

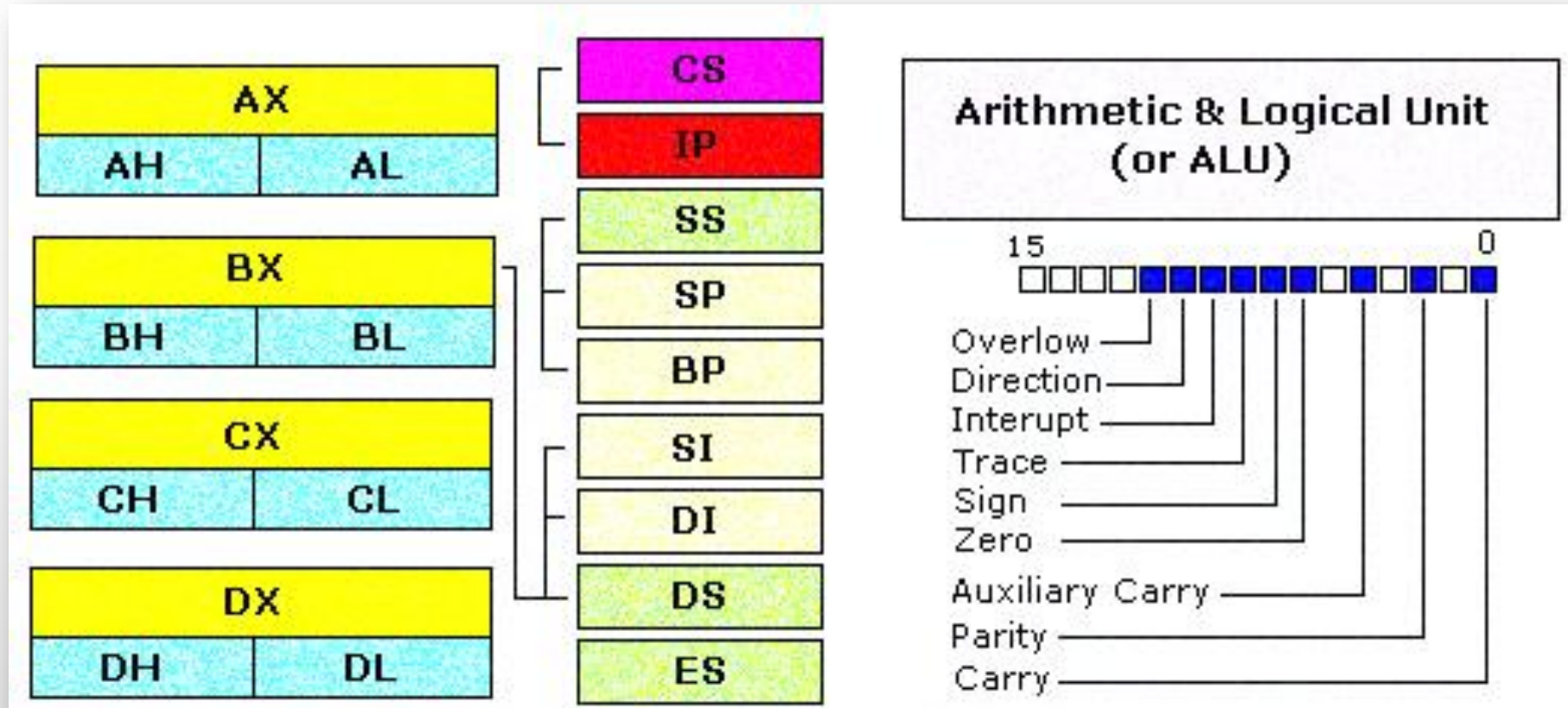


Bölüm 2: 8086 Yazmaçlar

Mikroişlemciler



8086 Yazmaçlar





Genel Amaçlı Yazmaçlar

- *General purpose registers*
- Geçici verileri saklamak için kullanılır.
- 8 adet genel amaçlı yazmaç bulunmaktadır.



AX, Accumulator

- 16 bit genişliğindedir.
 - AH ve AL olarak 8 bitlik iki bölüme ayrılmıştır.
- AH (Yüksek Sıra) ve AL (Düşük Sıra) olarak adlandırılır.
- 8-bitlik komutları gerçekleştirmek için AH ve AL ayrı ayrı kullanılabilir.
- Genellikle aritmetik ve mantıksal komutlar için kullanılır.
- Örnek: `ADD AX, AX` ($AX = AX + AX$)



BX, Base

- 16 bit genişliğindedir.
 - BH ve BL olarak 8 bitlik iki bölüme ayrılmıştır.
- BH (Yüksek Sıra) ve BL (Düşük Sıra) olarak adlandırılır.
- 8-bitlik komutları gerçekleştirmek için BH ve BL ayrı ayrı kullanılabilir.
- Dolaylı bellek adreslemede bağıl konum (offset) değerini saklar.
- Genellikle dizi ve matris işlemlerinde kullanılır.
- Örnek: MOV BL, [500] (BL = 500H)
 - Bellekteki 500H adresindeki veriyi BL yazmacına aktarır.



CX, Counter

- 16 bit genişliğindedir.
 - CH ve CL olarak 8 bitlik iki bölüme ayrılmıştır.
- CH (Yüksek Sıra) ve CL (Düşük Sıra) olarak adlandırılır.
- 8-bitlik komutları gerçekleştirmek için CH ve CL ayrı ayrı kullanılabilir.
- Döngü (*loop*) ve kaydırma (*rotation*) işlemlerinde kullanılır.
- Sayma görevlerinde kullanılır.
- Örnek: `MOV CX, 0005` ve `LOOP`
 - CX yazmacına 0005 atanır, `LOOP` komutu ile işlem tekrarlanır.



DX, Data

- 16 bit genişliğindedir.
 - DH ve DL olarak 8 bitlik iki bölüme ayrılmıştır.
- DH (Yüksek Sıra) ve DL (Düşük Sıra) olarak adlandırılır.
- 8-bitlik komutları gerçekleştirmek için DH ve DL ayrı ayrı kullanılabilir.
- Çarpma komutlarında, çarpma sonucu DX ve AX kayıtlarında saklanır.
- DX, çarpma sonucunun yüksek sırasını, AX ise düşük sırasını tutar.
- DX, giriş/çıkış portlarına adresleme yaparken kullanılır.
- Örnek: MUL BX (DX, AX = AX * BX)
 - DX ve AX, AX ve BX'nin çarpım sonucunu saklar.



SP, Stack Pointer

- Yığın İşaretçisi olarak adlandırılır.
- Toplam 16 bit genişliğindedir.
- Yığının en üst ögesini işaret eder.
- Eğer yığın boşsa, yığın işaretçisi (FFFE)H değerini alır.
- Yığın, verilerin sırasıyla eklenip çıkarıldığı bir veri yapısıdır.
- Veri eklemek için *PUSH*, veri çıkarmak için *POP* işlemleri yapılır.
- SP'nin belirttiği bağıl konum (offset) değeri,
 - Yığın kesimi (*Stack Segment*) ile ilişkilidir.



BP, Base Pointer

- Taban İşaretçisi olarak adlandırılır.
- Toplam 16 bit genişliğindedir.
- Yığın üzerinden fonksiyonlara geçirilen parametrelere erişim için kullanılır.
- Yığın üzerindeki belirli bir konumu gösterir.
- BP'nin belirttiği bağıl konum (offset) değeri,
 - Yığın kesimi (*Stack Segment*) ile ilişkilidir.



SI, Source Index

- Kaynak indis olarak adlandırılır.
- Toplam 16 bit genişliğindedir.
- Veri kesiminde (*Data segment*) bağıl konumu (*offset*) temsil eder.
- Bellekteki veri okunacak belirli bir konumu işaret eder.
- MOVSB (Move String Byte),
- CMPSB (Compare String Byte) gibi işlemlerde kaynak olarak görev alır.
- Örnek: MOV SI, 2000H
 - SI'yi belirli bir bellek konumuna ayarlar.



DI, Destination Index

- Hedef indis olarak adlandırılır.
- 16 bit genişliğindedir.
- Ekstra kesimde (*Extra Segment*) bağıl konumu (offset) temsil eder.
- Bellekte veri yazılacak belirli bir konumu işaret eder.
- MOVSW (Move String Word),
- STOSB (Store String Byte) gibi işlemlerde hedef olarak görev alır.
- Örnek: MOV DI, 3000H
 - DI'yi belirli bir bellek konumuna ayarlar.



Bayrak Yazmacı

- Durum yazmacı olarak da adlandırılır.
- Mikroişlemcinin yürüttüğü son komutun sonuçlarını içerir.
- Koşullu atlama ve dallanma komutlarının davranışını belirler.



Sign Flag (S)

- İşaret bayrağı, işlem sonucunun negatif mi pozitif mi olduğunu belirtir.
- İşlem sonucunun en önemli (MSB) biti kontrol edilir;
 - 1 ise, sayı negatif kabul edilir ve işaret bayrağına 1 atanır.
 - 0 ise, sayı pozitif kabul edilir ve işaret bayrağı sıfırlanır.
- 00H ile 7FH arasındaki sayılar için işaret bayrağı sıfır (0)'dir.
 - çünkü MSB 0'dır (pozitif sayı).
- 80H ile FFH arasındaki sayılar için işaret bayrağı bir (1)'dir.
 - çünkü MSB 1'dir (negatif sayı).



Zero Flag (Z)

- Sıfır bayrağı, işlemin sonucunun 00H olup olmadığını belirtir.
 - İşlem sonucu 0 ise, sıfır bayrağına 1 atanır.
 - Değilse, sıfır bayrağı sıfırlanır.



Auxiliary Carry Flag (AC)

- Yedek taşıma bayrağı,
 - işlemin sonucunda D(3) taşıma üretiyorsa ve
 - bu taşıma D(4)'e geçiyorsa 1 atanır;
 - aksi takdirde sıfırlanır.
- Bu bayrak, BCD (*Binary Coded Decimal*) sayı sistemine (0-9) özgüdür.
- Programcı tarafından doğrudan erişilemeyen tek bayraktır.



Parity Flag (P)

- Çiftlik bayrağı, işlemin sonucunda
 - çift sayıda 1 varsa, çiftlik bayrağına 1 atanır.
 - aksi takdirde sıfırlanır.



Carry Flag (CY)

- Taşıma bayrağı, n bitlik işlemler gerçekleştirilirken,
 - sonuç n biti aşıyorsa, taşıma bayrağına 1 atanır.
 - aksi takdirde sıfırlanır.



Overflow Flag (O)

- Taşma bayrağı, işlem sonucu
 - kullanılan bit sayısına sığmayacak kadar büyük ise 1 atanır.
 - aksi takdirde sıfırlanır.



Directional Flag (D)

- Yön bayrağı, özellikle dize (string) komutlarında,
 - veri erişim yönünü belirlemek için kullanılır.
- Eğer yön bayrağı 1 ise, verilere yüksekten düşük bellek konumuna erişilir.
- 0 ise, dize verilerine düşük bellek konumundan yükseğe doğru erişilir.



Interrupt Flag (I)

- Kesme bayrağı, kesmelerle ilişkilidir.
- Eğer kesme bayrağı 1 ise,
 - çevre birimlerinden gelen kesme talepleri ele alınır.
- 0 ise, herhangi bir kesme talebi ele alınmaz.



Trap Flag (T)

- Tuzak bayrağı, yonga içi hata ayıklama için kullanılır.
- Tuzak bayrağı ayarlandığında,
 - hata ayıklama (*debugging*) için tek adım (*single step*) moduna girilir.
- Eğer tuzak bayrağı 1 ise,
 - her komuttan sonra dahili bir kesme üretilir,
 - tek adım ISR'ye (*interrupt service routine*) girilir.
 - program komut komut incelenir.
- 0 ise, herhangi bir işlem gerçekleşmez.



Adresleme Modları

- İşlenen verinin, belirtilme yöntemine *adresleme modu* denir.
- İşlenenin, bir veri (*immediate data*) veya bir adres olduğunu belirtir.
- Ayrıca, işlenenin bir yazmaç veya yazmaç çifti olup olmadığını belirtir.



Register Mode

- Yazmaç Modu:
 - Her iki işlenen (*operand*) de yazmaçtır.
 - Örnekler:
 - MOV AX, BX
 - XOR AX, DX
 - ADD AL, BL



Immediate Mode

- Dolaysız Modu:
 - Kaynak işlenen 8 veya 16 bit veridir.
 - Hedef işlenen asla dolaysız olamaz.
 - Örnekler:
 - MOV AX, 2000
 - MOV CL, 0A
 - ADD AL, 45
 - AND AX, 0000
 - MOV AX, 2000
 - MOV CS, AX (*kesim yazmacına değer atamak için yazmaç gerekir!*)



Displacement or Direct Mode

- Kaydırma veya Doğrudan Modu:
 - Etkin adres doğrudan verilir.
 - Örnekler:
 - MOV AX, [DISP]
 - MOV AX, [0500]



Register Indirect Mode

- Yazmaç Dolaylı Modu:
 - Etkin adres SI, DI veya BX içindedir.
 - Örnekler: Fiziksel Adres = Kesim Adresi + Etkin Adres
 - MOV AX, [DI]
 - ADD AL, [BX]
 - MOV AX, [SI]



Based Indexed Mode

- Taban İndisli Modu:
 - Etkin adres taban yazmaç ve indis yazmacının toplamıdır.
 - Taban yazmaç: BX, BP
 - İndis yazmaç: SI, DI
 - Fiziksel bellek adresi, taban yazmacına göre hesaplanır.
 - Örnekler:
 - `MOV AL, [BP+SI]`
 - `MOV AX, [BX+DI]`



Indexed Mode

- İndis Modu:
 - Etkin adres indis yazmacı ve kaydırma (*displacement*) toplamıdır.
 - Örnekler:
 - MOV AX, [SI+2000]
 - MOV AL, [DI+3000]



Based Mode

- Taban Modu:
 - Etkin adres, taban yazmacı ve kaydırma toplamıdır.
 - Örnekler:
 - `MOV AL, [BP+ 0100]`



Based Indexed Displacement Mode

- Taban İndisli Kaydırma Modu:
 - Etkin adres, indis yazmacı, taban yazmacı ve kaydırma toplamıdır.
 - Örnekler:
 - `MOV AL, [SI+BP+2000]`



String Mode

- Dize Modu:
 - Dize komutları ile ilgilidir.
 - SI ve DI'nin değeri,
 - yön (*directional*) bayrağına bağlı olarak otomatik artar veya azalır.
 - Örnekler:
 - MOVS B
 - MOVS W



Input/Output Mode

- Giriş/Çıkış Modu:
 - Giriş/çıkış işlemleri ile ilgilidir.
 - Örnekler:
 - IN A, 45
 - OUT A, 50



Relative Mode

- Göreceli Mod:
 - Etkin adres, komut işaretçisine göre hesaplanır.
 - Örnekler:
 - *JNZ 8 bit adres*
 - $IP = IP + 8 \text{ bit adres}$



SON