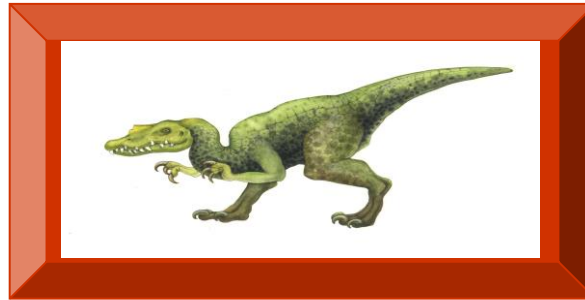


# GİRİŞ:

## İşletim Sistemlerine Genel Bakış





# Bölüm 1: Giriş

---

- İşletim Sistemleri Ne Yapar?
- Bilgisayar Sistemi Organizasyonu
- Bilgisayar Sistemi Mimarisi
- İşletim Sistemi Yapısı
- İşletim Sistemi İşleyişi
- Proses Yönetimi
- Bellek Yönetimi
- Depolama Birimi Yönetimi
- Koruma ve Güvenlik
- Dağıtık Sistemler
- Özel Amaçlı Sistemler
- Bilgisayar Ortamları





# Hedefler

---

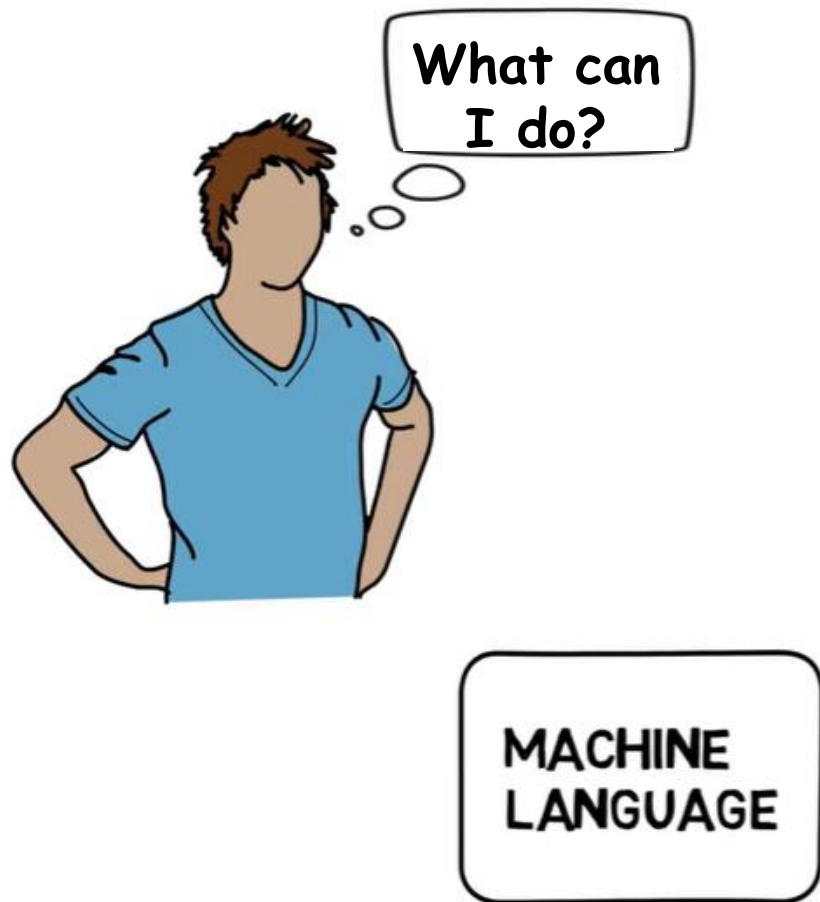
- Bir bilgisayar sisteminin genel organizasyonunu ve kesmelerin rolünü tanımlamak
- Modern, tek / çok işlemcili bir bilgisayar sistemindeki bileşenleri tanımlamak
- Kullanıcı modu x çekirdek modu kavramını anlamak
- İşletim sistemlerinin çeşitli bilgi işlem ortamlarında nasıl kullanıldığını incelemek
- Ücretsiz ve açık kaynaklı işletim sistemlerine genel olarak örnekler vermek





# What is an Operating System?

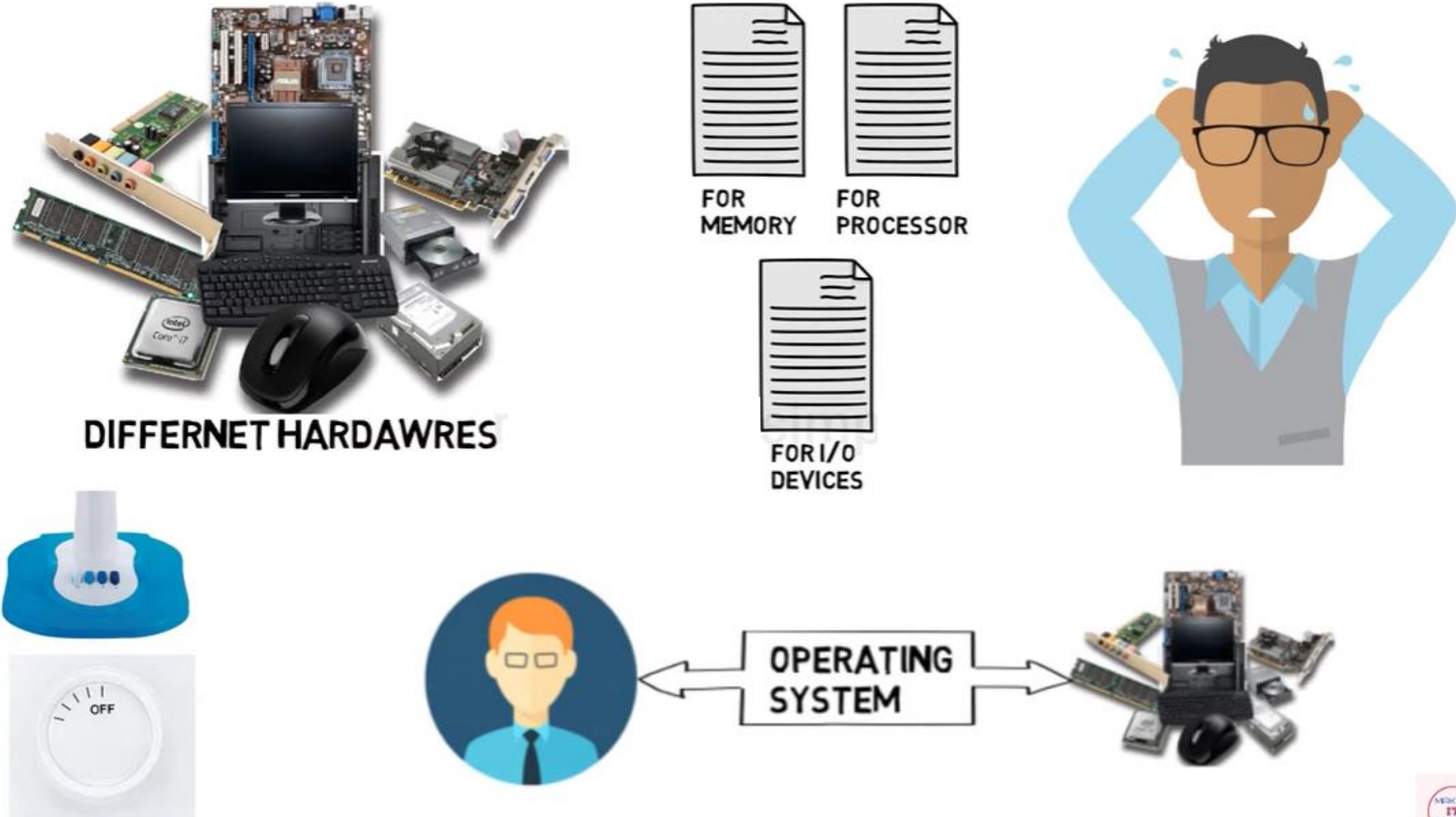
- How can I use easily a **large number of hardware with very different features?**





# What is an Operating System?

- İşletim sistemi, bilgisayar kullanıcısı ile bilgisayar donanımları arasında aracı görevi gören bir yazılımdır





# İşletim Sistemi Nedir?

---

- **İşletim sistemi**; bilgisayarda var olan **donanım kaynaklarını yöneten** ve çeşitli **uygulama yazılımları** için **yaygın olarak kullanılan servisleri sağlayan** bir yazılımlar bütünüdür.
- İşletim sisteminin hedefleri:
  - Kullanıcı programlarını çalıştırmak ve kullanıcı problemlerini çözmeyi kolaylaştırmak
  - Bilgisayar sisteminin kullanımını kolaylaştırmak
  - Bilgisayar donanımını verimli bir şekilde kullanmak





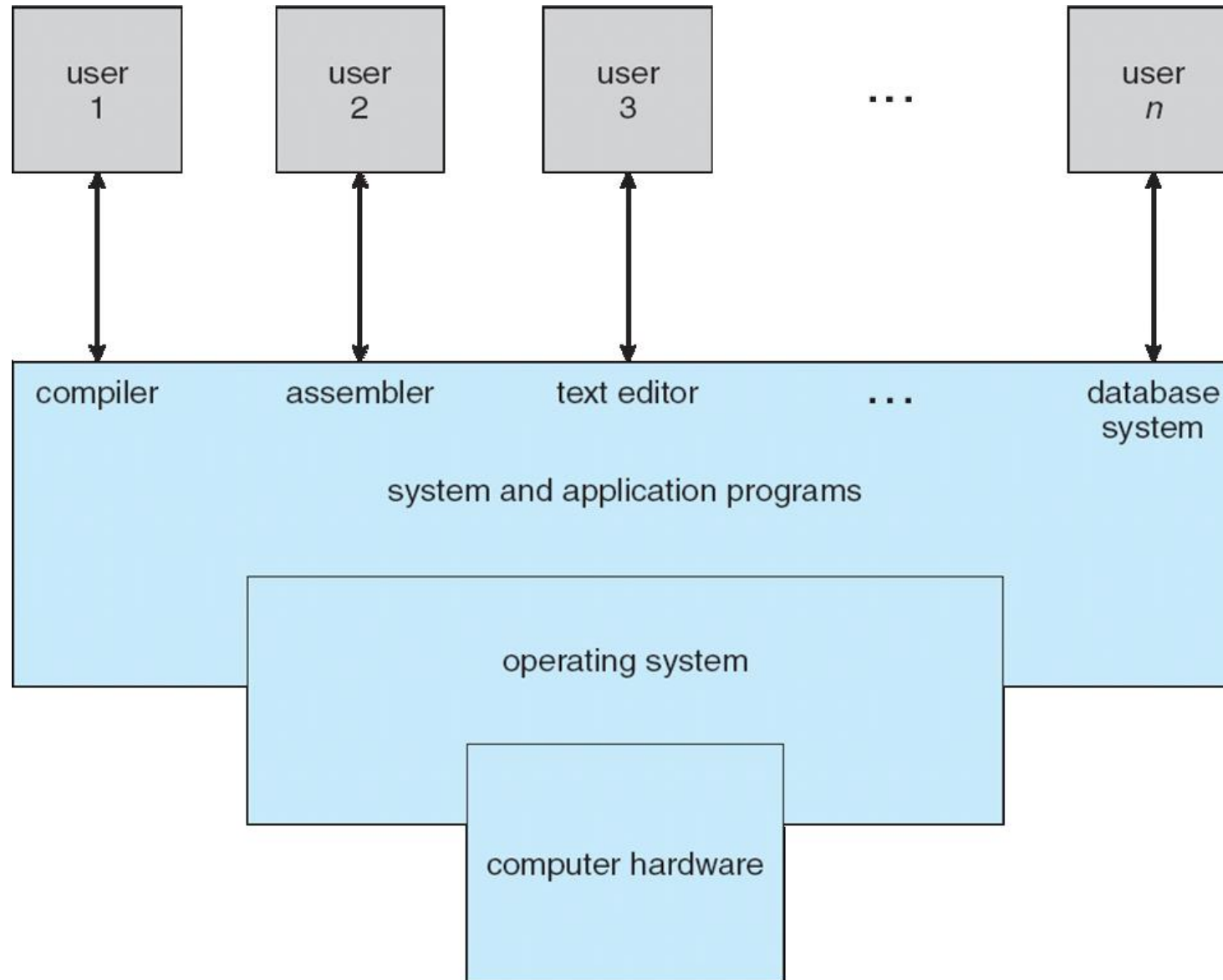
# Bilgisayar Sistemi Yapısı

- Bilgisayar sistemi dört bileşene ayrılabilir:
  - **Donanım (hardware)** – temel bilişim (**computing**) kaynaklarını sağlar
    - İşlemci (**CPU**), bellek, I/O cihazları
  - **İşletim sistemi (operating system)**
    - Donanımın pek çok sayıda uygulama ve kullanıcı arasında paylaşımlı kullanımını koordine eder
  - **Sistem ve Uygulama programları** – kullanıcıların bilişim problemlerini sistem kaynaklarını kullanarak çözmeye yardımcı olan yazılımlardır
    - Kelime işlemciler, derleyiciler (**compilers**), web tarayıcıları, veritabanı sistemleri, oyunlar, vs.
  - **Kullanıcılar**
    - İnsanlar, makineler, diğer bilgisayarlar





# Bilgisayar Sisteminin Dört Bileşeni







# İşletim Sistemi Ne yapar?

- **Bakış açısına bağlıdır!**
- Kullanıcılar **rahatlık, kullanım kolaylığı** ve **iyi performans** ister
  - Kaynak kullanımını umursamazlar
- Ancak **mainframe** veya **mini bilgisayar** gibi paylaşılan bilgisayarlar tüm kullanıcıları mutlu etmelidir

İşletim sistemi → **donanımı verimli bir şekilde kullanan** ve **kullanıcı programlarının sorunsuz yürütülmesini yöneten** bir **kaynak dağıtıcı** ve **kontrol programıdır**





# İşletim Sistemi Ne yapar?

- İş istasyonları gibi özel sistemlerin kullanıcıları **özel kaynaklara** sahiptir, ancak **sunuculardan** paylaşılan kaynakları da kullanırlar
- Akıllı telefonlar ve tabletler gibi mobil cihazlar **kaynak açısından zayıftır**, **kullanılabilirlik** ve **pil ömrü** için optimize edilmişlerdir
  - Dokunmatik ekranlar, ses tanıma gibi mobil kullanıcı arayüzleri
- Gömülü bilgisayarlarda (aygıtlara) **kullanıcı arayüzü** çok kısıtlıdır veya yoktur
  - Kullanıcı müdahalesi olmadan çalışırlar





# İşletim Sistemi Tanımı

---

- İşletim sistemi **kaynak dağıtıcıdır (resource allocator)**
  - Tüm kaynakları yönetir
  - Birbirine aykırı istekler arasında verimli ve adil kullanımı gözeterek karar verir
  
- İşletim sistemi bir **kontrol programıdır (control program)**
  - Programların çalışmasını, hatalara ve uygun olmayan kullanımlara engel olmak için kontrol eder





# İşletim Sistemi Tanımı (Devamı)

- Evrensel kabul gören bir tanım yok
- “İşletim sistemi üreticisinin bir işletim sistemine dahil ettiği herşeydir”  
doğruya yakın bir cevap 😊
  - Fakat büyük oranda değişmekte
- **Çekirdek (kernel):** Bilgisayarda her zaman çalışan tek programdır !
  - Diğer her şey ya sistem programı (işletim sistemi ile birlikte gelir) ya da uygulama programıdır
  - Çekirdek sistem kaynaklarını yönetir
  - Uygulamalar ve donanım seviyesindeki işlemleri arasında bir köprü görevi görür





# Bilgisayarın Başlatılması

---

- Bilgisayar yeniden başlatıldığında ya da açıldığında **önyükleyici program (bootstrap program)** çalıştırılır
  - Tipik olarak ROM veya EPROM'da tutulur ve genellikle **aygıt yazılımı (firmware)** olarak adlandırılır
  - Bilgisayar sistemi tüm yönleri ile başlatır
  - İşletim sistemi çekirdeğini yükler ve çalıştırır





# Bilgisayar Sistem Yapısına Genel Bakış

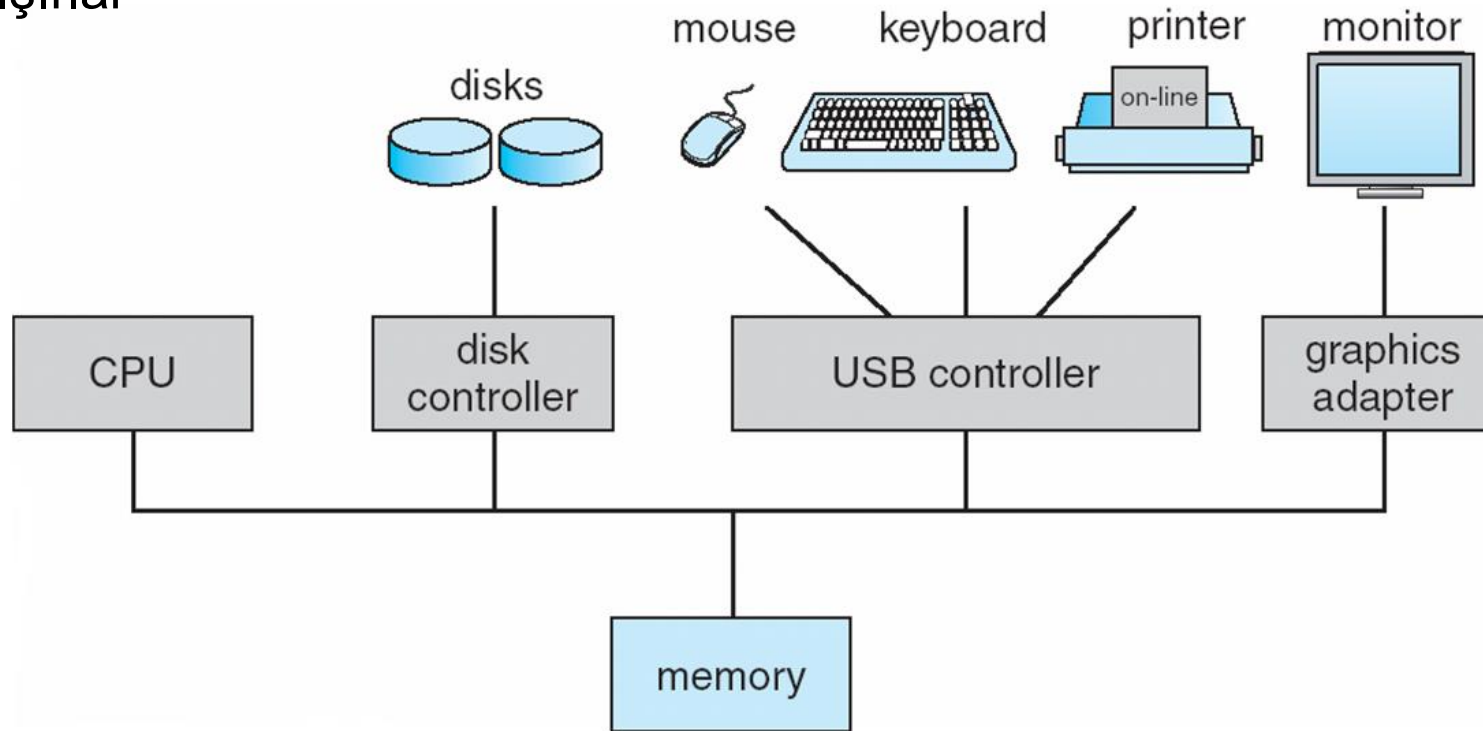




# Bilgisayar Sistemi Organizasyonu

## ■ Bilgisayar sisteminin işleyişi:

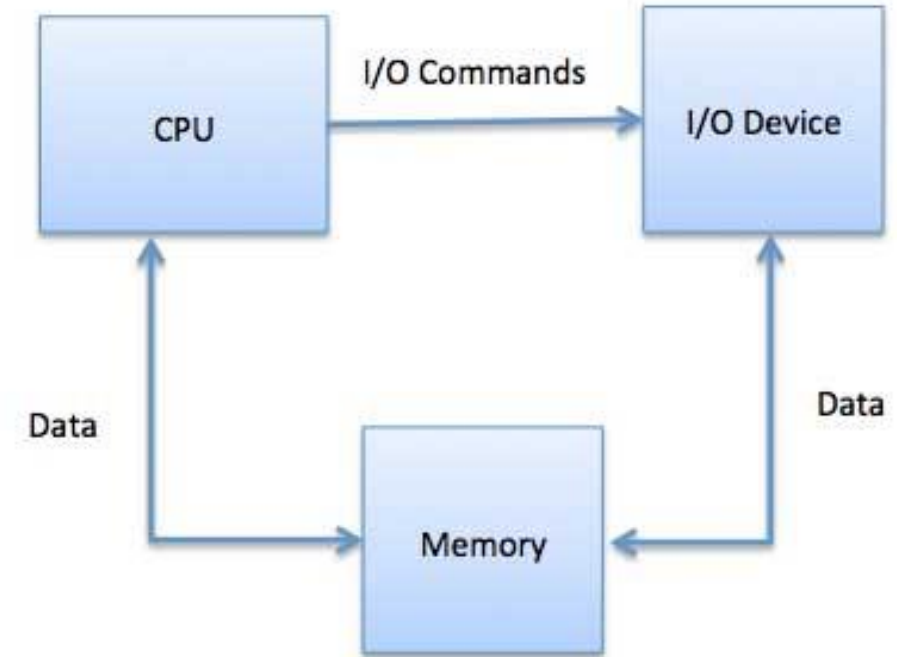
- Bir veya daha fazla **işlemci** (CPU) ve **cihaz denetleyici (device controller)** ortak bir veri yolu (**bus**) üzerinden paylaşılan **belleğe** bağlanır
- Aynı anda çalışan işlemciler ve cihazlar belleğe erişmek için birbirleriyle yarışır





# Bilgisayar Sistemi İşleyişi

- I/O cihazları ve CPU aynı anda çalışabilir
- Her bir cihaz denetleyicisi belli bir tip cihazın kontrolünden sorumludur
- Tüm cihaz denetleyicilerinin **yerel tampon belleği (local buffer)** vardır

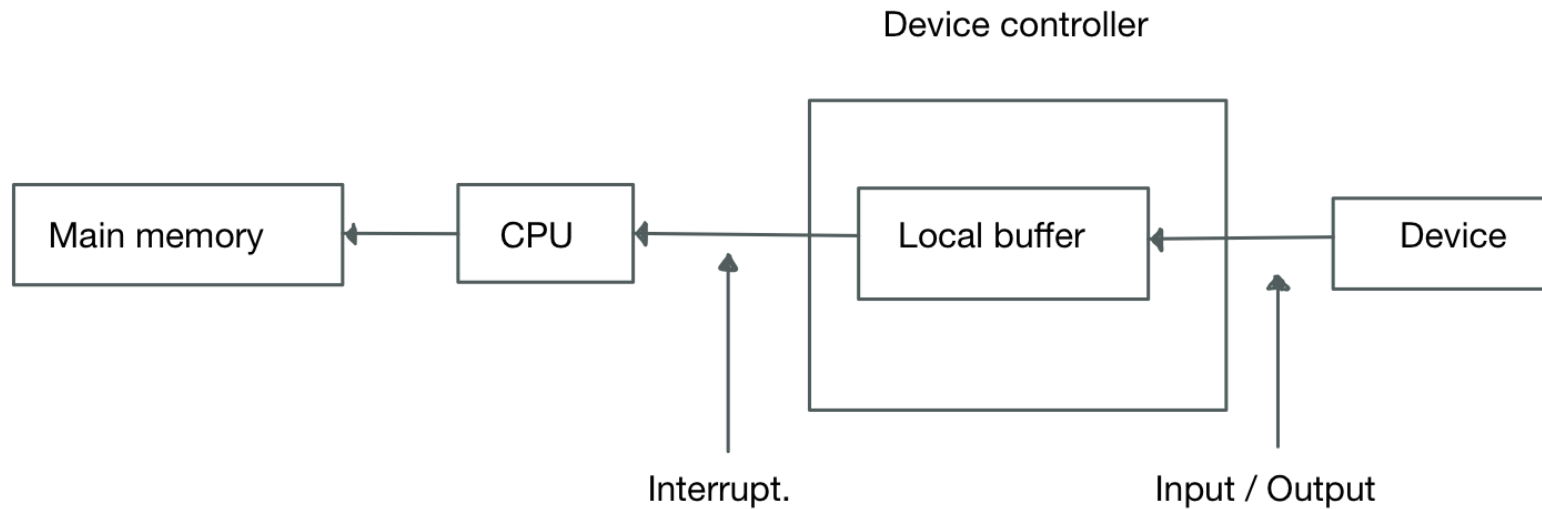






# Bilgisayar Sistemi İşleyişi

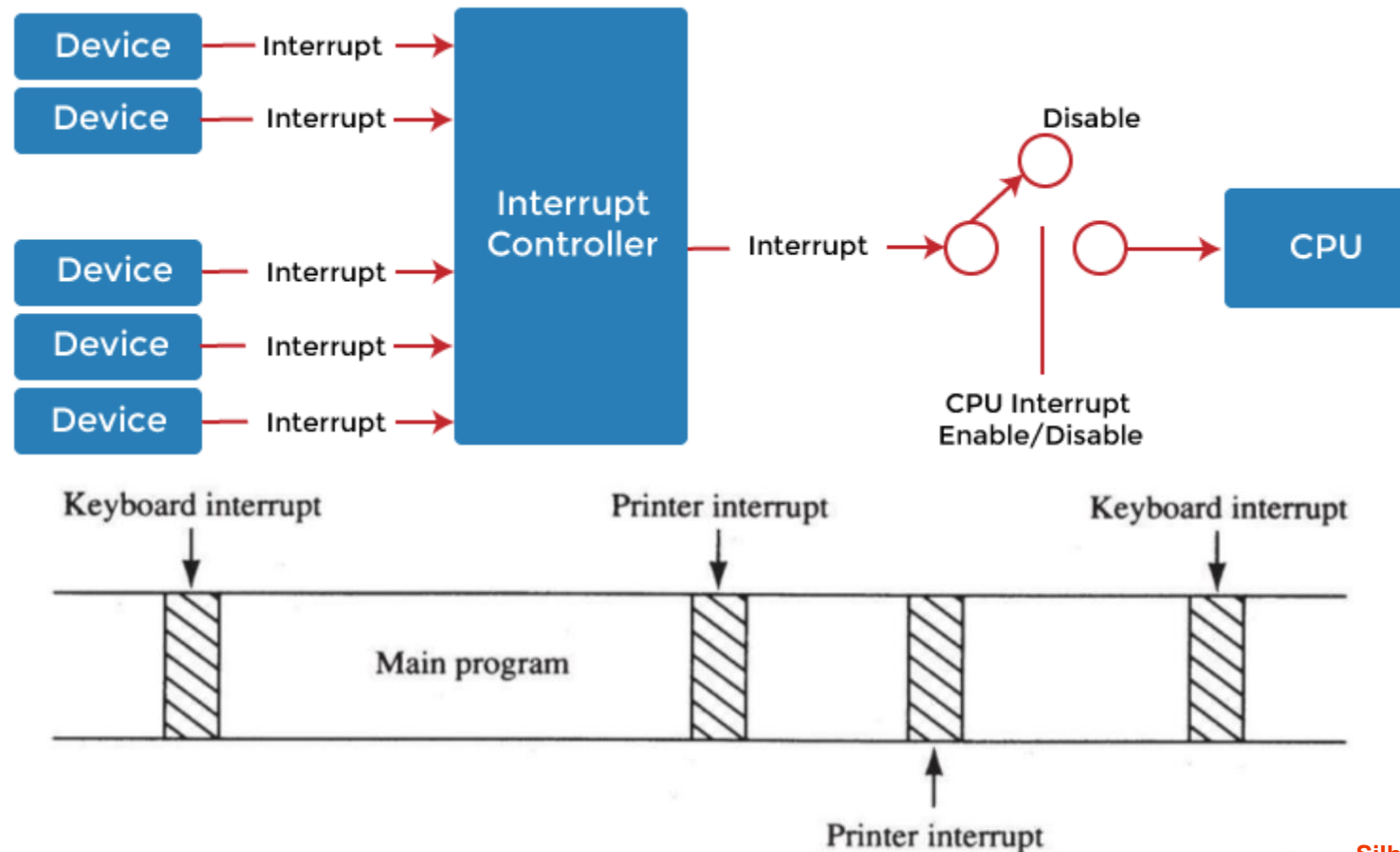
- CPU, ana bellek ile bu yerel tampon bellekler arasında çift yönlü veri taşır
- I/O işlemi cihazdan denetleyicinin yerel tampon belleğine doğrudur
- Cihaz denetleyicisi, işleminin bittiğini, işlemciye **kesme (interrupt)** göndererek bildirir





# Kesmelerin Genel Özellikleri

- **Kesme kontrolörü**, program denetimini o kesmeye ait **kesme hizmet programı (interrupt service routine-ISR)** yönlendirir
- **ISR**'ler kesme sonucu yapılması gereken işi gerçekleştiren kod parçalarıdır





# Kesmelerin Genel Özellikleri

- Hangi hizmet programının hangi bellek adresinde yer aldığı **kesme vektöründe (interrupt vector)** bulunmaktadır

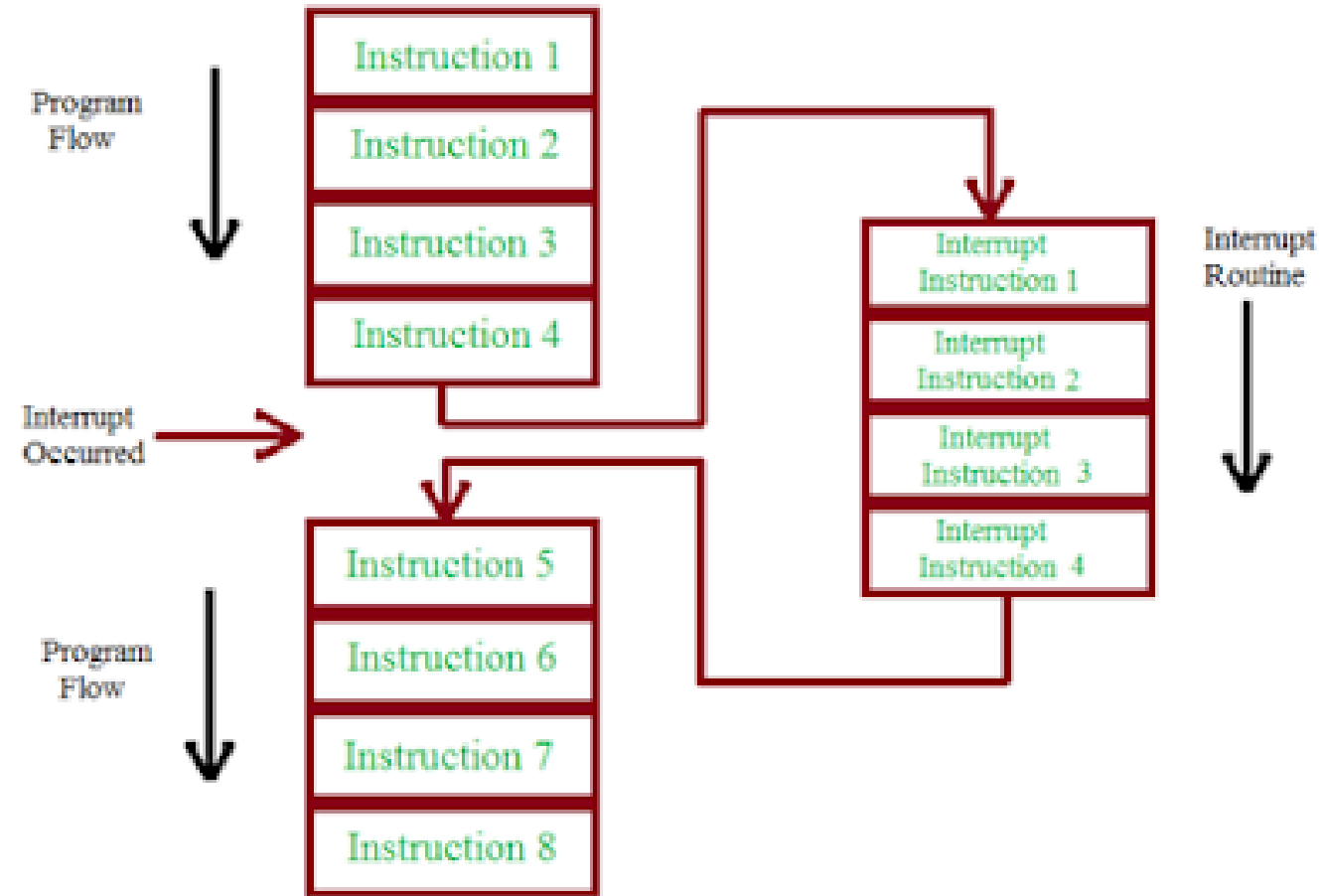
Vectors Numbers		Address		Space <sup>6</sup>	Assignment
Hex	Decimal	Dec	Hex		
0	0	0	000	SP	Reset: Initial SSP <sup>2</sup>
1	1	4	004	SP	Reset: Initial PC <sup>2</sup>
2	2	8	008	SD	Bus Error
3	3	12	00C	SD	Address Error
4	4	16	010	SD	Illegal Instruction
5	5	20	014	SD	Zero Divide
6	6	24	018	SD	CHK Instruction
7	7	28	01C	SD	TRAPV Instruction
8	8	32	020	SD	Privilege Violation
9	9	36	024	SD	Trace
A	10	40	028	SD	Line 1010 Emulator
B	11	44	02C	SD	Line 1111 Emulator
C	12 <sup>1</sup>	48	030	SD	(Unassigned, Reserved)
D	13 <sup>1</sup>	52	034	SD	(Unassigned, Reserved)
E	14	56	038	SD	Format Error <sup>5</sup>
F	15	60	03C	SD	Uninitialized Interrupt Vector
10–17	16–23 <sup>1</sup>	64	040	SD	(Unassigned, Reserved)
		92	05C		—
18	24	96	060	SD	Spurious Interrupt <sup>3</sup>
19	25	100	064	SD	Level 1 Interrupt Autovector
1A	26	104	068	SD	Level 2 Interrupt Autovector
1B	27	108	06C	SD	Level 3 Interrupt Autovector
1C	28	112	070	SD	Level 4 Interrupt Autovector
1D	29	116	074	SD	Level 5 Interrupt Autovector
1E	30	120	078	SD	Level 6 Interrupt Autovector
1F	31	124	07C	SD	Level 7 Interrupt Autovector
20–2F	32–47	128	080	SD	TRAP Instruction Vectors <sup>4</sup>
		188	0BC		—
30–3F	48–63 <sup>1</sup>	192	0C0	SD	(Unassigned, Reserved)
		255	0FF		—
40–FF	64–255	256	100	SD	User Interrupt Vectors
		1020	3FC		—





# Kesmelerin Genel Özellikleri

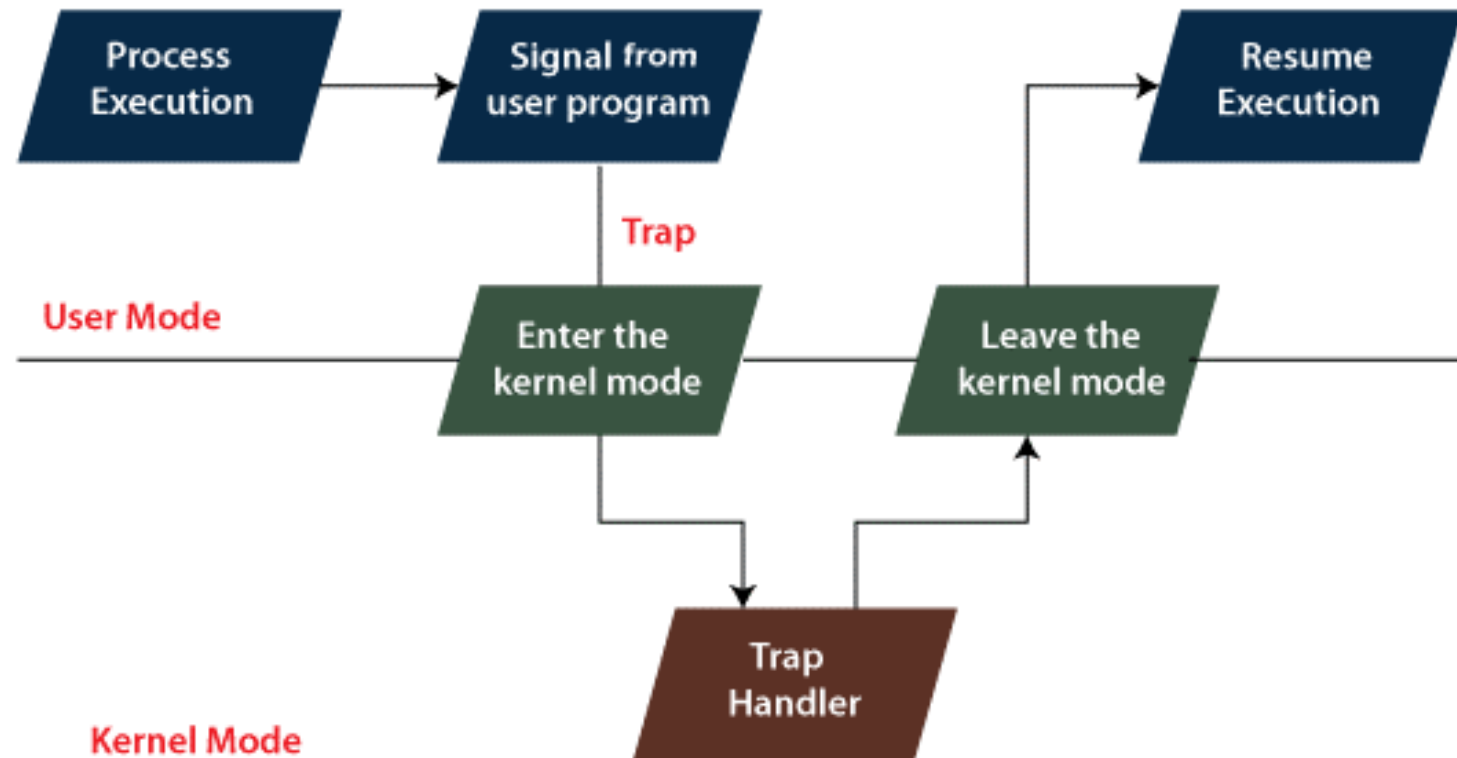
- Bilgisayar, kesme sonunda yarıda kesilen prosese geri dönebilmek için, kesilen prosesin işletilen son komutunun adresini saklamalıdır
- Kayıp kesmelere engel olmak için, kesmeye ilişkin kod çalıştığı sürece yeni kesme gönderimine izin verilmez (gelen kesmeler kuyruğa alınır)





# Kesmelerin Genel Özellikleri

- **Tuzak (trap)** yazılım tarafından oluşturulan kesmelerdir
- Tuzaklara **yazılım hataları** ya da **kullanıcı istekleri** neden olur
- İşletim sistemleri kesmelerle yönlendirilirler (**interrupt driven**)





# Kesmelerin İşletilmesi

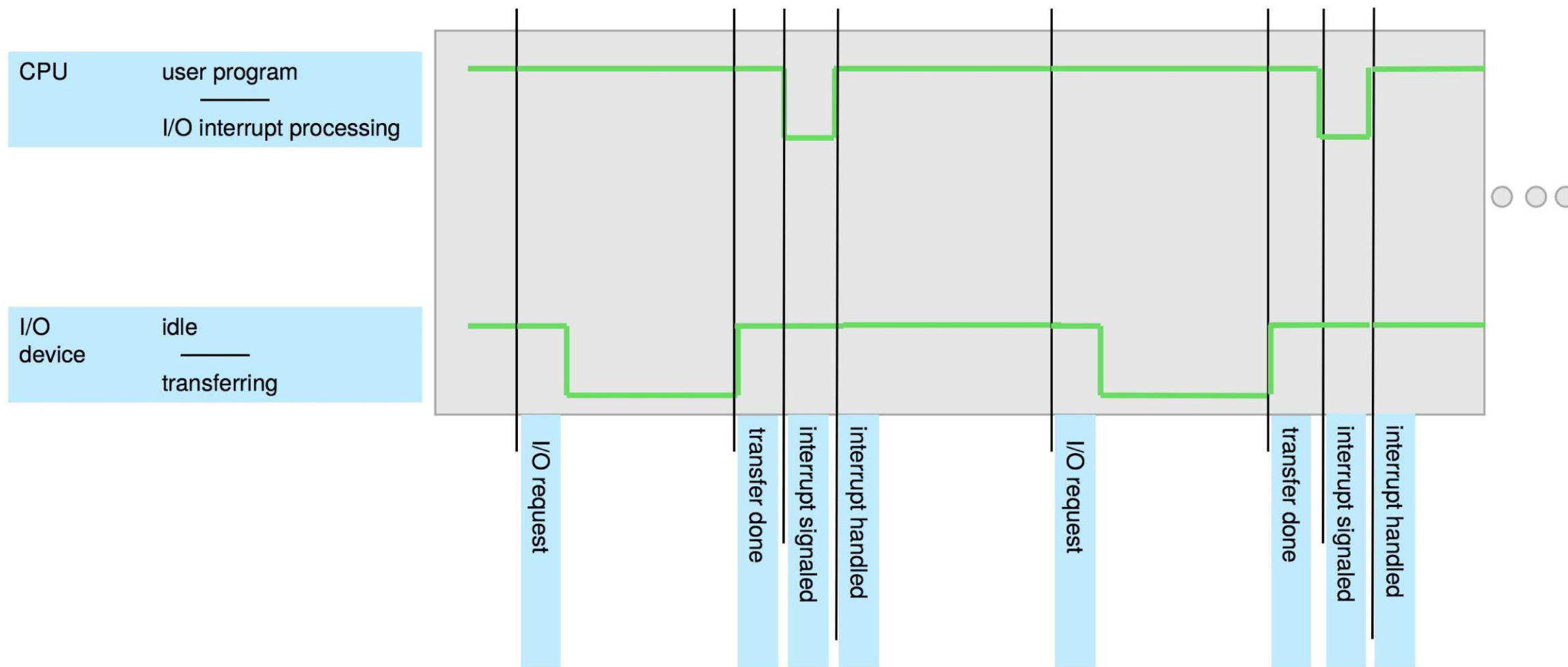
Bir kesme isteği geldiğinde;

- İşletim sistemi CPU'nun durumunu kaydeder: **saklayıcılar (registers)** ve **program sayacı (program counter)**
- Hangi tür kesmenin gerçekleştiğini belirler:
  - **sorgulama (polling)** – hangi cihazdan gerçekleştiği bulunmalıdır
  - **vektörlü kesme sistemi (vectored interrupt system)** – cihazı belirten kod, kesme ile birlikte gönderilir
- Her bir kesme için hangi işlemin gerçekleştirileceğini ayrı bir kod parçası belirler





# Kesme Zaman Çizelgesi





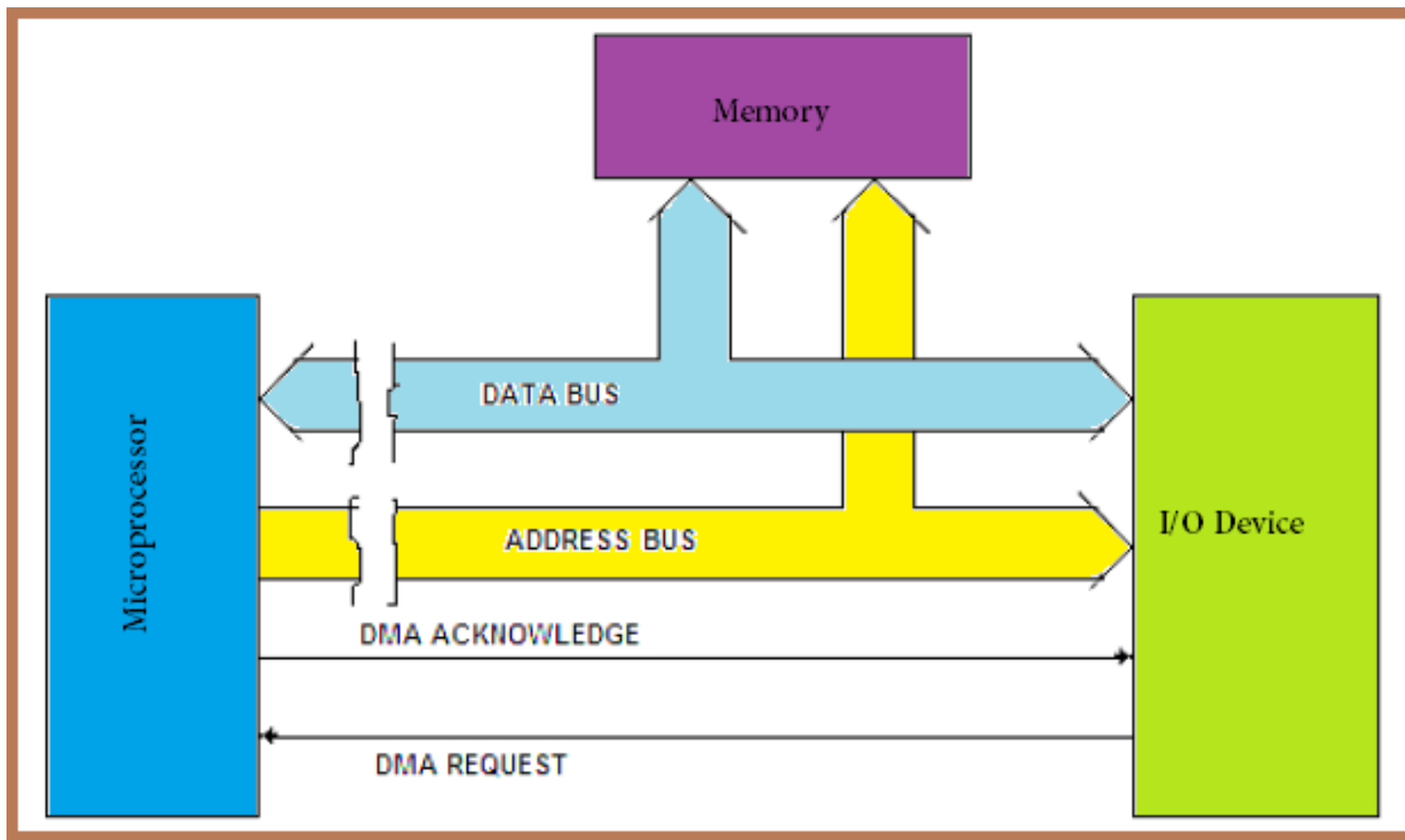
# Depolama Yapısı







# Bellek Eriřim Yapısı





# Doğrudan Bellek Erişim Yapısı

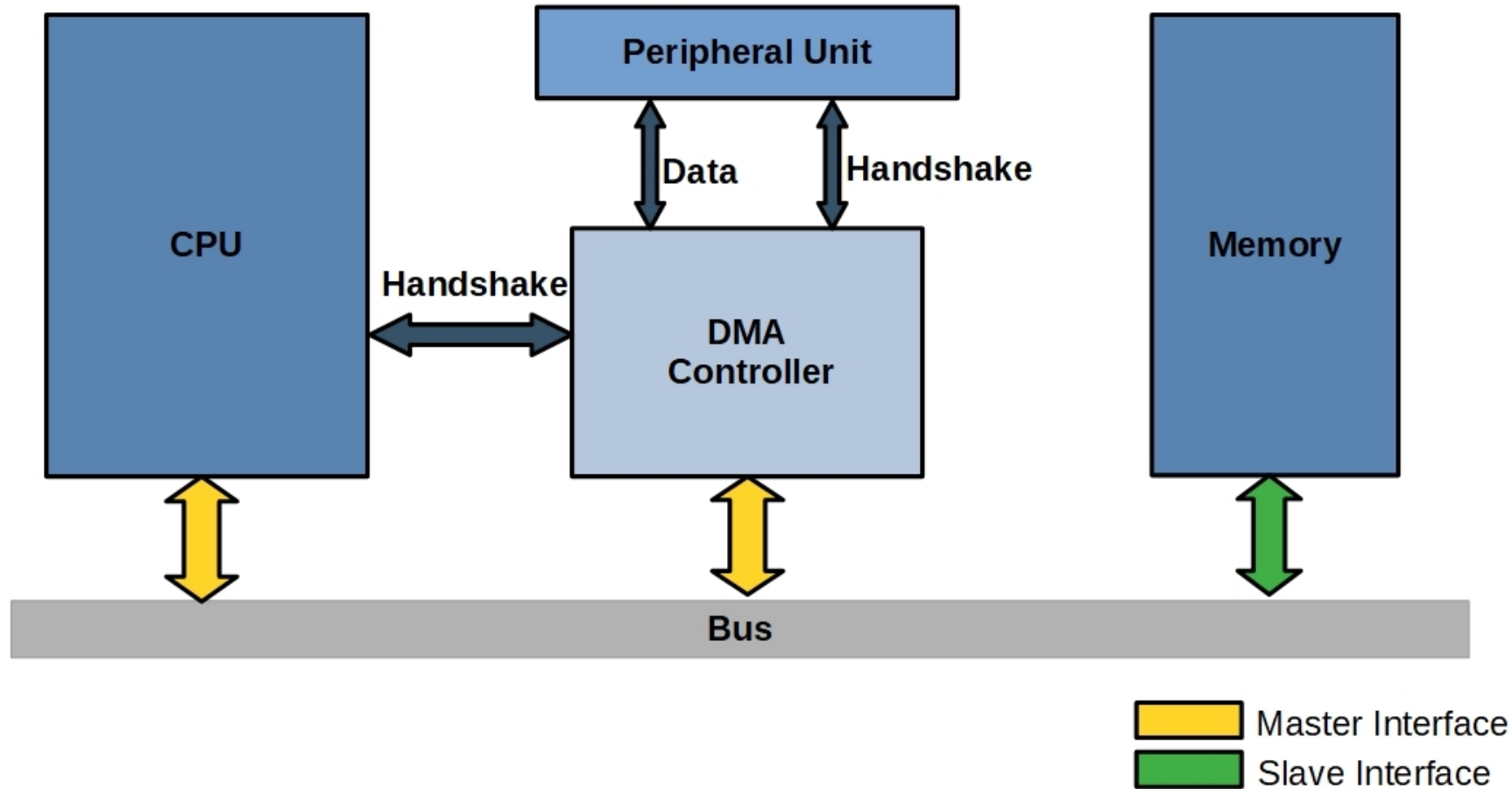
---

- **Doğrudan Bellek Erişimi – Direct Memory Access (DMA)**
- Cihaz denetleyicisinin, CPU'nun çalışmasını bölmeden, veri bloklarını cihazın tampon belleğinden doğrudan ana belleğe aktarması işlemidir
- Her byte için kesinti göndermek yerine, her bir blok için bir kesinti gönderilir
- Bellek hızına yakın bilgi aktarması yapabilen yüksek hızlı I/O cihazları için kullanılır





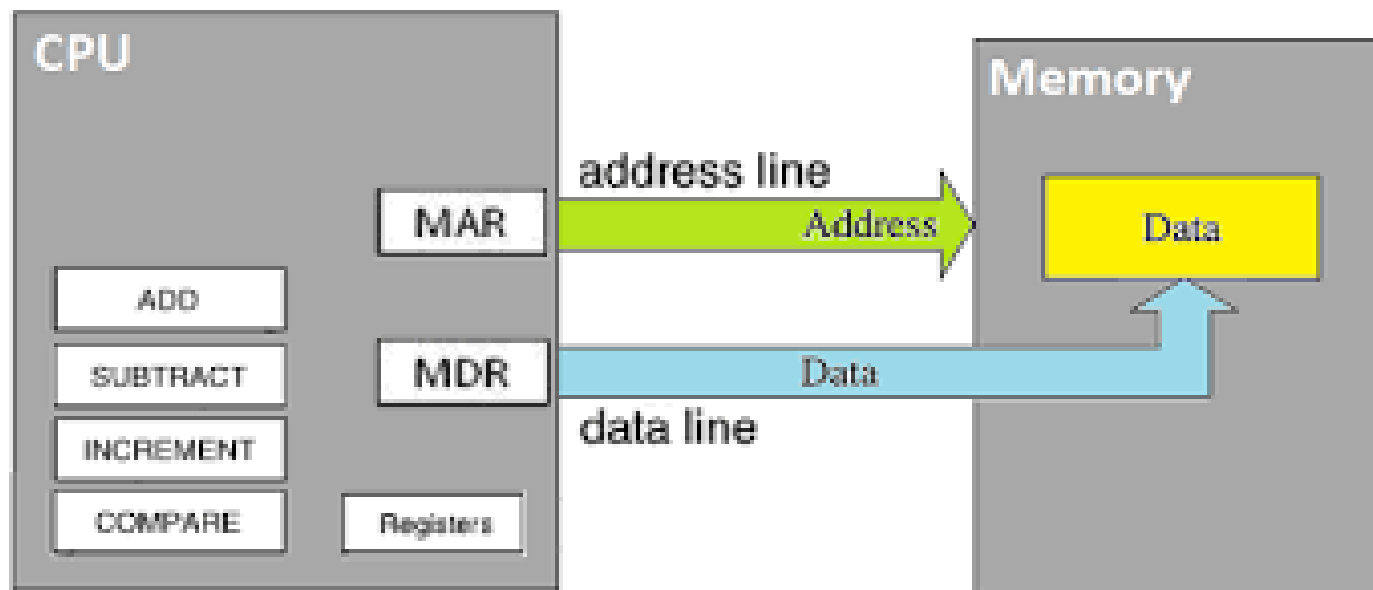
# Doğrudan Bellek Erişim Yapısı





# Depolama Birimi Yapısı

- **Ana bellek (main memory)** – CPU'nun direk erişebileceği tek geniş depolama birimidir
  - Random access
  - Typically **volatile**
  - Dynamic Random-access Memory (DRAM)





# Depolama Birimi Yapısı

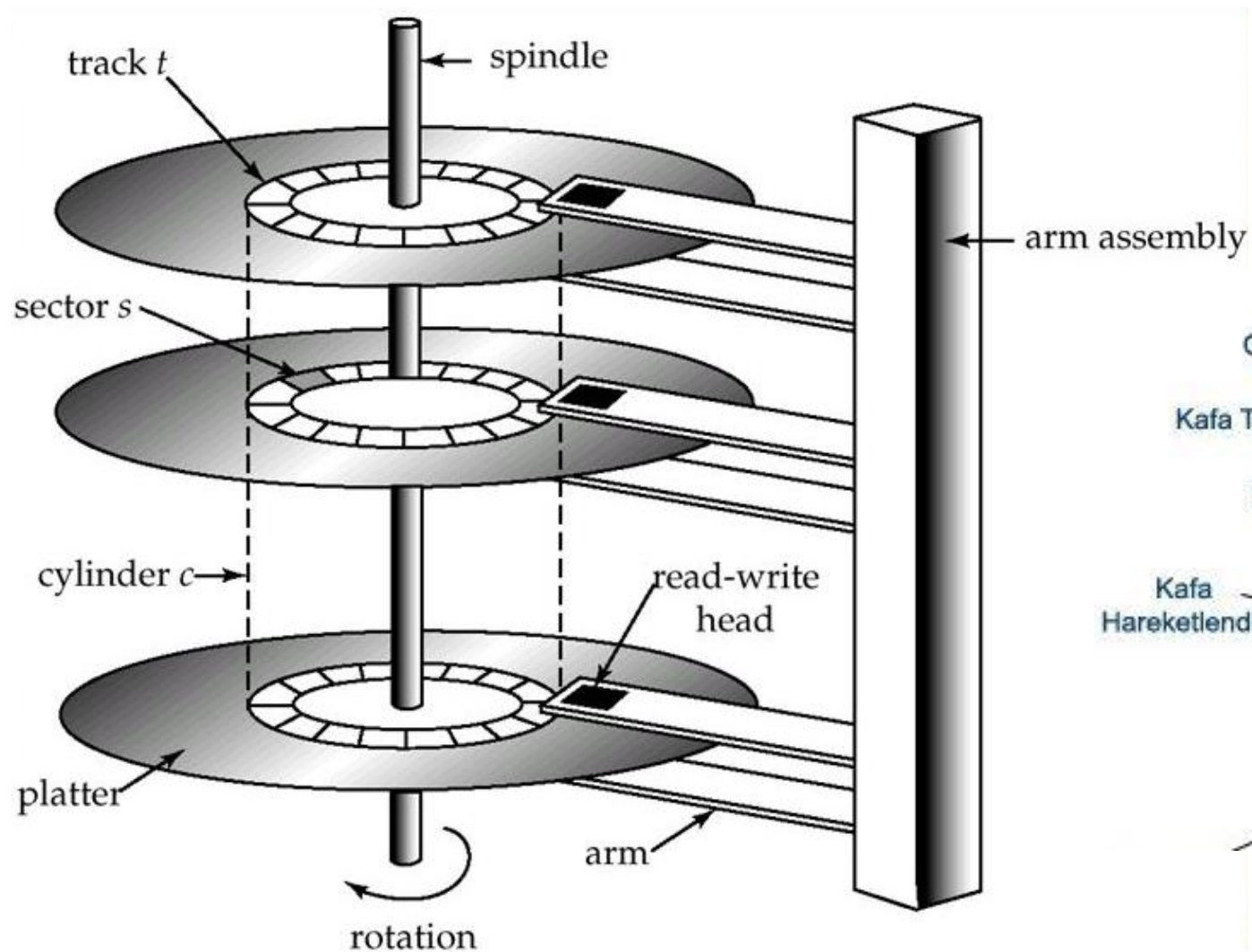
---

- **İkincil depolama birimi (secondary storage)** – kalıcı bir şekilde bilgilerin depolandığı, ana belleğin uzantısı olan depolama birimidir
  - **Manyetik diskler (magnetic disks)** – yüzeyi manyetik kayıt malzemesiyle kaplı sert metal veya cam tabakalar
  - Disk yüzeyi genellikle mantıksal olarak **izlere (tracks)** bölünür
  - Her bir iz **sektörlere (sectors)** bölünür
  - **Disk denetleyicisi (disk controller)** bilgisayar ile disk arasındaki mantıksal etkileşimi sağlar





# Manyetik Disk Yapısı

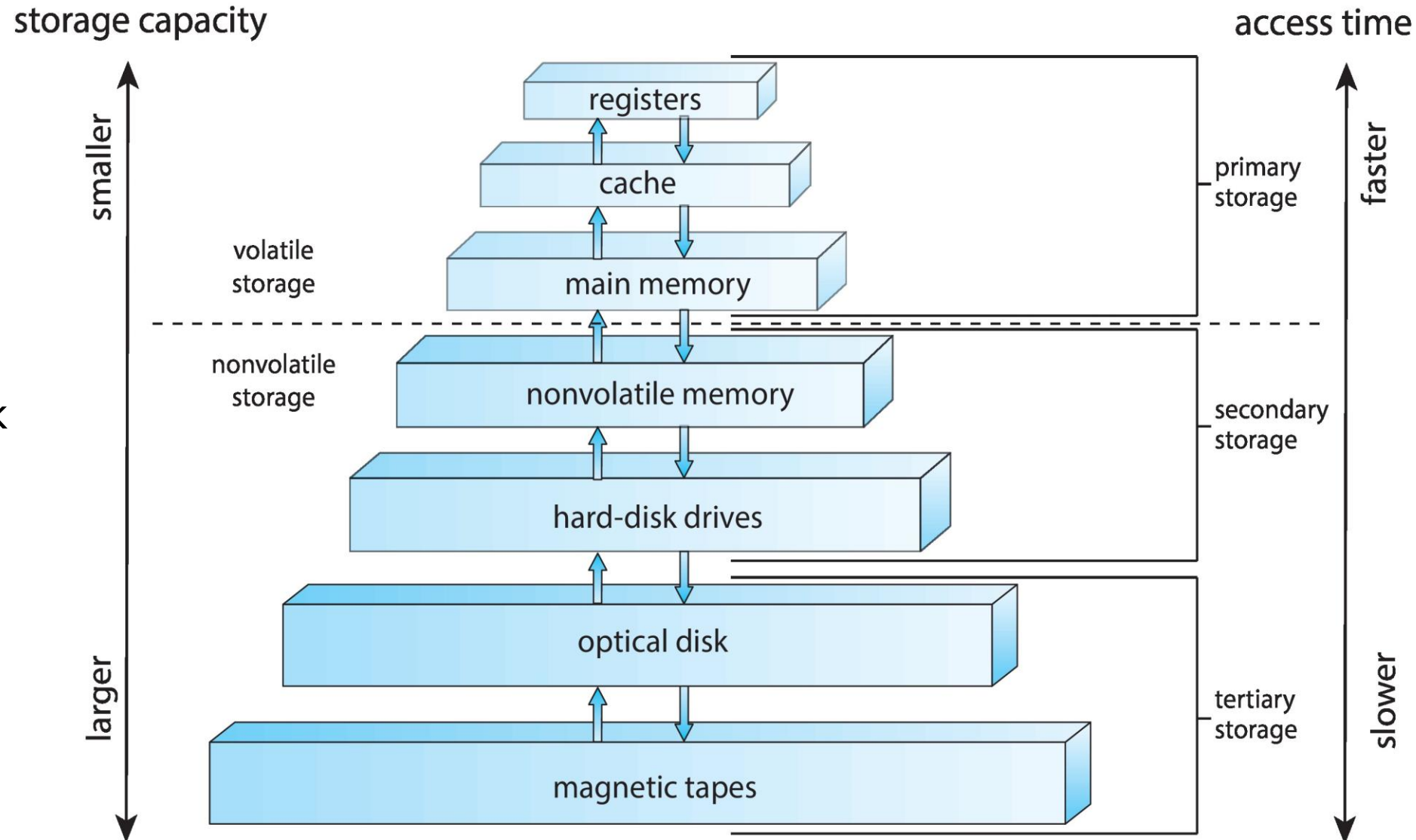




# Depolama Aygıtı Hiyerarşisi

Depolama birimleri hiyerarşik bir şekilde organize edilirler

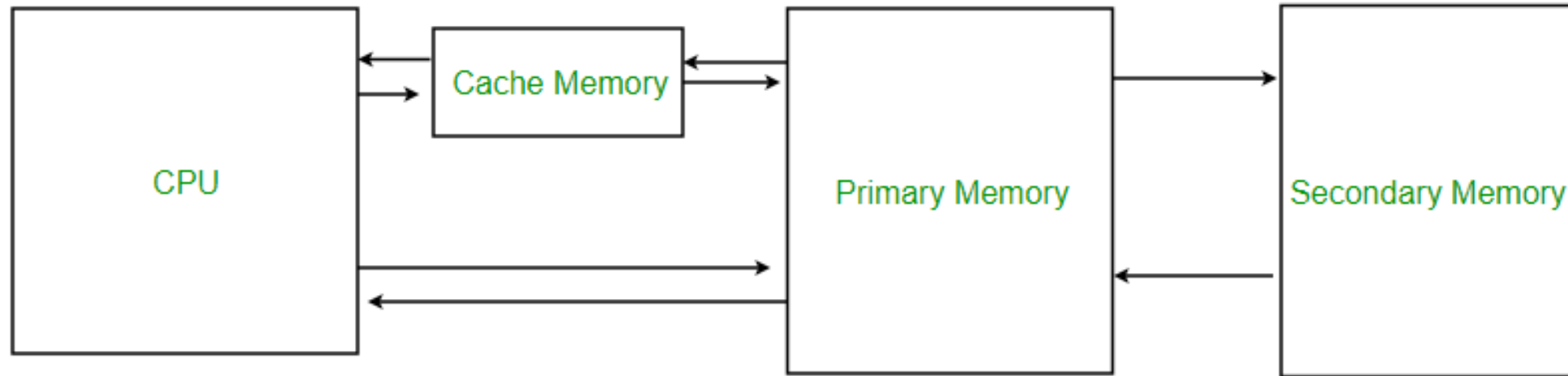
- Hız (Speed)
- Maliyet (Cost)
- Gelgeçlik-uçuculuk (volatility)





# Ön Belleğe Alma (Caching)

- Ön belleğe alma (**Caching**) işlemi, bilgisayarda pek çok seviyede (donanım, işletim sistemi, yazılım) gerçekleştirilen önemli bir prensiptir
- Kullanılan bilgi yavaş depolama biriminden hızlı olana kopyalanır



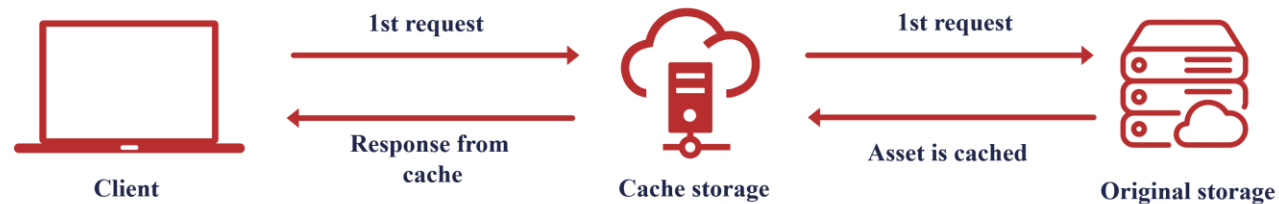




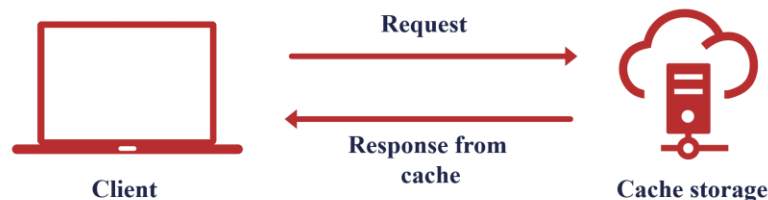
# Ön Belleğe Alma

- Aranan bilgi öncelikle daha hızlı depolama biriminde mi (cache)? kontrol edilir
  - Eğer oradaysa, bilgi direkt ön bellekten alınır (hızlı)
  - Eğer değilse, ön belleğe alınır ve oradan kullanılır
- **Ön bellek**, ön belleğe alınacak bilgiden daha küçük olabilir
  - **Ön bellek yönetimi (cache management)** önemli bir tasarım problemidir
  - Ön bellek boyutu ve **yenileme politikası (replacement policy)**

First request



Subsequent requests





# Bilgisayar Sistem Mimarisi





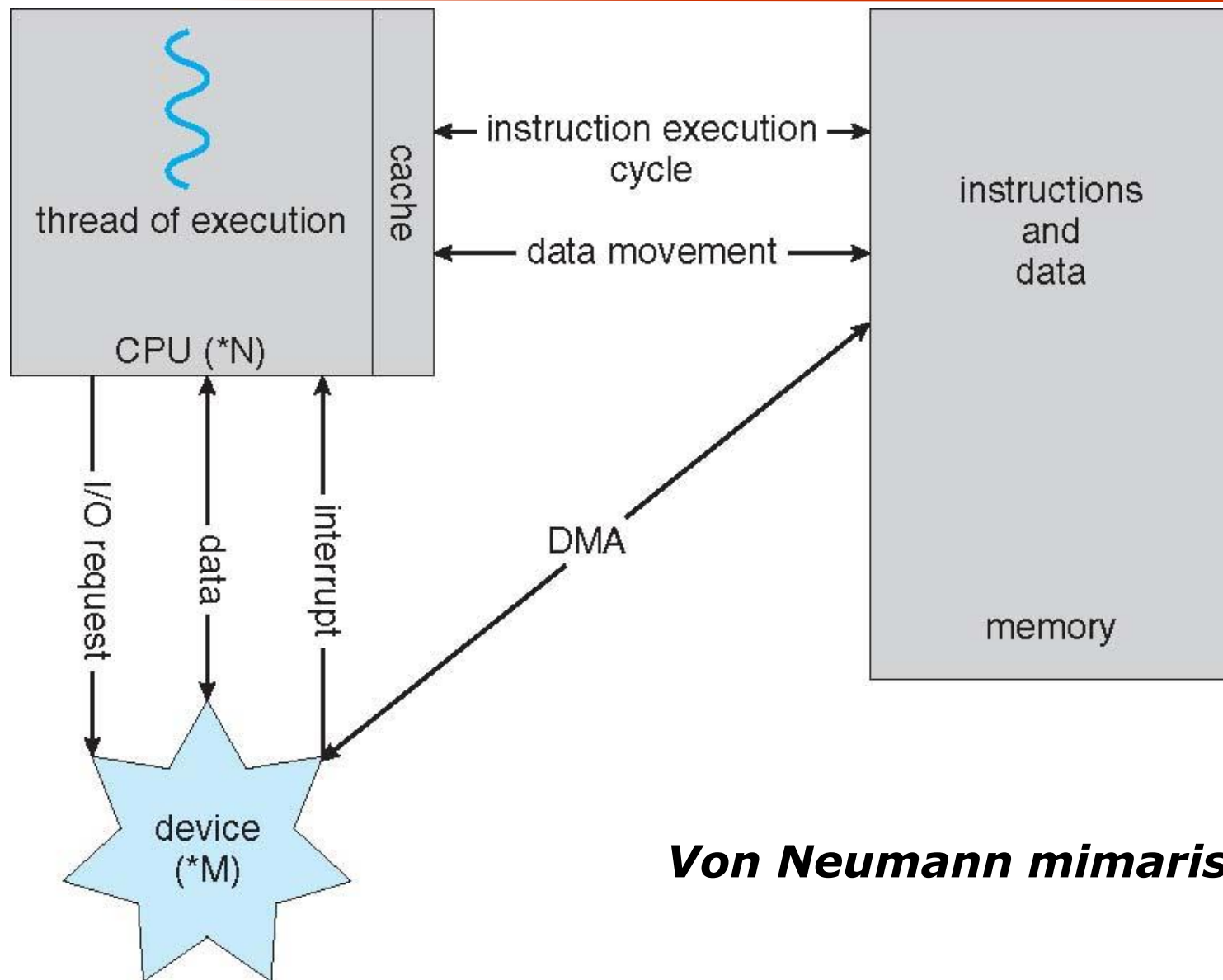
# Tekli veya Çoklu İşlemciler

- Pek çok sistem tek bir **genel amaçlı işlemci** kullanır (örn: PC, sunucu).
  - Aynı zamanda, pek çok sistem de **özel amaçlı işlemciler** kullanır (örn: gömülü sistemler)
- **Çok işlemcili sistemler (multiprocessors systems)** giderek yaygınlaşmakta ve önem kazanmaktadır
  - **Paralel sistemler (parallel systems)** ve **sıkı bağlı sistemler (tightly-coupled systems)** olarak da bilinirler
  - Avantajlar
    1. Artan **üretilebilirlik (throughput)**
    2. Ekonomik olarak ölçeklenebilirlik (**economy of scale**)
    3. Artan **güvenilirlik (reliability)** – alt parçalara bölme veya **hata toleransı (fault tolerance)**





# Modern Bilgisayarlar Nasıl Çalışır?

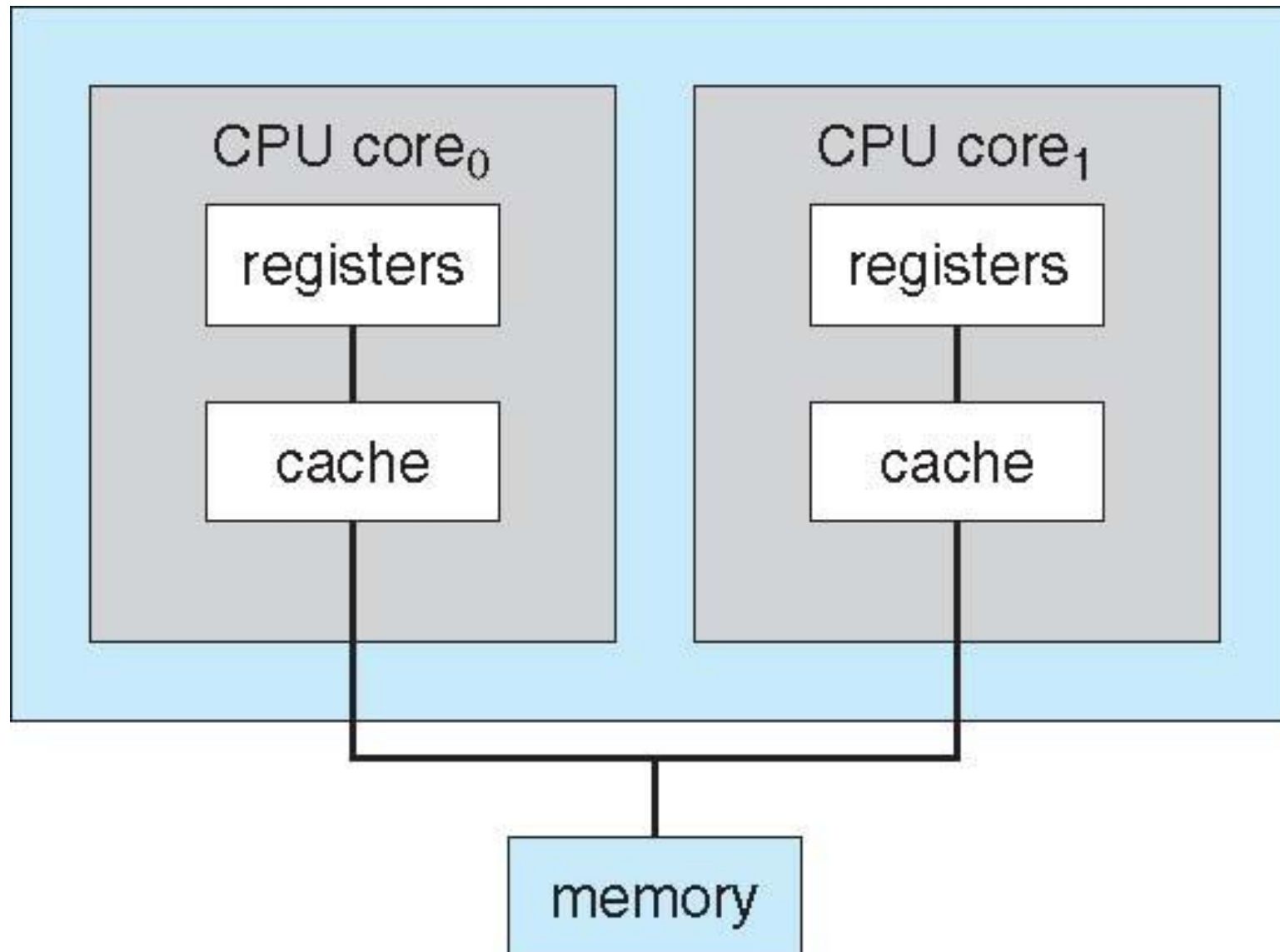


***Von Neumann mimarisi***



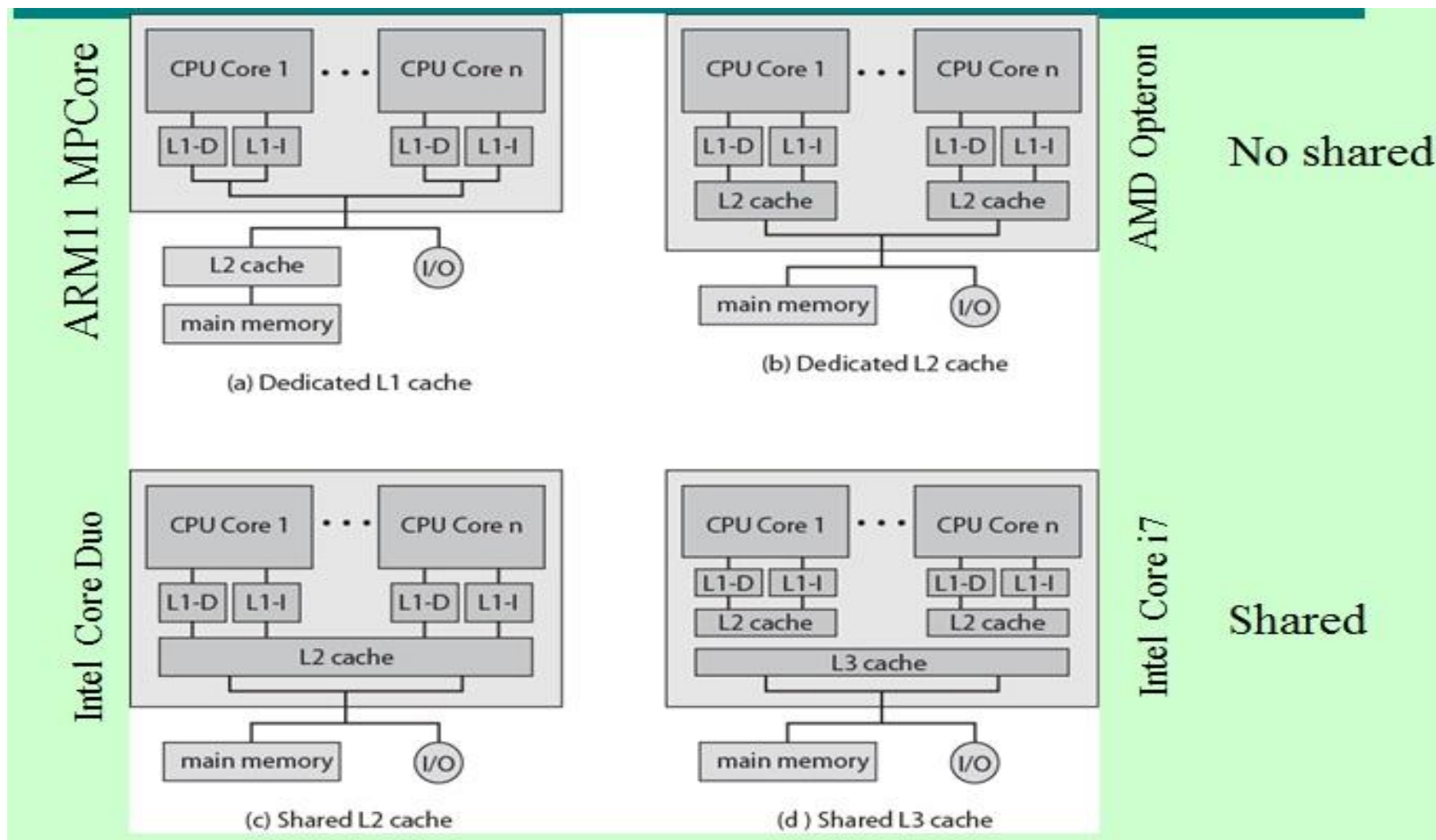


# Çok Çekirdekli Tasarımlar





# Çok Çekirdekli Tasarımlar

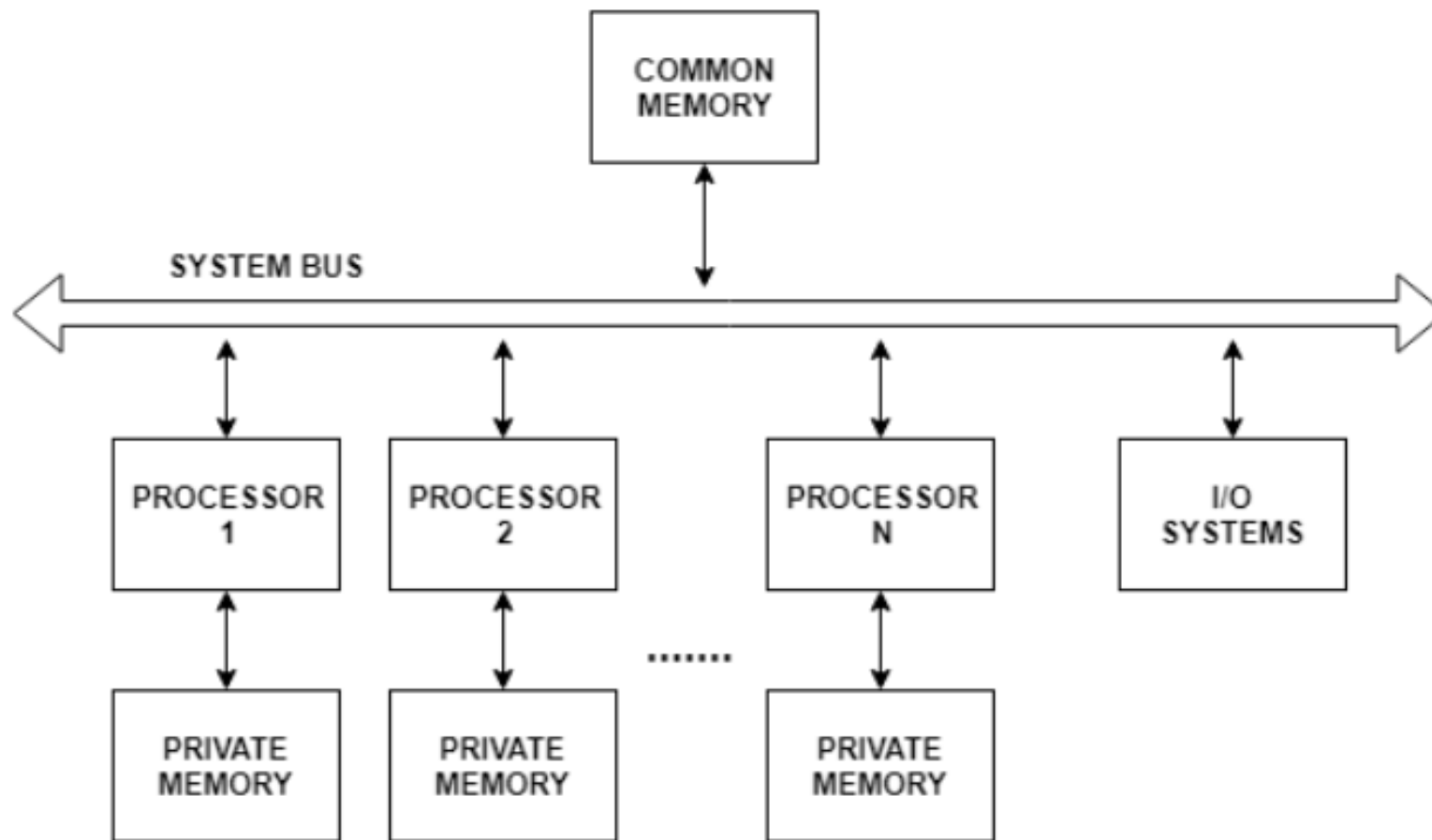


Avantajlar/Dezavantajlar?





# Çok İşlemcili Sistemler





# Çok İşlemcili Sistemler

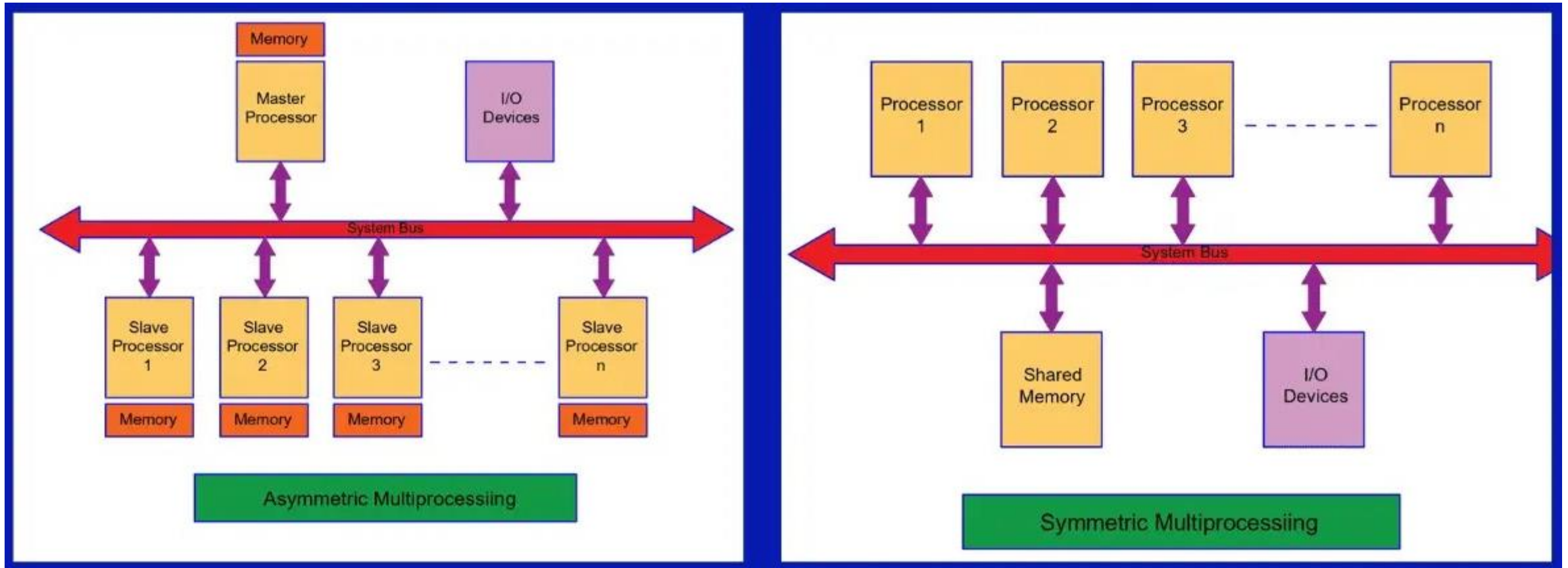
- İki farklı tür:
  1. **Asimetrik Çoklu İşlemciler (Asymmetric Multiprocessing)**
  2. **Simetrik Çoklu İşlemciler (Symmetric Multiprocessing)**
- Asimetrik çoklu işlemciler – Görev dağıtan bir işlemci var, diğerleri görev bekliyor (**master-slave**)
- Asimetrik çoklu işlemciler özellikle ilk zamanlarda kullanılıyordu
- Simetrik Çoklu İşlemciler (SMP) – tüm işlemciler her tür işi yapıyor





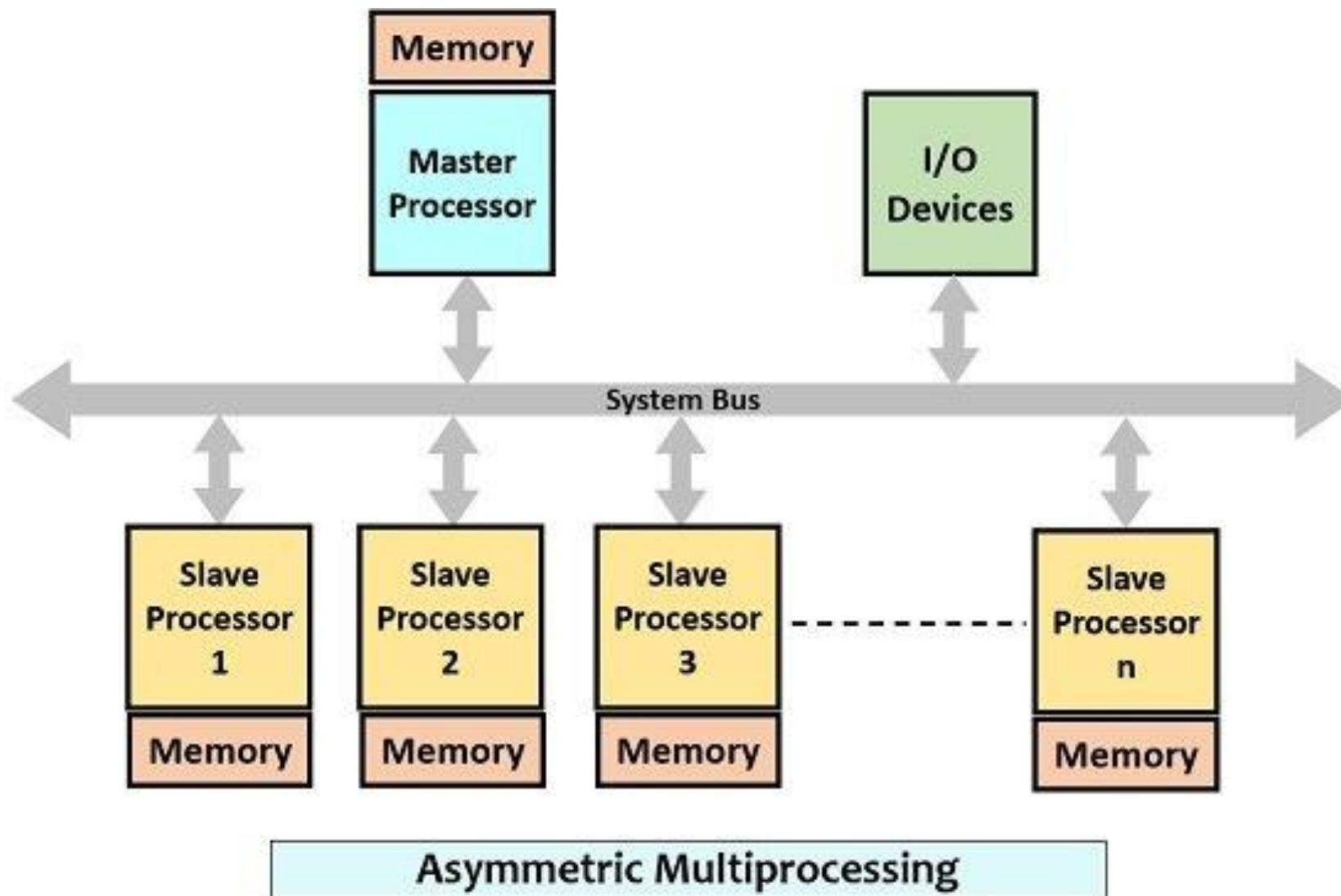


# Çok İşlemcili Sistemler



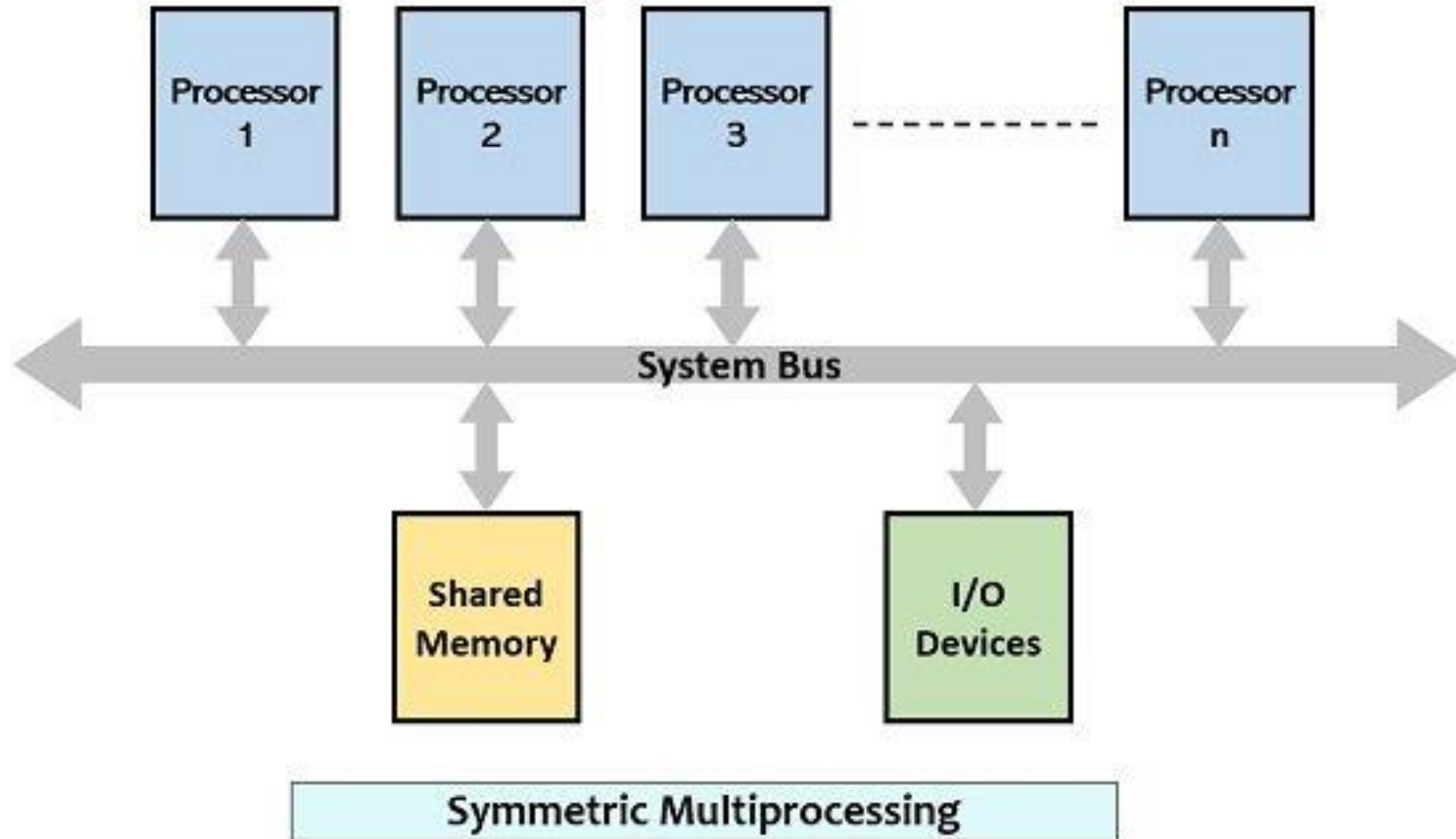


# Çok İşlemcili Sistemler



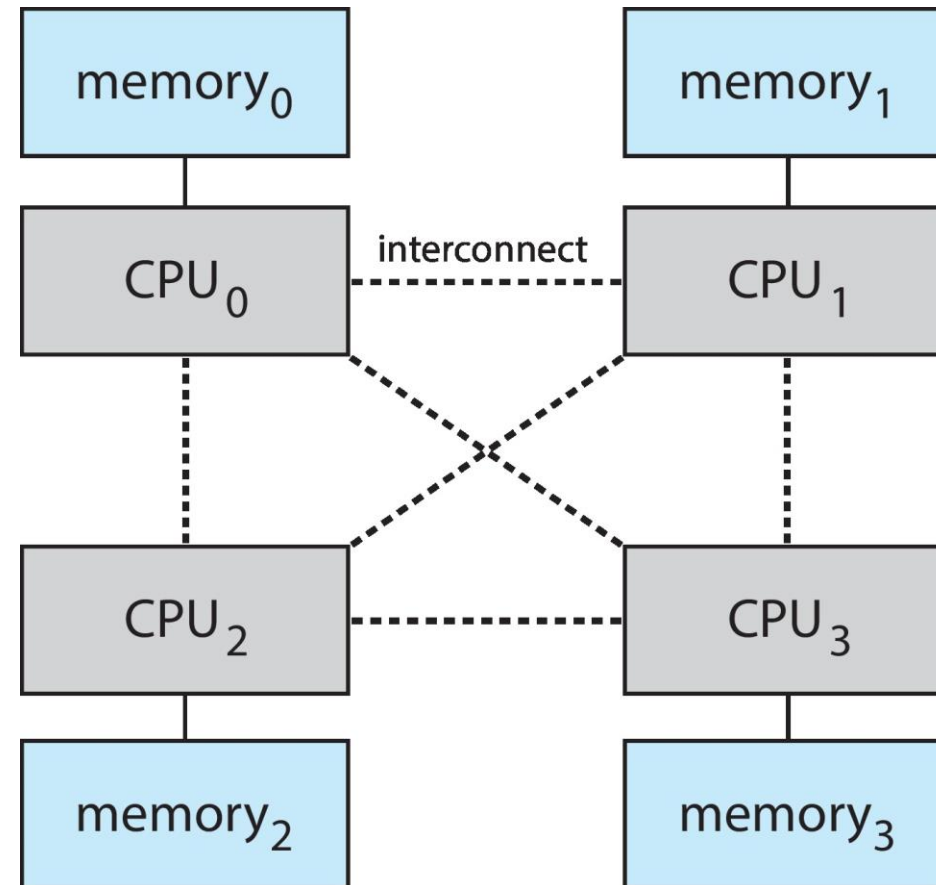


# Çok İşlemcili Sistemler



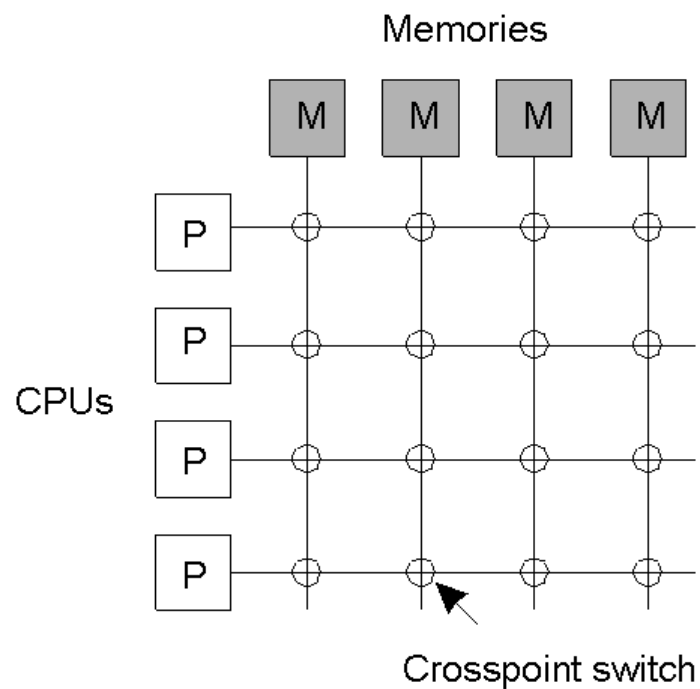


# Non-Uniform Bellek Erişimli Sistemler



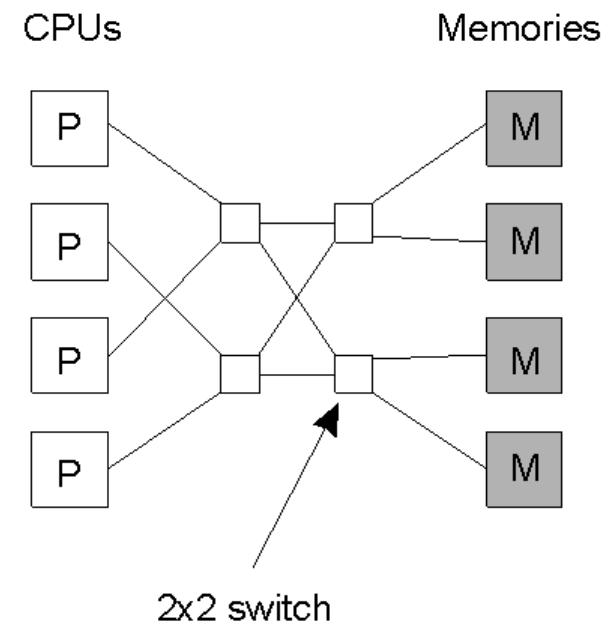


# Uniform Bellek Erişimli Sistemler



(a)

a) A crossbar switch



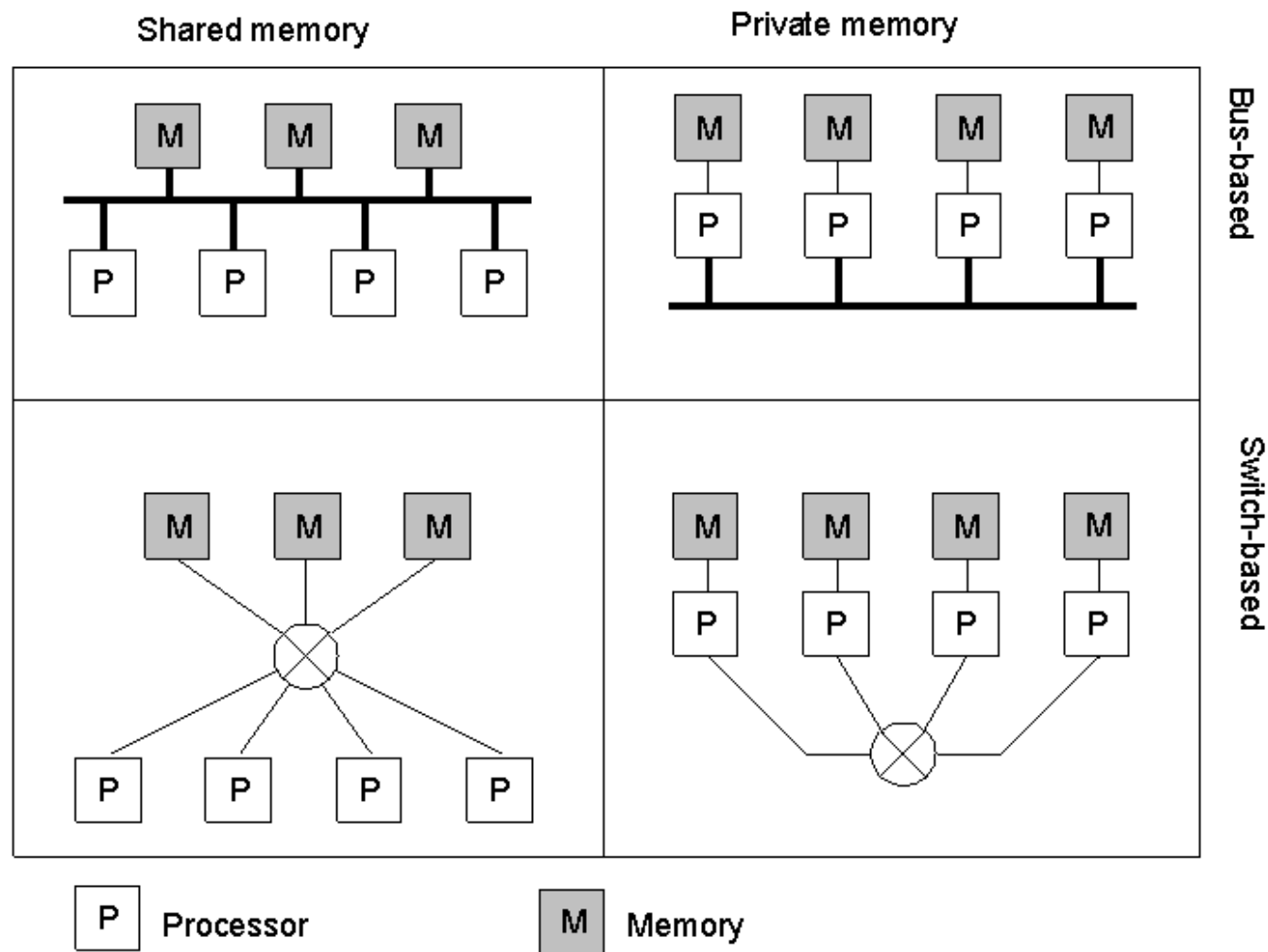
(b)

b) An omega switching network



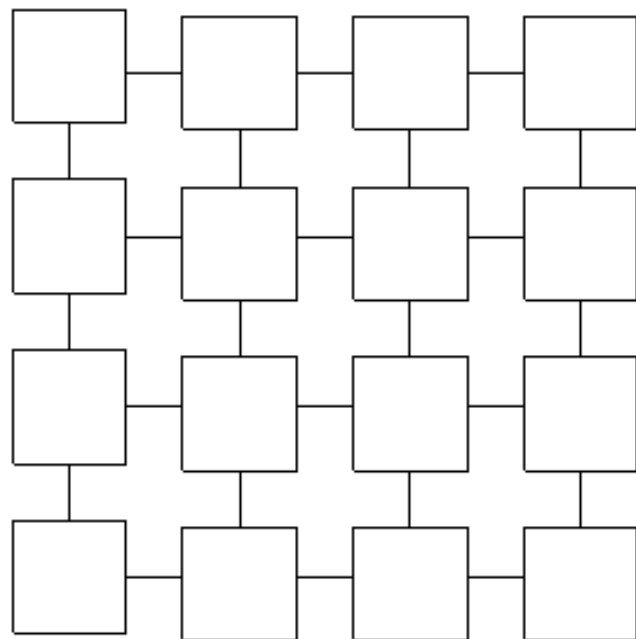


# Multiprocessors ve Multicomputers

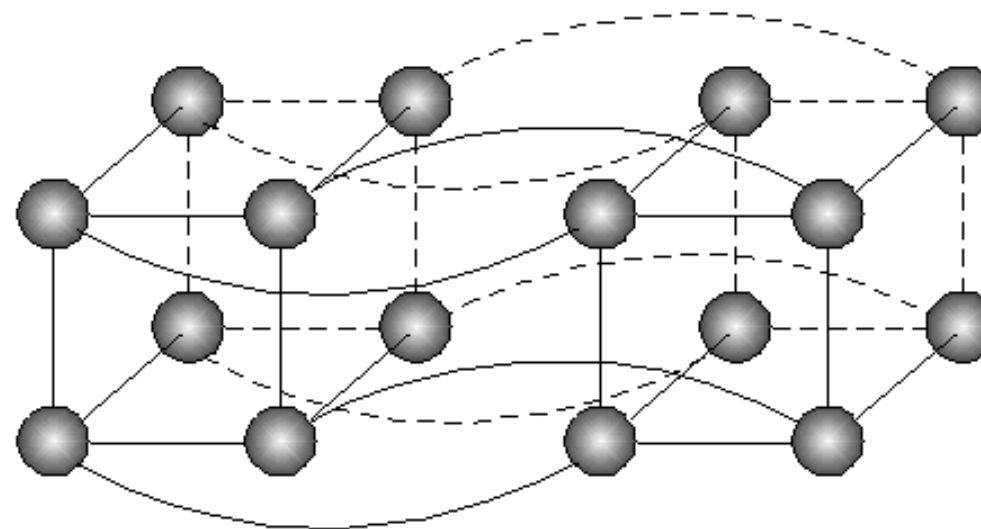




# Homojen Multicomputers Mimarisi



(a)



(b)

- a) Grid
- b) Hypercube

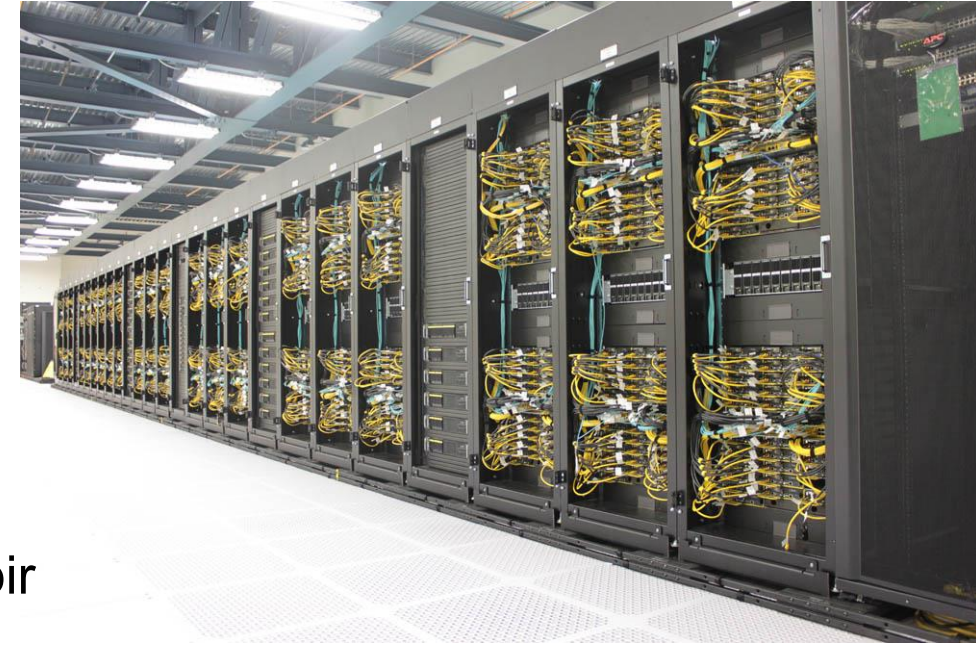






# Küme Bilgisayarlar

- Küme Bilgisayarlar (**clustered computers**)
- Çok işlemcili sistemler gibi, fakat çok sayıda sistem birlikte çalışıyor
  - Genellikle depolama birimi, **storage-area network (SAN)** ile paylaşıyor
  - **yüksek kullanılabilirlik (high-availability)** sağlayan bir servis (fault-tolerant)
    - ▶ **Asimetrik kümeleme (asymmetric clustering)** – bir tane gözlem makinası, diğerleri çalışıyor
    - ▶ **Simetrik kümeleme (symmetric clustering)** – birden fazla uygulama çalıştıran ve aynı zamanda birbirini gözlemleyen makinalara (**node**) sahip







# Yüksek Performanslı Hesaplama

- Bazı kümeler **yüksek performanslı hesaplama (high-performance computing-HPC)** sağlıyor
- Uygulamalar **paralelleştirmeyi (parallelization)** kullanacak şekilde yazılmalı





# İşletim Sistemleri İşlemleri





# Multiprogramming (Batch Sistem)

---

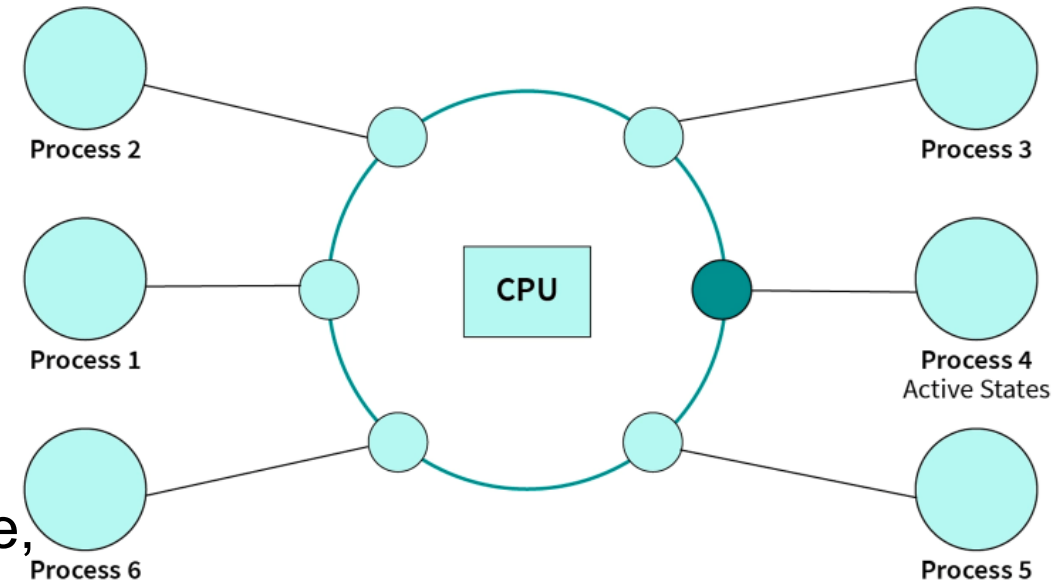
- Çoklu program desteği (multiprogramming) verimlilik için gerekli
  - Tek kullanıcı, CPU ve I/O cihazlarını her zaman meşgul edemez !
  - Çoklu program desteği, işleri (kod ve veri) CPU'nun her zaman çalıştıracağı bir proses olacak şekilde düzenler
  - Sistemdeki tüm prosesler (ya da bir kısmı) bellekte tutulur
  - İş sıralama (job scheduling) ile bir iş seçilir ve çalıştırılır
  - Çalışan iş beklemek zorunda kaldığında (örneğin I/O işlemi için) işletim sistemi başka bir işe geçer





# Multitasking (Zaman Paylaşımı)

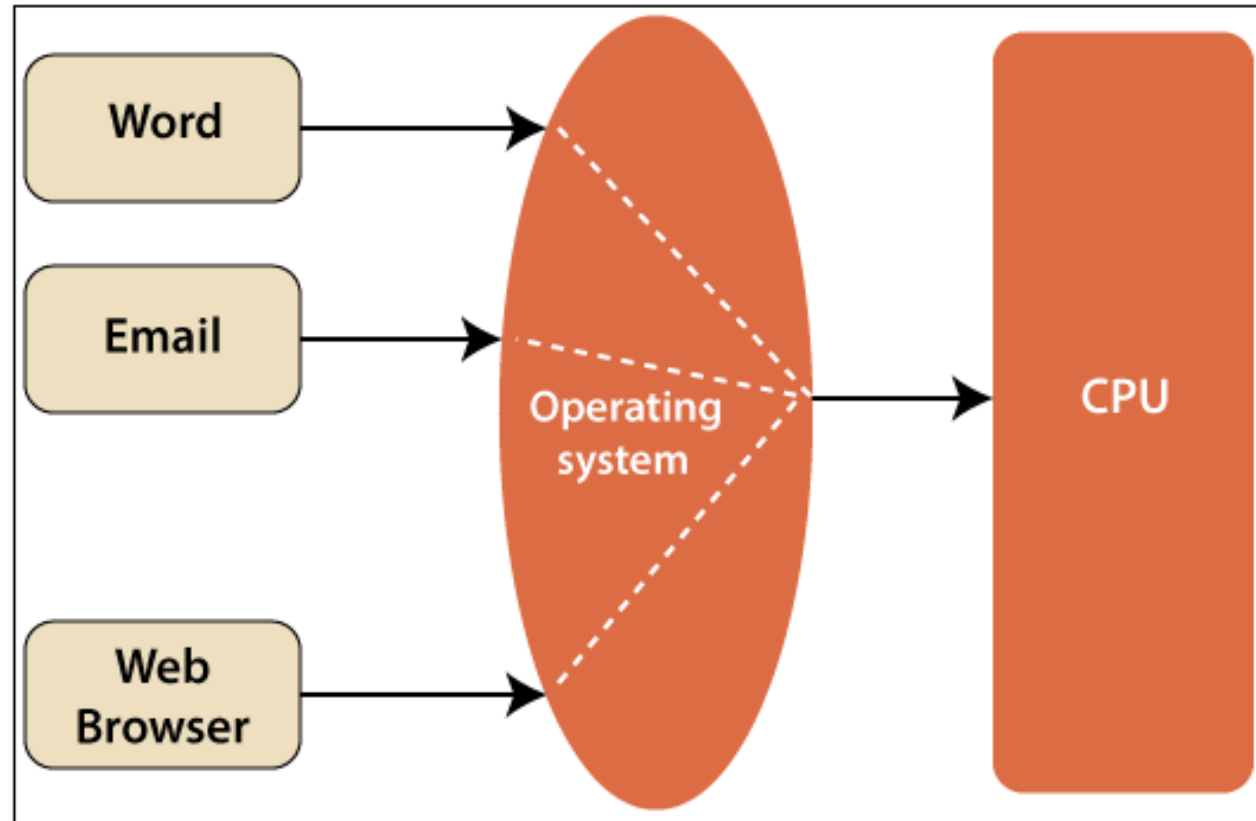
- **Yanıt süresi (response time)**  $\ll$  1 saniyeden olmalıdır
- Her bir kullanıcı bellekte çalışan en az bir programa sahiptir
- Eğer aynı anda birden fazla iş çalışmak için hazırsa  $\Rightarrow$  **İş sıralaması (job scheduling)**
- Eğer çalışacak olan işlerin tamamı belleğe sığmıyorsa, **değiş-tokuş (swapping)** işlemleri ile, çalışması gereken iş belleğe alınır ya da gerektiğinde bellekten çıkarılır
- **Sanal bellek (virtual memory)** tamamı bellekte bulunmayan işleri çalıştırmayı sağlar





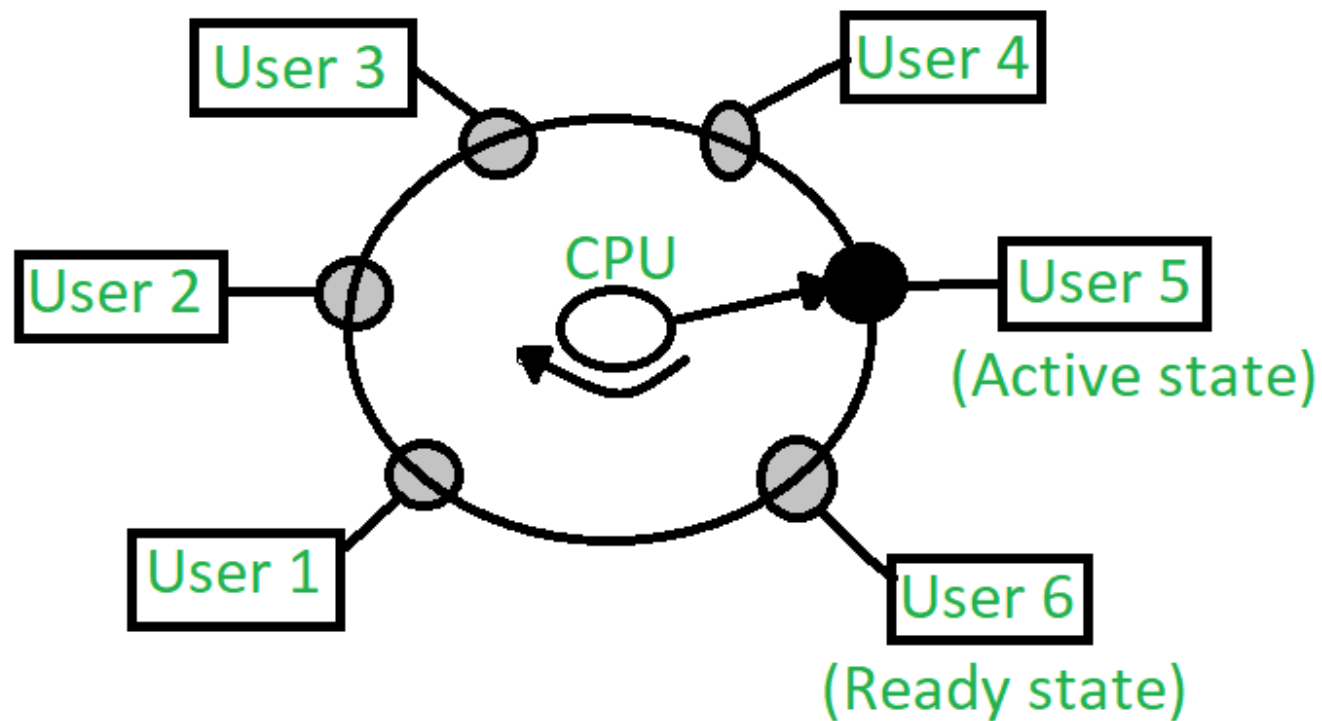
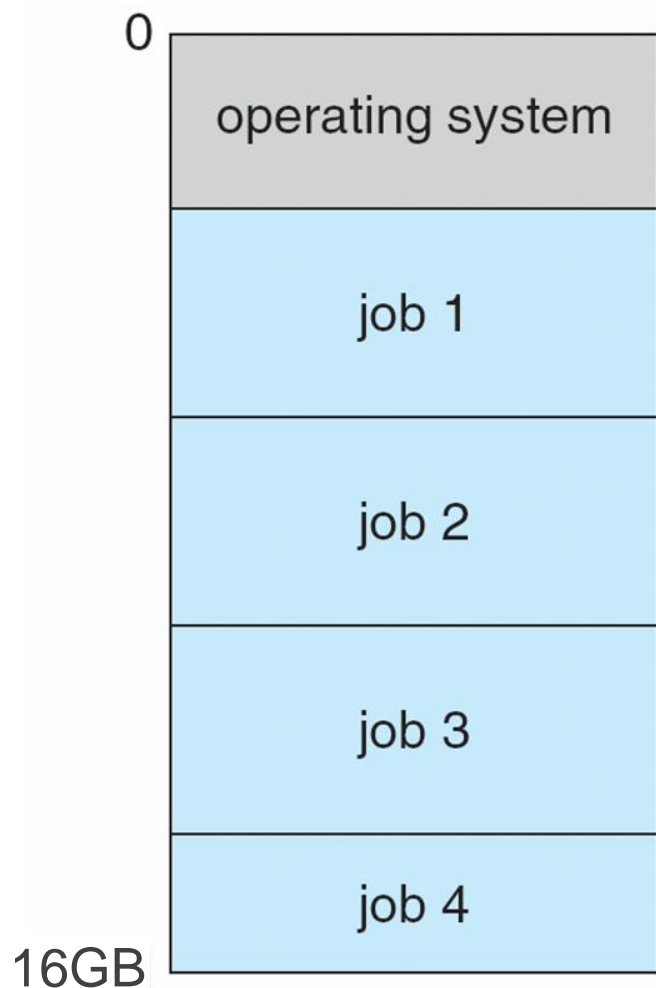
# Zaman Paylaşımı

- **Zaman Paylaşımı (timesharing veya multitasking)**, CPU'nun, iş (job)'leri çalıştırırken, işler arasında çok hızlı geçiş sağlayarak kullanıcıya bilgisayarı **interaktif (interactive)** şekilde kullanıyormuş hissi vermesidir





# Çok Programlı Sistemlerde Bellek Yapısı





# Çift-Modlu (Dual-Mode) Çalışma

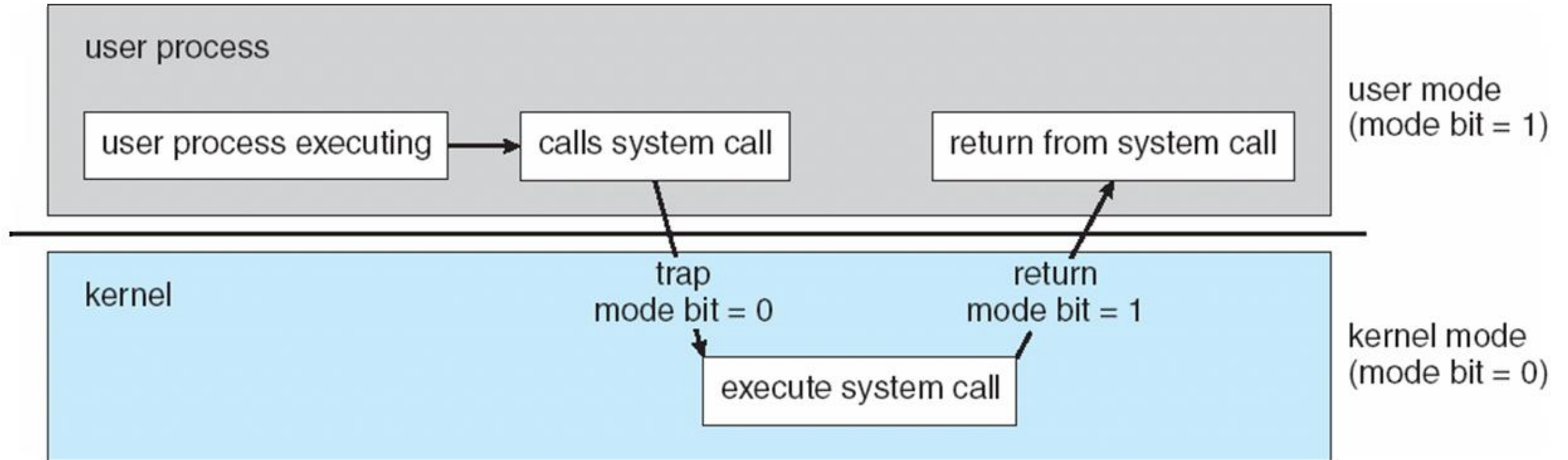
- **Çift-modlu işleme (dual-mode operation)**; işletim sistemini ve diğer sistem bileşenleri korumayı sağlar
  - **Kullanıcı modu (user mode)** ve **çekirdek modu (kernel mode)**
  - Donanım tarafından sağlanan **mod biti (mode bit)** vardır
    - Sistemin kullanıcı kodu mu, yoksa çekirdek kodu mu çalıştırdığını ayırt etmekte kullanılır
  - Bazı komutlar **ayrıcalıklı (privileged)** olarak tanımlıdırlar ve sadece çekirdek modunda çalıştırılabilirler
  - **Sistem çağrılarında** mod, çekirdek moduna çevirir
  - Sistem çağrısı bittiğinde mod, tekrar kullanıcı moduna çevrilir







# Kullanıcı Moddan Çekirdek Moduna Geçiş







# 1. Proses Yönetimi

---

- **Proses (process)** → çalışmakta olan program
- Program ***pasif*** bir yapı iken, proses ***aktif*** bir yapıdır
- Prosesler görevlerini yerine getirmek için kaynaklara ihtiyaç duyarlar
  - CPU, bellek, I/O, dosyalar
  - Başlangıç verisi
- Prosesin sonlandırılması kullanılan kaynakların sisteme iade edilmesini gerektirir





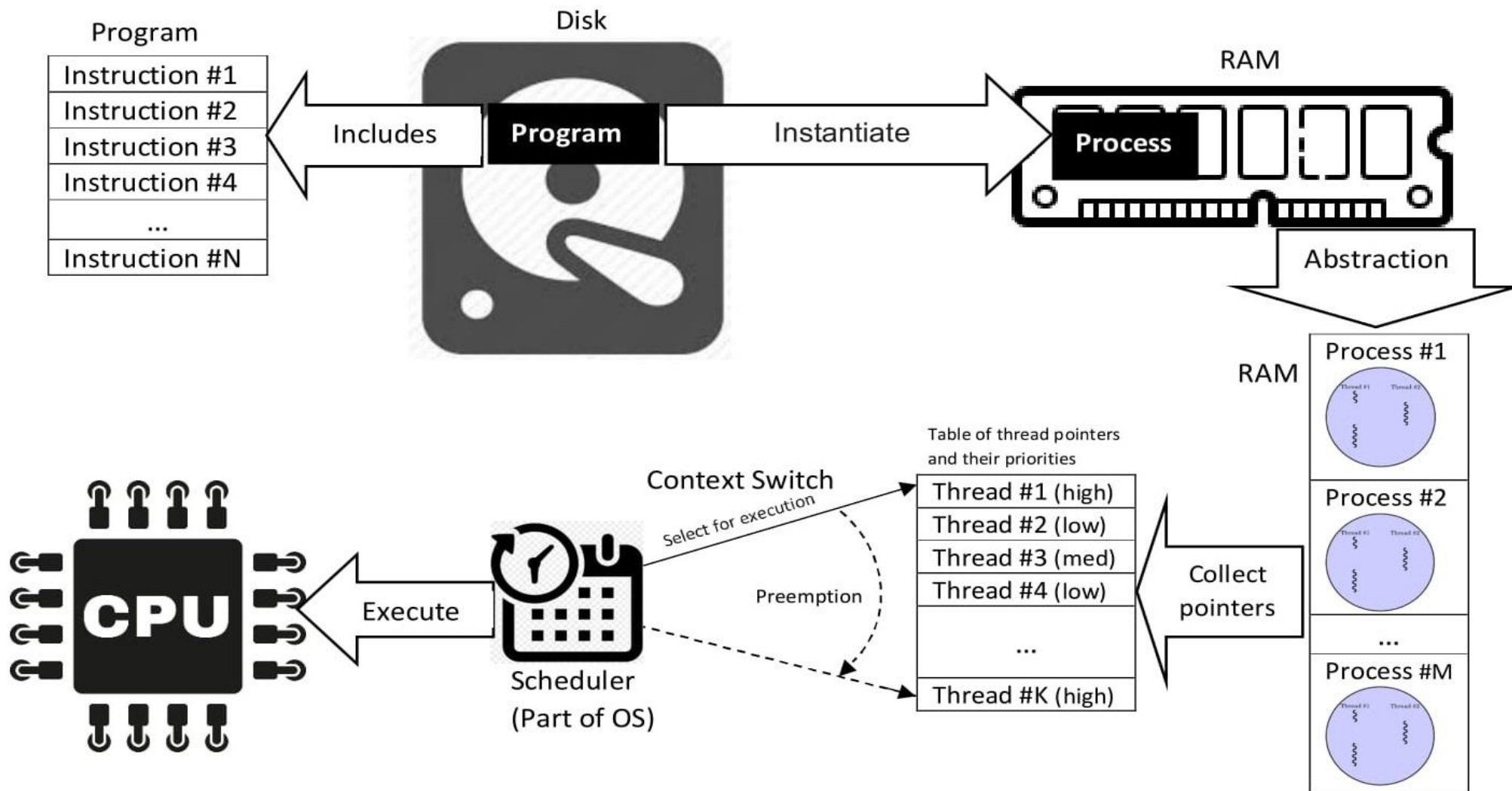
# Proses ve Thread Yönetimi

- **İş parçacığı (thread)** bir program çalışırken aynı anda yapılması gereken başka işler de varsa bunları çalıştırmak için kullanılır
- **Tek iş parçacıklı (single-threaded)** prosesler, çalıştırılacak bir sonraki komutun bellekteki adresini belirten tek bir **program sayacına (program counter-PC)** sahiptir
  - Proses sonlanana kadar, komutları tek tek sırayla çalıştırır
- **Çok iş parçacıklı (multi-threaded)** prosesler her bir ithread için ayrı bir PC'e sahiptirler
- Tipik olarak sistemlerde; çok sayıda proses ve bu proseslerin threadleri aynı anda bir veya birden fazla işlemcide çalıştırılır
  - **Eşzamanlı kullanım (concurrency)** işlemci(ler)in birden fazla proses veya thread arasında ortak kullanımını gerektirir





# Program vs Proses





# Proses Yönetim Faaliyetleri

---

İşletim sisteminin, proses yönetimi ile ilişkili faaliyetleri:

- Kullanıcı ve sistem proseslerinin **oluşturulması** ve **sonlandırılması**
- Proseslerin **duraklatılması** ve **devam ettirilmesi**
- Proseslerin senkronizasyonu için mekanizmalar sağlaması
- Proseslerin birbiri ile iletişim kurabilmesi için mekanizmalar sağlaması
- **Kilitlenmelerin (deadlock)** sağlıklı yönetilmesi için mekanizmalar sağlaması





# Bellek Yönetimi

---

- Proseslerin çalıştırılabilmesi için kod ve verinin bellekte olması gerekir
- Bellek yönetimi:
  - Neyin ve ne zamana kadar bellekte olması gerektiğine karar verir
  - Hedefi, işlemci kullanımını ve kullanıcılara verilen yanıt sürelerini optimize etmektir





# Bellek Yönetimi Faaliyetleri

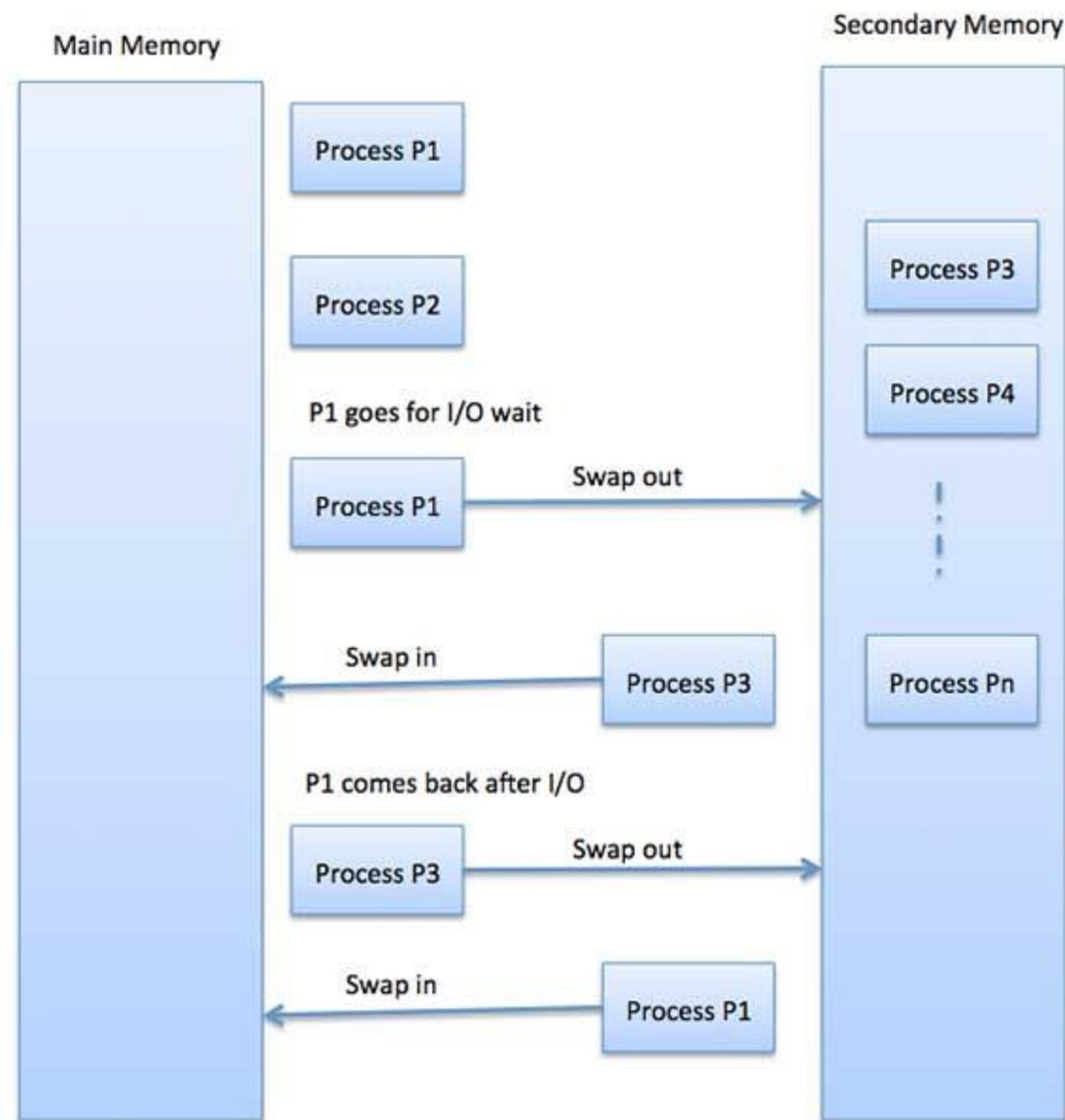
---

- Belleğin hangi bölümlerinin kim tarafından kullanıldığını takip etmek
- Hangi proseslerin ve verilerin belleğe alınacağına ya da bellekten çıkarılacağına karar vermek
- Gerektiğinde yeni bellek alanı ayırmak ya da kullanılmış alanları iade etmek





# Bellek Yönetimi Faaliyetleri





### 3. Depolama Birimi (Disk) Yönetimi

---

- İşletim sistemi, depolama üniteleri için tek ve mantıksal bir arayüz sunar
  - Kullanıcılar kendilerine sunulan verilerin mantıksal saklama yapısı (**dosya (file)**) sayesinde fiziksel cihazların karmaşıklığından soyutlanırlar
  - Tüm depolama üniteleri (i.e., disk, DVD, vs.) cihaz denetleyicisi tarafından kontrol edilir
  - Değişken özellikler:
    - erişim hızı
    - kapasite
    - veri transfer hızı
    - erişim yöntemi - ardışıl veya doğrudan

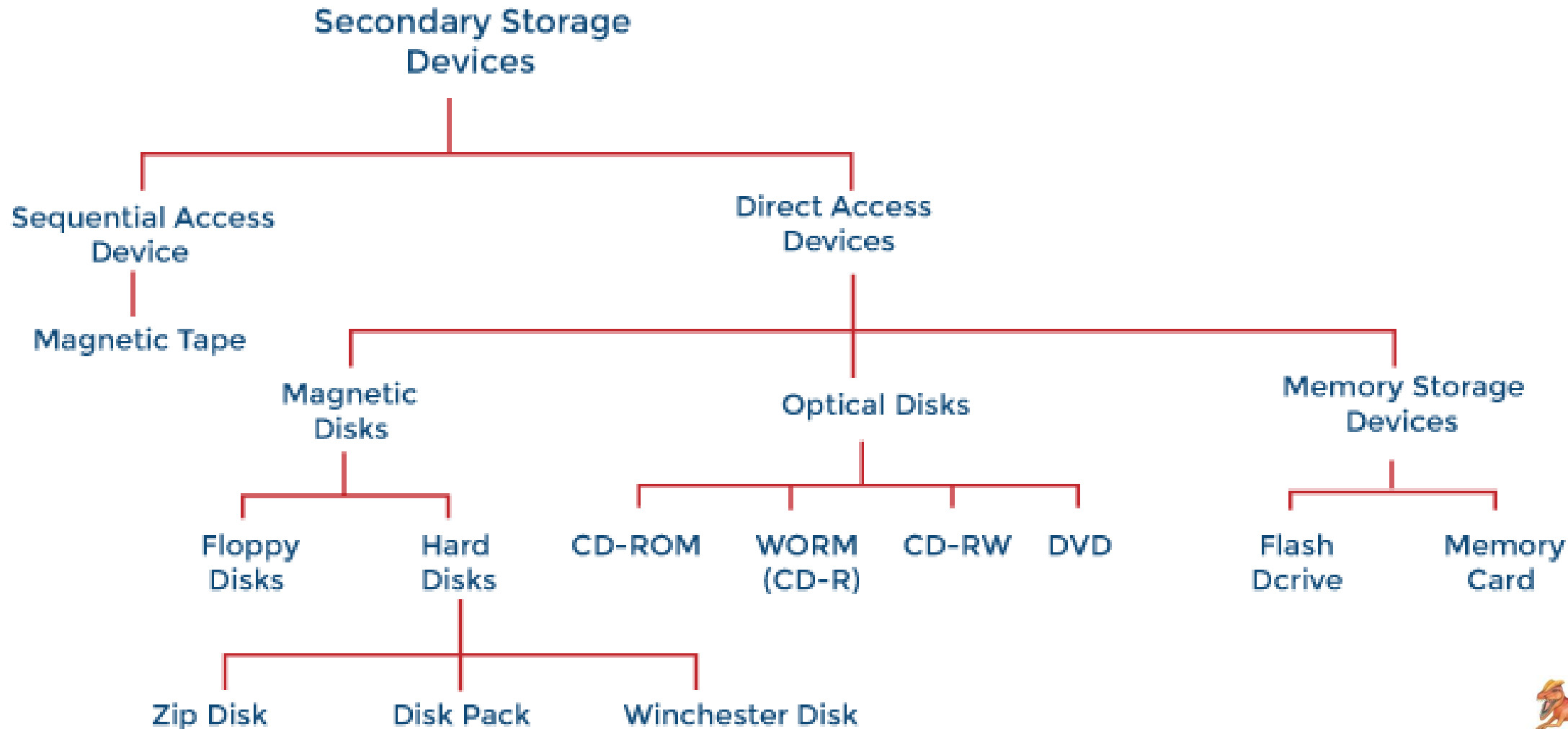






# Veri Depolama Birimleri

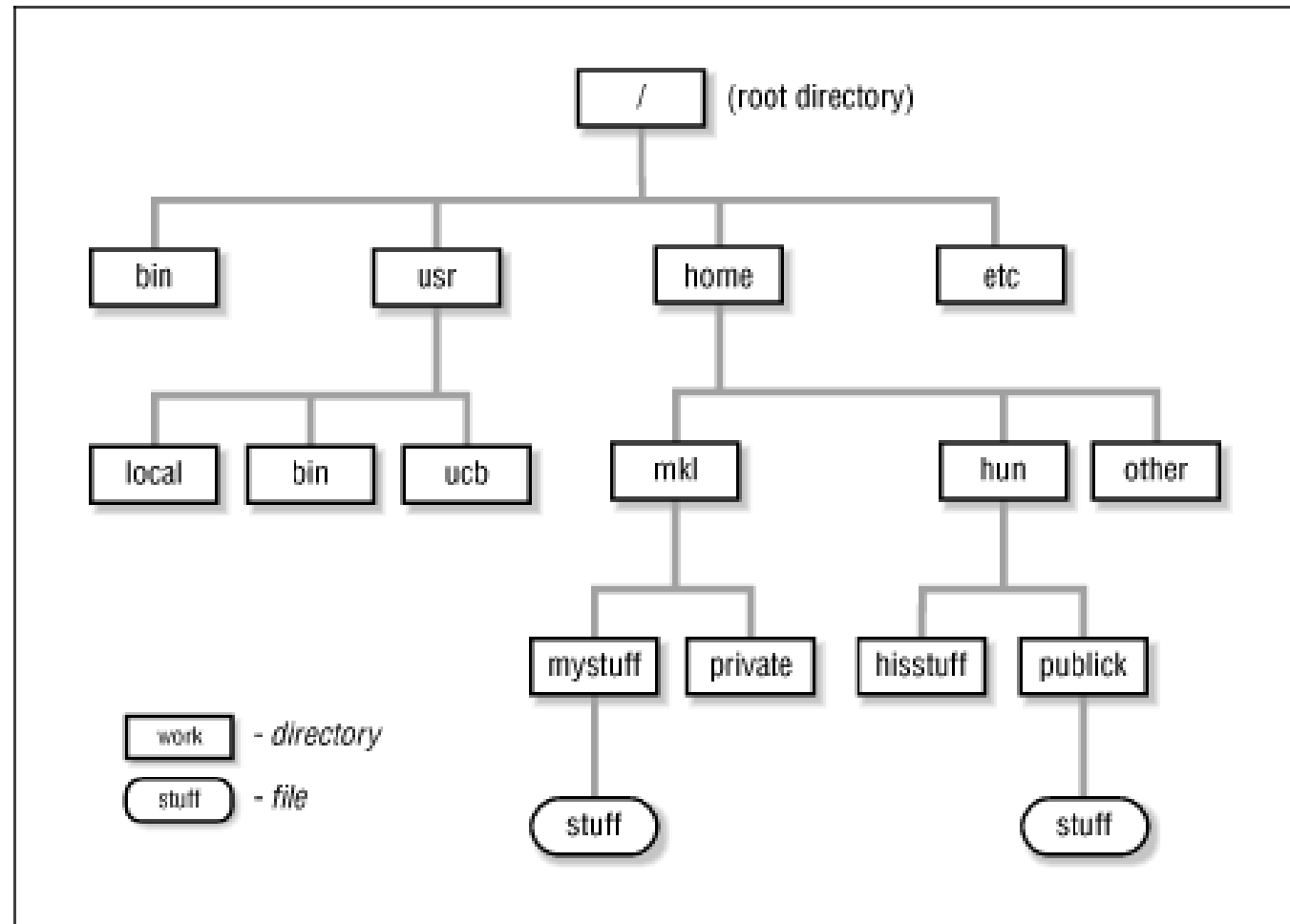
## Classification of Secondary Storage Devices





# Depolama Birimi (Disk) Yönetimi

- Dosyalar **dizinler (directory)** kullanılarak bir ağaç yapısı içerisinde organize edilirler





# Depolama Birimi (Disk) Yönetimi

---

- Özellikle çok kullanıcıli sistemlerde dizinlere veya dosyalara erişim kontrol edilmelidir: **erişim kontrolu** (**access control**)
- Dosya sistemi ile ilişkili işletim sistemi aktiviteleri
  - Dosya ve dizinlerin **oluşturulması** (**create**) veya **silinmesi** (**delete**)
  - Dosyaların veya dizinlerin **değiştirilmesi** (**editing**) için mekanizmanın sağlanması
  - Dosyaların ikincil depolama birimi ile **eşleştirilmesi** (**synchronization**)
  - Dosyaların kalıcı depolama birimlerine **yedeklenmesi** (**backup**)





# Depolama Birimi (Disk) Yönetimi

- Genellikle diskler, belleğe sığmayan verileri ya da uzun süre saklanacak verileri tutmakta kullanılırlar
- Verilerin **tutarlı (consistent)** yönetimi çok önemlidir
- Bilgisayarın genel hızı disk altsistemi ve algoritmalarının performansına çok bağlıdır
- Disk yönetimi ile ilgili işletim sistemi faaliyetleri:
  - **Boş alan yönetimi (free-space management)**
  - **Depolama alanı ayrımı (disk space allocation)**
  - **Disk zamanlaması-sıralama (disk scheduling)**
- Bazı depolama birimlerinin hızlı olması gerekmez
  - CD, DVD, Manyetik teypler (yine de yönetilmelidir)
  - **WORM (write-once, read-many-times)** ve **RW (read-write)** erişim modlarında çalışabilirler





# Depolama Birimi Performansları

- Depolama birimi seviyeleri arasında bilgi aktarımı, **kullanıcının isteğine bağlı** ya da **kullanıcı isteğinden bağımsız** gerçekleşebilir

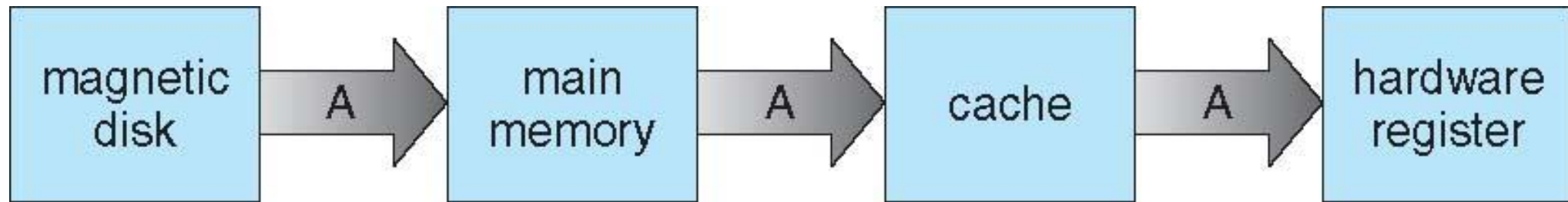
Level	1	2	3	4	5
Name	registers	cache	main memory	solid-state disk	magnetic disk
Typical size	< 1 KB	< 16MB	< 64GB	< 1 TB	< 10 TB
Implementation technology	custom memory with multiple ports CMOS	on-chip or off-chip CMOS SRAM	CMOS SRAM	flash memory	magnetic disk
Access time (ns)	0.25-0.5	0.5-25	80-250	25,000-50,000	5,000,000
Bandwidth (MB/sec)	20,000-100,000	5,000-10,000	1,000-5,000	500	20-150
Managed by	compiler	hardware	operating system	operating system	operating system
Backed by	cache	main memory	disk	disk	disk or tape





# 'A' Değerinin Diskten Saklayıcıya Aktarımı

- Multiprocessor ortamlar, **en güncel değeri kullanmak** konusunda dikkatli olmalıdır (veri depolama hiyerarşisinin neresinde tutuluyorsa tutulsun)



- **Çok işlemcili sistemlerde ön bellek tutarlılığı (cache consistency)** donanım seviyesinde sağlanmalı ve tüm işlemciler en güncel değere sahip olmalıdır
- Dağıtık ortamlarda durum daha da karmaşıktır
  - genelde verinin birden fazla kopyası bulunur





# I/O Alt Sistemi

- İşletim sisteminin amaçlarından biri donanım cihazlarının karmaşıklıklarını kullanıcıdan gizlemektir
- I/O alt sisteminin sorumlulukları:
  - I/O işlemlerinin bellek yönetimini yapmak
    - ▶ **Tampon bellek işlemleri (buffering)** – veriyi bir yerden diğerine aktarırken geçici olarak saklamak
    - ▶ **Ön bellek işlemleri (caching)** – veriyi geçici olarak daha hızlı depolama birimine aktarmak
    - ▶ **Kuyruklama (spooling - Simultaneous Peripheral Operation On-Line)** – Özellikle CPU ve G/Ç cihazları arasında önemli bir hız uyumsuzluğu varsa, G/Ç işlemlerinin verimliliğini artırın. Geçici bir depolama alanı, CPU ile yazıcılar, diskler veya ağ kaynakları gibi çeşitli çevresel aygıtlar arasında aktarılan verileri tutar.
  - Genel **cihaz sürücüsü (device driver)** arayüzü
  - Özel donanım cihazları için sürücüler





# Koruma ve Güvenlik

- **Koruma (protection)** – Proseslerin veya kullanıcıların herhangi bir kaynağa erişiminin işletim sistemi tarafından kontrol edilmesi
- **Güvenlik (security)** – Sistemin içeriden ve dışarıdan gelen saldırılara karşı savunulması
  - Geniş kapsamlı: DoS saldırıları, virusler, solucanlar, kimlik bilgileri hırsızlığı







# Koruma ve Güvenlik

- Sistemler öncelikle kullanıcıları, kimin ne yapabileceğine göre sınıflandırır
  - **Kullanıcı adı (user\_ID, security\_ID)** her kullanıcı için isim ve ilişkili parolayı içerir
  - Bu çift daha sonra, erişim kontrolü amacıyla, kullanıcının sahip olduğu tüm dosya ve işlemlerle ilişkilendirilir
  - **Grup adı (group\_ID)** benzer izinlere veya sorumluluklara sahip kullanıcıları kategorilere ayırmak ve gruplandırmak için kullanılır. Aynı gruba ait olan kullanıcılar, yönetimi basitleştiren ortak bir erişim hakları kümesini paylaşırlar.
  - Ayrıcalık yükseltme (**Privilege escalation**), kullanıcının geçerli kimliği ile daha fazla hakka geçmesine olanak tanır





# Sanallaştırma (Virtualization)

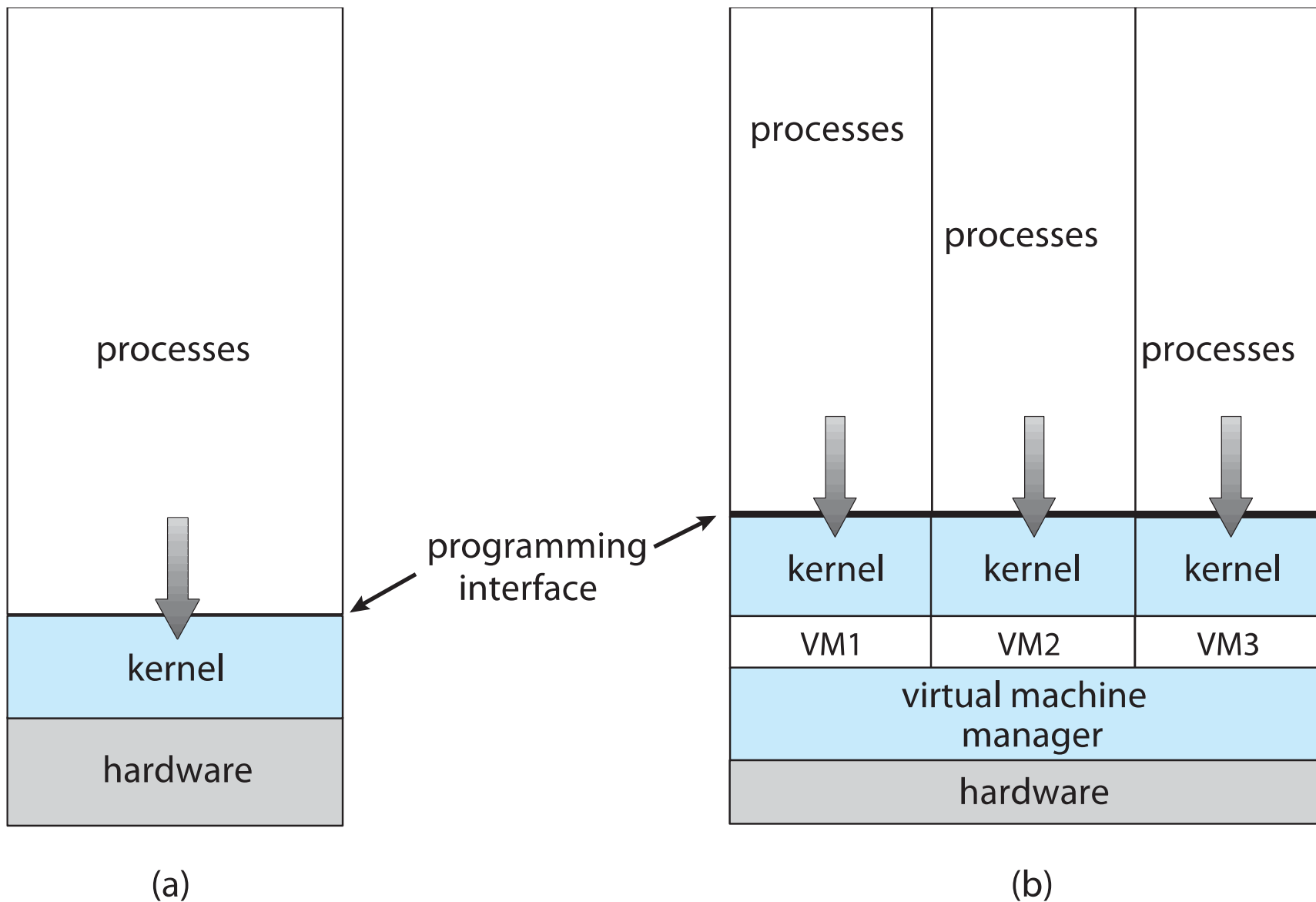
---

- Sanallaştırma, donanım, işletim sistemleri, depolama aygıtları veya ağ kaynakları gibi **fiziksel bilgi işlem kaynaklarının sanal ortamlarının veya simülasyon sürümlerinin oluşturulmasını** sağlayan bir teknolojidir.
- Birden çok işletim sisteminin (**misafir - guest-os**) temel donanım kaynaklarını paylaşarak tek bir fiziksel makinede aynı anda çalışmasına izin verir.





# Sanallaştırma





# Sanallaştırma

---

- **Sanallaştırma** guest-os'ler arasında izolasyon sağlar ve kendi kaynakları, konfigürasyonları ve uygulamalarıyla bağımsız olarak çalışmalarına olanak tanır.
- Ayrıca, birden çok sanal makineyi (VM) tek bir fiziksel sunucuda birleştirerek donanım kaynaklarının verimli bir şekilde kullanılmasına olanak tanır. Böylece;
  - esneklik
  - maliyet tasarrufu
  - ölçeklenebilirlik ve
  - yönetim kolaylığı sunar.



# 1. Bölümün Sonu

---

