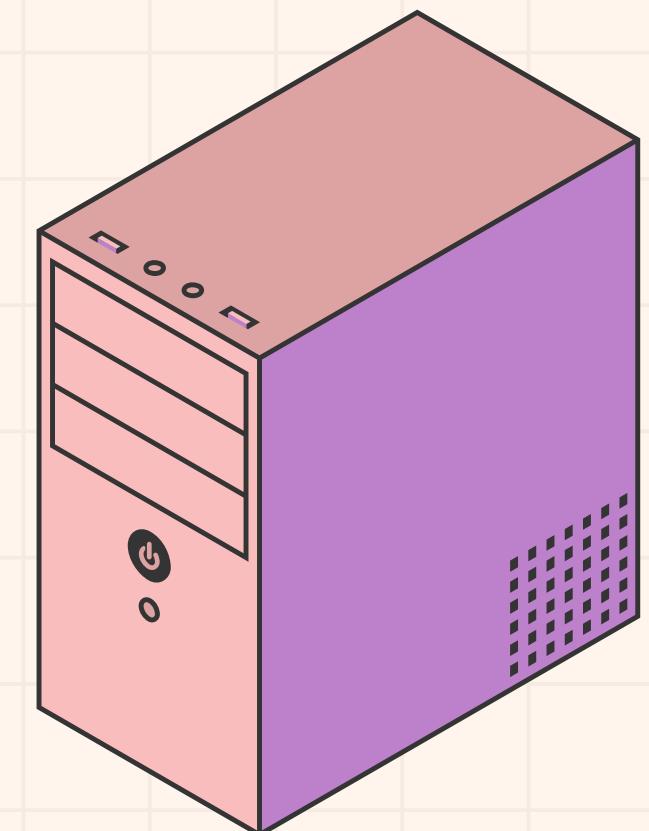
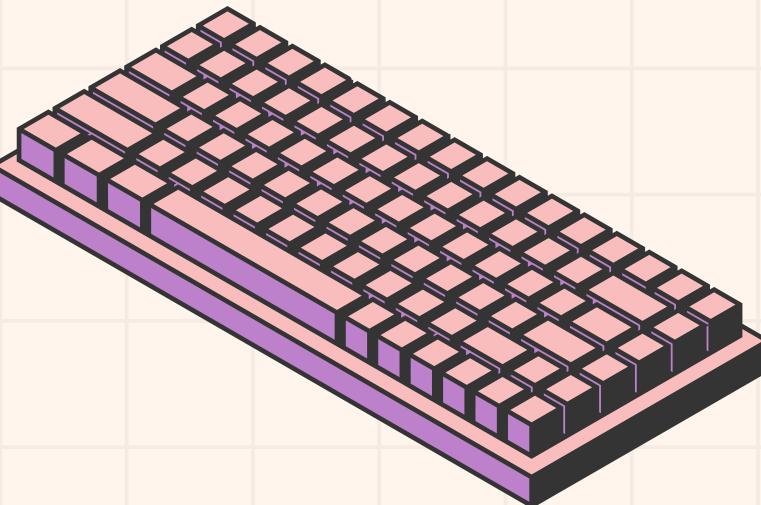
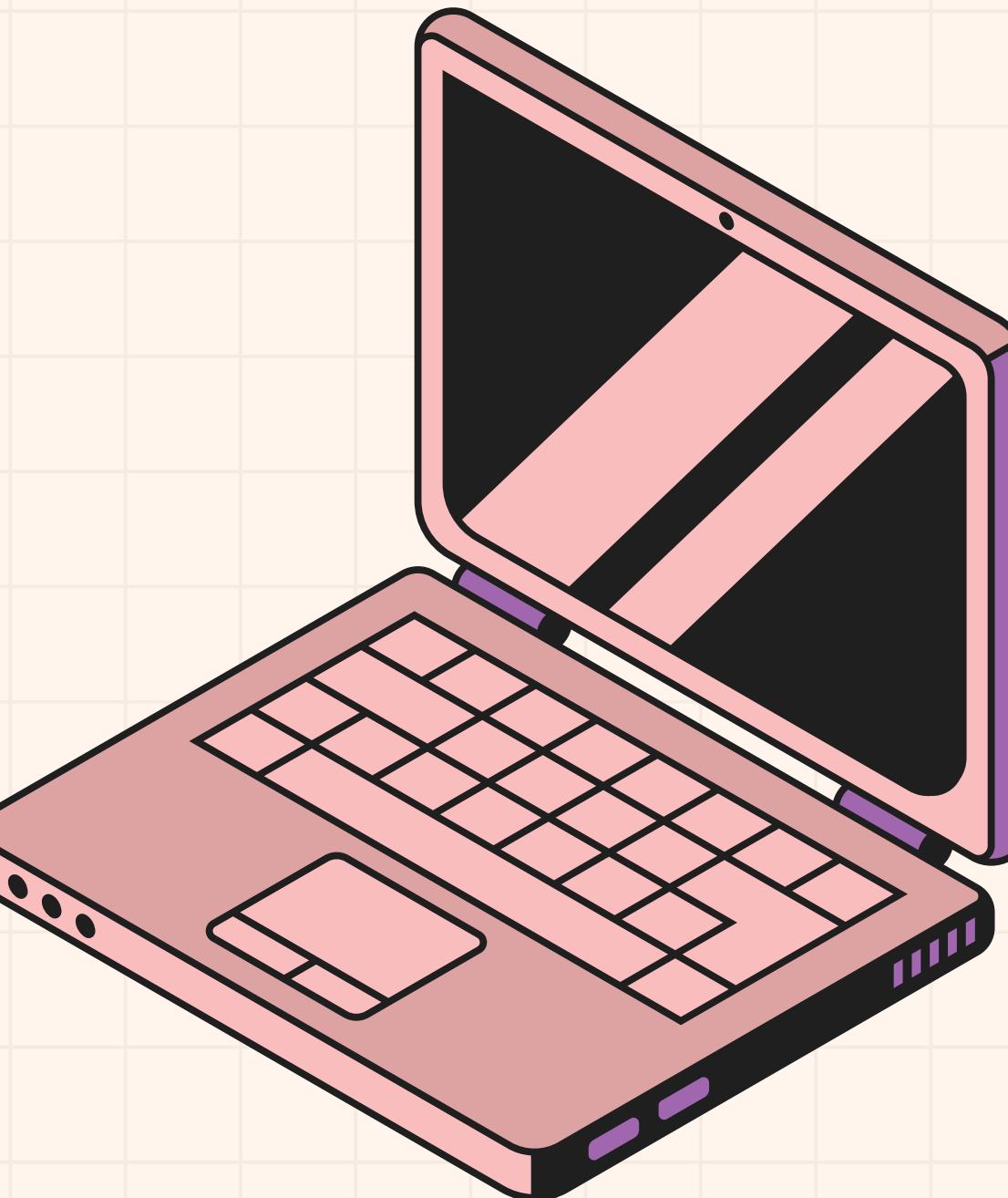


# BİLGİSAYAR MİMARİSİ VE MAKİNE DİLİ



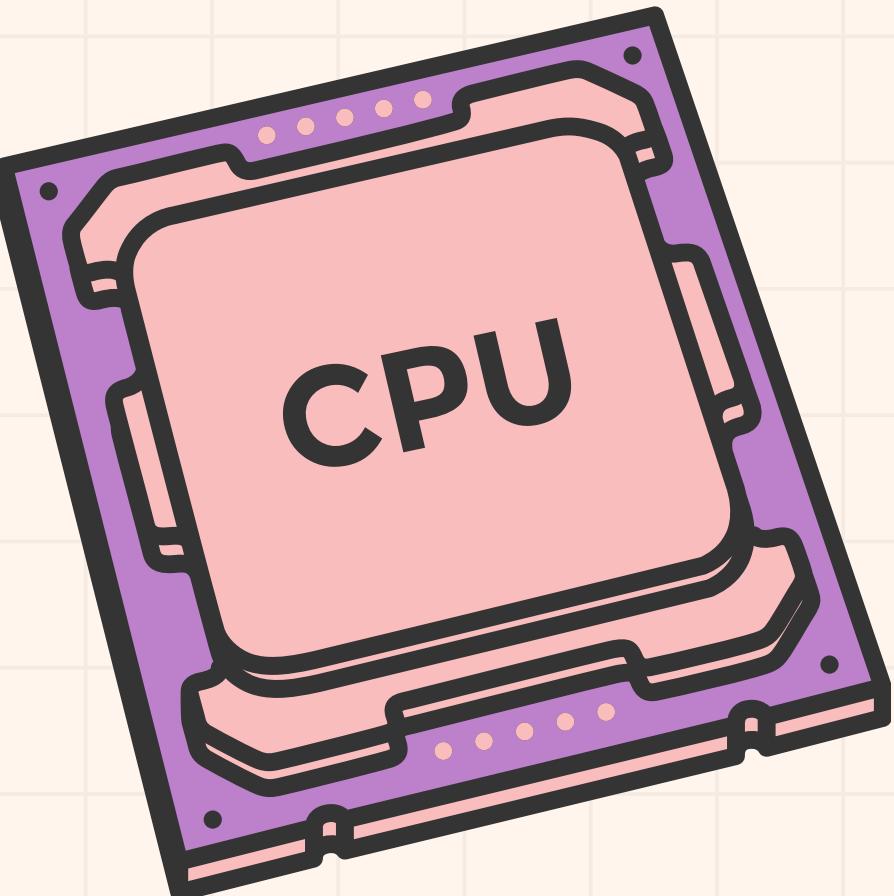
# Bilgisayar Mimarisi Nedir?

- Bilgisayar sadece kod yazmaktan ibaret değildir; donanım ve yazılımın dansıdır.
- Verinin bir yerden bir yere taşınması, aritmetik hesaplamalar ve mantıksal kararlar bütündür.
- Mimari, bir binanın planı gibidir. Programcıya "bu bilgisayar neleri yapabilir" sorusunun cevabını verir.



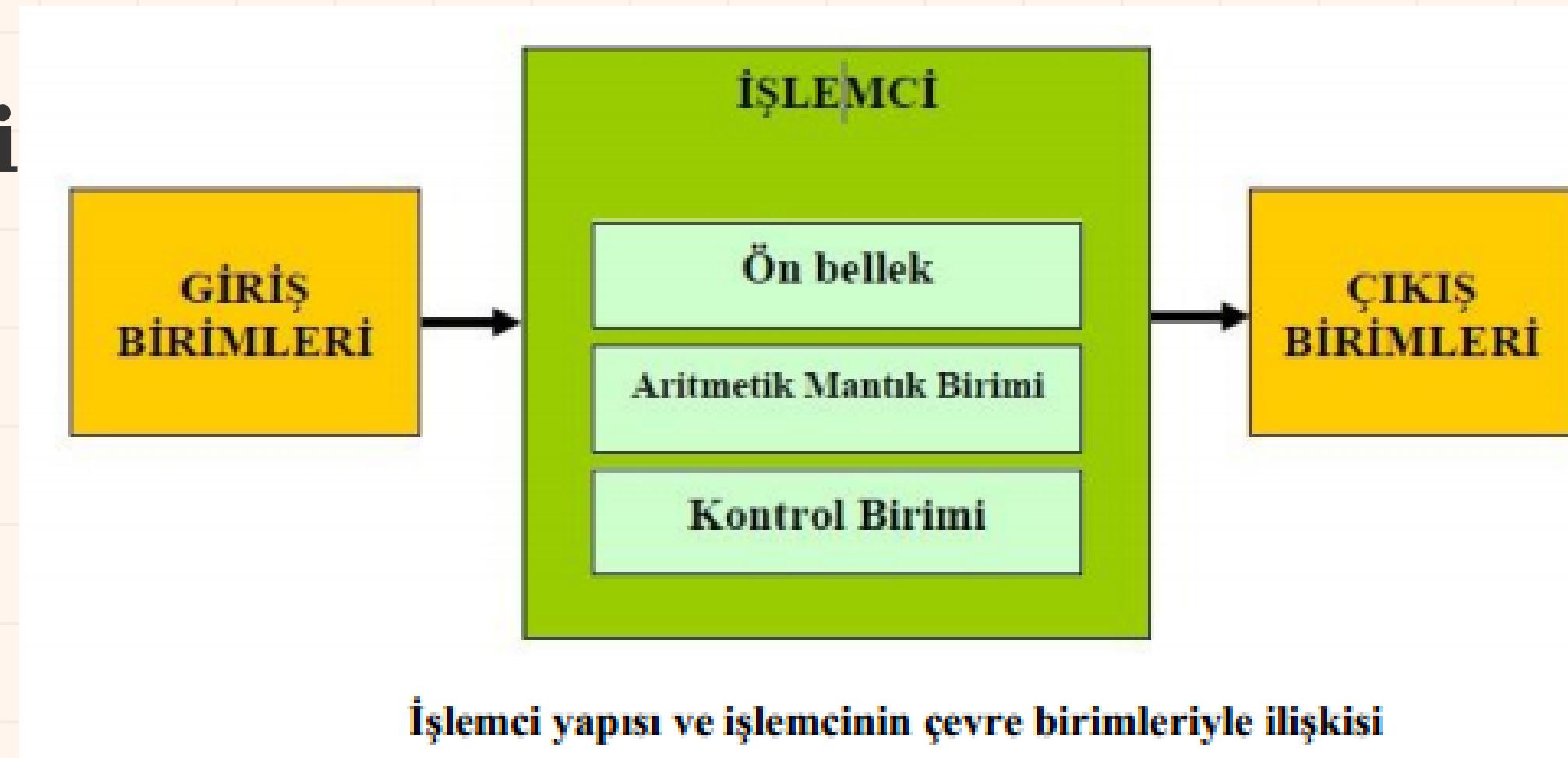
# Merkezi İşlem Birimi (CPU)

- Bir bilgisayarda verinin işlenmesini kontrol eden devreye merkezi işlem birimi (CPU –central processing unit) veya kısaca işlemci adı verilir.
- 20.yüzyılın ortalarında CPU'lar birden çok raf sistemine yerleştirilmiş elektronik devrelerden oluşmaktadır.
- Günümüzde masaüstü veya dizüstü bilgisayarda bulunan CPU'lar küçük, yassı, kare formlu entegreler şeklinde üretilmektedir. Bu entegrelerin bağlantı bacakları makinenin ana devre kartı anakart üzerinde bulunan bir sokete yerleştirilir.
- Akıllı telefonlar ve diğer Mobil İnternet Cihazlarında (MID) bulunan CPU'lar ise küçük boyutlarından ötürü bu işlemcilere mikroişlemci adı verilir.



# ÇPU'nun iç yapısı

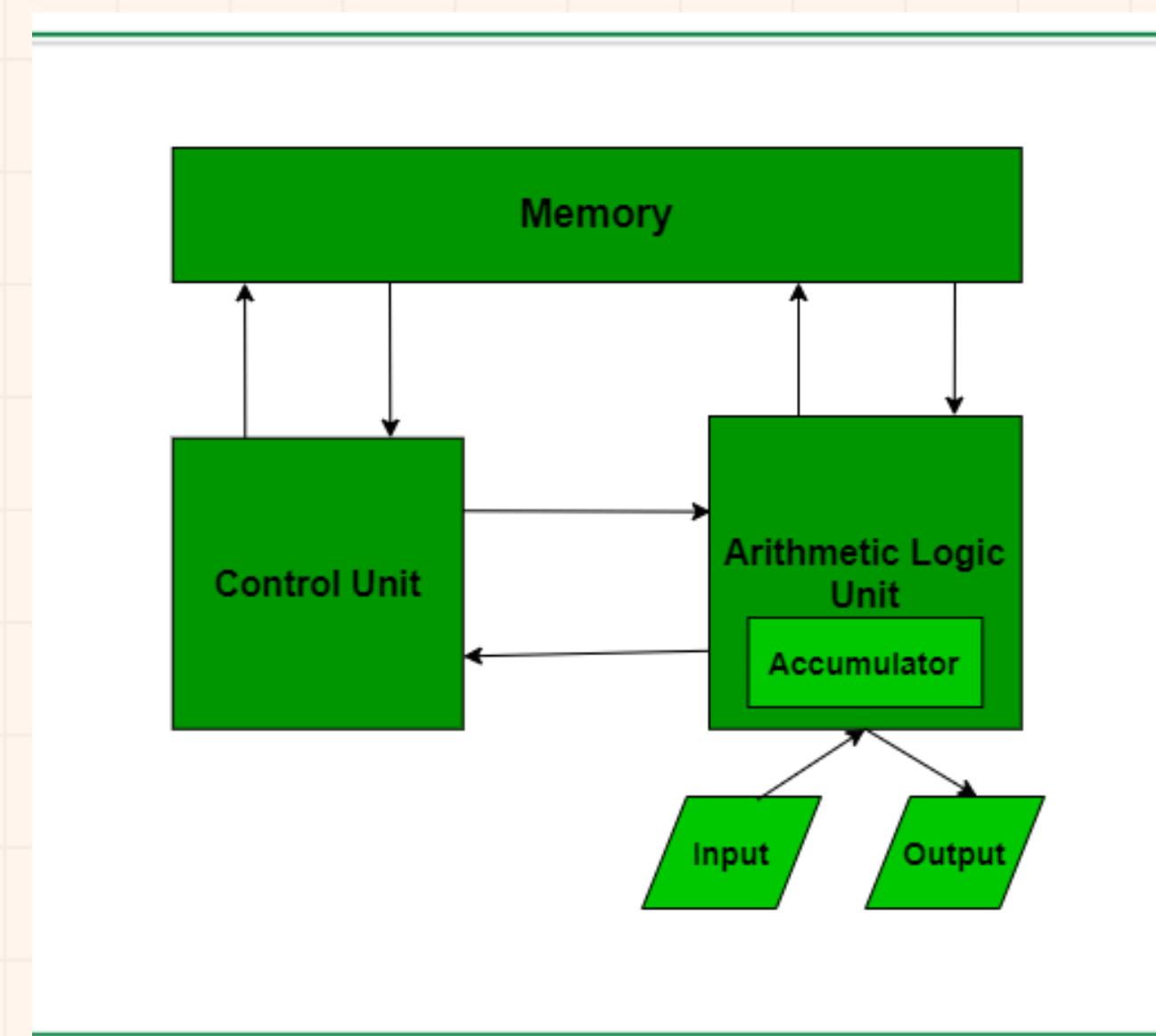
- Aritmetik/Mantık Birimi
- Kontrol Birimi
- Yazmaç Birimi



# 1. Aritmeik/Mantık Komutları

CPU'nun çok önemli bir parçasıdır. Tüm temel aritmetik (toplama, çıkarma, çarpma ve bölme) işlemlerini ve mantıksal işlemleri gerçekleştirir. ALU, genellikle rastgele erişimli belleğe (RAM) doğrudan giriş ve çıkış erişimine sahip olacak şekilde tasarlanmıştır . Modern bilgisayarlarda, ALU'nun kendisi iki kategoriye ayrılır:

- Aritmetik Birim (AU)
- Mantık Birimi (LU)



# ALU'nun Fonksiyonları

Aşağıda Aritmetik Mantık Biriminin (ALU) bazı fonksiyonları yer almaktadır.

- Giriş işlenenleri, eklenen işlenenler, birikmiş sonuçlar ve kaydırma sonuçları, ALU'da saklanabilir.
- ALU alt birimleri, veri akışını ve işlemleri düzenleyen kapılı devreler kullanır. Bölme ve çarpma işlemleri için bir dizi toplama, çıkarma ve kaydırma işlemi kullanılır.
- Hem toplama, çıkarma, çarpma ve bölme gibi aritmetik işlemleri, hem de mantıksal işlemleri gerçekleştirir.

# Komut Birimi

- İşlemcinin aktivitelerini düzenleyen devreleri içeren birimdir.

# Yazmaç Birimi

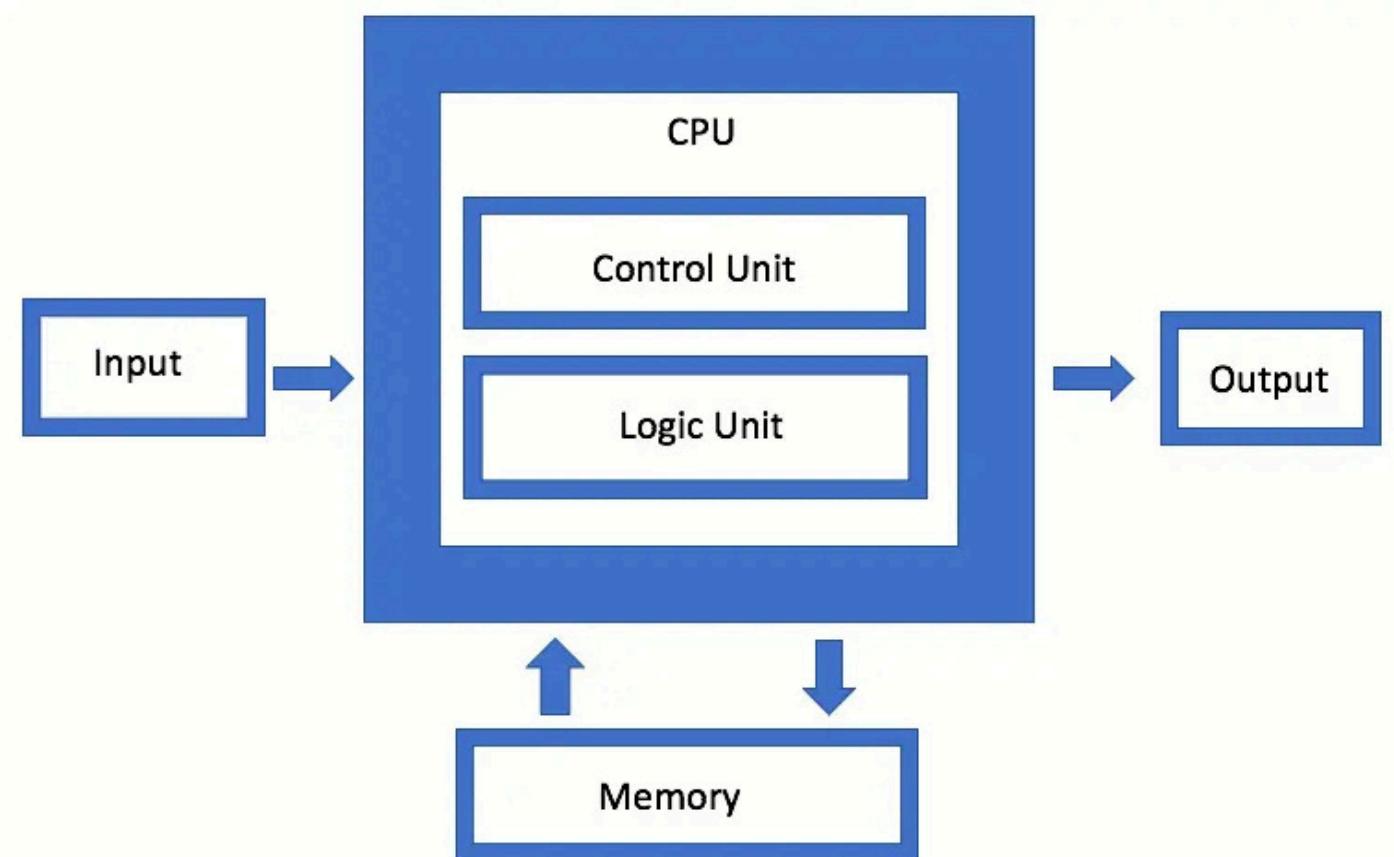
- CPU'nun içindeki bilgilerin geçici olarak depolanması için kullanılan ve yazmaç adı verilen veri depolama hücrelerinden oluşur.
- Yazmaçların bazıları genel-amaçlı yazmaç olarak adlandırılırken diğerlerine özel amaçlı yazmaç adı verilir.

# Genel Amaçlı Yazmaçlar

- Genel-amaçlı yazmaçlar CPU tarafından işlenen verinin geçici olarak saklanması için kullanılırlar. Bu yazmaçlar hem aritmetik/mantık biriminin giriş verilerini tutarlar hem de bu birim tarafından üretilen sonuçlar için geçici depolama alanı sağlarlar.
- Bitler halindeki verinin taşınması için bir makinedeki CPU ve ana bellek birbirine veriyolu adı verilen tel grubuya bağlıdır.
- Bu veri yolu üzerinden CPU, ilgili bellek hücresinin adresini belirterek ve bunun yanı sıra bellek devresine o adressteki veriyi almak istediğini söyleyen bir elektronik sinyal göndererek, ana bellekten veri çekebilir(bu işleme okuma adı verilir). Aynı şekilde CPU, hedef hücrenin adresini belirterek ve ana belleğe, göndermekte olduğu veriyi saklamasını istediğini söyleyen bir elektronik sinyal aracılığıyla, belleğe veri yerleştirebilir (bu işleme ise yazma adı verilir).

# Von Neumann Mimarisi

- Saklı Programlar Kavramı, bilgisayara verdığımız komutların veriler gibi bellekte muhafaza edilerek bellekten ilgili komutları değiştirme imkanı sağlar. Böylelikle programda değişiklikler yapabiliriz veya başka bir program çalıştırılmak için yeni bir bilgisayar tasarıımına ihtiyacımız kalmaz.
- Von Neumann Mimarisi verilerin program komutları ile tek bir bellekte sıralı olarak saklandığı ve sıralı şekilde bellekten çekiliip işlemcide işlemlere tabi tutulduktan sonra çıktı veren bilgisayar mimarisidir. Başka bir ifade ile komut belleği ve veri belleği aynı belleği paylaşmaktadır.



# Bellek Hiyerarşisi

- Bellek hiperarşisi çeşitli veri depolama birimlerinin veri iletim hızı/işlem gücüne göre hiperarşik olarak sıralanmasına verilen addır. İşlemcilerin hızı ve işlem gücü arttıkça bilgisayar mimarisini oluşturan veri saklama birimlerinin işlemciyle arasındaki senkron farkı gitgide artmaktadır.
- Hiperarşideki bellek türleri hızlılık, düşük gecikme ve maliyete göre yukarıdan aşağıya doğru şu şekilde sıralanabilir:
  1. Yazmaçlar
  2. Ön bellek
  3. RAM
  4. Disk

# Önbellek

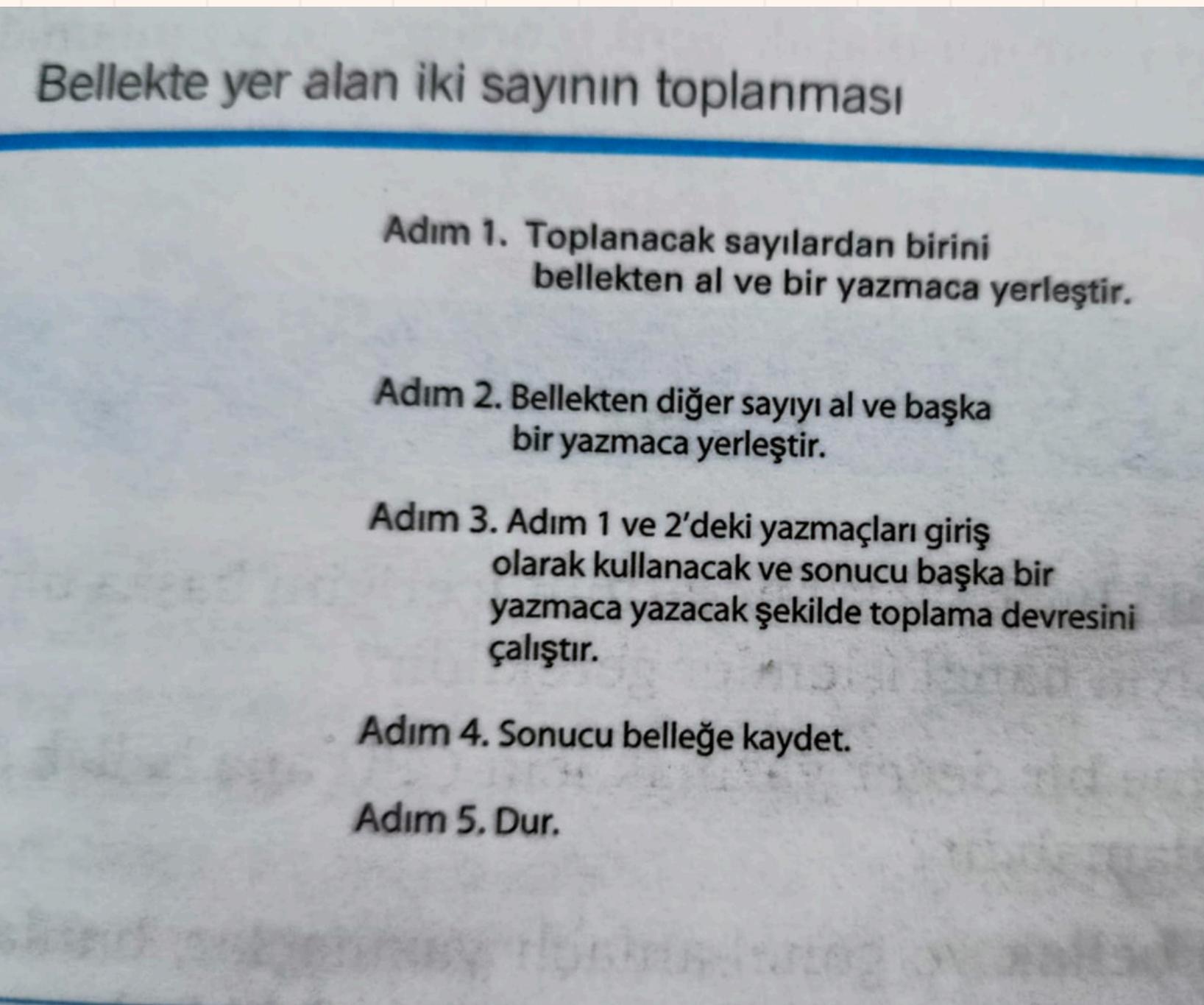
- Önbellek, fiziksel olarak CPU'nun içinde yer alan yüksek-hızlı bir bellek parçasıdır (genellikle bir kaç yüz KB büyüklüğünde). Makine, bu özel bellek alanında, ana belleğin o an kullandığı bölümünün bir kopyasını saklar. Bu sayede, normal yazmaçlar ve ana bellek arasında gerçekleşmesi gereken veri transferi, yazmaçlar ve önbellek arasında gerçekleşir. Önbellekte yapılan değişiklikler uygun bir zamanda toplu olarak ana belleğe aktarılır. Sonuç olarak CPU, nispeten daha yavaş olan ana bellek haberleşmesiyle vakit kaybetmediği için, komutları çok daha hızlı bir şekilde yürütür.

# Ana Bellek

- Ana bellek (main memory), bilgisayarın “çalışma masası” gibidir. Tıpkı bir çalışma masasında projeleriniz için ihtiyaç duyduğunuz araçları ve belgeleri hazır tutmanız gibi, ana bellek de işlemcinin (CPU) hızlı bir şekilde erişmesi gereken verileri ve program talimatlarını geçici olarak saklar. Genellikle RAM (Rastgele Erişim Belleği) olarak bilinen bu bellek, uçucu bir yapıdadır; yani, bilgisayar kapatıldığında masanın üzerindeki her şey kaybolur.
- Eğer masanız küçükse veya dağınıksa, işler yavaşlar. Bu yüzden, modern bilgisayarlarda ana belleğin kapasitesine ve hızına büyük önem verilir.
- RAM ve önbellek gibi hızlı bellekler, CPU'nun veriyi anında işlemesini sağlar.

# Bellek Adresleme Örneği

**Bellekte yer alan iki sayının toplanması**



**Adım 1.** Toplanacak sayılarından birini bellekten al ve bir yazmaca yerleştir.

**Adım 2.** Bellekten diğer sayıyı al ve başka bir yazmaca yerleştir.

**Adım 3.** Adım 1 ve 2'deki yazmaçları giriş olarak kullanacak ve sonucu başka bir yazmaca yazacak şekilde toplama devresini çalıştır.

**Adım 4.** Sonucu belleğe kaydet.

**Adım 5.** Dur.

# Makine Dili

- Saklı-program kavramını uygulayabilmek için CPU'lar, bit desenleri şeklinde kodlama komutları tanımlayabilecek şekilde tasarılanırlar. Kodlama sistemiyle birlikte bu komutlar bütününe makine dili denir. Bu dilde kullanılan bir komuta makine-seviyesi komut veya daha yaygın olarak, makine komutu denir.

# Komut Listesi

- Sıradan bir CPU'nun çözmek ve yürütmek zorunda olduğu makine komutlarının listesi oldukça kısadır. Gerçekte, bir bilgisayar sistemi belirli temel görevleri yerine getirebiliyorsa yeni özellikler eklemek sistemin teorik kapasitesini artırmaz.
- Bilgisayar tasarıımı sırasında yukarıda bahsettiğimiz noktanın nerede seçileceğinin kararı, bilgisayar mimarisinde iki farklı felsefenin gelişmesine yol açmıştır. Bunlar:
  1. İndirgenmiş komut takımı bilgisayar (RISC -reduced instruction set computer)
  2. Karmaşık komut takımı bilgisayar(CISC- complex instruction set computer)

# İndirgenmiş Komut Takımlı Bilgisayar (RISC)

- CPU'nun en az sayıda makine komutu yürütebilecek şekilde tasarlanması yaklaşımına indirgenmiş komut takımlı bilgisayar(RISC- reduced instruction set computer) adı verilir.
- RISC mimarisinin savunduğu argüman, bu tür işlemcilerin daha verimli ve hızlı olması ve üretim maliyetlerinin düşük olmasıdır.

# Karmaşık Komut Takımlı Bilgisayar (CISC)

- Öte yandan bazıları, teknik olarak birçoğu gereksiz olsa da, çok sayıda karmaşık komutu yürütebilen CPU'ların faydalarını savunmaktadır.
- Bu yaklaşım ise karmaşık komut takımlı bilgisayar (CISC-complex instruction set computer) olarak adlandırılır.
- CISC mimarisinin dayanağı ise daha karmaşık bir CPU'nun günümüzün sürekli gelişmekte olan yazılımlarıyla daha iyi baş edebileceği şeklindedir.

Bir işlemcinin komutları, RISC veya CISC olmasına bakılmaksızın üç ayrı gruba ayrılabilir. Bunlar:

- 1. Veri Aktarma Komutları**
- 2. Aritmetik/Mantık Komutları**
- 3. Kontrol Komutları**

# Veri Aktarma Komutları

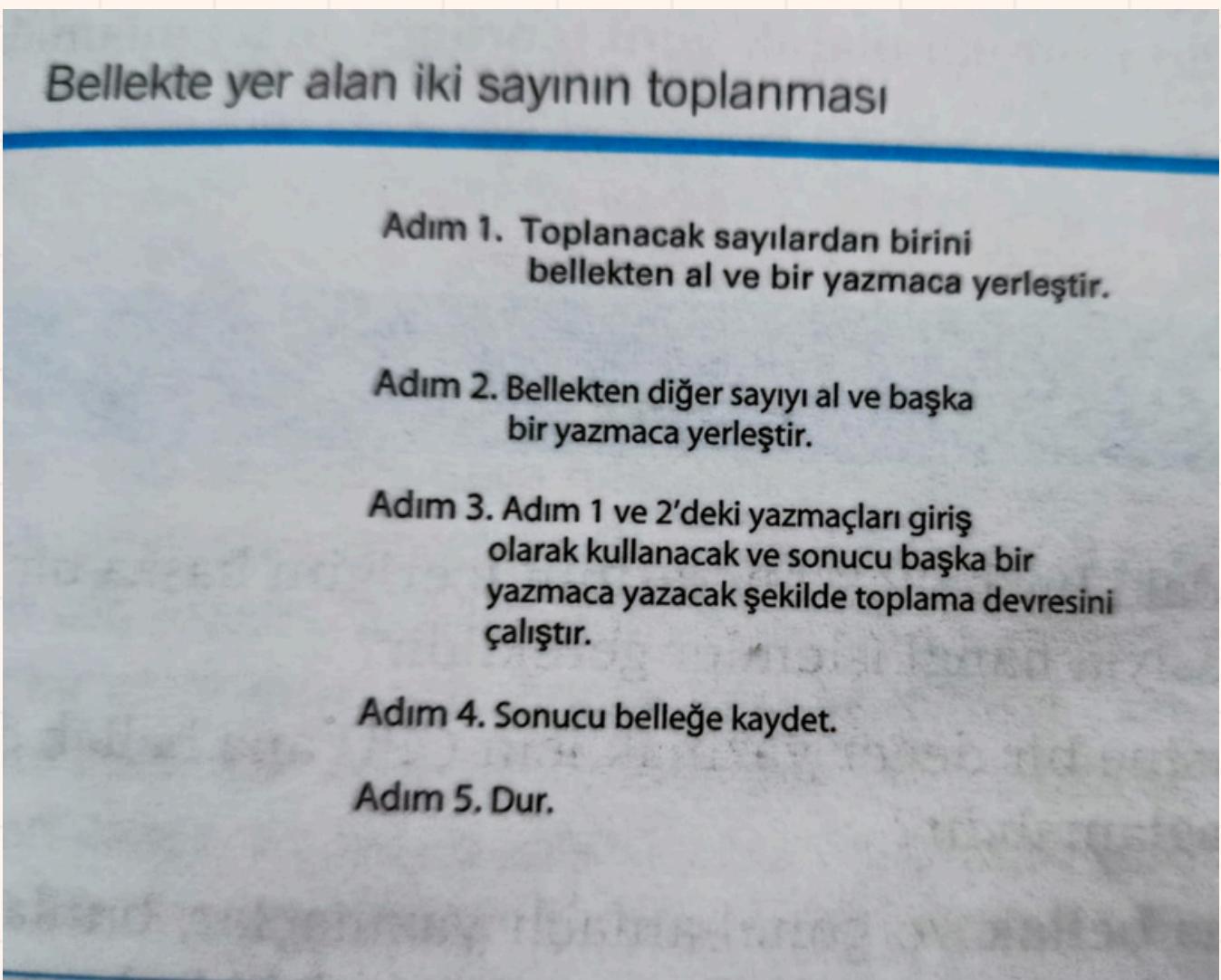
- Bu grupta verinin bir yerden başka bir yere taşınmasını sağlayan komutlar yer alır.
- Bir genel-amacı yazmacı, bir bellek hücresinin içeriğiyle doldurma isteğine YÜKLEME (LOAD) komutu denir. Öte yandan bu işlemin tersine, yani bir yazmacın içeriğinin bir bellek hücresına aktarılması isteğine ise KAYIT (STORE) komutu adı verilir.
- İşlemciye veri giriş ve çıkışını sağlayan komutlara GİRİŞ/ÇIKIŞ (G/Ç) komutları denir. G/Ç komutlarını veri aktarım komutlarının bir parçası olarak düşünmeliyiz.

# Aritmetik/Mantık Komutları

- Bu grup, işlemci içerisindeki kontrol birimine, aritmetik mantık biriminde bir işlem gerçekleştirmesini söyleyen komutlardan oluşur.
- Bu işlemlerden bazıları AND, OR VE XOR gibi Boolean işlemleridir. Bir diğer işlem grubu ise bir yazmacın içeriğinin kendi içerisinde sağa veya sola kaydırılmasını sağlayan komutlardan oluşan gruptur. Bu işlemler, en sona gelen bitin durumuna bağlı olarak, KAYDIRMA(SHIFT) veya DÖNDÜRME (ROTATE) işlemi olarak adlandırılırlar.

# Kontrol Komutları

- Bu gruptaki komutlar veriyi işlemek yerine programın çalışmasını yönetirler.
- CPU'ya bellekteki bir sonraki komut yerine başka bir komut işlemesini söyleyen ATLAMA (JUMP veya BRANCH) komutlarıdır. ATLAMA komutları iki farklı şekilde karşımıza çıkabilir; şartsız (koşulsuz) atlama veya şartlı (koşullu) atlama. Aradaki fark, şartlı atlama komutlarının ancak belli bir şartın sağlanması durumunda programın akışını değiştirmesidir.



## Şekilde;

- 1,2 ve 4. adımlar Veri aktarma komutları aracılığı ile gerçekleştirilir. Adım 1 ve 2 YÜKLEME komutu; Adım 4 ise KAYIT komutudur.
- Adım 3 Aritmetik/mantık komutları ile gerçekleştirilen bir işlemidir.
- Adım 5 Kontrol komutlarına verilebilecek en temel örnektir.

# Sanal Bir Makine Dili

- Bir makine komutunun kodlanmış hali iki bölümden oluşur:

- 1. İşlem kodu (op-code)**
- 2. İşlenen (operand)**

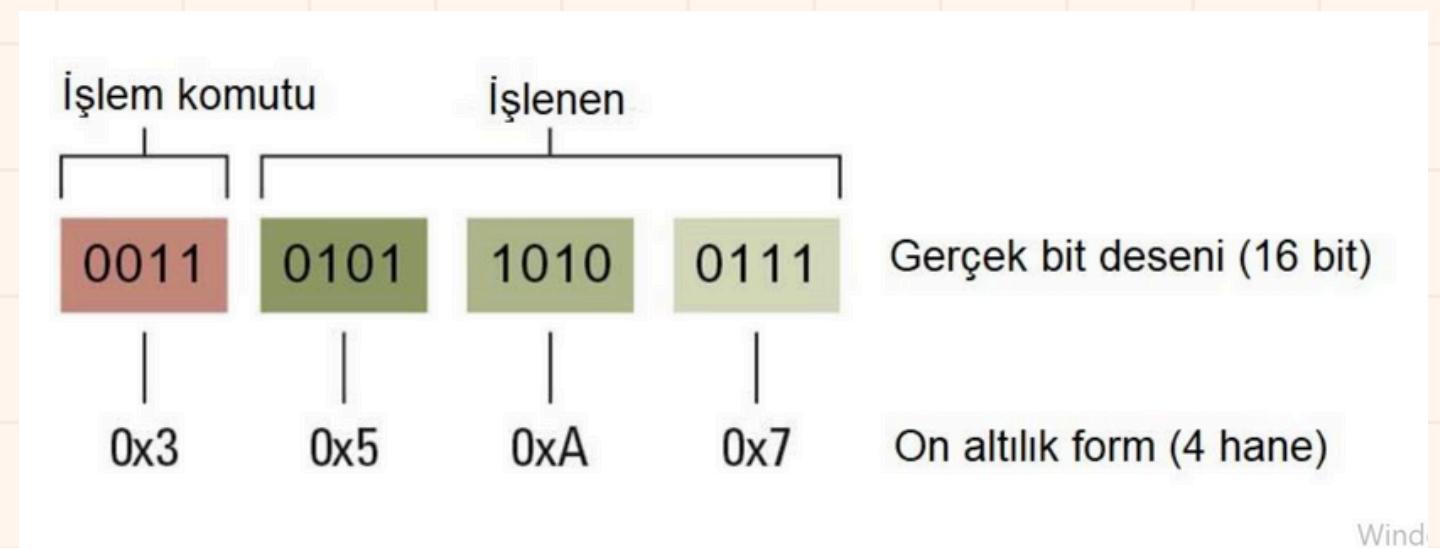
## İşlem kodu (op-code):

- Bu bölümde yer alan bitler komutun hangi temel işlemi (KYADIR, KAYDET ve ATLA gibi) gerçekleştireceğini belirtir.

## İşlenen (operand):

- Bu bölüm ise işlem koduyla belirlenen işlem hakkında detaylı bilgi verir. Örneğin KAYDET işleminde, işlenen bölümü kaydedilecek verinin hangi yazmaçta olduğunu ve hangi bellek hücresına kaydedileceğini belirtir.

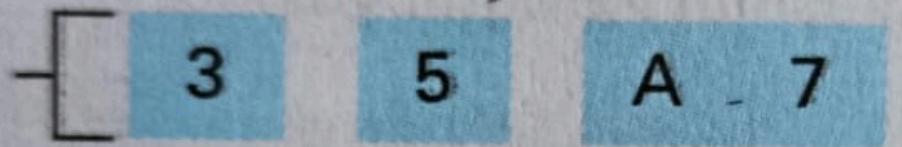
## Bir Makine dili komutunun yapısı:



Wind

## Şekil 2.6 "35A7" komutunun çözümlenmesi

Komut



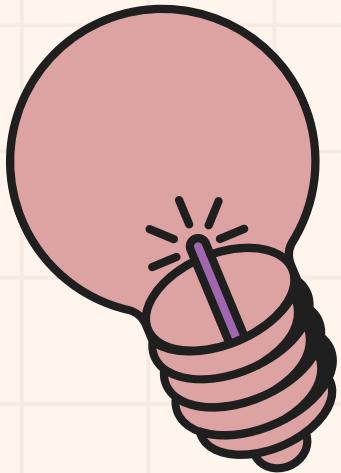
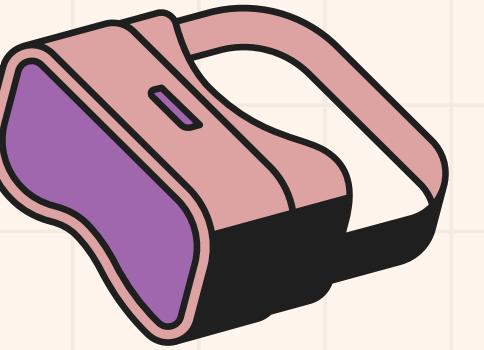
"3" işlem kodu bir yazmacın içeriğinin, bir bellek hücresine yazılıacağı anlamına gelir.

İşlenenin bu bölümü verinin kayıt edileceği bellek hücresinin adresini içerir.

İşlenenin bu bölümü hangi yazmacın içeriğinin belleğe kayıt edileceğini belirtir.

# Kaynakça

- <https://www.geeksforgeeks.org/computer-science-fundamentals/difference-between-alu-and-cu/>
- <https://alperenonal.com/blog/von-neumann-mimaris-vs-hardvard-mimaris>
- [https://tr.wikipedia.org/wiki/Bellek\\_hiyerar%C5%9Fisi#:~:text=Bellek%20hiyerar%C5%9Fisi%20%C3%A7e%C5%9Fitli%20veri%20depolama,aras%C4%B1ndaki%20senkron%20fark%C4%B1%20gitgide%20artmaktad%C4%B1r](https://tr.wikipedia.org/wiki/Bellek_hiyerar%C5%9Fisi#:~:text=Bellek%20hiyerar%C5%9Fisi%20%C3%A7e%C5%9Fitli%20veri%20depolama,aras%C4%B1ndaki%20senkron%20fark%C4%B1%20gitgide%20artmaktad%C4%B1r)
- [https://prezi.com/p/sk2\\_mbvfcz6f/bilgisayar-mimaris-bellek-hiyerarsisi/](https://prezi.com/p/sk2_mbvfcz6f/bilgisayar-mimaris-bellek-hiyerarsisi/)
- **Brookshear, J.G. (2022). Bilgisayar Bilimine Giriş. Ankara: Nobel Akademik Yayıncılık.**



# THANK YOU



+123-456-7890



WWW.REALLYGREATSITE.COM



123 ANYWHERE ST., ANY CITY

