

# **BÖLÜM 5. KOMUT SETİ MİMARİSİ**

## **Temel Bilgisayar Sistemimizin Bileşenleri**

Düşük ve Yüksek Anlamlı Veri Yolunun Tahsisi

## **Komut Tasarımı**

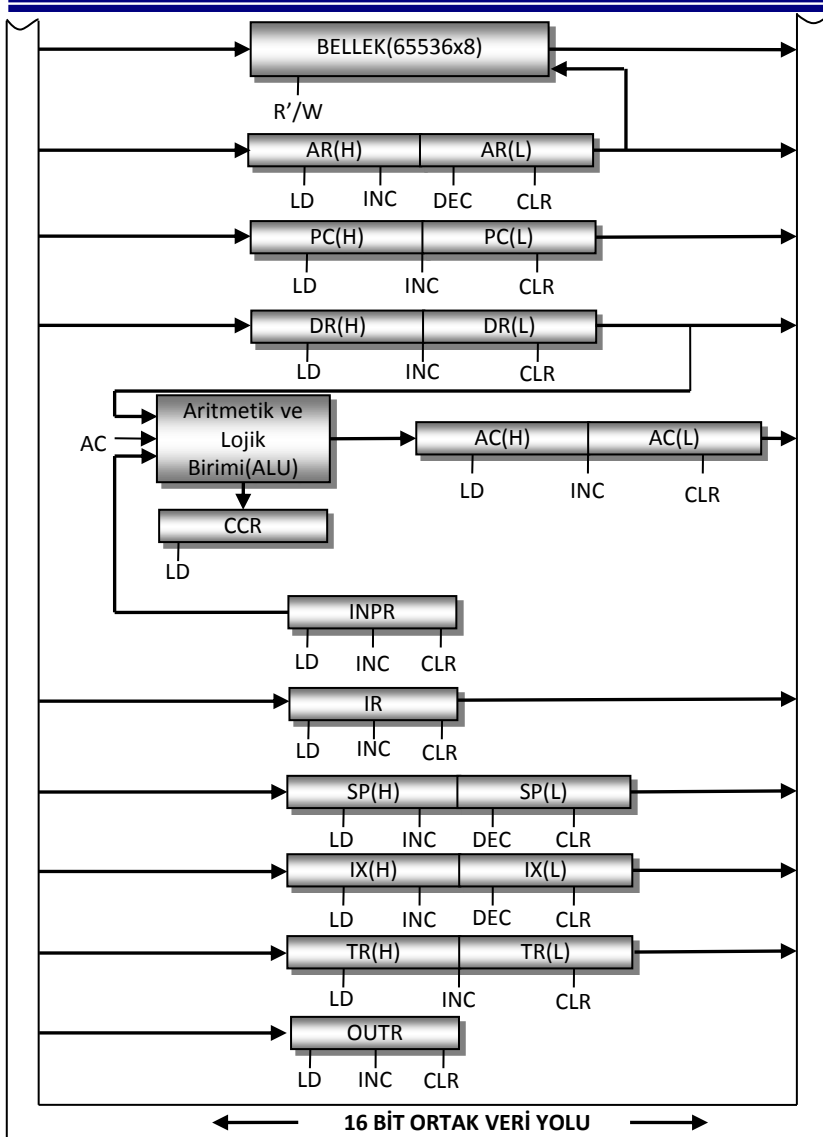
Adresleme Modu Seçim Bitleri ve OPCODE

Adresleme Modu Bilgisinin Çözümlemesi

Komutun İşlem Tipi Bilgisinin Çözümlemesi

Komutun Çözümlemesi

# Temel Bilgisayar Sistemimizin Bileşenleri



Bellek, 64 KB olarak düşünülmüştür. Belleğin her gözü de 8 bitlik veriyi tutmaktadır.

Ayrıca komut tasarımında bazı komutların 1 byte yer kaplayabileceği de düşünülerek, belleğin her bir gözünün 8 bitlik olmasına karar verilmiştir (Ayrıca komutların OPCODE kısmı da 8 bitlik olduğundan).

11 adet kaydedici ve bellek ortak veri yoluna bağlanmışlardır. Hangi elemanın hangi durumda veri yoluna bilgi aktaracağına,  $4 \times 16$ 'lık kod çözücülerin girişlerine gelen veri ile karar verilir.

ALU'da gerçekleştirilecek işlemler 16 bitlik olacağından ve bellek de 8 bitlik olduğundan, veri yolu 2 kısım olarak düzenlenmiştir; veri yolunun düşük ve yüksek anlamlı kısmı diye. Bu sebepten ötürü 2 kod çözücü kullanılmıştır.

# Düşük ve Yüksek Anlamlı Veri Yolunun Tahsisi

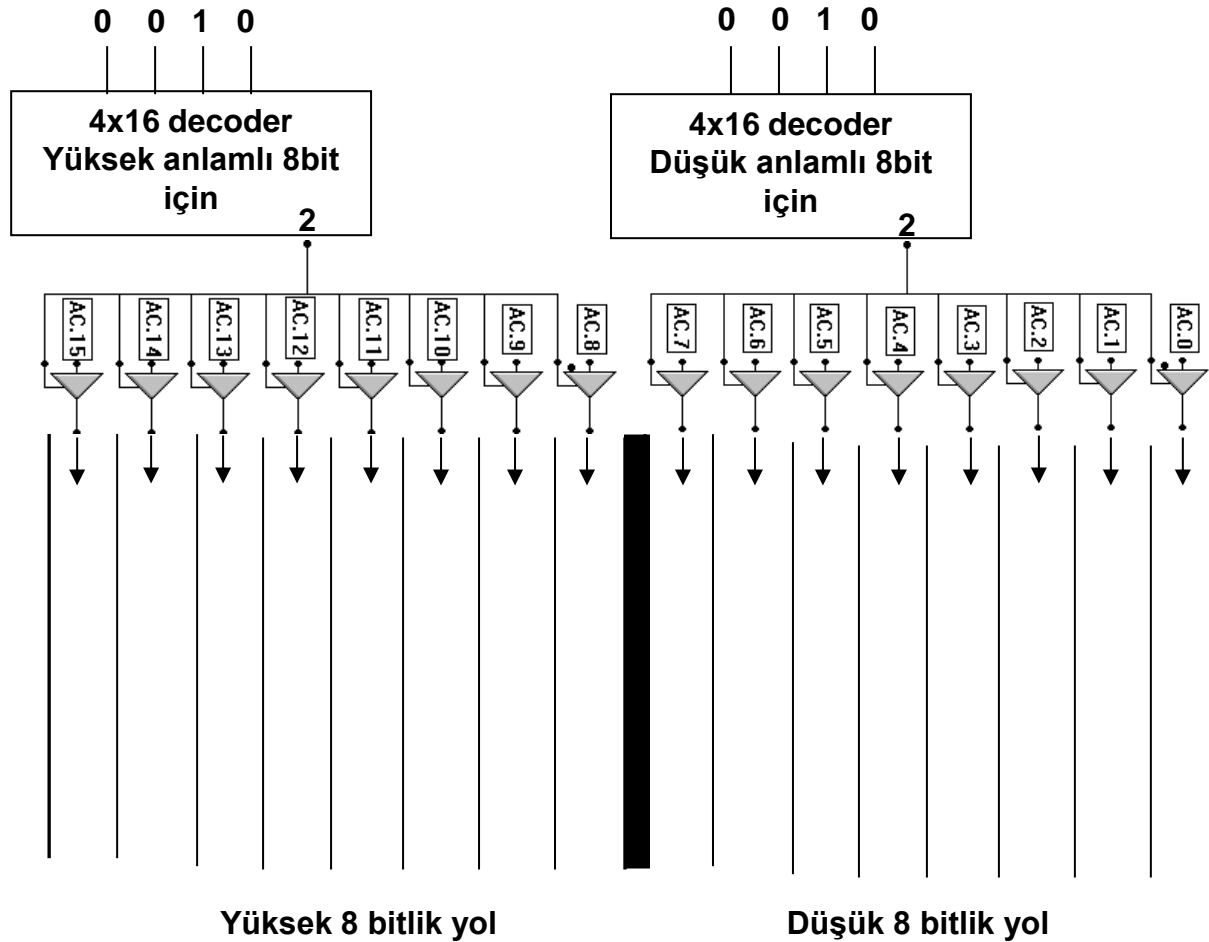
Bir kaydedici veya belleğin verisini yola aktarmak için kod çözücülerin girişlerine uygulanması gereken ikili değerler aşağıdaki tablodaki gibi seçilmiştir.

Kod Çözücü Girişi	Yolu Kullanan Birim
0001	Yığın göstericisi (SP)
0010	Akümülatör (AC)
0011	Program sayacı (PC)
0100	Komut kaydedicisi (IR)
0101	Veri kaydedicisi (DR)
0110	İndis (Index) kaydedicisi (IX)
0111	Geçici Kaydedici (TR)
1000	Adres Kaydedicisi (AR)
1001	Bellek (MEM)
1010	İndis ve Göreceli Adreslemede kullanılır
1100	Durum Kod Kaydedicisi (CCR)

*Durum kod kaydedicisi ve komut kaydedicisi dışındaki kaydediciler 16 bitlidir. 16 bitlik kaydediciler için ikinci bir kod çözücü daha kullanılmıştır (Yüksek anlamlı veri yolu).*

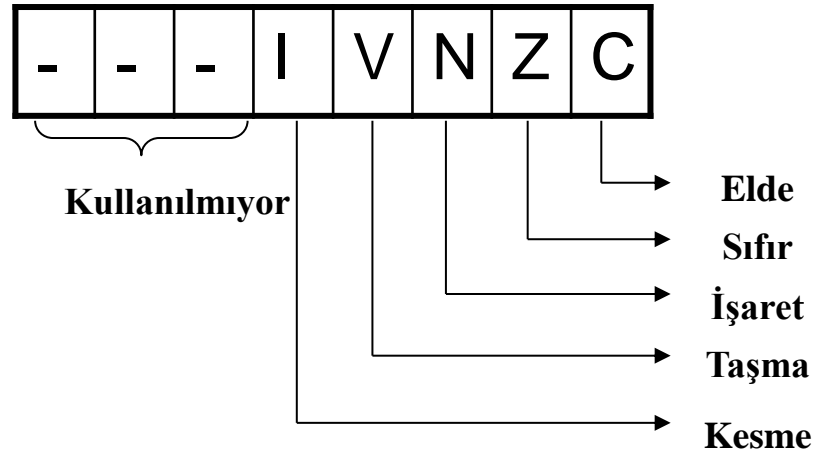
Düşük anlamlı yol tahsisini yapan kod çözücü giriş bitleri ile yüksek anlamlı yol tahsisini yapan kod çözücünün giriş bitleri aynıdır.

# Örnek Gerçekleştirim: Akünün İçeriğinin Yola Aktarılması



## Durum Kod Kaydedicisi (CCR)

ALU ile birlikte çalışır. Bayrak kaydedicisi olarak da anılır. Bayrak kaydedicisi, bütün mikroişlemcilerde olduğu gibi, bir işlemin sonucuyla ilgili bazı bilgileri içerir ve bu bilgiler daha sonra işletilecek komutlar üzerinde etkiye sahiptirler. Özellikle karar vermeye dayalı komutların yürütülmesinde sıklıkla kullanılırlar.



# Komut Tasarımı

---

Program, CPU'nun yapması gereken işlemleri tanımlayan komutlar dizisidir. Ayrıca işlemlerin hangi sıra dahilinde yapılacağını da gösterir. Bir bilgisayar komutu, ikili (binary) bir koddur ve bir dizi mikro işlem adımını tanımlar.

Komutlar verilerle birlikte bellekte bulunurlar. Bilgisayar her komutu bellekten okur bunu komut kaydedicisine yazar. Kontrol birimi bu kodu çözer ve gereken mikro işlemleri sırasıyla icra eder.

Komut kodu bir grup bitten ibarettir. Bunlar bilgisayara belli bir işlemi yapmasını söylerler. Bir komut birkaç kısımdan meydana gelir. Komutun en önemli parçası *işlem* parçasıdır. İşlem kısmını oluşturan bitler toplama, çıkarma, çarpma, kaydırma ve tümleme gibi bazı işlemleri tanımlarlar. Bir komuttaki işlem bitlerinin sayısı, bilgisayarda kullanılacak işlem sayısına bağlıdır.  $n$  tane bit ile  $2^n$  (veya daha az) işlem tanımlanabilir.

Komut kodunda yer alan diğer önemli kısım ise adresleme modunu belirleyen kısımdır. Temel bilgisayar sistemimizde komut kodunun 5 biti işlem koduna ayrılmıştır. Beş bitlik bir işlem koduyla 32 adet işlem yerine getirilebilmesine rağmen, bazı komutların bazı adresleme modlarına gereksinimi olmamasından ötürü, kullanılmayan durumlar başka komutlara tahsis edilmiş ve yaklaşık 60 farklı işlem tanımlanabilmiştir.

# Komut Tasarımı

---

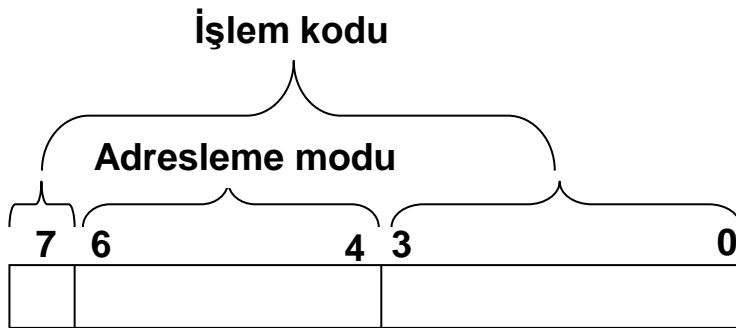
6 tip adresleme modu kullanıldığından dolayı komut kodunun 3 biti, adres alanı için tahsis edilmiştir.

Temel bilgisayar sistemimizde ivedi, direkt ve dolaylı adresleme modlarına sahip komutlar 3 byte,

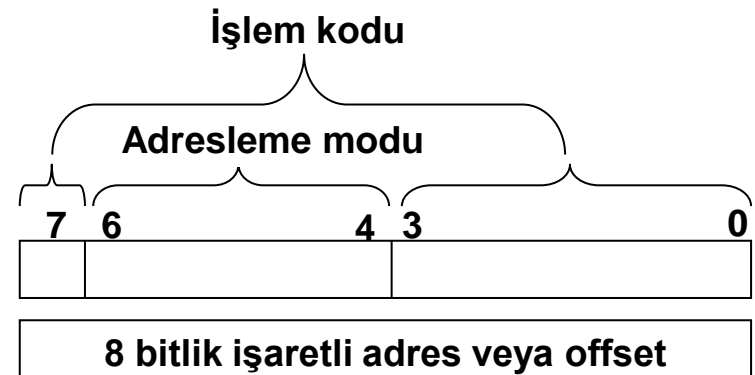
indis ve göreceli adresleme modlarına sahip komutlar 2 byte,

doğal adresleme moduna sahip komutlar ise 1 byte yer kaplamaktadır.

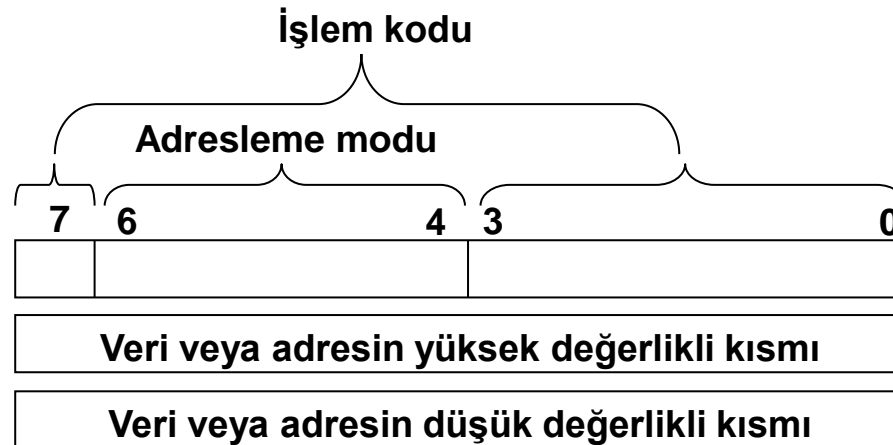
# Komut Formatı



Doğal adreslemede



İndis ve göreceli adreslemede



İvedi, direkt ve dolaylı adreslemede



# Adresleme Modu Seçim Bitleri

Adresleme modunun seçimi için 3 bit gerekmektedir.

Seçim Bitleri	Adresleme Modu
000	Doğal Mod
001	İvedi Mod
010	Direkt Mod
011	Dolaylı Mod
100	İndis Mod
101	Göreceli Mod

**Adresleme modu değiştiğinde opcode'un da sıra ile değişmesini temin etmek için, işlem tipinin 1 biti opcode'un baş tarafına alınmıştır**

**Örnek:** Temel bilgisayar sistemimizde toplama (ADD) komutunun işlem tipi  $00000_2$  olarak seçilmiştir.

İvedi adresleme için opcode kısmı;

0	001	0000
---	-----	------

Opcode'un karşılığı 10h

Direkt adresleme için opcode kısmı;

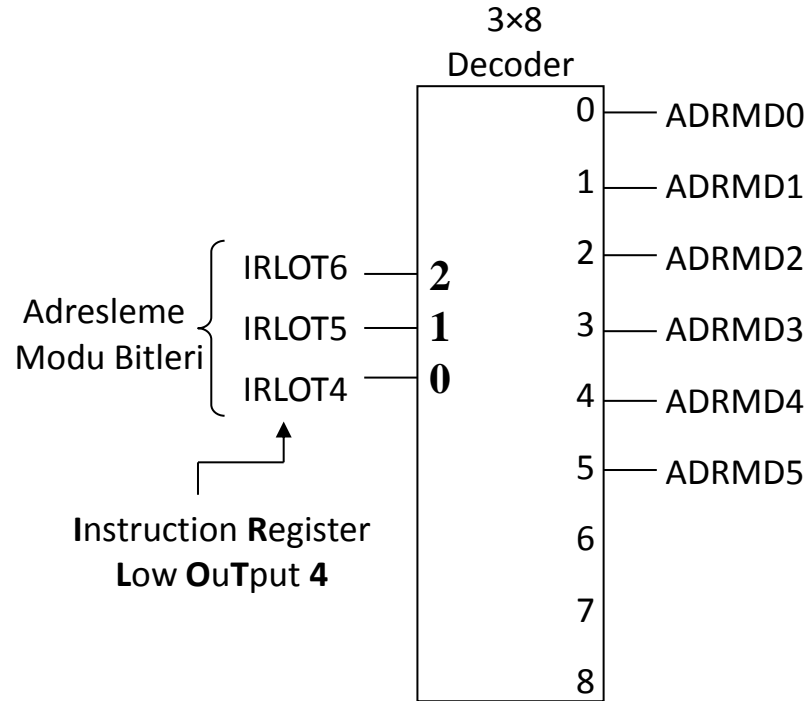
0	010	0000
---	-----	------

Opcode'un karşılığı 20h

# Adresleme Modu Bilgisinin Çözümlemesi

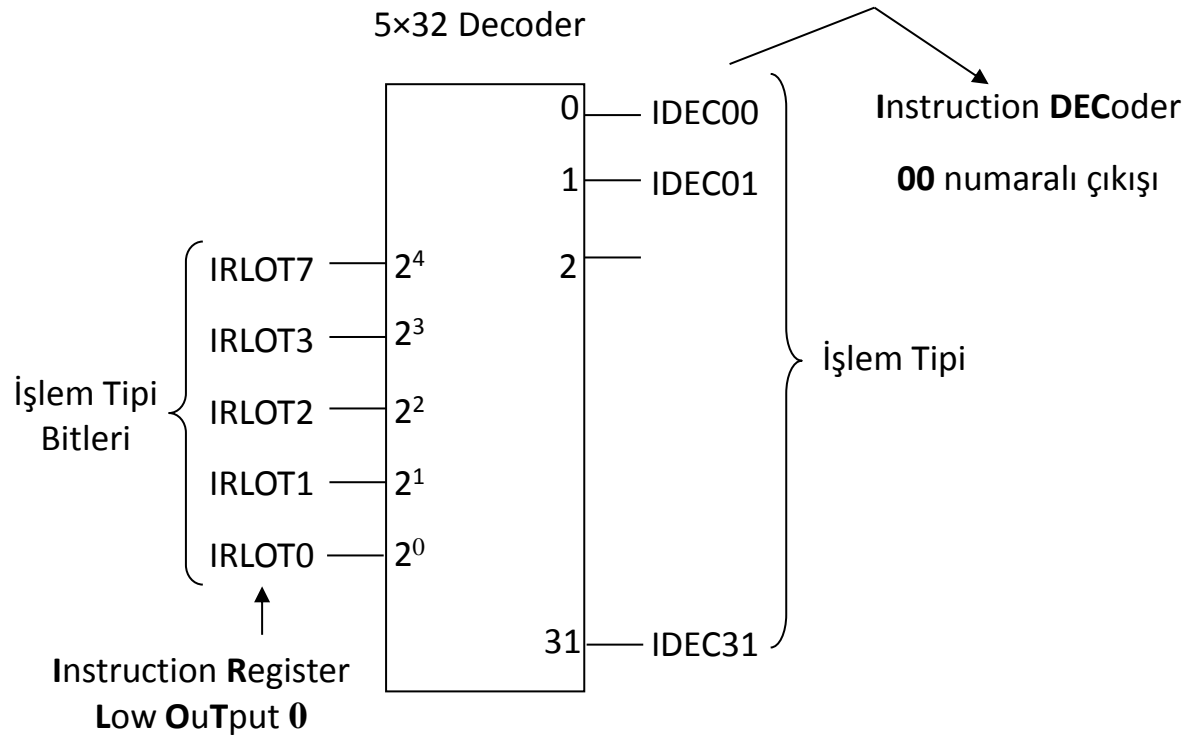
Bellekteki komutlar işletilirken, opcode kısımları komut kaydedicisine aktarılmaktadır. Opcode'un 4, 5 ve 6. bitleri adresleme modunu tayin etmeye yaramaktadır. Bu üç bit,  $3 \times 8$ 'lik bir kod çözücünden geçirilerek 8 adet çıkış elde edilmiştir. Temel bilgisayar sistemimizde bunların 6 tanesi kullanılmıştır. Adresleme modları aşağıdaki kısaltmalarla isimlendirilmiştir;

Kontrol Adı	Adresleme Modu Tipi
ADRMD0	Doğal Mod
ADRMD1	İvedi Mod
ADRMD2	Direkt Mod
ADRMD3	Dolaylı Mod
ADRMD4	İndis Mod
ADRMD5	Göreceli Mod



# Komutun İşlem Tipi Bilgisinin Çözülmesi

Komut kaydedicisindeki opcode'un 0, 1, 2, 3 ve 7. bitleri işlem tipini tayin etmeye yaramaktadır. Bu beş bit,  $5 \times 32$ 'lik bir kod çözücünden geçirilerek 32 adet çıkış elde edilmiştir.



## Komutun Çözülmesi

---

Adresleme modunu ve işlem tipini çözen donanımsal düzeneklerden gelen sinyallere göre komut çözülmektedir.

Örneğin komut kaydedicisindeki işlem tipini gösteren 5 bitin  $00101_2$  olduğunu varsayalım. Bu durumda işlem tipini çözen kod çözücünün 5 numaralı çıkışı yani IDEC05 sinyali aktif olacaktır.

Bu sinyal, tek başına komutun tipini belirlemeye yeterli olmayacaktır. Çünkü aynı sinyali üreten başka komutlar da olabilir. Temel bilgisayar sistemimizin komut setini oluştururken, hem bölme komutu (DIV) hem de ikiye tümleyen alma komutu (NEG) için aynı işlem tipi seçilmiştir. Çünkü bölme komutu doğal adresleme modunu kullanmamaktadır (her zaman operand bilgisi aldığından). Yani işlem tipinin 5 ve adresleme modunun 0 olduğu kombinasyon ile NEG komutu tanımlanmıştır.

# Komutun Çözümlemesi

Komut	Komutun Opcode'u				
	Doğal	İvedi	Direkt	Dolaylı	İndis
DIV	-	15h	25h	35h	45h
NEG	05h	-	-	-	-

IDEC05.ADRMD0 şartı NEG komutunu aktive edecek,

IDEC05.ADRMD1  
IDEC05.ADRMD2  
IDEC05.ADRMD3  
IDEC05.ADRMD4

} şartlarından herhangi biri ise DIV komutunu aktive edecektir.

Bu aktivasyon sinyalleri, komutların icrası ile ilgili donanımsal düzenekleri harekete geçirmek için kullanılacaktır.