

Operation Systems

Dr. A. Taghinezhad



Operating System Concepts

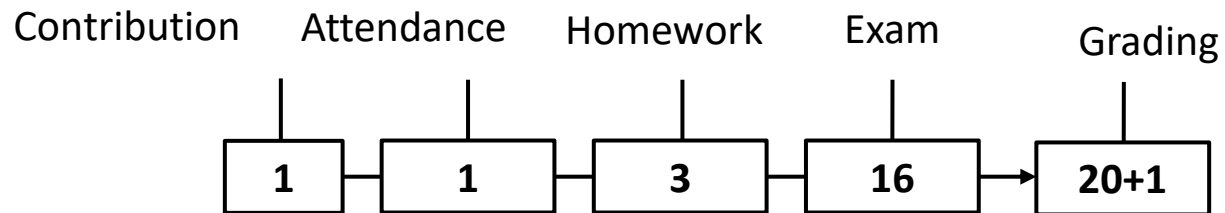
TENTH EDITION

ABRAHAM SILBERSCHATZ • PETER BAER GALVIN • GREG GAGNE



WILEY

Website: ataghinezhad.github.io, Email: a0taghinezhad@gmail.com



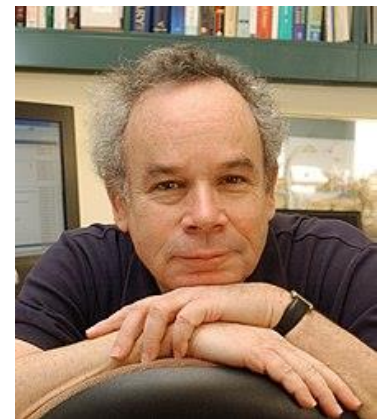
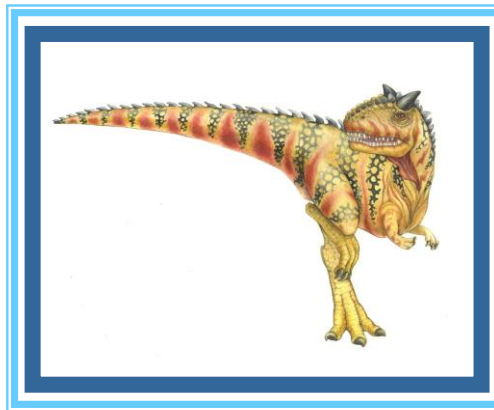


Outline of this Course

- فرآیندها (Processes)
- رشته‌ها و هم‌زمانی (Threads and Concurrency)
- زمان‌بندی (CPU Scheduling) CPU
- بن‌بست‌ها (Deadlocks)
- حافظه اصلی (Main Memory)
- حافظه مجازی (Virtual Memory)
- ساختار ذخیره‌سازی حجیم (Mass-Storage Structure)



فصل ١: مقدمه





چیزهایی که در این بخش میخوانید:

- سیستم عامل ها چه کاری انجام می دهند
- سازمان سیستم های کامپیوتری
- معماری سیستم های کامپیوتری
- ساختار سیستم عامل
- عملیات سیستم عامل
- مدیریت فرآیند
- مدیریت حافظه
- مدیریت ذخیره سازی
- حفاظت و امنیت
- ساختارهای داده ای کرنل
- محیط های محاسباتی
- سیستم های عامل متن باز





Objectives

□ اهداف:

- توصیف ساختار سیستم‌های کامپیوتری
- ارائه مروری کلی بر اجزای اصلی سیستم‌عامل‌ها
- بررسی انواع محیط‌های محاسباتی
- کاوش چندین سیستم‌عامل متن‌باز





What is an Operating System?

□ **تعریف سیستم عامل:**

□ برنامه‌ای که به عنوان واسطه بین کاربر کامپیوتر و سخت افزار عمل می کند.

□ **اهداف سیستم عامل:**

- اجرای برنامه‌های کاربر و تسهیل حل مسائل کاربران
- راحت تر کردن استفاده از سیستم کامپیوتری
- استفاده بهینه از سخت افزار کامپیوتر (مدیریت منابع)





Computer System Structure

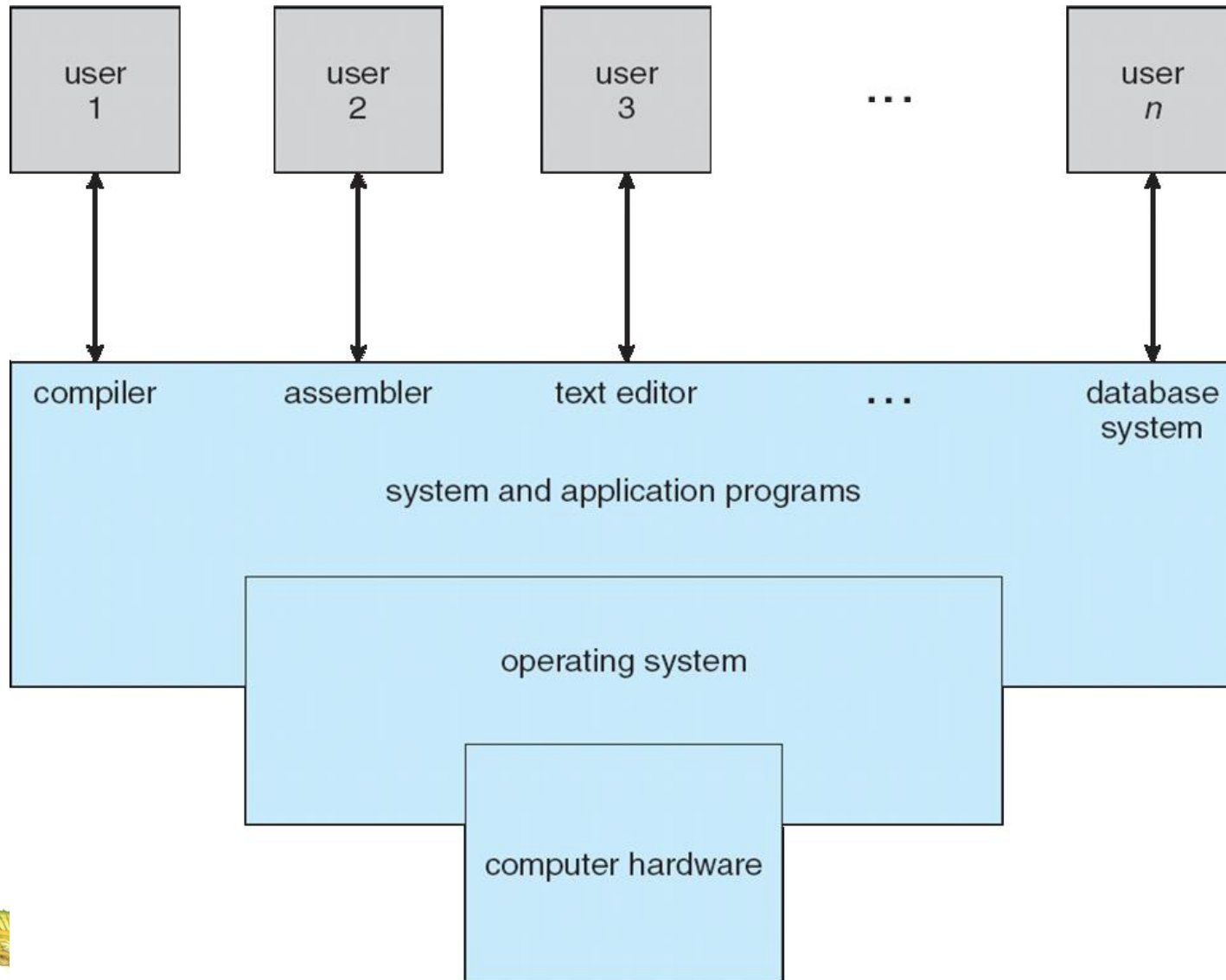
□ اجزای سیستم کامپیوتری:

1. **سخت افزار** – منابع پایه‌ای محاسباتی را فراهم می‌کند:
 - پردازنده (CPU)، حافظه، دستگاه‌های ورودی/خروجی
2. **سیستم عامل** – کنترل و هماهنگی استفاده از سخت افزار بین برنامه‌های مختلف و کاربران
3. **برنامه‌های کاربردی** – تعیین روش استفاده از منابع سیستم برای حل مسائل کاربران:
 - پردازشگر متن، کامپایلرها، مرورگرهای وب، سیستم‌های پایگاه داده، بازی‌های ویدیویی
4. **کاربران** – شامل افراد، ماشین‌ها و سایر کامپیوترها





Four Components of a Computer System





What Operating Systems Do



بستگی به دیدگاه دارد

- کاربران راحتی، سهولت استفاده و عملکرد خوب را می خواهند
- به بهره‌برداری از منابع اهمیتی نمی دهند
- اما در سیستم‌های مشترک مانند **مین فریم** یا **مینی کامپیوتر**، باید تمام کاربران راضی باشند
- کاربران سیستم‌های اختصاصی مانند **ایستگاه‌های کاری**، منابع اختصاصی دارند اما اغلب از منابع مشترک سرورها نیز استفاده می کنند
- **کامپیوترهای دستی** منابع محدودی دارند و برای **قابلیت استفاده و عمر باتری** بهینه شده اند
- برخی از کامپیوترها **رابط کاربری کمی دارند یا اصلاً ندارند**، مانند کامپیوترهای تعبیه شده در دستگاه‌ها و خودروها





Operating System Definition

□ سیستم عامل به عنوان مدیریت منابع

- تمامی منابع را مدیریت می کند
- بین درخواست های متضاد تصمیم گیری می کند تا استفاده از منابع کارآمد و عادلانه باشد

□ سیستم عامل به عنوان برنامه کنترلی

- اجرای برنامه ها را کنترل می کند تا از خطاها و استفاده نادرست از کامپیوتر جلوگیری کند





Operating System Definition (Cont.)

□ تعریف سیستم عامل

- هیچ تعریف جهانی پذیرفته شده‌ای ندارد
- "هر چیزی که یک فروشنده هنگام خرید سیستم عامل ارائه می‌دهد" یک تقریب خوب است، اما بسیار متغیر است
- "برنامه‌ای که همیشه در حال اجرا است" کرنل نامیده می‌شود
- سایر اجزا شامل:
 - برنامه‌های سیستمی همراه سیستم عامل ارائه می‌شوند
 - برنامه‌های کاربردی





Computer Startup

□ برنامه بوت استرپ

- هنگام روشن شدن یا ری استارت سیستم بارگذاری می شود
- معمولاً در **ROM** یا **EPROM** ذخیره شده و به عنوان فرمور شناخته می شود
- تمامی بخش های سیستم را مقداردهی اولیه می کند
- کرنل سیستم عامل را بارگذاری کرده و اجرای آن را آغاز می کند

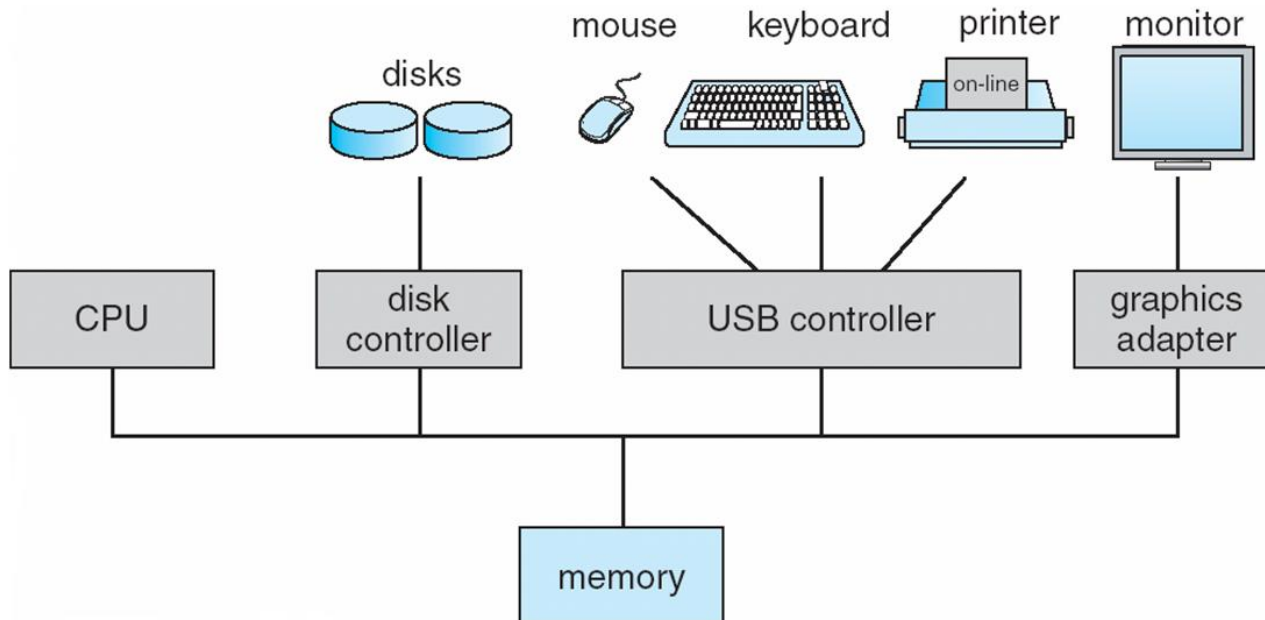




Computer System Organization

□ عملیات سیستم کامپیوتری

- شامل یک یا چند پردازنده (CPU) و کنترل کننده‌های دستگاه که از طریق یک گذرگاه مشترک به حافظه اشتراکی دسترسی دارند
- پردازنده‌ها و دستگاه‌های مختلف به‌طور همزمان اجرا می‌شوند و برای دسترسی به حافظه رقابت می‌کنند





Computer-System Operation

- مدیریت ورودی/خروجی (I/O) و نقش کنترل کننده های دستگاه
- CPU و دستگاه های I/O می توانند همزمان اجرا شوند
- هر کنترل کننده دستگاه مسئول یک نوع خاص از دستگاه ها است
- هر کنترل کننده دارای یک بافر محلی است
- CPU داده ها را بین حافظه اصلی و بافر کنترل کننده جابه جا می کند
- عملیات I/O از دستگاه به بافر محلی کنترل کننده انجام می شود
- کنترل کننده دستگاه از طریق وقفه به CPU اطلاع می دهد که عملیاتش به پایان رسیده است



Device controller

□ تعریف کنترل کننده دستگاه

- یک جزء الکترونیکی است که ارتباط بین CPU و دستگاه‌های ورودی/خروجی (I/O) را مدیریت می‌کند
- به عنوان یک واسطه بین سیستم‌عامل و دستگاه‌های I/O عمل می‌کند
- وظایف کنترل کننده:
 - تبدیل جریان بیت‌های سریال به بلوک‌های بایت
 - اصلاح خطا در صورت نیاز
 - ذخیره داده در بافر داخلی و بررسی صحت آن قبل از انتقال به حافظه اصلی



Interrupt

□ وقفه (Interrupt) در سیستم عامل

• سیگنالی که سخت افزار یا نرم افزار هنگام نیاز به رسیدگی فوری ارسال می کند

□ انواع وقفه ها:

1. وقفه سخت افزاری: ناشی از وضعیت دستگاه های سخت افزاری

◦ مثال: خطوط IRQ در رایانه شخصی، تایمر پردازنده

2. وقفه نرم افزاری: توسط نرم افزار یا سیستم عامل ایجاد می شود

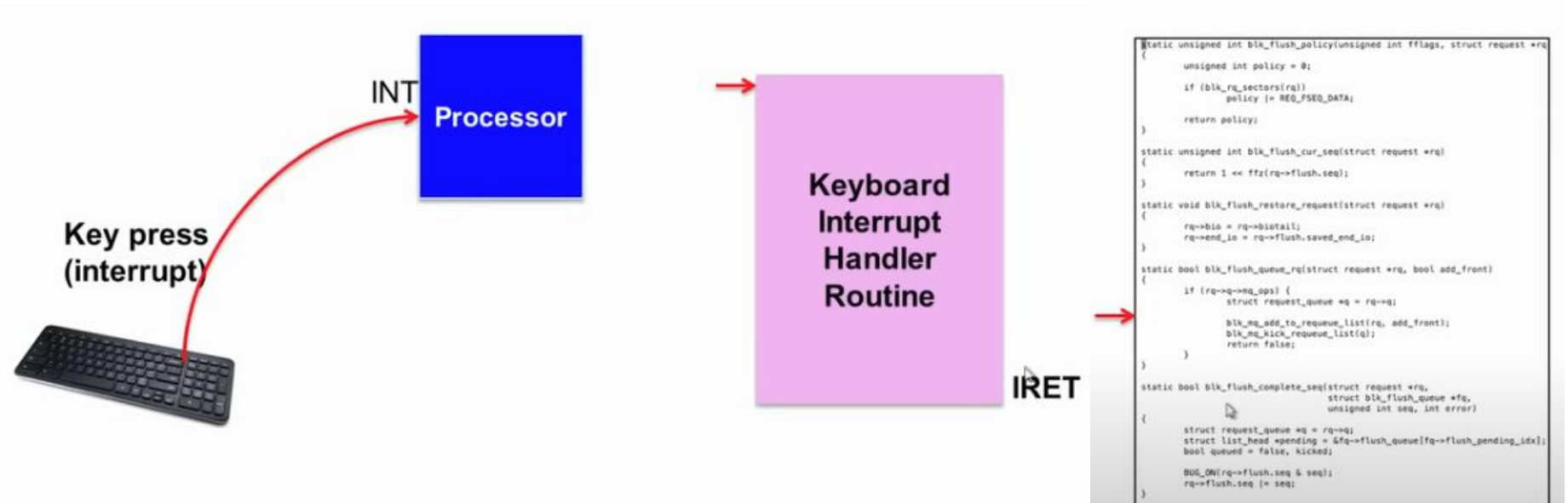
◦ مثال: خطاها (Trap)، درخواست های سیستمی (System Calls)

□ زمان تأخیر وقفه: (Interrupt Latency)

• فاصله زمانی بین دریافت وقفه و شروع اجرای سرویس وقفه



Hardware Interrupt





Common Functions of Interrupts

- مدیریت وقفه در سیستم عامل
- کنترل به روال سرویس وقفه (ISR) منتقل می شود، که شامل آدرس تمامی روال های خدماتی است
- معماری وقفه باید آدرس دستورالعمل قطع شده را ذخیره کند
- یک تله (Trap) یا استثنا، یک وقفه نرم افزاری است که به دلیل خطا یا درخواست کاربر رخ می دهد
- سیستم عامل مبتنی بر وقفه ها اجرا می شود

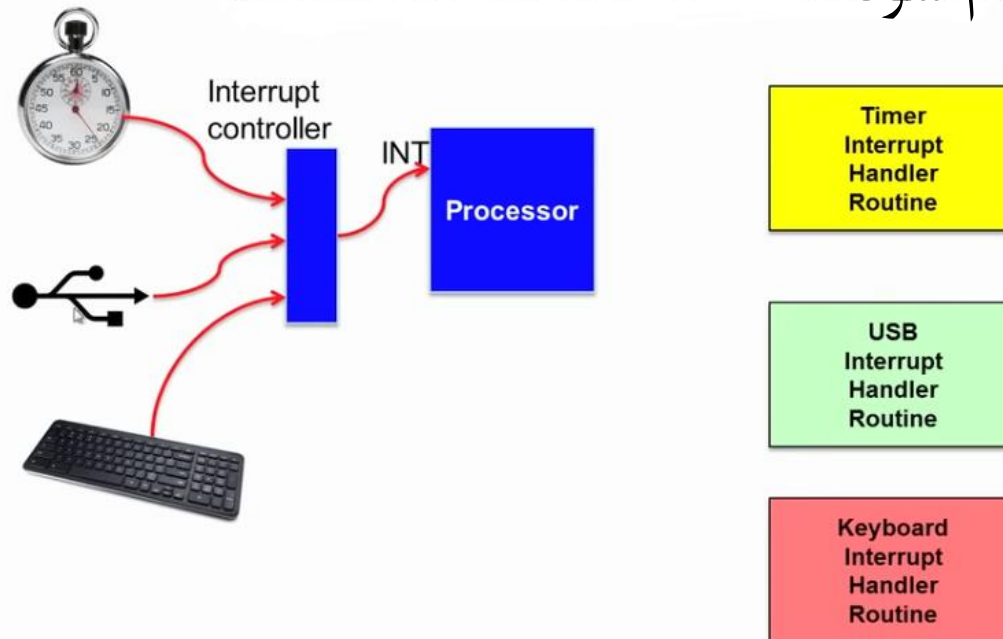




Interrupt Handling

□ سیستم عامل وضعیت پردازنده را با ذخیره‌ی ثبات‌ها (Registers) و شمارنده‌ی برنامه (Program Counter) حفظ می‌کند.

تشخیص می‌دهد که چه نوع وقفه‌ای رخ داده است. بخش‌های جداگانه‌ای از کد مشخص می‌کنند که برای هر نوع وقفه چه اقدامی باید انجام شود



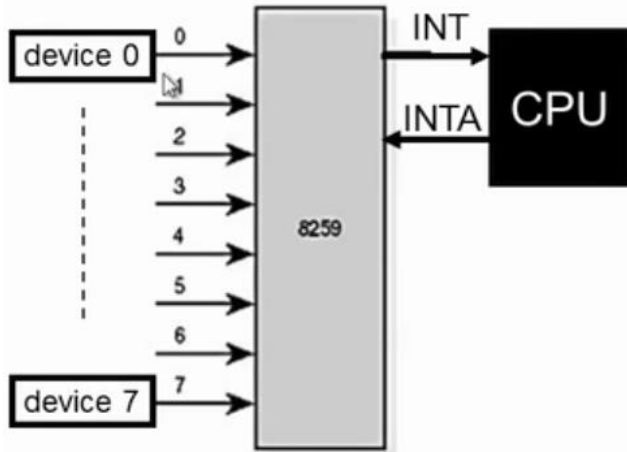


□ هنگامی که پردازنده دچار وقفه می‌شود، آنچه را که در حال انجام آن است متوقف کرده و بلافاصله اجرای خود را به یک مکان ثابت منتقل می‌کند. این مکان معمولاً شامل آدرس شروع روال خدمات وقفه (Interrupt Service Routine - ISR) است. پس از اجرای ISR، پردازنده به محاسبات متوقف‌شده بازمی‌گردد.



Programmable Interrupt Controller

کنترل وقفه توسط ۸۲۵۹ (کنترل کننده وقفه برنامه پذیر)



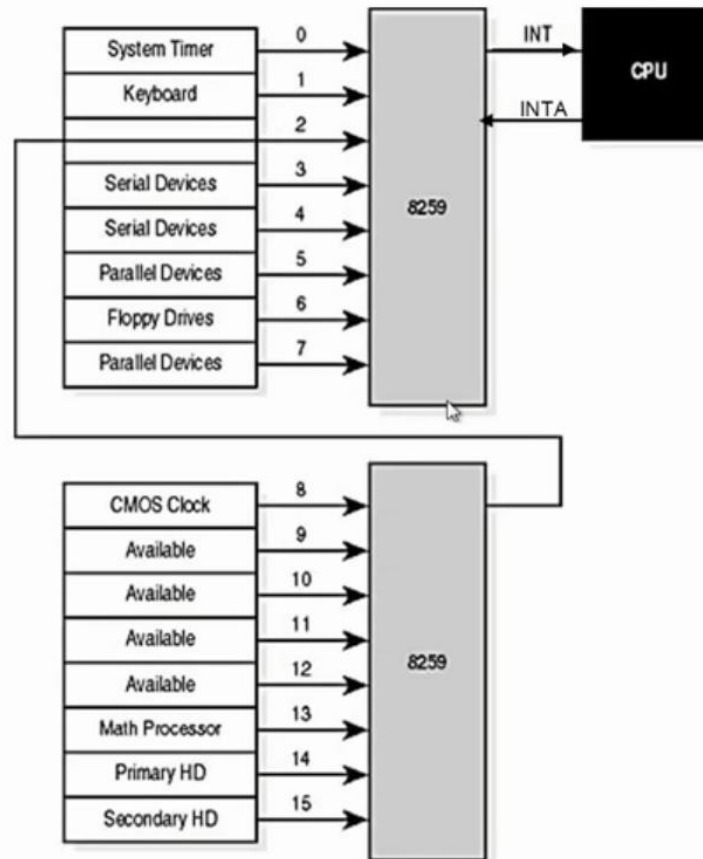
- قابلیت ارسال حداکثر ۸ وقفه به CPU
- دستگاه‌ها از طریق IRQ وقفه ایجاد می‌کنند
- CPU وقفه را تأیید کرده و 8259 را برای شناسایی منبع وقفه پرس‌وجو می‌کند
- اولویت‌ها می‌توانند به هر خط IRQ اختصاص داده شوند
- چندین ۸۲۵۹ می‌توانند به صورت آبشاری برای پشتیبانی از تعداد بیشتری وقفه استفاده شوند

The [Intel 8259](#) is a [programmable interrupt controller](#) (PIC) designed for the [Intel 8085](#) and [Intel 8086 microprocessors](#).



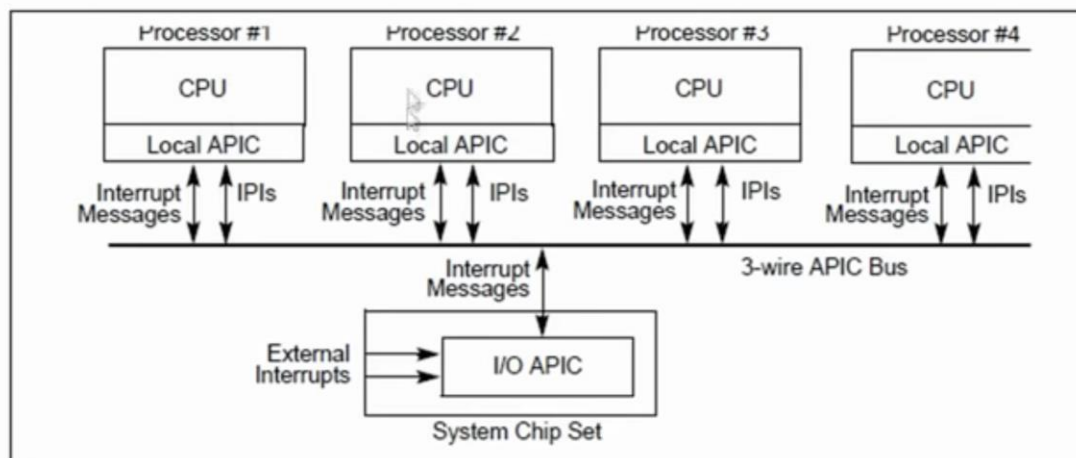
Interrupts in legacy CPUs

- 15 IRQs (IRQ0 to IRQ15), so 15 possible devices
- Limitations
 - Limited IRQs
 - Not suited for multi-processor / multi-core platforms



New systems Interrupt handler per CPU

Advanced Programmable Interrupt Controller (APIC)



وقفه‌های خارجی (External Interrupts) در سیستم‌های چندپردازنده‌ای از طریق APIC از دستگاه‌های جانبی به پردازنده‌ها منتقل می‌شوند.

APIC وقفه‌ها را به پردازنده‌ها توزیع و اولویت‌بندی می‌کند.

APIC شامل دو بخش است:

- APIC محلی (LAPIC)

- APIC ورودی/خروجی (I/O APIC)

ارتباط بین APIC ها از طریق یک گذرگاه (Bus) ویژه‌ی سه‌سیمی APIC انجام می‌شود.



I/O Structure

□ عملیات ورودی/خروجی (I/O) و بازگشت کنترل به برنامه‌ی کاربر

- پس از شروع I/O، کنترل تنها پس از تکمیل I/O به برنامه‌ی کاربر بازمی‌گردد.
- دستور *Wait* پردازنده را تا زمان وقوع وقفه‌ی بعدی در حالت بیکار قرار می‌دهد.
- حلقه‌ی انتظار (Wait Loop) موجب رقابت برای دسترسی به حافظه می‌شود.
- در هر لحظه، تنها یک درخواست I/O فعال است و پردازش همزمان I/O وجود ندارد.
- پس از شروع I/O، کنترل بدون انتظار برای تکمیل عملیات I/O به برنامه‌ی کاربر بازمی‌گردد.






I/O Structure

- **System Call** (سیستم کال) درخواستی به سیستم عامل است که اجازه می دهد کاربر برای تکمیل I/O منتظر بماند.
- جدول وضعیت دستگاه (Device-Status Table) برای هر دستگاه ورودی/خروجی یک ورودی دارد که نوع، آدرس و وضعیت آن را نشان می دهد.
- سیستم عامل با استفاده از این جدول وضعیت دستگاه را تعیین کرده و در صورت نیاز مدخل مربوطه را برای ثبت وقفه تغییر می دهد.







vector number	description
0	divide error
1	debug exception
2	null interrupt
3	breakpoint
4	INTO-detected overflow
5	bound range exception
6	invalid opcode
7	device not available
8	double fault
9	coprocessor segment overrun (reserved)
10	invalid task state segment
11	segment not present
12	stack fault
13	general protection
14	page fault
15	(Intel reserved, do not use)
16	floating-point error
17	alignment check
18	machine check
19–31	(Intel reserved, do not use)
32–255	maskable interrupts




Figure 1.5 Intel processor event-vector table.



Storage Definitions and Notation Review

واحدهای ذخیره‌سازی کامپیوتر

- واحد پایه ذخیره‌سازی در کامپیوتر بیت است. یک بیت می‌تواند یکی از دو مقدار ۰ یا ۱ را ذخیره کند.
- تمامی سایر انواع ذخیره‌سازی در کامپیوترها بر اساس مجموعه‌ای از بیت‌ها ساخته شده است. با داشتن تعداد کافی بیت‌ها، می‌توان اعداد، حروف، تصاویر، فیلم‌ها، صداها، اسناد و برنامه‌ها را نمایندگی کرد.
- یک بایت برابر با ۸ بیت است، و در بیشتر کامپیوترها کوچک‌ترین واحد ذخیره‌سازی قابل استفاده است. اصطلاح کمتر رایج "کلمه" (word) است که واحد داده‌ای بومی معماری خاص کامپیوتر می‌باشد.
- ذخیره‌سازی کامپیوتر و بیشتر توان عملیاتی آن معمولاً در واحد بایت و مجموعه‌هایی از بایت‌ها اندازه‌گیری و پردازش می‌شود.
- یک کیلوبایت (KB) برابر با ۱,۰۲۴ بایت است.
- یک مگابایت (MB) برابر با ۱,۰۲۴۲ بایت است.
- یک گیگابایت (GB) برابر با ۱,۰۲۴۳ بایت است.
- یک ترابایت (TB) برابر با ۱,۰۲۴۴ بایت است.
- یک پتابایت (PB) برابر با ۱,۰۲۴۵ بایت است.





Storage Structure

□ حافظه اصلی (Main Memory) و ذخیره‌سازی ثانویه (Secondary Storage)

- حافظه‌ی اصلی، تنها رسانه‌ی ذخیره‌سازی بزرگی است که پردازنده مستقیماً به آن دسترسی دارد.
- دسترسی تصادفی دارد.
- معمولاً فرّار (Volatile) است.
- حافظه‌ی ثانویه، گسترشی از حافظه‌ی اصلی است که ظرفیت ذخیره‌سازی بزرگ و غیرفرّاری را فراهم می‌کند.





Storage Structure

□ دیسک‌های سخت و ذخیره‌سازی Solid-state disk حالت جامد

- دیسک‌های سخت از صفحات فلزی یا شیشه‌ای سختی تشکیل شده‌اند که با مواد مغناطیسی پوشیده شده‌اند.
- سطح دیسک به صورت منطقی به شیارها (Tracks) تقسیم شده است که هر شیار به بخش‌هایی (Sectors) تقسیم می‌شود.
- کنترل کننده‌ی دیسک، تعامل منطقی بین دستگاه و رایانه را تعیین می‌کند.
- دیسک‌های حالت جامد (SSD) سریع‌تر و غیرفرّار هستند.
- دارای فناوری‌های مختلفی بوده و روزبه‌روز محبوب‌تر می‌شوند.





Storage Hierarchy

□ سیستم‌های ذخیره‌سازی در یک سلسله‌مراتب سازمان‌دهی می‌شوند بر اساس:

□ سرعت

□ هزینه

□ فرآر بودن

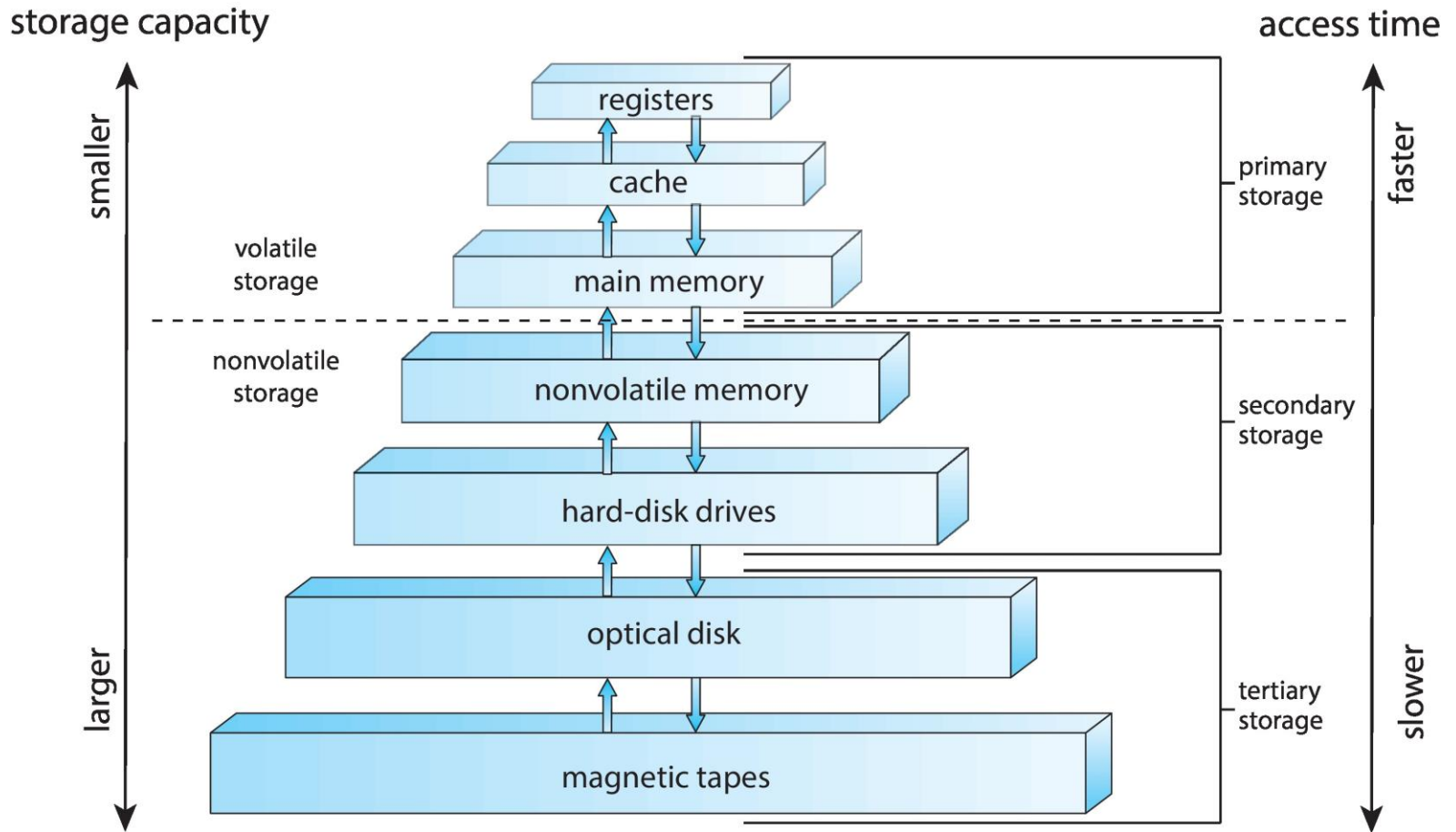
□ **Caching** – کپی کردن اطلاعات در یک سیستم ذخیره‌سازی سریع‌تر. حافظه‌ی اصلی می‌تواند به‌عنوان یک حافظه‌نهمان (Cache) برای ذخیره‌سازی ثانویه دیده شود. **دراپور دستگاه** برای هر کنترل‌کننده‌ی دستگاه ورودی/خروجی جهت مدیریت I/O وجود دارد.

این دراپور یک رابط یکپارچه بین کنترل‌کننده‌ی دستگاه و کرنل فراهم می‌کند.



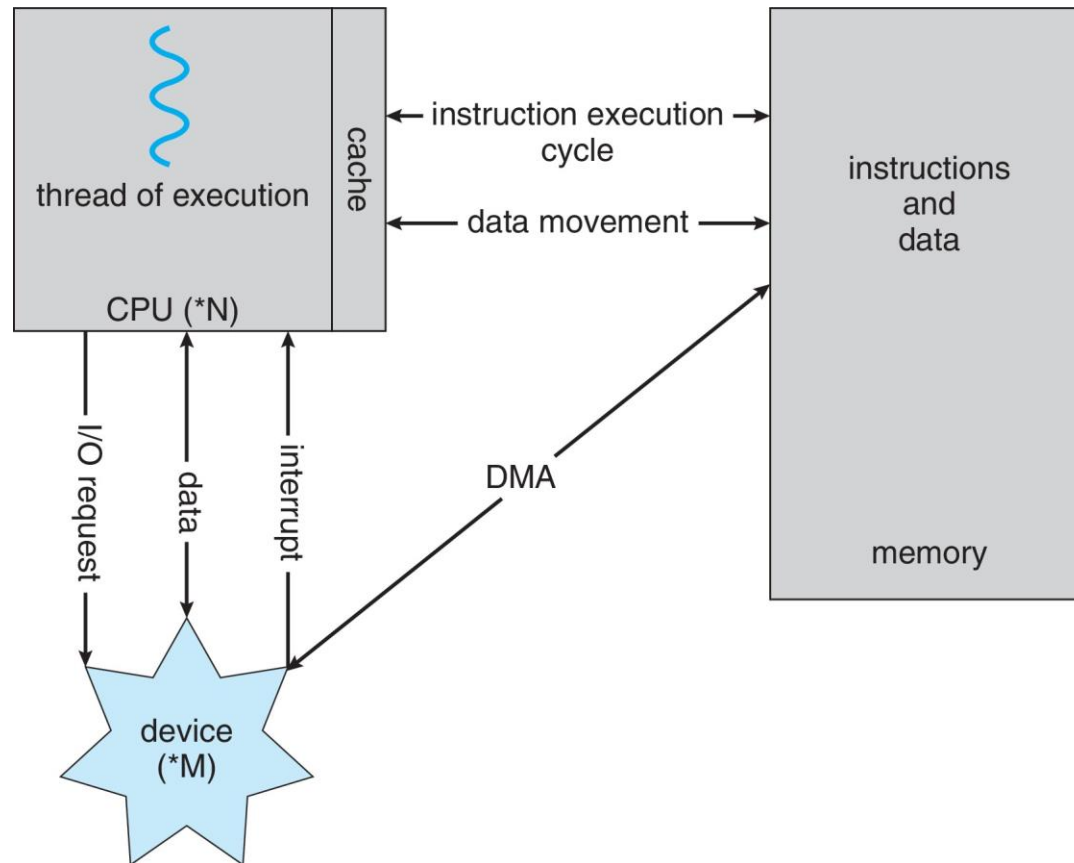


Storage-Device Hierarchy





How a Modern Computer Works



A von Neumann architecture





Direct Memory Access Structure

□ دسترسی مستقیم به حافظه (DMA - Direct Memory Access)

این یک دستگاه سخت‌افزاری است که به برخی از زیرسیستم‌های سخت‌افزاری اجازه می‌دهد به حافظه‌ی اصلی (RAM) بدون دخالت پردازنده مرکزی (CPU) دسترسی پیدا کنند.

این ویژگی مفید است زیرا پردازنده می‌تواند به انجام کارهای دیگر بپردازد، در حالی که انتقال داده‌های I/O که معمولاً کند است، در حال انجام است.

سیستم‌های سخت‌افزاری که از DMA استفاده می‌کنند:

- درایو دیسک
- کارت‌های گرافیک
- کارت‌های شبکه
- کارت‌های صدا





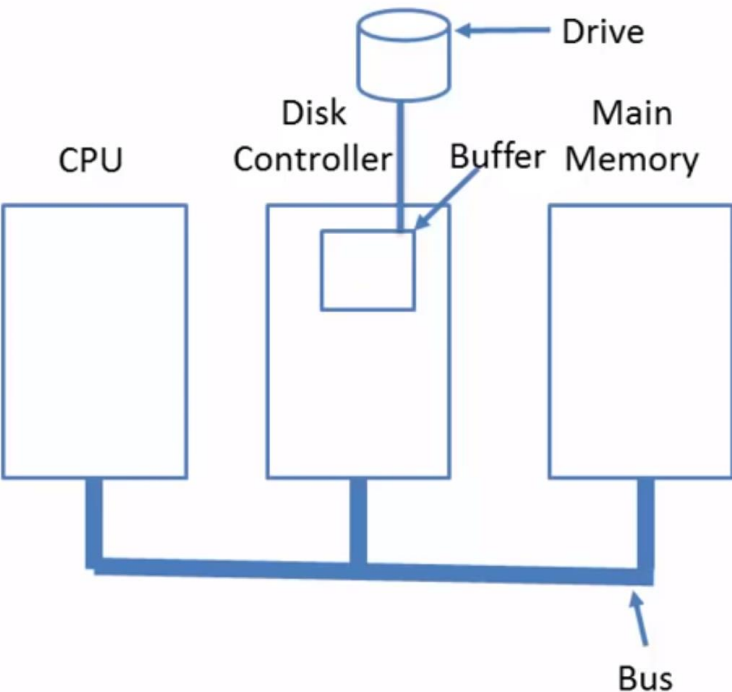
Direct Memory Access Structure

□ مزایای استفاده از DMA

- بدون استفاده از DMA ، زمانی که پردازنده از I/O برنامه‌ریزی شده استفاده می‌کند، برای کل مدت عملیات خواندن یا نوشتن اشغال می‌شود و نمی‌تواند کار دیگری انجام دهد.
- با استفاده از DMA ، پردازنده ابتدا انتقال را آغاز می‌کند، سپس به انجام سایر عملیات می‌پردازد و در نهایت از کنترل‌کننده‌ی DMA وقفه‌ای دریافت می‌کند که نشان‌دهنده‌ی پایان عملیات است.

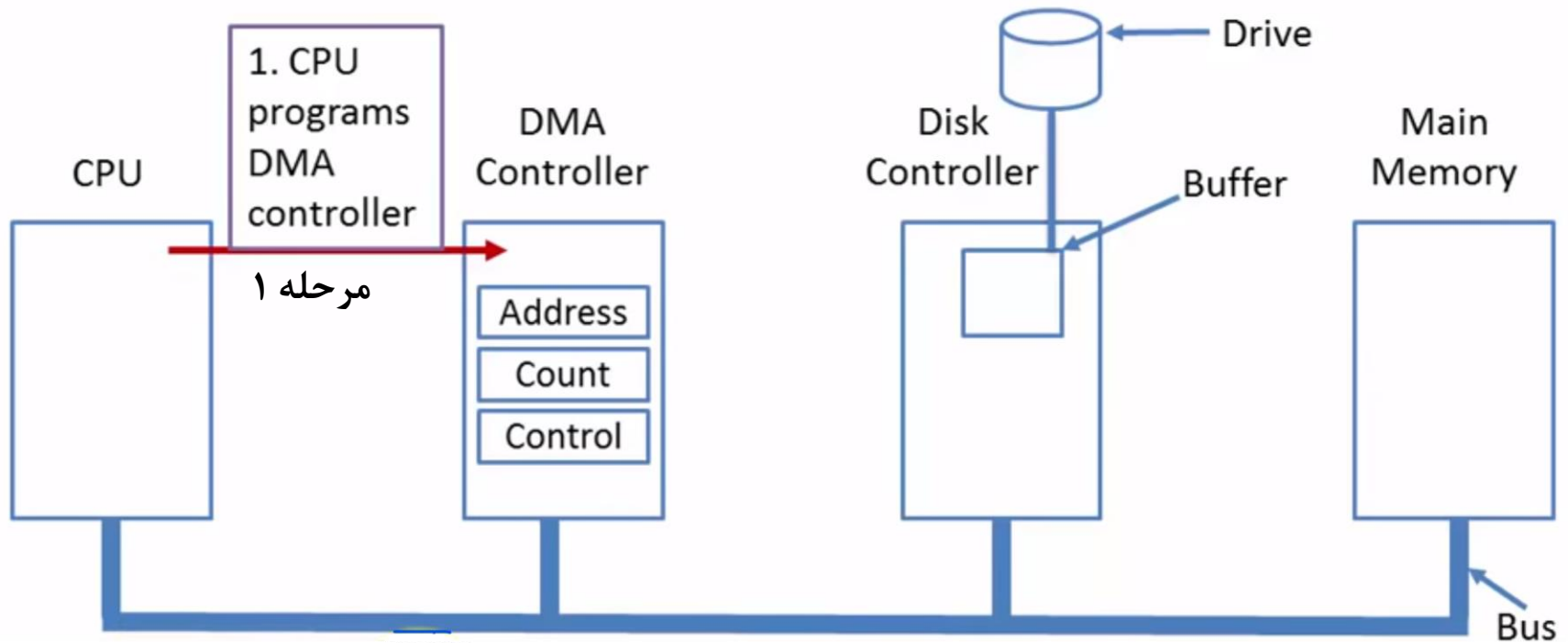


Disk read-write without a DMA



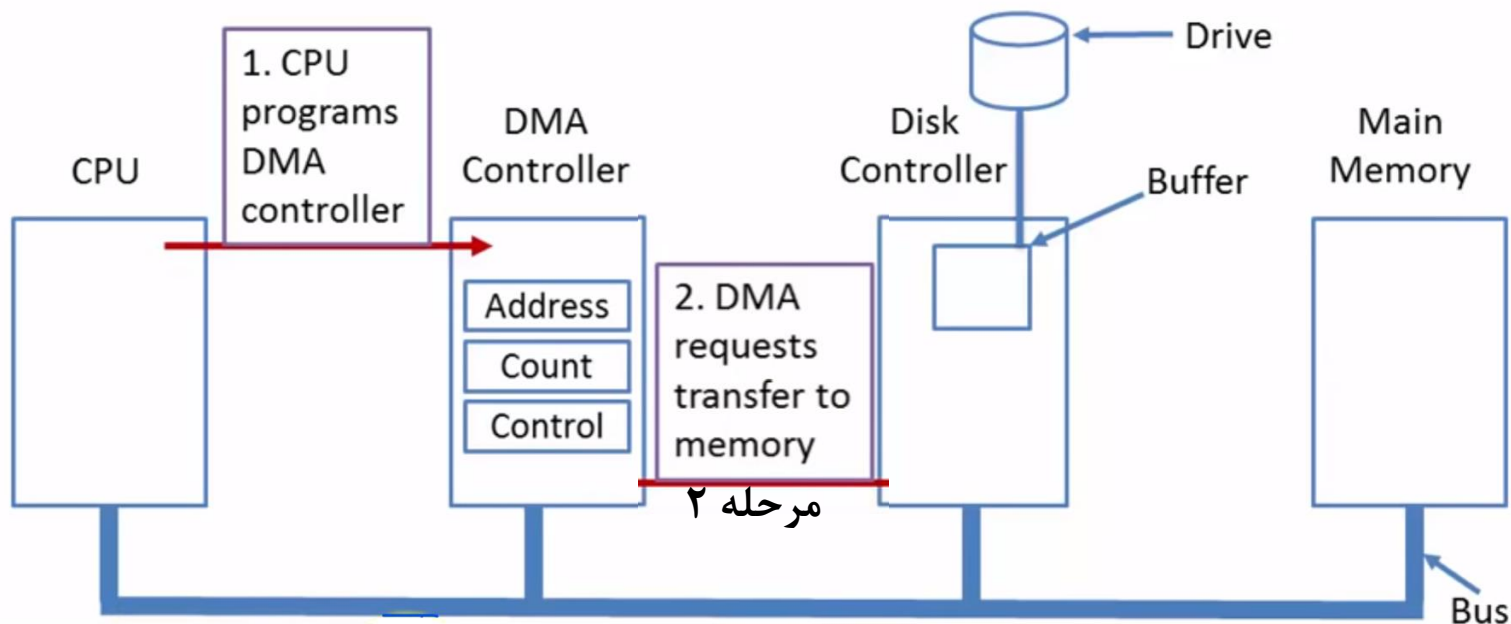
- کنترل کننده دیسک، بلوک را به صورت سریالی، بیت به بیت از درایو می خواند تا زمانی که کل بلوک در بافر کنترل کننده ذخیره شود.
- سپس یک مقدار بررسی (checksum) را محاسبه می کند تا اطمینان حاصل کند که هیچ خطای خواندنی رخ نداده است.
- پس از تأیید داده ها، کنترل کننده یک وقفه ایجاد می کند تا سیستم عامل بتواند بلوک را از بافر کنترل کننده بخواند (به صورت یک بایت یا یک کلمه در هر مرحله) با اجرای یک حلقه.
- پس از خواندن هر بخش از بلوک از ثبات (register) دستگاه کنترل کننده، سیستم عامل آن را در حافظه اصلی ذخیره می کند.

DMA فرآیند خواندن داده از دیسک توسط



مرحله ۱:

- پردازنده ابتدا کنترل کننده‌ی DMA را با تنظیم ثبات‌های آن پیکربندی می‌کند تا بداند چه چیزی را به کجا انتقال دهد.
- سپس فرمانی به کنترل کننده‌ی دیسک می‌دهد تا داده‌ها را بخواند و در بافر داخلی خود ذخیره کند.

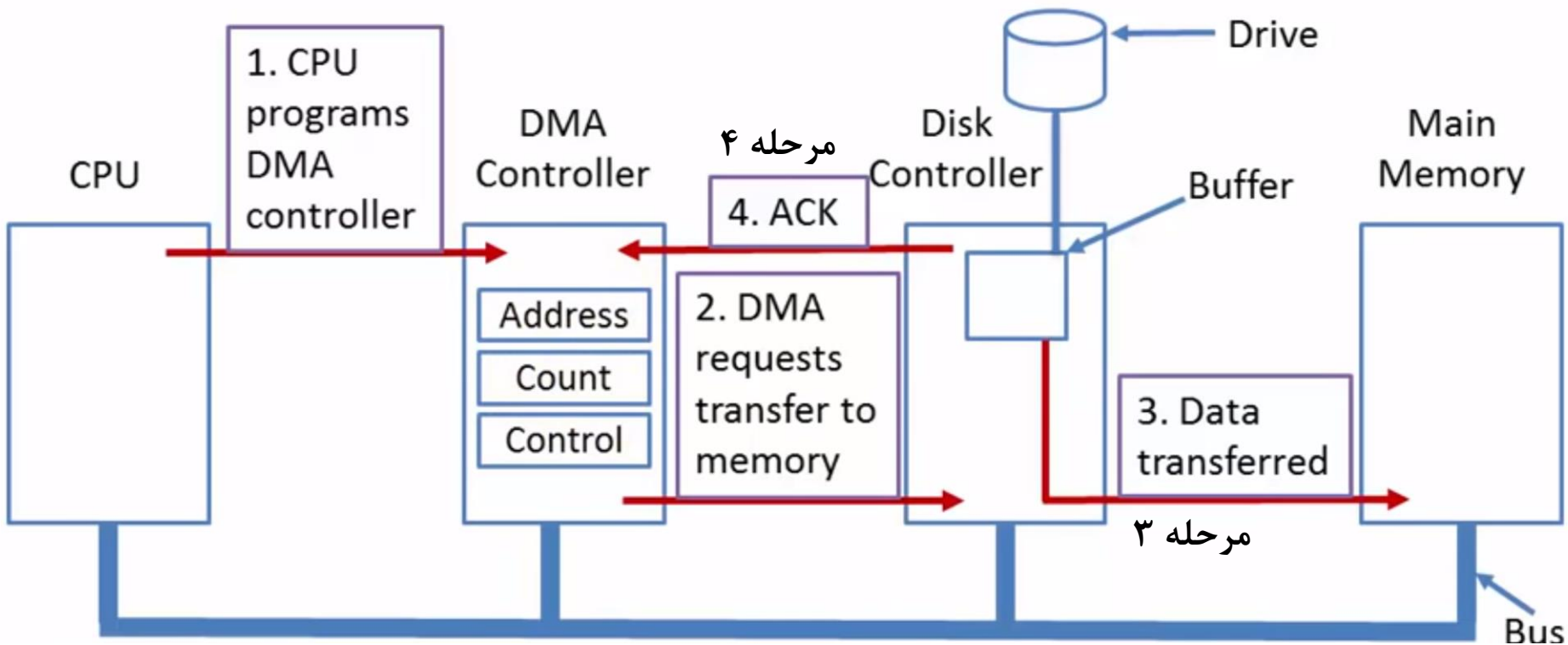


مرحله ۲:

- کنترل کننده DMA با ارسال درخواست خواندن، انتقال داده را آغاز می کند.

مرحله ۳:

- داده های بافر شده از کنترل کننده دیسک به حافظه اصلی منتقل می شوند.



مرحله ۴:

- پس از تکمیل انتقال، کنترل کننده دیسک سیگنال «ASK» را به کنترل کننده DMA ارسال می کند.
- کنترل کننده DMA آدرس حافظه را افزایش داده و مقدار شمارش بایت را کاهش می دهد.
- در صورت وجود داده های بیشتر، مراحل ۲ تا ۴ تکرار می شوند تا تمام داده ها منتقل شوند. در نهایت، وقفه ای برای اطلاع پردازنده ارسال می شود.

Modes of Bus operation

مدهای عملکرد گذرگاه‌ها (Buses)

مد انتقال کلمه‌ای: (Word-at-a-time Mode)

- DMA برای انتقال یک کلمه درخواست می‌کند و آن را دریافت می‌کند.
- اگر پردازنده در همان لحظه به گذرگاه نیاز داشته باشد، باید منتظر بماند.
- به این روش *Cycle* گفته می‌شود.

مد انتقال بلوکی: (Block Mode)

- کنترل‌کننده‌ی DMA گذرگاه را در اختیار می‌گیرد، داده‌ها را در چندین انتقال ارسال می‌کند و سپس گذرگاه را آزاد می‌کند.
- این روش *Burst Mode* نام دارد و کارایی بیشتری نسبت به *Cycle* دارد.



Operating-System Operations

- بوت استرپ (Bootstrap Program) و فرآیند راه اندازی سیستم
- برنامه‌ای ساده برای راه اندازی سیستم و بارگذاری کرنل
- کرنل اجرا شده و سرویس‌های سیستمی (Daemons) را راه اندازی می‌کند
- سیستم عامل مبتنی بر وقفه‌ها است





Multiprogramming (Batch system)

- یک کاربر نمی‌تواند همیشه CPU و دستگاه‌های ورودی/خروجی را مشغول نگه دارد.
- چندبرنامگی (Multiprogramming) کارها (کد و داده) را سازمان‌دهی می‌کند تا CPU همیشه یک کار برای اجرا داشته باشد.
- زیرمجموعه‌ای از کل کارهای سیستم در حافظه نگه داشته می‌شود.
- یک کار از بین کارهای موجود انتخاب شده و از طریق زمان‌بندی کار (Job Scheduling) اجرا می‌شود.
- هنگامی که یک کار باید منتظر بماند (مثلاً برای ورودی/خروجی)، سیستم عامل به کار دیگری سوئیچ می‌کند.





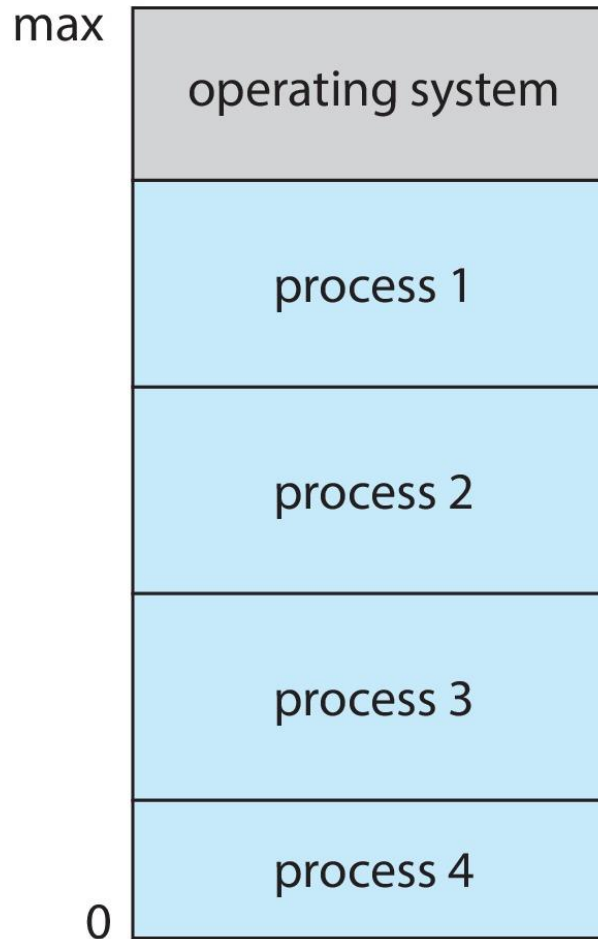
Multitasking (Timesharing)

- CPU کارها را به طور مداوم تغییر می دهد تا کاربران بتوانند هنگام اجرای هر کار با آن تعامل داشته باشند و این فرآیند منجر به محاسبات تعاملی (Interactive Computing) می شود.
 - زمان پاسخ باید کمتر از ۱ ثانیه باشد.
 - هر کاربر حداقل یک برنامه در حال اجرا در حافظه دارد که به آن **پردازه (Process)** گفته می شود.
 - اگر چندین کار به طور همزمان آماده اجرا باشند ، زمان بندی CPU (CPU Scheduling) انجام می شود.
 - اگر پردازه ها در حافظه جا نشوند، عملیات تعویض (Swapping) آنها را جابجا می کند تا اجرا شوند.
 - حافظه مجازی (Virtual Memory) امکان اجرای پردازه هایی را فراهم می کند که به طور کامل در حافظه اصلی قرار ندارند.





Memory Layout for Multiprogrammed System





Dual-mode Operation

- عملکرد دو حالت (Dual-Mode) به سیستم عامل اجازه می دهد که از خود و سایر اجزای سیستم محافظت کند.

○ دو حالت موجود عبارتند از **حالت کاربر (User Mode)** و **حالت کرنل (Kernel Mode)**.

- بیت حالت (Mode Bit) توسط سخت افزار ارائه می شود تا امکان تشخیص اجرای کد کاربر یا کد کرنل فراهم شود.

○ هنگامی که یک کاربر در حال اجراست ، بیت حالت "کاربر" است.

○ هنگامی که کد کرنل در حال اجراست ، بیت حالت "کرنل" است.

○ چگونه تضمین کنیم که کاربر به طور دستی بیت حالت را به "کرنل" تغییر ندهد؟

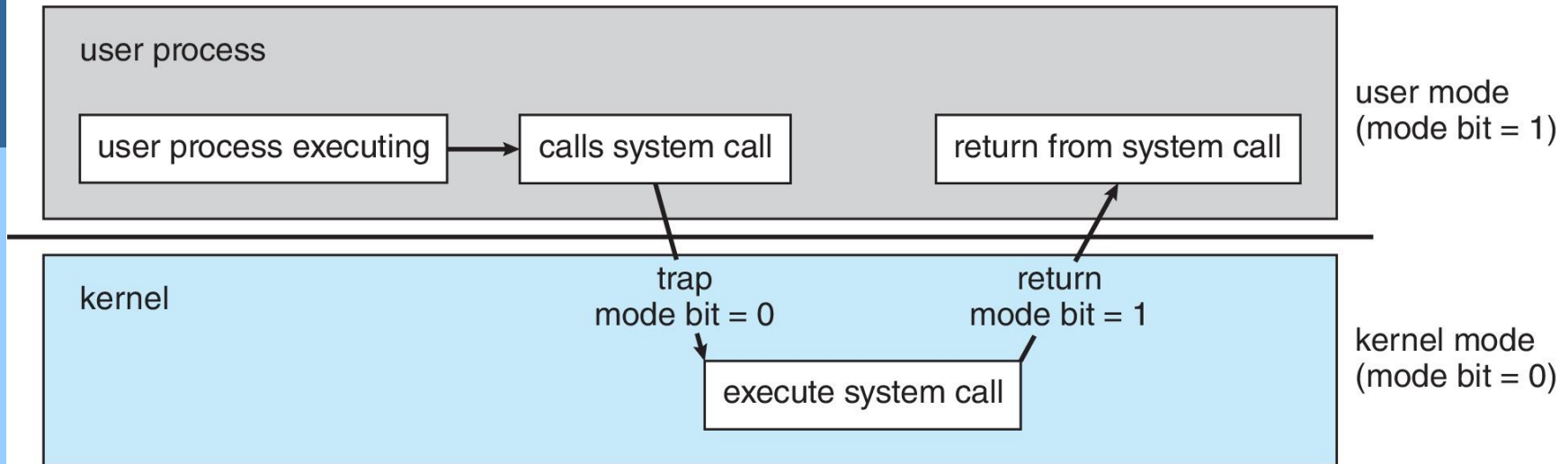
- **فراخوانی سیستمی (System Call)** حالت را به کرنل تغییر می دهد و بازگشت از این فراخوانی، آن را مجدداً به حالت کاربر تنظیم می کند.

○ برخی از دستورات به عنوان **دستورات ویژه (Privileged Instructions)** تعیین شده اند و فقط در حالت کرنل قابل اجرا هستند.





Transition from User to Kernel Mode



در زمان راه اندازی سیستم (Boot Time) ، سخت افزار در **حالت کرنل** شروع به کار می کند.

سپس سیستم عامل بارگذاری شده و اجرای برنامه های کاربر را در **حالت کاربر** آغاز می کند.





Timer

- تایمر برای جلوگیری از حلقه‌های بی‌نهایت (یا پردازش‌هایی که منابع را اشغال می‌کنند)
 - تایمر برای ایجاد وقفه در رایانه پس از مدت‌زمان مشخص تنظیم می‌شود.
 - یک شمارنده وجود دارد که با ساعت فیزیکی کاهش می‌یابد.
 - سیستم‌عامل مقدار این شمارنده را تنظیم می‌کند (یک دستور ویژه).
 - هنگامی که مقدار شمارنده به صفر برسد، یک وقفه (**Interrupt**) ایجاد می‌شود.
 - این تایمر قبل از زمان‌بندی پردازش تنظیم می‌شود تا کنترل سیستم را بازپس گیرد یا پردازش‌های را که از زمان اختصاص یافته فراتر رفته است، خاتمه دهد.





Process Management

□ یک پردازش (Process) برنامه‌ای در حال اجرا است و به عنوان یک واحد کاری در سیستم شناخته می‌شود.
برنامه یک موجودیت غیرفعال است، در حالی که پردازش یک موجودیت فعال است.

- پردازش برای انجام وظایف خود به منابع نیاز دارد:
 - CPU، حافظه، ورودی/خروجی، فایل‌ها
 - داده‌های اولیه برای مقداردهی اولیه
- پایان پردازش نیازمند بازپس‌گیری منابع قابل بازیابی است.





Process Management

□ یک پردازش تک‌نخی (Single-Threaded Process) تنها یک شمارنده برنامه دارد که محل دستور بعدی برای اجرا را مشخص می‌کند.

- پردازش به صورت ترتیبی، یک دستور را در هر لحظه اجرا می‌کند تا به پایان برسد.

□ یک پردازش چندنخی (Multi-Threaded Process) برای هر نخ یک شمارنده برنامه جداگانه دارد.

□ معمولاً سیستم دارای پردازش‌های متعددی است که برخی مربوط به کاربر و برخی مربوط به سیستم عامل هستند و به طور هم‌زمان روی یک یا چند CPU اجرا می‌شوند.

- هم‌زمانی (Concurrency) از طریق تسهیم CPU (Multiplexing) بین پردازش‌ها و نخ‌ها انجام می‌شود.





Process Management Activities

- مدیریت پروسس (پردازه) در سیستم عامل
- سیستم عامل مسئول فعالیت‌های زیر در ارتباط با مدیریت پردازه است:
 - ایجاد و حذف پردازه‌های کاربر و سیستم
 - متوقف کردن و از سرگیری پردازه‌ها
 - فراهم‌سازی مکانیزم‌هایی برای همگام‌سازی پردازه‌ها (Process Synchronization)
 - فراهم‌سازی مکانیزم‌هایی برای ارتباط پردازه‌ها (Process Communication)
 - مدیریت بن‌بست (Deadlock Handling)





Memory Management

□ مدیریت حافظه

- برای اجرای یک برنامه، تمام (یا بخشی از) دستورالعمل‌ها باید در حافظه باشند.
- تمام (یا بخشی از) داده‌هایی که برنامه نیاز دارد، باید در حافظه باشند.
- **مدیریت حافظه** مشخص می‌کند چه چیزی در حافظه باشد و چه زمانی بارگذاری یا جابجا شود.
- بهینه‌سازی استفاده از CPU و پاسخ‌گویی به کاربران.





Memory Management

□ فعالیت‌های مدیریت حافظه

- پیگیری بخش‌های مورد استفاده حافظه و کاربرانی که از آن استفاده می‌کنند.
- تصمیم‌گیری درباره انتقال پردازنده‌ها (یا بخش‌هایی از آن‌ها) و داده‌ها به داخل و خارج از حافظه.
- تخصیص و آزادسازی فضای حافظه بر اساس نیاز.





File-system Management

□ مدیریت فایل در سیستم عامل

□ سیستم عامل یک نمای منطقی و یکنواخت از ذخیره سازی اطلاعات فراهم می کند:

- ویژگی های فیزیکی را به یک واحد ذخیره سازی منطقی **فایل** تبدیل می کند.
- هر رسانه ذخیره سازی تحت کنترل یک دستگاه خاص است (مانند دیسک، نوار مغناطیسی).
- ویژگی های متغیر شامل سرعت دسترسی، ظرفیت، نرخ انتقال داده و روش دسترسی (ترتیبی یا تصادفی) است.





File-system Management

□ مدیریت فایل

- فایل‌ها معمولاً درون دایرکتوری‌ها (**Directories**) سازماندهی می‌شوند.
- کنترل دسترسی (**Access Control**) مشخص می‌کند که چه کسانی می‌توانند به چه فایل‌هایی دسترسی داشته باشند.
- فعالیت‌های سیستم‌عامل شامل موارد زیر است:
 - ایجاد و حذف فایل‌ها و دایرکتوری‌ها
 - ارائه عملیات پایه برای مدیریت فایل‌ها و دایرکتوری‌ها
 - نگاشت فایل‌ها روی حافظه ذخیره‌سازی ثانویه
 - پشتیبان‌گیری از فایل‌ها روی رسانه‌های ذخیره‌سازی پایدار (غیر فرار)





Mass-Storage Management

□ مدیریت دیسک

- معمولاً دیسک‌ها برای ذخیره داده‌هایی که در حافظه اصلی جا نمی‌شوند یا برای مدت طولانی نیاز است، استفاده می‌شوند.
- کل سرعت عملکرد رایانه به زیرسیستم دیسک و الگوریتم‌های آن بستگی دارد.

□ فعالیت‌های سیستم‌عامل در مدیریت دیسک

- سوار کردن (Mounting) و پیاده کردن (Unmounting) سیستم فایل
- مدیریت فضای آزاد (Free-Space Management)
- تخصیص حافظه ذخیره‌سازی
- زمان‌بندی دیسک (Disk Scheduling)
- تقسیم‌بندی (Partitioning)
- تأمین مکانیزم‌های حفاظتی (Protection)





Caching

□ مدیریت کش (Cache)

- اطلاعات در حال استفاده به طور موقت از ذخیره سازی کندتر به ذخیره سازی سریع تر (کش) منتقل می شود.
- ابتدا کش بررسی می شود تا مشخص شود آیا اطلاعات مورد نظر در آن وجود دارد:
 - اگر اطلاعات در کش باشد -> مستقیماً از آن استفاده می شود (سریع).
 - در غیر این صورت -> داده ها در کش بارگذاری شده و از آنجا استفاده می شوند.
- اندازه کش کوچک تر از کل فضای ذخیره سازی است، بنابراین مدیریت کش یک مسئله مهم در طراحی سیستم است:
 - اندازه کش و سیاست جایگزینی داده ها (Replacement Policy)





Characteristics of Various Types of Storage

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

Movement between levels of storage hierarchy can be explicit or implicit

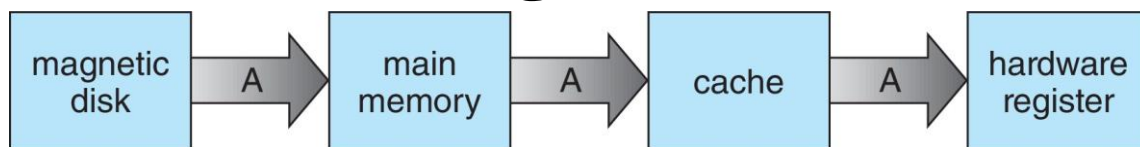




Migration of data “A” from Disk to Register

□ مدیریت همزمانی در سیستم‌های چندپردازشی

- در محیط‌های چندوظیفه‌ای (Multitasking)، سیستم باید آخرین مقدار ذخیره‌شده را بدون توجه به مکان ذخیره‌سازی، بازیابی کند.



- در محیط‌های چندپردازنده‌ای (Multiprocessor Environment)، سخت‌افزار باید یکپارچگی کش (Cache Coherency) را تضمین کند تا همه CPUها مقدار به‌روز را در کش خود داشته باشند.
- در محیط‌های توزیع‌شده (Distributed Environment)، این چالش پیچیده‌تر می‌شود:

- نسخه‌های متعددی از یک داده ممکن است وجود داشته باشد.
- راهکارهای مختلفی برای حل این مشکل وجود دارد.





Migration of data “A” from Disk to Register

□ یکپارچگی کش (Cache Coherency)

- یک مکانیزم یکپارچگی داده‌ها است که تضمین می‌کند تمام پردازنده‌ها در یک سیستم چندپردازنده‌ای، یک نمای واحد و سازگار از داده‌ها مشاهده کنند.
- زمانی که یک CPU مقدار ذخیره‌شده در کش خود را تغییر می‌دهد، مقدار معادل در کش سایر پردازنده‌ها باید به‌روزرسانی یا نامعتبر شود تا یکپارچگی حفظ شود.
- مثال: دو پردازنده **P1** و **P2** دارای کش‌های مستقل هستند و هر دو به یک مکان حافظه **M** دسترسی دارند:
 - مقدار اولیه **M = 10**: در کش‌های **P1**، **P2** و حافظه اصلی
 - اگر **P1** مقدار **M** را در کش خود به **20** تغییر دهد، بدون وجود پروتکل یکپارچگی کش، **P2** همچنان مقدار قدیمی (۱۰) را می‌بیند.
 - اما با وجود پروتکل یکپارچگی کش، مقدار جدید **M = 20** در حافظه اصلی و کش **P2** به‌روزرسانی می‌شود.





I/O Subsystem

□ مدیریت ورودی/خروجی (I/O Management)

□ یکی از وظایف سیستم عامل، پنهان سازی پیچیدگی های سخت افزار از کاربر است.

زیرسیستم I/O مسئول موارد زیر است:

- مدیریت حافظه I/O شامل بافرینگ (ذخیره موقت داده ها هنگام انتقال)، کشینگ (ذخیره بخش هایی از داده در حافظه سریع تر برای بهبود عملکرد) و اسپولینگ (همپوشانی خروجی یک کار با ورودی کار دیگر)
- رابط استاندارد درایورهای دستگاه
- درایورهای مخصوص سخت افزارهای مختلف





Protection and Security

□ امنیت و حفاظت در سیستم عامل

• **حفاظت: (Protection)** مکانیزمی برای کنترل دسترسی پردازنده‌ها و کاربران به منابع سیستم.

• **امنیت: (Security)** دفاع در برابر حملات داخلی و خارجی شامل:

- حملات منع سرویس (DoS)
- ویروس‌ها و کرم‌های رایانه‌ای
- سرقت هویت و سرقت خدمات





Protection and Security

□ سیستم‌ها معمولاً ابتدا کاربران را شناسایی می‌کنند تا

مشخص شود هر کاربر چه مجوزهایی دارد:

- هر کاربر دارای شناسه کاربری (**User ID**) است که به فایل‌ها و پردازش‌های وی مرتبط می‌شود.

- شناسه گروهی (**Group ID**) اجازه مدیریت مجموعه‌ای از کاربران را فراهم می‌کند.

- افزایش سطح دسترسی (**Privilege Escalation**) به کاربران امکان می‌دهد موقتاً مجوزهایی بیشتری دریافت کنند.





Virtualization

□ مجازی سازی (Virtualization) و شبیه سازی (Emulation)

- مجازی سازی به سیستم عامل امکان می دهد که برنامه ها را در محیط های دیگر اجرا کند.
- **شبیه سازی (Emulation):** زمانی استفاده می شود که نوع پردازنده مبدا و مقصد متفاوت باشد مثلاً PowerPC به Intel x86
 - معمولاً کندترین روش است.
 - در صورتی که زبان برنامه نویسی مستقیماً به کد بومی ترجمه نشود، **تفسیر (Interpretation)** انجام می شود.

□ مجازی سازی: سیستم عامل مهمان و میزبان هر دو به صورت بومی برای پردازنده کامپایل شده اند.

- مثال **VMware**: که سیستم عامل **Windows XP** را به عنوان مهمان روی **Windows XP** میزبان اجرا می کند.
- **مدیر ماشین مجازی (VMM - Virtual Machine Manager)** سرویس های مجازی سازی را ارائه می دهد.





Virtualization (cont.)

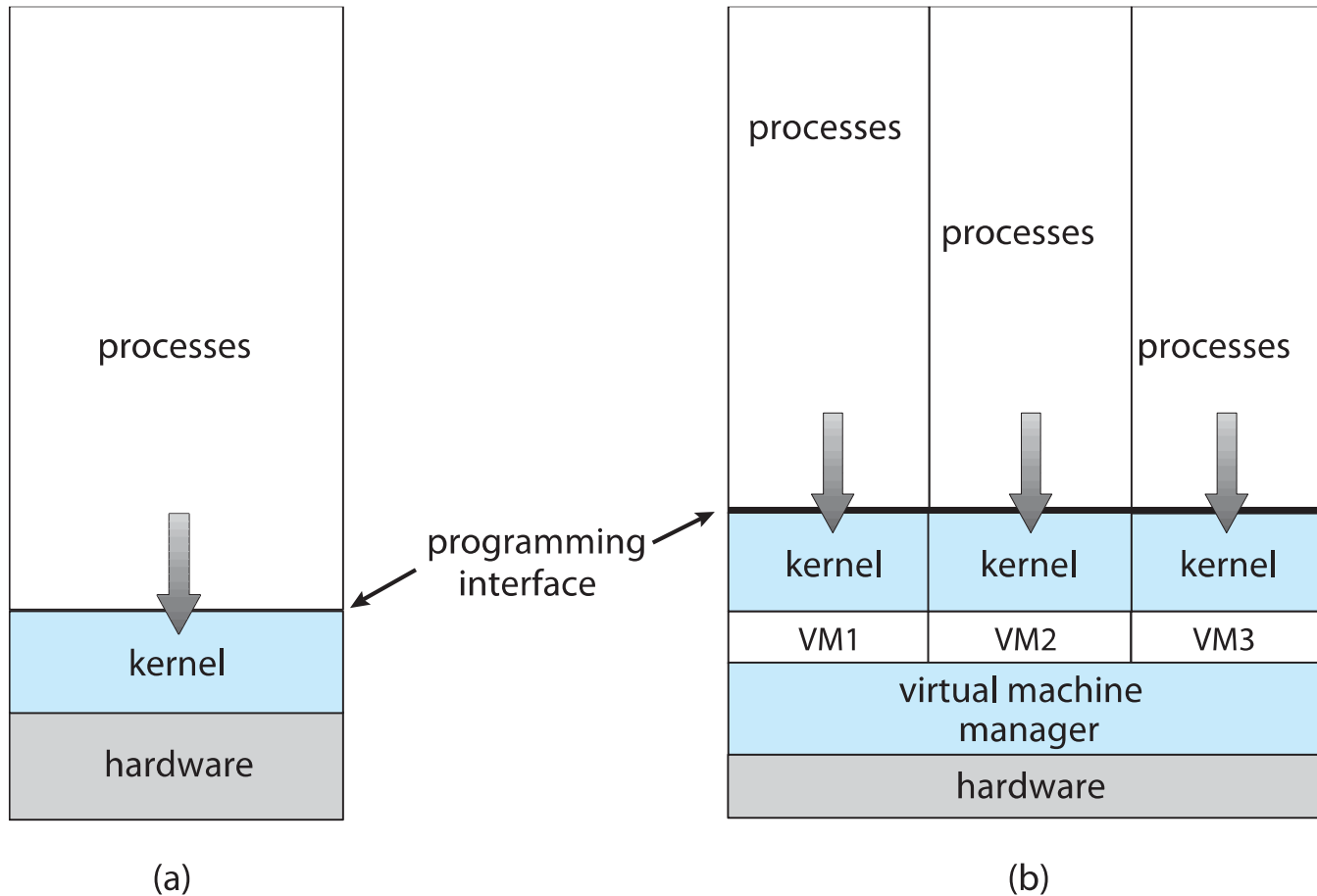
□ کاربردهای مجازی سازی

- اجرای چندین سیستم عامل روی لپ تاپ یا دسکتاپ مثلاً اجرای Windows در Mac OS
- توسعه اپلیکیشن برای چند سیستم عامل بدون نیاز به سخت افزارهای متعدد
- تست نرم افزارها در محیط های مختلف بدون نیاز به سخت افزار اضافی
- مدیریت محیط های پردازی در مراکز داده
- برخی VMM ها مانند **VMware ESX** و **Citrix XenServer** مستقیماً روی سخت افزار اجرا می شوند و نیازی به سیستم عامل میزبان ندارند.





Computing Environments - Virtualization





Distributed Systems

□ رایانش توزیع شده (Distributed Computing)

• مجموعه‌ای از سیستم‌های مستقل که از طریق شبکه به هم متصل شده‌اند.

• پروتکل TCP/IP رایج‌ترین روش ارتباط است.

• سیستم عامل شبکه‌ای (Network OS) امکاناتی برای تعامل بین سیستم‌ها فراهم می‌کند.

• امکان تبادل پیام بین سیستم‌ها

• ایجاد توهم یک سیستم یکپارچه برای کاربر



Network Operating System VS Distributed Systems

□ نمونه‌هایی از سیستم‌های توزیع شده

• **Windows Server:** مدیریت منابع شبکه مانند فایل‌ها، چاپگرها و کاربران به صورت متمرکز.

• **Android OS:** سیستم عاملی برای دستگاه‌های موبایل که با ساعت‌های هوشمند و اسپیکرهای هوشمند ارتباط برقرار کرده و تجربه‌ای یکپارچه برای کاربر ایجاد می‌کند.



Computer-System Architecture





Computer-System Architecture

□ پردازش چندگانه (Multiprocessing)

• بیشتر سیستم‌ها دارای یک پردازنده عمومی (General-Purpose Processor) هستند.

• اما امروزه استفاده از پردازنده‌های چندگانه (Multiprocessors) رو به افزایش است.

□ مزایای پردازش چندگانه:

1. افزایش توان عملیاتی (Throughput)

2. مقیاس‌پذیری اقتصادی (Economy of Scale)

3. افزایش قابلیت اطمینان – (Reliability) کاهش اثر خرابی‌ها (Fault Tolerance)

□ انواع سیستم‌های چندپردازنده‌ای:

1. پردازش نامتقارن: (Asymmetric Multiprocessing - AMP) هر پردازنده یک وظیفه خاص را اجرا می‌کند.

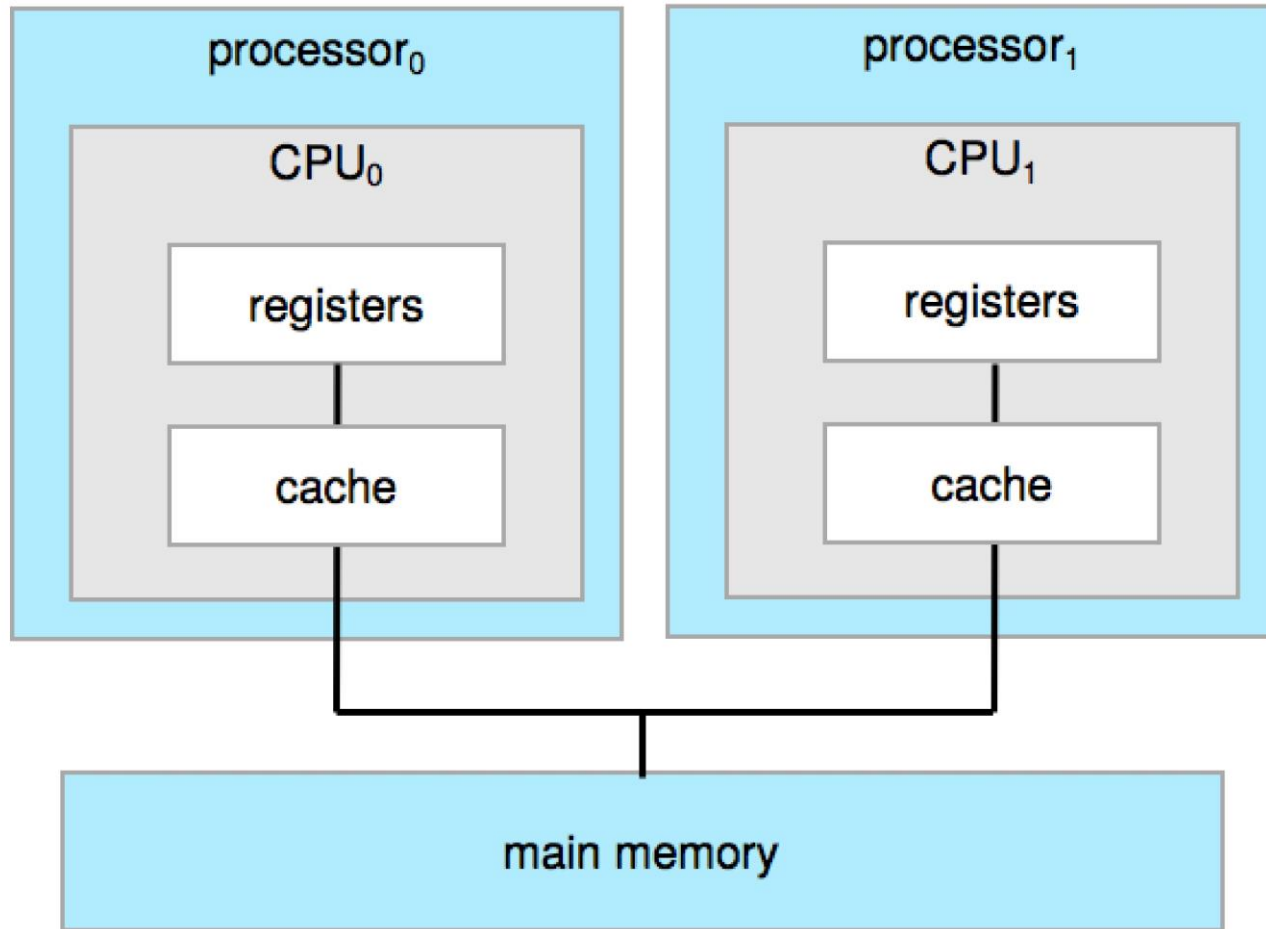
2. پردازش متقارن: (Symmetric Multiprocessing - SMP) تمام پردازنده‌ها

وظایف مشابهی دارند.





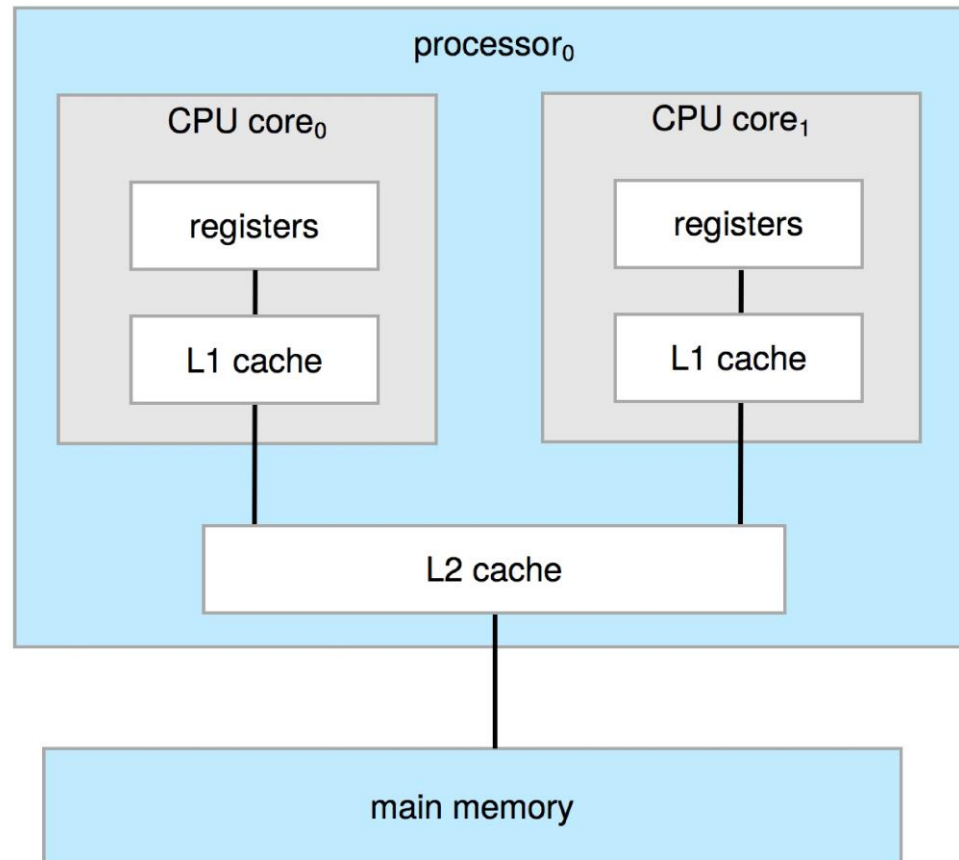
Symmetric Multiprocessing Architecture





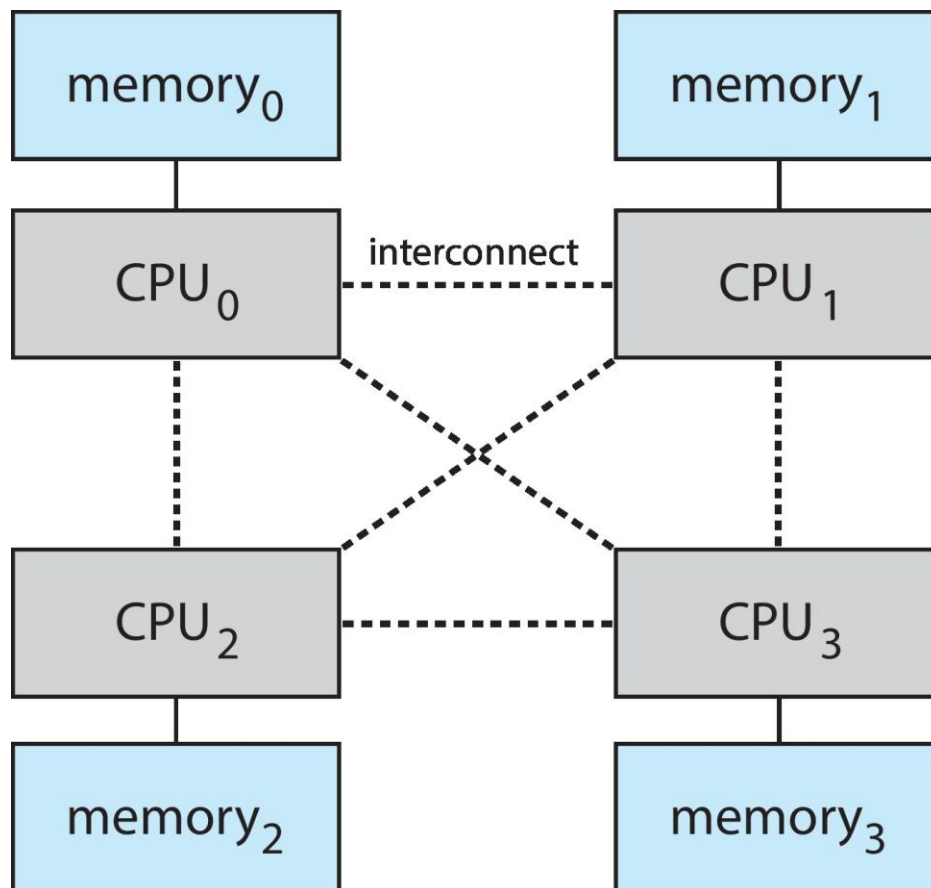
Dual-Core Design

- ❑ Multi-chip and **multicore**
- ❑ Systems containing all chips
 - ❑ Chassis containing multiple separate systems





Non-Uniform Memory Access System





Clustered Systems

خوشه‌بندی (Clustering) □

- مانند سیستم‌های چندپردازنده‌ای، اما شامل چندین سیستم مستقل که با هم کار می‌کنند.
 - معمولاً از **SAN (Storage-Area Network)** برای ذخیره‌سازی اشتراکی استفاده می‌شود.
 - ارائه سرویس‌های در دسترس پذیری بالا (**High Availability**) که در برابر خرابی‌ها مقاوم هستند.

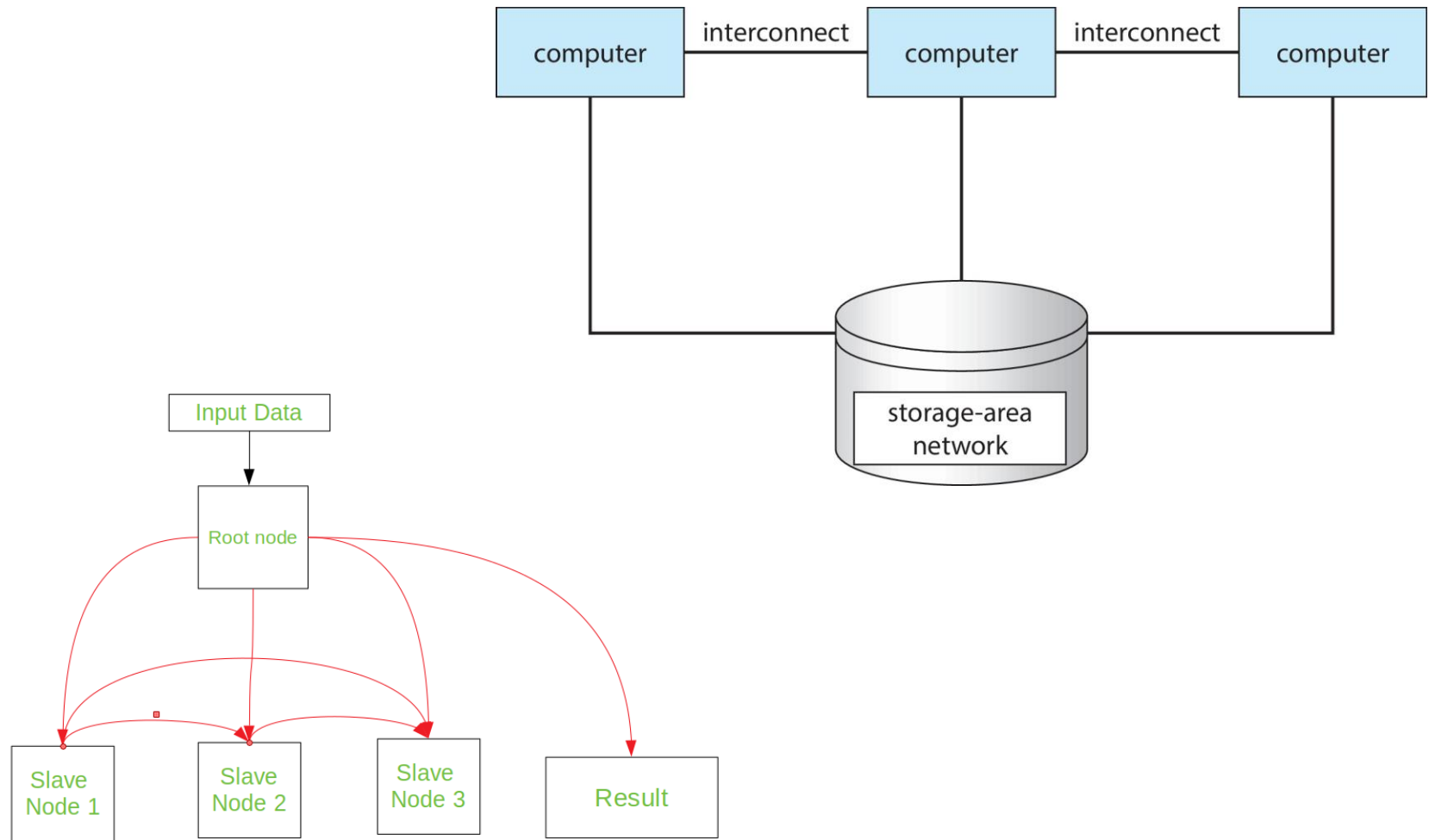
انواع خوشه‌بندی: □

- **خوشه‌بندی نامتقارن (Asymmetric Clustering):** یک ماشین در حالت آماده‌باش است و فقط هنگام خرابی وارد عمل می‌شود.
- **خوشه‌بندی متقارن (Symmetric Clustering):** چندین ماشین هم‌زمان در حال اجرا هستند و یکدیگر را پایش می‌کنند.
- **خوشه‌های پردازش با کارایی بالا (HPC - High Performance Computing):** برای اجرای برنامه‌های موازی.





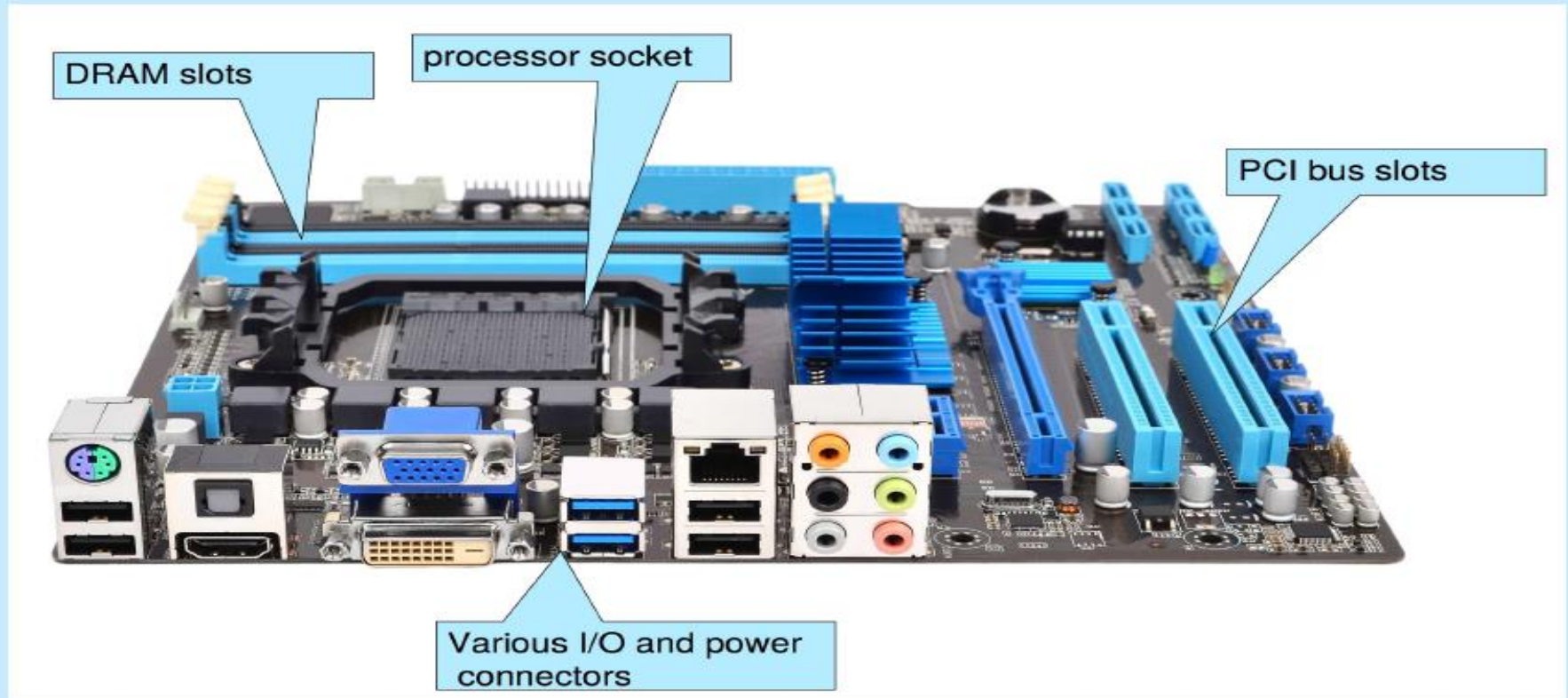
Clustered Systems





PC Motherboard

Consider the desktop PC motherboard with a processor socket shown below:



This board is a fully-functioning computer, once its slots are populated. It consists of a processor socket containing a CPU, DRAM sockets, PCIe bus slots, and I/O connectors of various types. Even the lowest-cost general-purpose CPU contains multiple cores. Some motherboards contain multiple processor sockets. More advanced computers allow more than one system board, creating NUMA systems.



Computer System Environments





Computing Environments

□ مدل‌های محاسباتی مختلف

1. سنتی (Traditional)

2. موبایل (Mobile)

3. کلاینت-سرور (Client-Server)

4. هم‌تا به هم‌تا (Peer-to-Peer)

5. رایانش ابری (Cloud Computing)

6. سیستم‌های بی‌درنگ (Real-time Embedded)





Mobile Computing

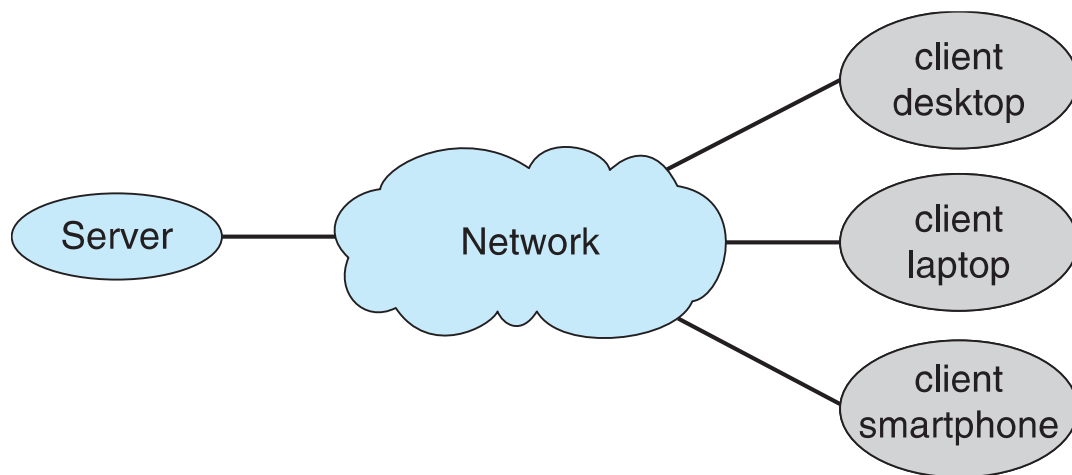
- تفاوت موبایل و سیستم‌های سنتی
 - تلفن‌های هوشمند، تبلت‌ها و سایر دستگاه‌های همراه
 - تفاوت عملکردی با لپ‌تاپ‌های سنتی شامل:
 - ویژگی‌های اضافی مانند GPS و ژيروسکوپ
 - اجرای اپلیکیشن‌های واقعیت افزوده (AR)
 - استفاده از شبکه‌های بی‌سیم IEEE 802.11 یا داده‌های سلولی
- سیستم‌عامل‌های پیشرو:
 - Apple iOS
 - Google Android





Client Server Computing

- تحول سیستم‌های رایانه‌ای
- ترمینال‌های ساده (Dumb Terminals) جای خود را به رایانه‌های هوشمند (Smart PCs) داده‌اند.
- امروزه بسیاری از سیستم‌ها به‌عنوان سرور عمل می‌کنند.
 - سیستم‌های سرویس‌دهنده محاسباتی (Compute-Server): ارائه خدماتی مانند پایگاه داده به کلاینت‌ها.
 - سیستم‌های سرویس‌دهنده فایل (File-Server): ارائه فضای ذخیره‌سازی و بازیابی فایل‌ها برای کلاینت‌ها.

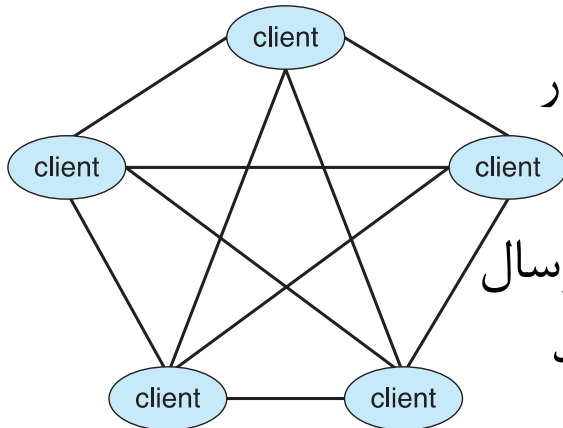




Peer-to-Peer

□ مدل دیگری از سیستم توزیع شده

- در P2P تفاوتی بین کلاینت‌ها و سرورها وجود ندارد
 - در عوض، تمام گره‌ها به عنوان همتا در نظر گرفته می‌شوند
 - هر گره می‌تواند به عنوان کلاینت، سرور یا هر دو عمل کند
 - گره باید به شبکه P2P متصل شود



- سرویس خود را با سرویس جستجوی مرکزی در شبکه ثبت می‌کند، یا
- درخواست سرویس را از طریق پروتکل کشف ارسال کرده و به درخواست‌های سرویس پاسخ می‌دهد
- مثال‌ها شامل Napster و Gnutella، و VoIP مانند

Skype هستند





Cloud Computing

□ محاسبات ابری

- ارائه محاسبات، ذخیره‌سازی و حتی برنامه‌ها به عنوان سرویس از طریق یک شبکه
- توسعه منطقی از مجازی‌سازی است زیرا از مجازی‌سازی به عنوان پایه برای عملکرد خود استفاده می‌کند
- Amazon EC2 هزاران سرور، میلیون‌ها ماشین مجازی و پتابایت‌ها ذخیره‌سازی در سراسر اینترنت دارد و پرداخت بر اساس استفاده است





Cloud Computing – Many Types

□ انواع مختلف ابر

- **ابر عمومی** – در دسترس هر کسی که حاضر به پرداخت باشد از طریق اینترنت
- **ابر خصوصی** – توسط یک شرکت برای استفاده داخلی همان شرکت اجرا می شود
- **ابر هیبریدی** – شامل اجزای ابر عمومی و خصوصی
- **نرم افزار به عنوان سرویس (SaaS)** – یک یا چند برنامه در دسترس از طریق اینترنت (مانند پردازشگر متن)
- **پلتفرم به عنوان سرویس (PaaS)** – پشته نرم افزاری آماده برای استفاده از برنامه ها از طریق اینترنت (مثلاً سرور پایگاه داده)
- **زیرساخت به عنوان سرویس (IaaS)** – سرورها یا ذخیره سازی در دسترس از طریق اینترنت (مانند ذخیره سازی برای استفاده پشتیبان)

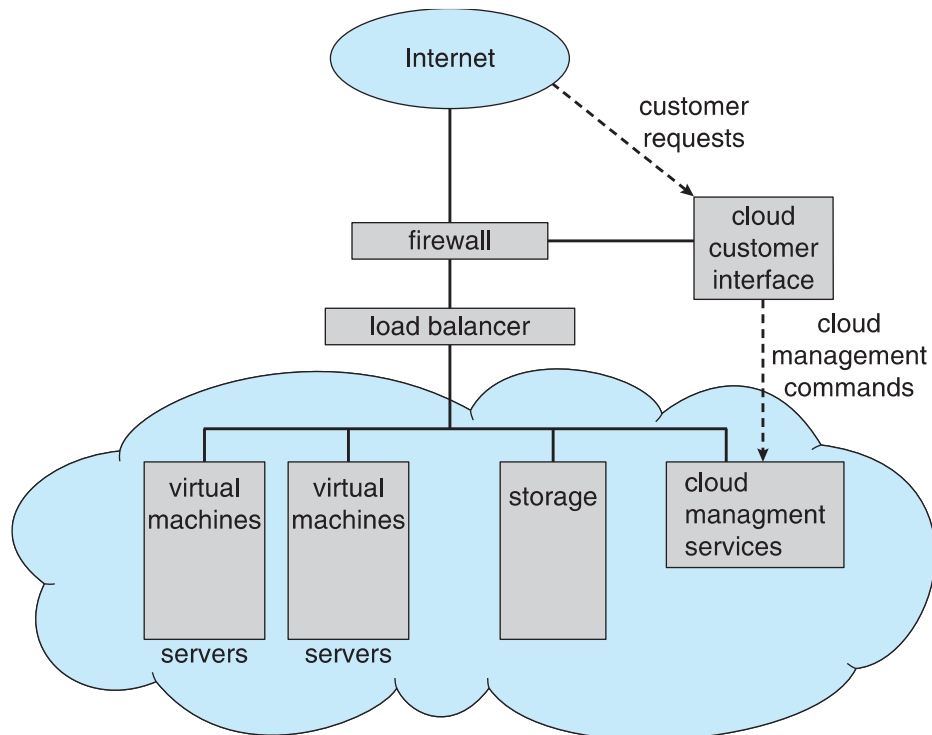




Cloud Computing (cont.)

□ محیط‌های محاسبات ابری

- محیط‌های محاسبات ابری شامل سیستم‌عامل‌های سنتی، به علاوه ماشین‌های مجازی، به علاوه ابزارهای مدیریت ابری
 - اتصال اینترنتی به امنیت نیاز دارد مانند فایروال‌ها
 - متعادل‌کننده بار ترافیک را بین چندین برنامه تقسیم می‌کند





Real-Time Embedded Systems

□ سیستم‌های تعبیه (جاساز) شده بلادرنگ

- رایج‌ترین نوع کامپیوترها
 - بسیار متنوع هستند، سیستم‌عامل‌های خاص، سیستم‌عامل‌های محدود و سیستم‌عامل‌های بلادرنگ
 - استفاده آن‌ها در حال گسترش است
- بسیاری دیگر از محیط‌های محاسباتی خاص نیز وجود دارند
 - بعضی از آن‌ها سیستم‌عامل دارند، برخی بدون سیستم‌عامل وظایف خود را انجام می‌دهند
- سیستم‌عامل بلادرنگ دارای محدودیت‌های زمانی مشخص و ثابت است
 - پردازش باید درون محدودیت زمانی انجام شود
 - عملکرد صحیح تنها در صورتی است که محدودیت‌ها رعایت شوند





Free and Open-Source Operating Systems

□ نرم افزارهای متن باز

- سیستم عامل هایی که به صورت کد منبع در دسترس قرار دارند، نه فقط در قالب باینری بسته و اختصاصی
- مخالف با حرکت حفاظت از کپی و مدیریت حقوق دیجیتال (DRM)
- این حرکت توسط بنیاد نرم افزار آزاد (FSF) و ریچارد استالمن آغاز شد که مجوز عمومی گنو (GPL) با مفهوم "کپی لفت" دارد
 - نرم افزار آزاد و نرم افزار متن باز دو ایده متفاوت هستند که توسط گروه های مختلف از افراد حمایت می شوند

• [لینک فلسفه نرم افزار آزاد](http://gnu.org/philosophy/open-source-misses-the-point.html/)

► <http://gnu.org/philosophy/open-source-misses-the-point.html/>



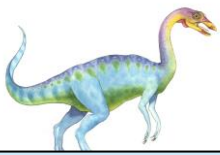


Free and Open-Source Operating Systems

□ نمونه‌ها شامل GNU/Linux (و BSD UNIX شامل هسته macOS) ، و بسیاری دیگر

- می‌توان از ماشین مجازی مانند VMware Player رایگان برای ویندوز Virtualbox متن‌باز و رایگان در بسیاری از پلتفرم‌ها - [لینک Virtualbox](#) استفاده کرد
- برای اجرای سیستم‌عامل‌های مهمان به منظور کاوش





The Study of Operating Systems

- جنبش نرم افزار متن باز باعث شده است که بسیاری از سیستم عامل ها هم در قالب کد منبع و هم باینری (قابل اجرا) در دسترس قرار گیرند • فهرست سیستم عامل هایی که در هر دو فرمت در دسترس هستند شامل لینوکس، BSD UNIX، Solaris و بخشی از macOS می شود • دسترسی به کد منبع به ما این امکان را می دهد که سیستم عامل ها را از درون بررسی کنیم. سوالاتی که قبلاً تنها با مطالعه مستندات یا رفتار یک سیستم عامل قابل پاسخ گویی بودند، اکنون می توانیم با بررسی کد خود سیستم عامل پاسخ دهیم • فهرست گسترده اما ناقص پروژه های سیستم عامل متن باز از [این لینک](#) در دسترس است • علاوه بر این، ظهور مجازی سازی به عنوان یک عملکرد رایج (و اغلب رایگان) رایانه ای، امکان اجرای چندین سیستم عامل را بر روی یک سیستم اصلی فراهم می کند • ظهور سیستم عامل های متن باز همچنین مهاجرت از دانش آموز به توسعه دهنده سیستم عامل را آسان تر کرده است. با برخی دانش، تلاش و اتصال به اینترنت، یک دانش آموز حتی می تواند توزیع جدیدی از سیستم عامل ایجاد کند.

End of Chapter 1

