

# Makine Dili ve Brookshear Mimarisi

FATMA AHSEN YAZICI

24360859033

# İÇİNDEKİLER:

1-Bilgisayar Mimarisi

2-Makine Dili

3-Komut Yapısı

4-Brookshear Mimarisi

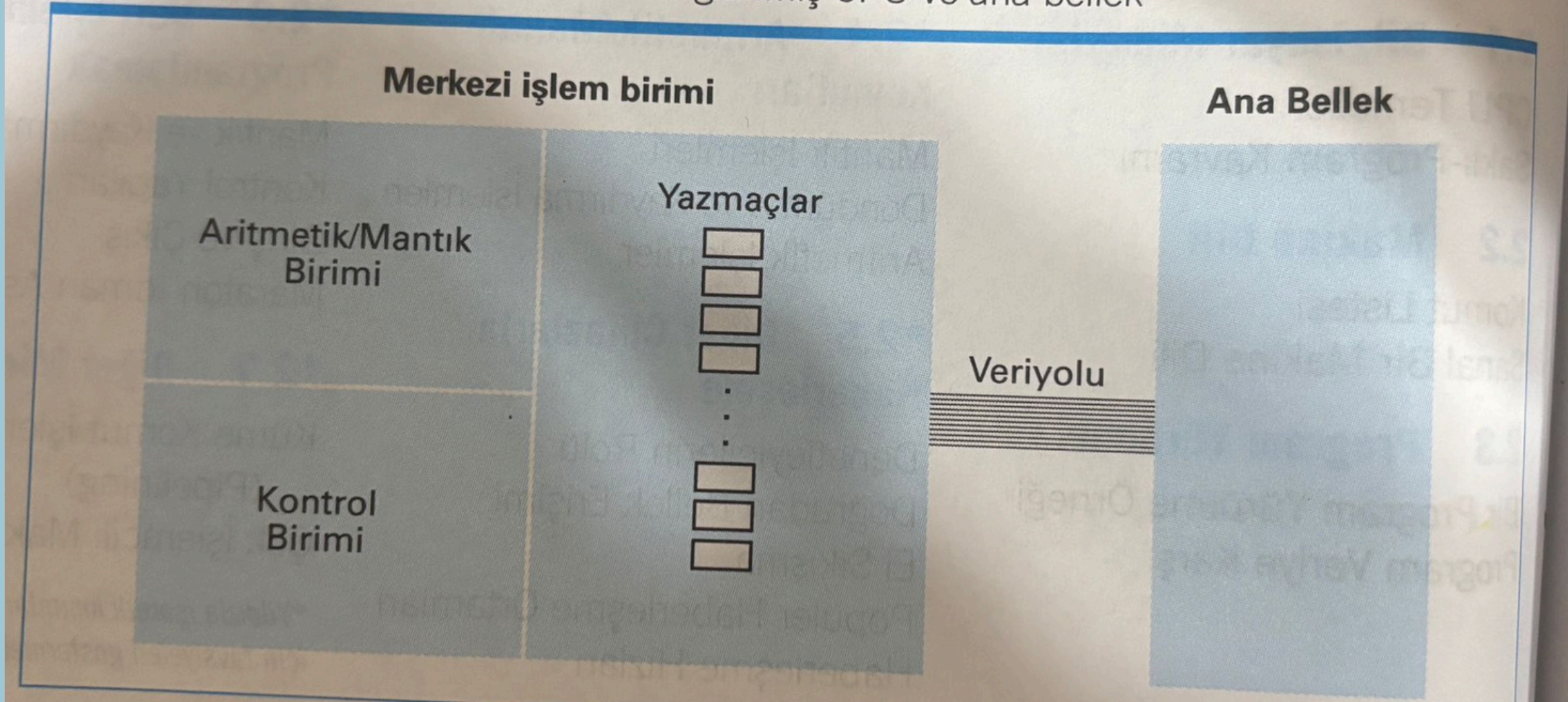
5-Programın Yürütülmesi

# Bilgisayar Mimarisi

Bir bilgisayarda verinin işlenmesini kontrol eden devreye **merkezi işlem birimi(CPU)** veya kısaca işlemci denilir.



**Şekil 2.1** Bir veriyolu ile birbirine bağlanmış CPU ve ana bellek

















# Makine Dili

Bilgisayarlar komutları nasıl anlar?

-Bilgisayarlar sadece 0 ve 1'den oluşan ikili (binary) dili anlayabilir.Yani tüm programlar ve komutlar makine diline çevrilir. Bu dil CPU tarafından doğrudan işlenir.











# Komut Yapısı

- CPU'nun anlayacağı her komut belirli bir yapıya sahiptir.
  - Genel olarak bir komut iki ana bölümden oluşur:
    1. Opcode (Operation Code / İşlem Kodu): CPU'ya hangi işlemi yapacağını söyler.
    2. Operand (İşlenen / Veri): İşlem yapılacak verinin adresini belirtir.
      - Kısaca:  $\text{Komut} = \text{Opcode} + \text{Operand}$





# Brookshear Makinesi

- Brookshear Makinesi, bilgisayar mimarisini öğretmek için tasarlanmış basit bir modeldir.
  - Gerçek bilgisayarlar gibi çalışır ama çok daha basit ve anlaşılırdır.

## Avantajları:

- Karmaşık bilgisayar mimarisini anlamayı kolaylaştırır.
- Öğrencilere CPU, ALU, register ve instruction set kavramlarını öğretir.
- Brookshear makinesiyle basit programlar yazıp, CPU'nun komutları nasıl işlediğini gözlemleyebilirsiniz.



# Sanal Makinenin 12 Temel Komutu

İşlem-kodu (Op-code)	İşlenen (Operand)	Açıklama
1	RXY	Adresi XY olan hafıza hücresinde bulunan bit deseni ile R'yi YÜKLE (LOAD). Örnek: 14A3, A3 adresindeki hafıza hücresinin içeriğinin 4 numaralı yazmaca yerleştirilmesini sağlayacaktır.
2	RXY	R yazmacını XY bit deseni ile YÜKLE. Örnek: 20A3, A3 değerini yazmaç 0'a yerleştirmeye sebep olur.

kodu (Op-code)	İşlenen (Operand)	Açıklama
3	RXY	R yazmacında bulunan bit desenini adresi XY olan hafıza hücresine KAYDET (STORE). Örnek: 35B1, 5 numaralı yazmacın içeriğinin belleğin B1 adresine yerleştirilmesine sebep olur.
4	ORS	R yazmacındaki bit desenini S yazmacına TAŞI (MOVE). Örnek: 40A4, A yazmacının içeriğinin yazmaç 4'e kopyalanmasını sağlar.
5	RST	Yazmaç S ve T'deki bit örüntülerini ikinin tümleyeni gösterimindeymiş gibi TOPLA (ADD) ve sonucu R yazmacına yaz. Örnek: 5726, 2 ve 6 numaralı yazmaçlardaki ikili sayıları toplar ve sonucun 7 numaralı yazmaca yerleştirilmesini sağlar.
6	RST	S ve T yazmaçlarındaki bit örüntülerini sanki değerleri kayan-nokta gösteriminde temsil ediyorlar gibi TOPLA (ADD) ve kayan-nokta sonucu R yazmacına yaz. Örnek: 634E, 4 numaralı yazmaç ve E yazmacındaki değerlerin kayan-nokta değerleri olarak toplanmasına ve sonucun 3 numaralı yazmaca yerleştirilmesine neden olur.
7	RST	S ve T yazmaçlarındaki bit desenlerine OR işlemini uygula ve sonucu R yazmacında tut. Örnek: 7CB4, yazmaç B ve 4'teki içeriğin OR işlemine alınmasını ve sonucun C yazmacında tutulmasını sağlar.
8	RST	S ve T yazmaçlarındaki bit desenlerini TOPLA ve sonucu R yazmacına yerleştirir. Örnek: 8045, 4 ve 5 numaralı yazmaçlardaki içeriklerin AND işlemine sokulmasıyla elde edilen sonucun 0 numaralı yazmaca yerleştirilmesini sağlar.



# Sanal Makinenin 12 Temel Komutu

9	RST	S ve T yazmaçlarındaki bit desenlerini XOR işlemine alarak sonucu R yazmacına yerleştirir. <i>Örnek:</i> 95F3 F ve 3 numaralı yazmaçlardaki içeriğin XOR işlemine alınarak sonuç 5 numaralı yazmaca yerleştirilir.
A	ROX	R yazmacındaki bit desenini X kere 1 bit sağa kaydır. Her seferinde, düşük değer ucunda başlayan biti, yüksek seviye uca yerleştir. <i>Örnek:</i> A403 4 numaralı yazmacın içeriğinin döngüsel biçimde 3 bit sağa kaydırılmasını sağlar.
B	RXY	Eğer R yazmacındaki bit deseni 0 numaralı yazmaçtaki bit desenine eşit ise, XY adresindeki hafıza hücresinde bulunan komuta ATLA (JUMP). Aksi durumda, yürütmeye normal sırada devam et. (atlama, yürütme sırasında program sayacına XY'yi kopyalayarak gerçekleştir) <i>Örnek:</i> B43C , önce 4 numaralı yazmaç ile 0 numaralı yazmaçtaki içerikleri karşılaştırır. Eşit ise, 3C deseni program sayacına yüklenir ve böylece çalıştırılan sonraki komut hafıza adresinde tutulan olacaktır. Aksi durumda, hiçbir şey yapılmaz ve programın yürütülmesi normal seyrinde devam eder.
C	000	Çalışmayı DURDURMA (HALT). <i>Örnek:</i> C000 programın yürütülmesinin durdurulmasına neden olur.

# Programın Yürütülmesi

- CPU görevini, bilgisayar çevrimi adı verilen üç adımlık bir işlemi sürekli tekrar ederek gerçekleştirir.
- Bilgisayar çevrimindeki bu üç adım, **getir(fetch)**, **çöz(decode)** ve **yürüt(execute)** adımlarıdır.
- Bu sürece Makine Döngüsü (Fetch-Decode-Execute) denir.
- Kısaca: CPU, programdaki her komutu alır, çözer ve çalıştırır.





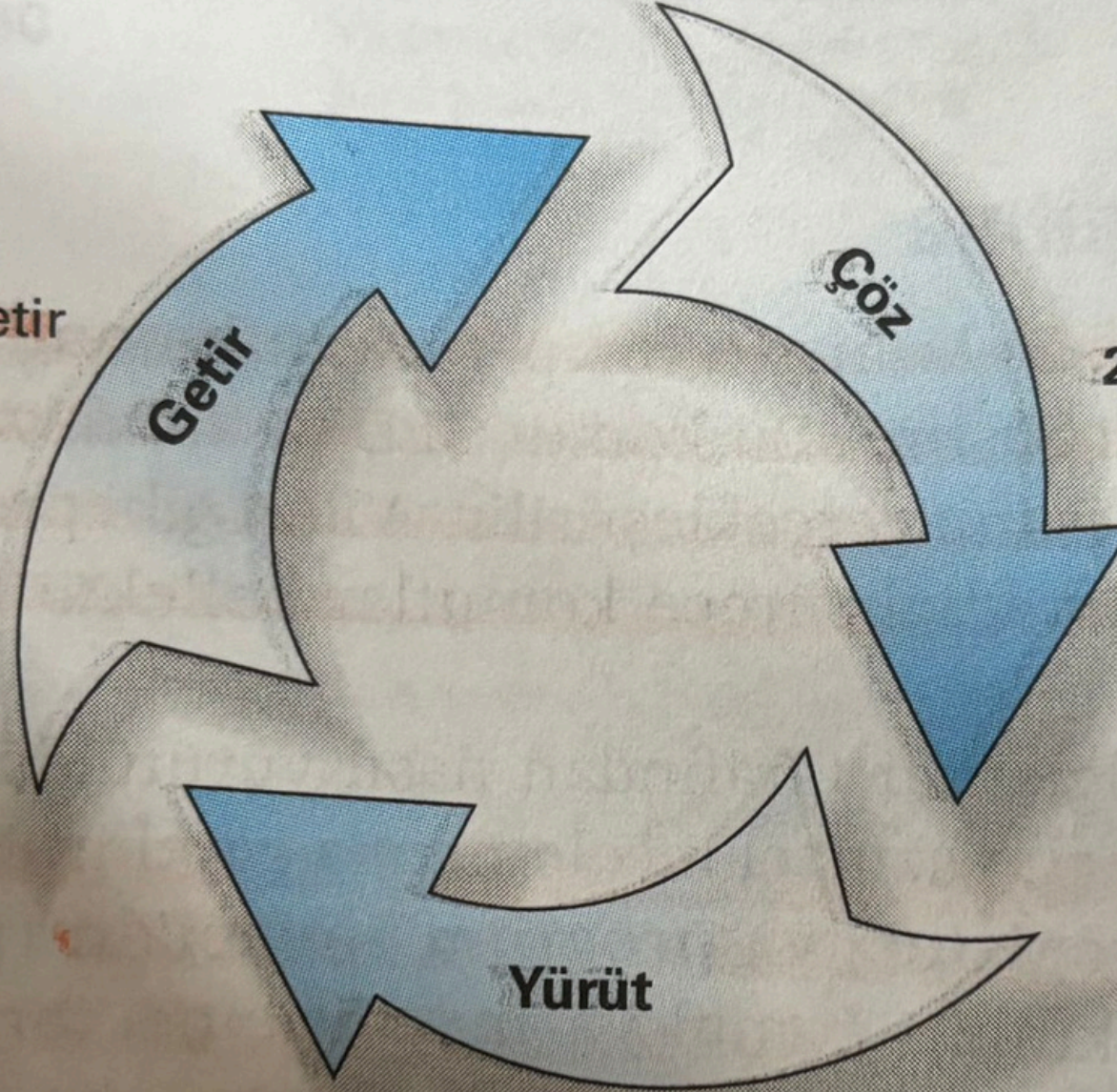








1. Program sayacının gösterdiği adresteki komutu bellekten getir ve sonra program sayacını artır.



2. Komut yazmacındaki bitleri çöz.

3. Komut yazmacındaki komutun gerektiği işlemleri gerçekleştir.



TESİKKÜRLER...