BX, [4326];

- 4326H-BL
- 4327H-BH
- 4328H-4329H-DS

LES(load extra segment register)

- LES regw,mem
 - LDS komutu ile aynı özellikleri göstermektedir. Ancak işlem diğerinden farklı olarak ES yazmacı kullanılarak gerçekleştirilir.
 - **Yapılan işlem:**
 - Regw=[mem]
 - ES=[mem+2]
 - Bu komut bir seferde ES:DI ikilisinin değerlerinin bellek alanından alınmasını sağlamaktadır. Genel olarak DI yazmacı kullanılmasının arkasında bellek erişimlerinde ES ve DI yazmaçlarının birlikte kullanılması yatmaktadır. Benzer şekilde SI ve BX yazmaçları da bu komut ile birlikte kullanılabilir. Mem değeri doğrudan bir bellek adresi olarak verilebildiği gibi bir değişken ile de verilebilir. Bu durumda kullanılacak olan değişkenin 32 bit olacak şekilde tanımlanmış olması gerekmektedir.

ÖRNEK

- MOV [4326], 1520H;
- MOV [4328], 2021H;
- LES BX, [4326];

- 4326H-BL
- 4327H-BH
- 4328H-4329H-ES

Kaynaklar

- Feza Buzluca, İTÜ Ders Notları, Bilgisayar Mimarisi
- Wikipedia
- Emel Soylu, Kadriye Öz, Karabük Üniversitesi, Mikroişlemciler Ders Notları
- 1) [Bilgisayar Mimarisi – Doç. Dr. Şirzat KAHRAMANLI](#)
- 2) [Ders Notları – Yrd. Doç. Dr. Rifat KURBAN](#)
- Wikipedia
- <https://edukedar.com/difference-between-cisc-and-risc/>
- Dr. B. B. Hegde First Grade College, Kundapura

