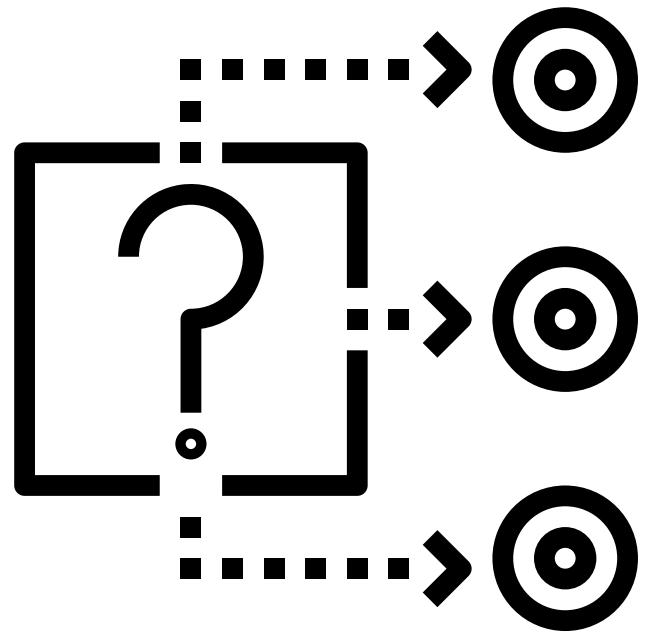


OVERVIEW

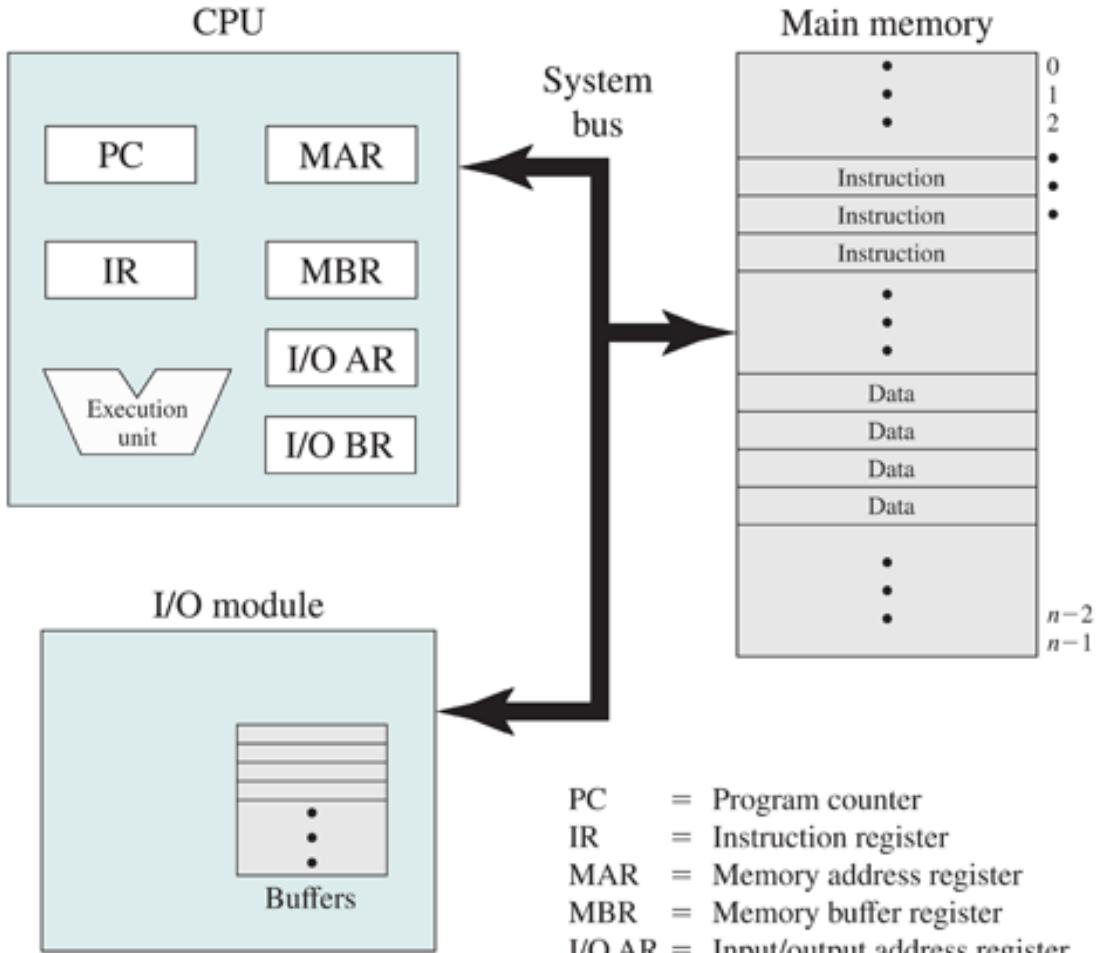
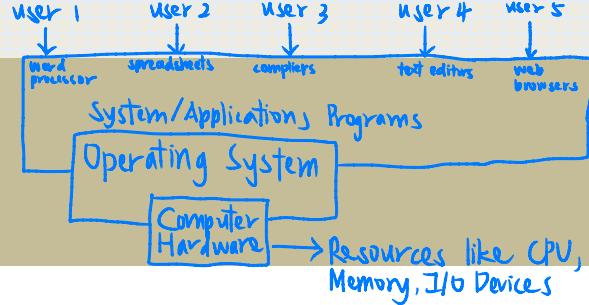
DCS4103 Operating System



Computer System



Basic Elements of Computer System



- Controls the operation 操作
- Data processing



Main Memory /
Real Memory /
Primary Memory

- Stores data and programs



- Move data between the computer and its external environment.



- Communication among processors, main memory, and I/O modules



Registers

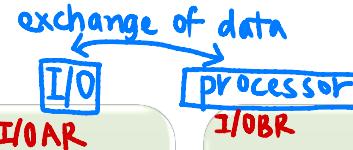


Memory Address Register (MAR)

指定
Specifies the address in memory for the next read or write

缓存
Memory Buffer Register (MBR)

Contains the data to be written into memory



I/O Address Register (I/OAR)

Specifies a particular I/O device

I/O Buffer Register (I/OBR)

Exchange of data between an I/O module and the processor



Program Counter

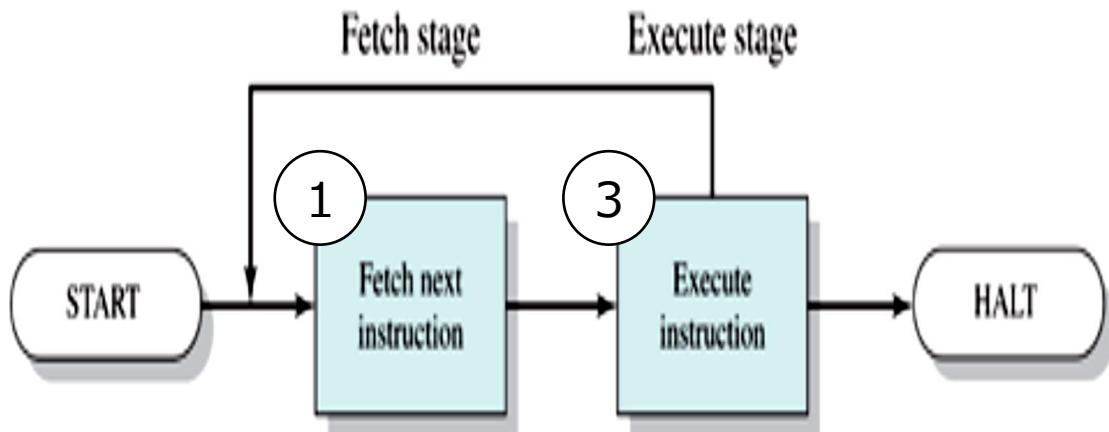
Hold the address of the next instruction to be executed
执行

Instruction Register

Hold the instruction that is currently being executed

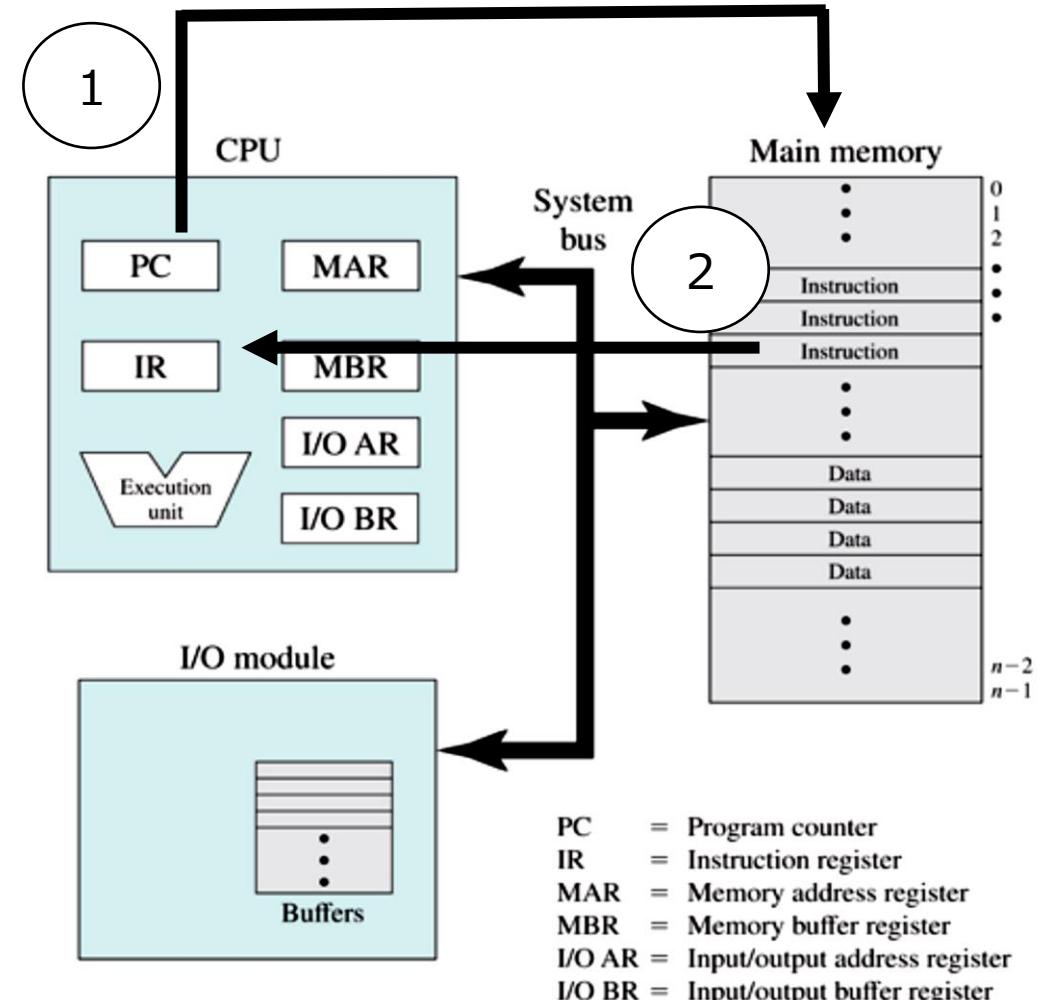


Instruction Cycles



Execution :

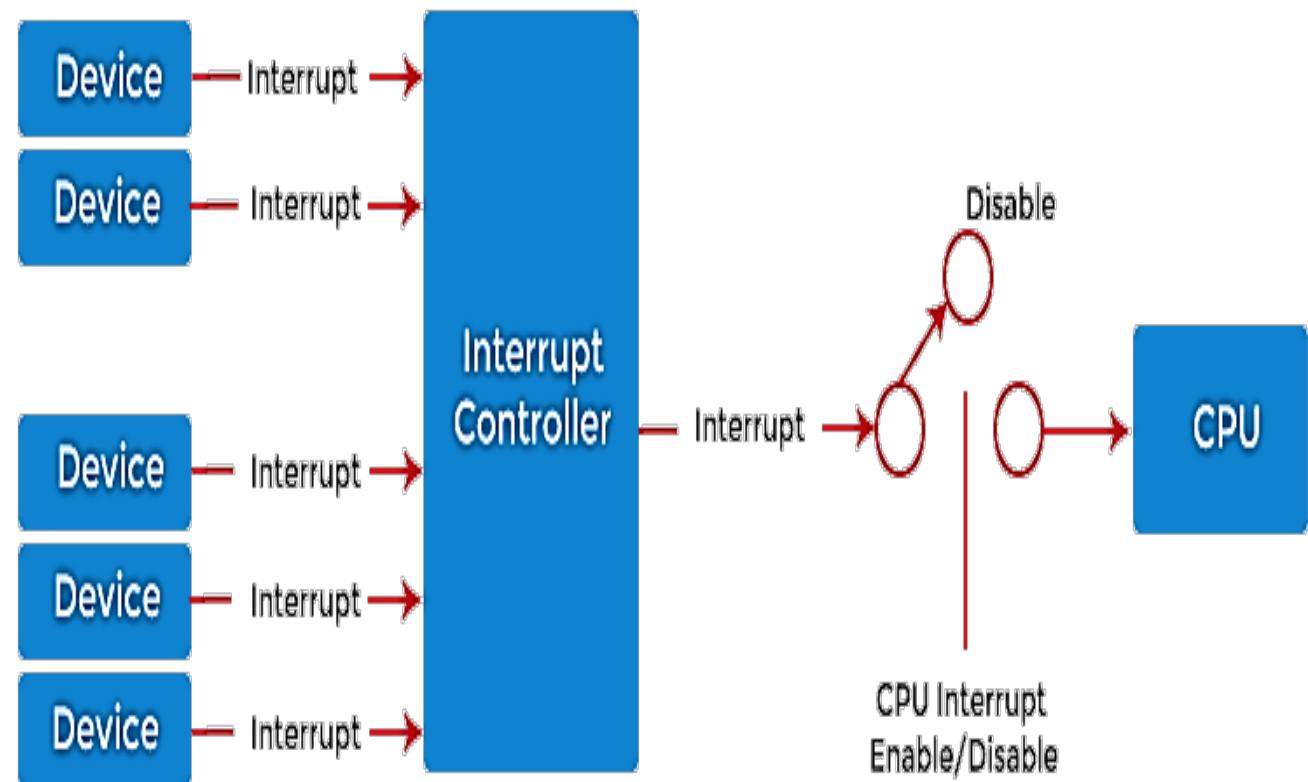
- Processor-memory
- Processor-I/O
- Data processing
- Control





Interrupts

- A signal emitted by hardware or software when a process or an event needs immediate attention
释放
- Alerts the processor to a high-priority process requiring interruption of the current working process
提醒 *need to* *中断*
- To improve processor utilization
利用
- In the operating system, interrupts are essential
 - It give a reliable technique for the OS to communicate and react to their surroundings.
可靠



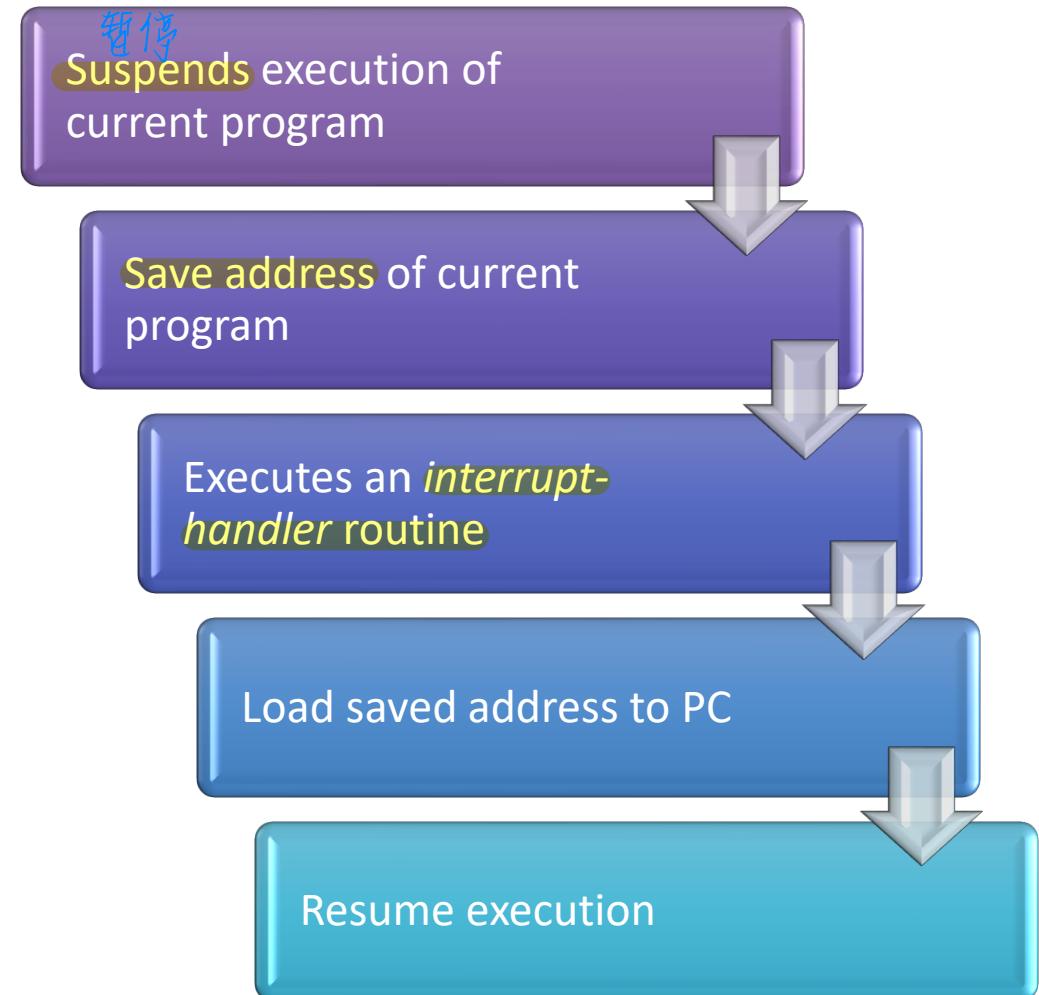
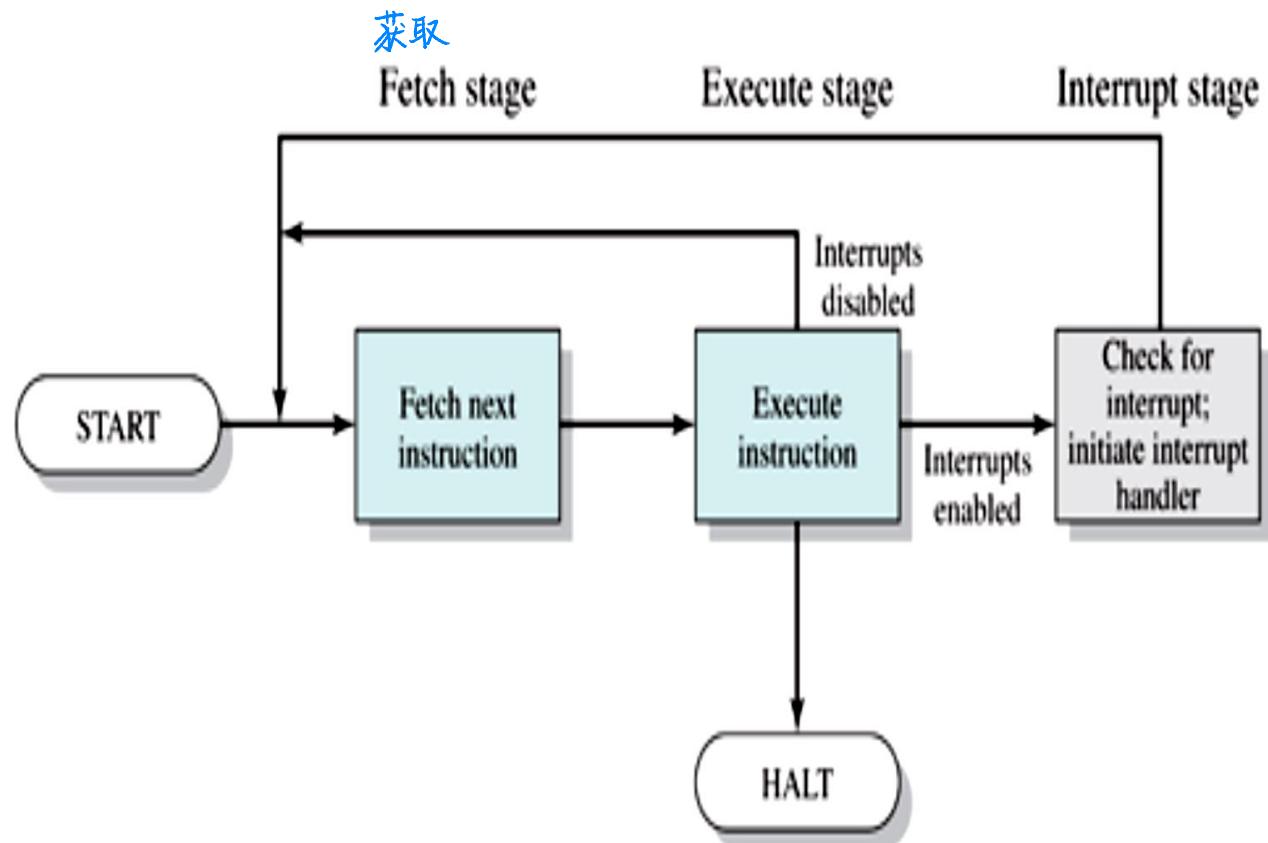


Interrupts

Classes of Interrupts	Description
Program	<p>Result of an instruction 执行 指令 execution</p> <ul style="list-style-type: none">• Arithmetic overflow 算术溢流• Division by zero 非法• Attempt to execute an illegal machine instruction• Reference outside a user's allowed memory space.
Timer	<p>Timer within the processor</p> <ul style="list-style-type: none">• Perform certain functions on a regular basis 执行 定期
I/O	<ul style="list-style-type: none">• To signal normal completion of an operation• To signal a variety of error conditions
Hardware failure	<ul style="list-style-type: none">• Power failure 奇偶• Memory parity error

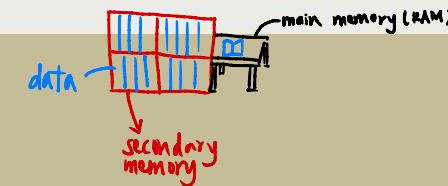


Interrupts and Instruction Cycle





Memory Hierarchy



Faster access time
访问时间

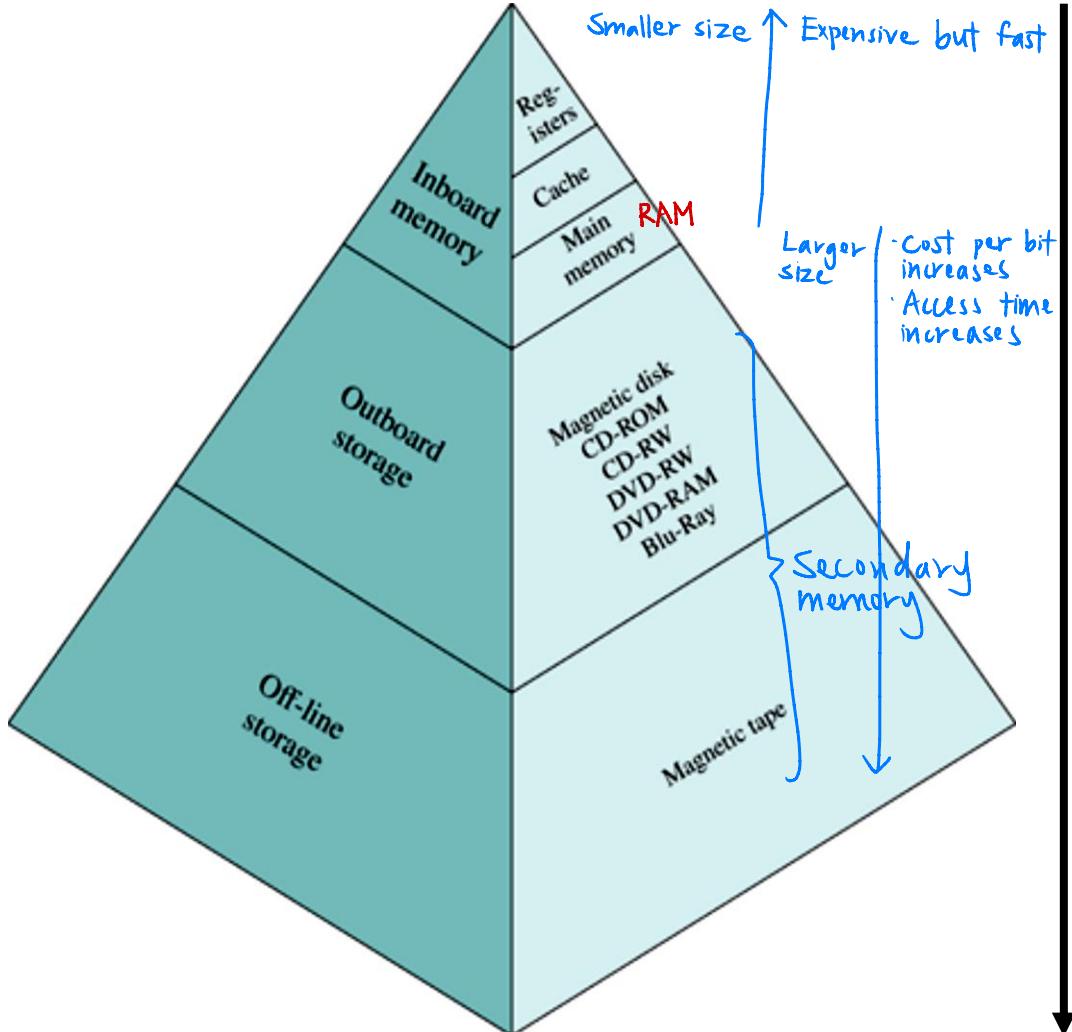
- Greater cost per bit 单价

Greater capacity 容量

- Smaller cost per bit

Greater capacity

- Slower access speed

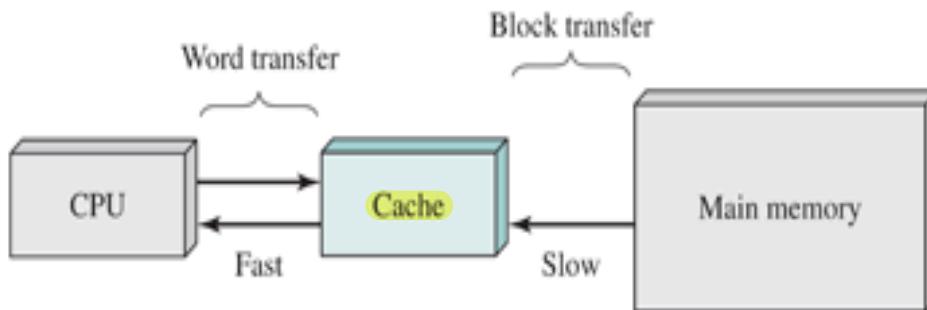


- Decreasing cost per bit
- Increasing capacity
- Increasing access time
- Decreasing frequency of access to the memory by the processor

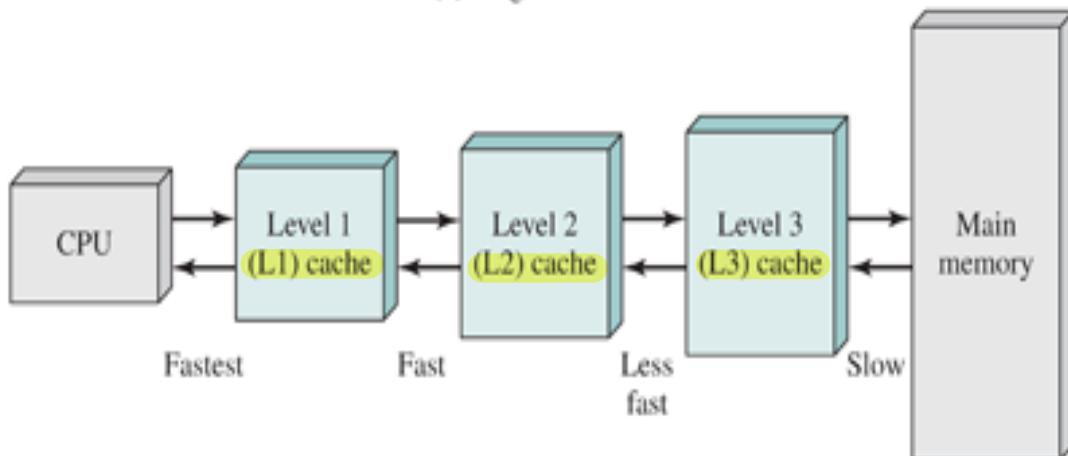
↓ 处理器访问
内存的频率



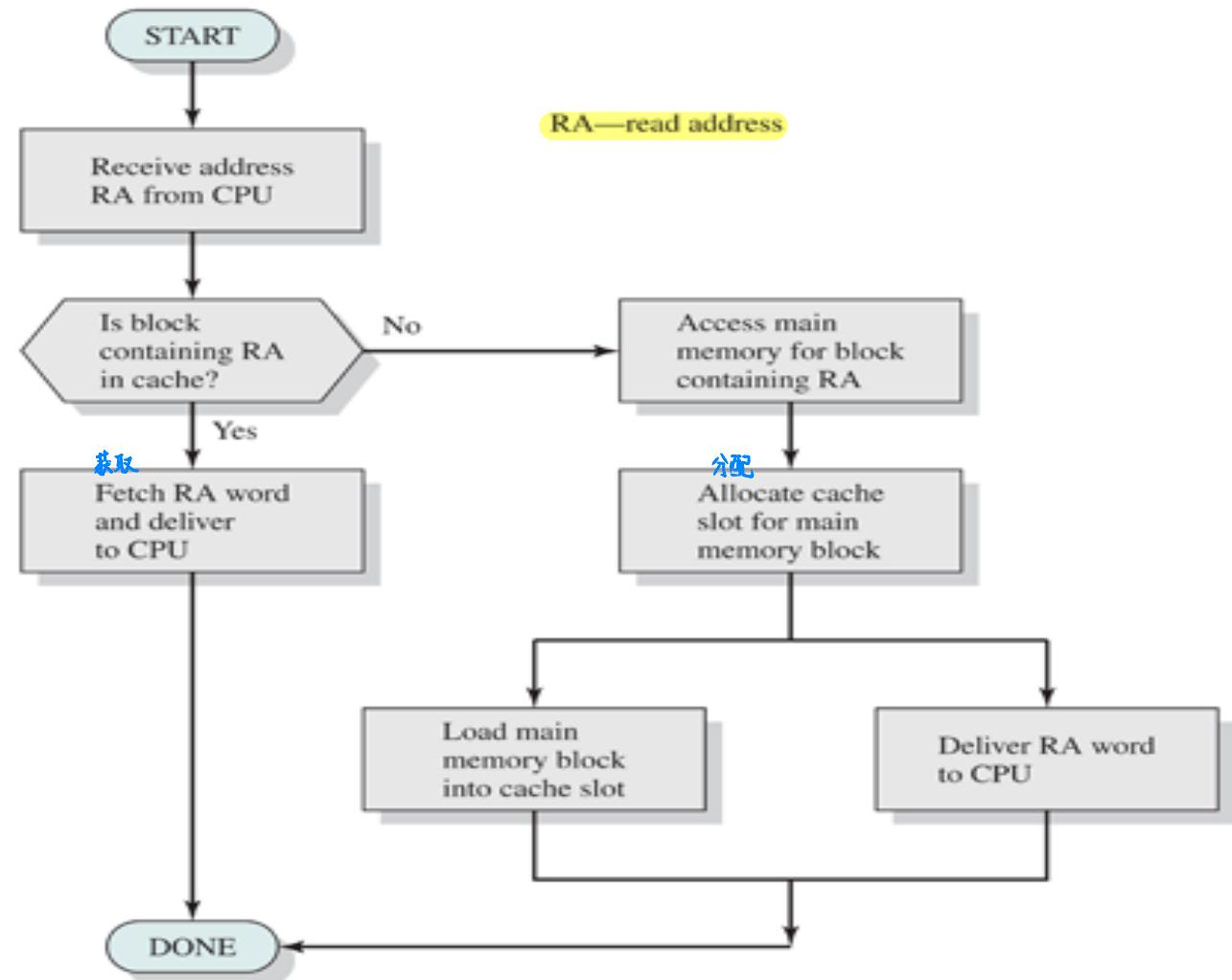
Cache Memory



(a) Single cache



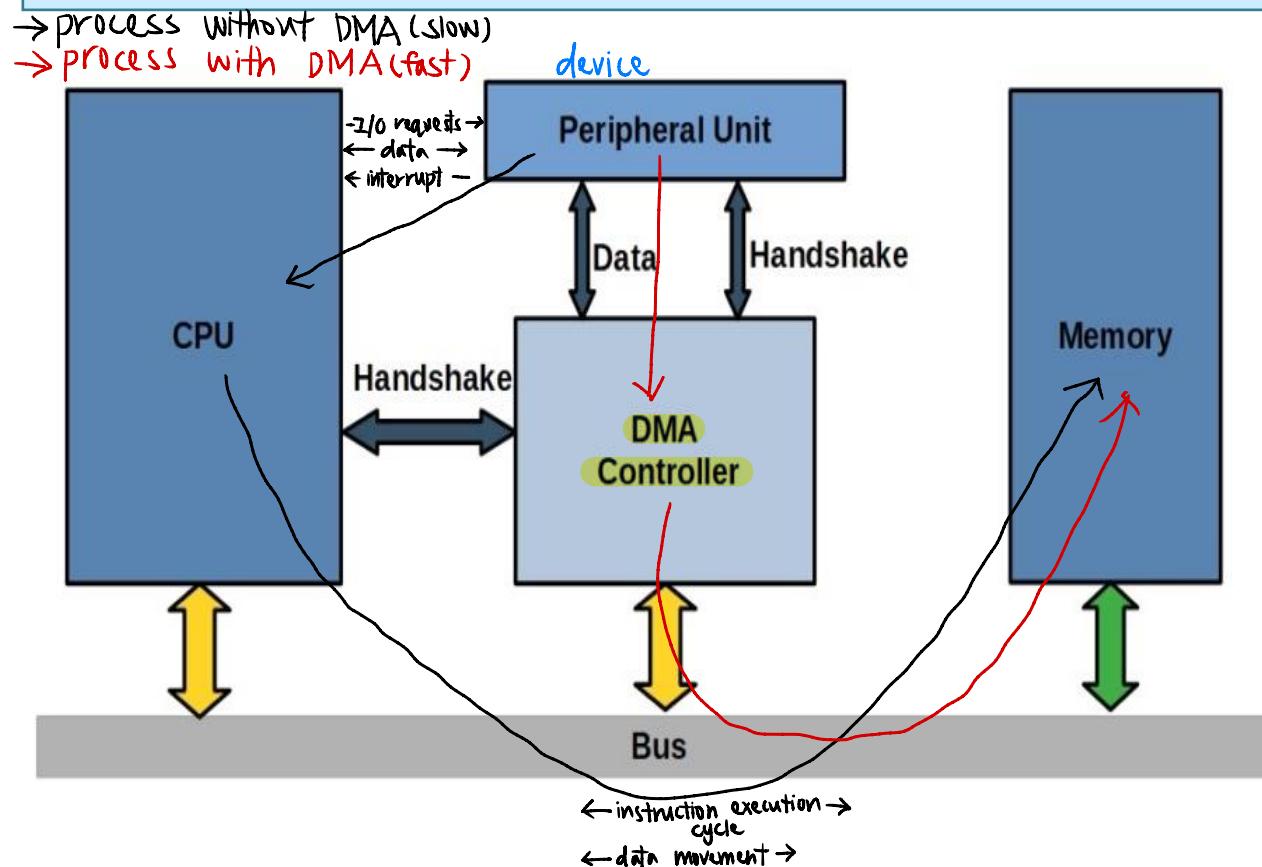
(b) Three-level cache organization





Direct Access Memory (DMA)

Process of transferring data from one memory location to another without the direct involvement of the processor (CPU)

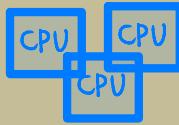


Benefits

- ❑ More efficient data movement in the **嵌入式** embedded system
- ❑ Improve the program's speed
- ❑ Reduce the power consumption **功耗**



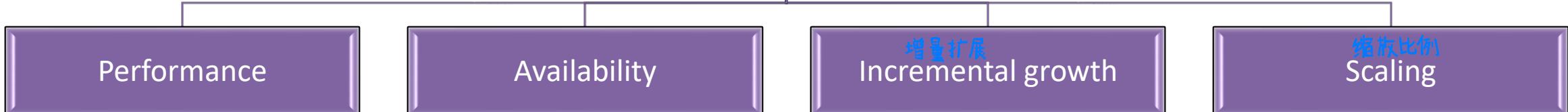
Multiprocessors



Characteristics

- Two or more similar processors
- Share the same main memory and I/O facilities
- Share access to I/O devices
- All processors can perform the same functions
- Controlled by an integrated operating system

Advantages





多核

芯片多处理器

Multicores (Chip Multiprocessor)

Characteristics

- Combine two or more processors on a single piece of silicones
- Each core consists of:
 - Registers
 - ALU
 - Pipeline hardware and control unit
 - L1 instruction and data
 - Caches (L2 & L3)

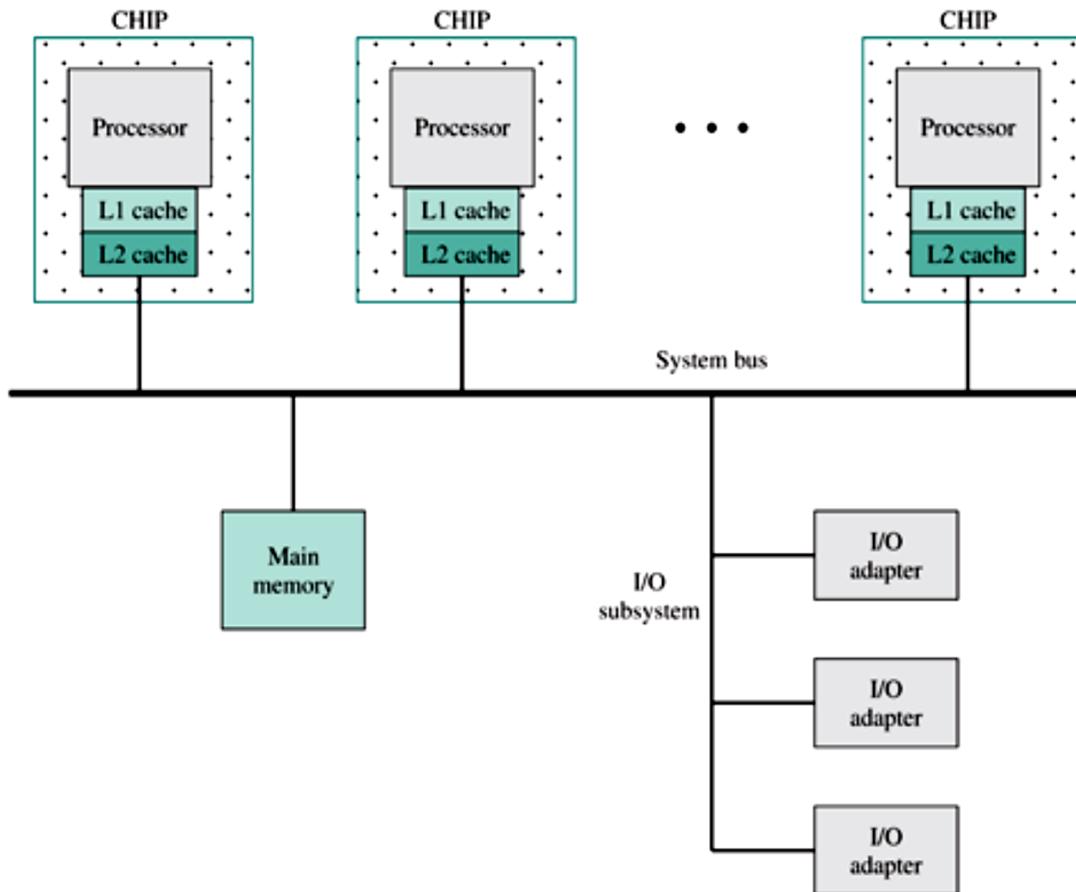
Advantages



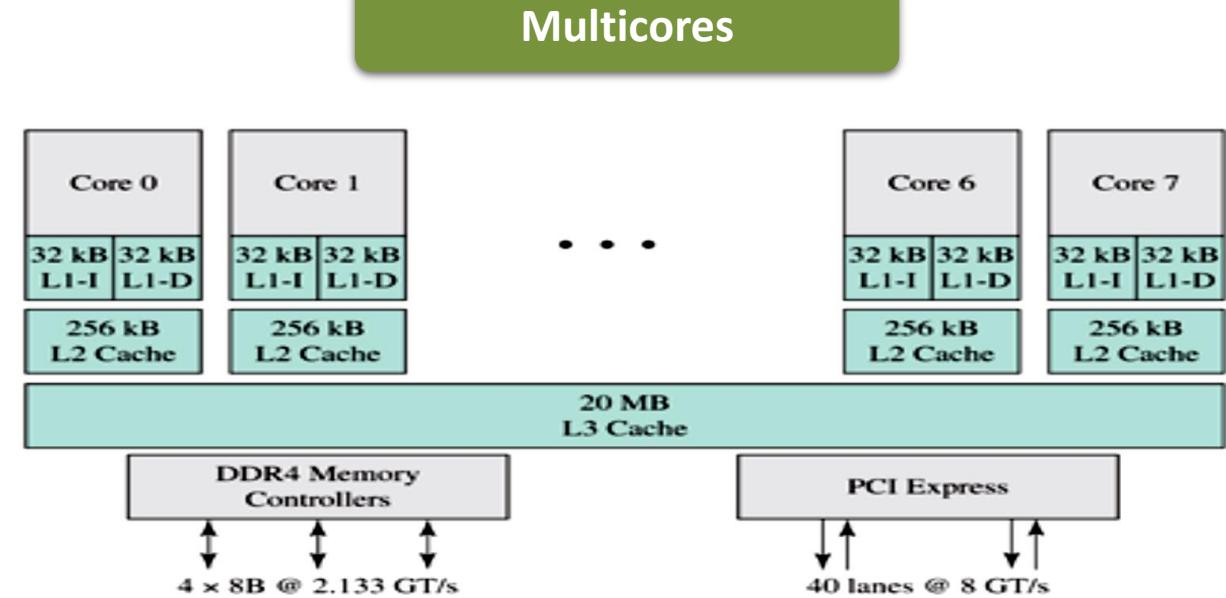


Multiprocessors vs Multicores

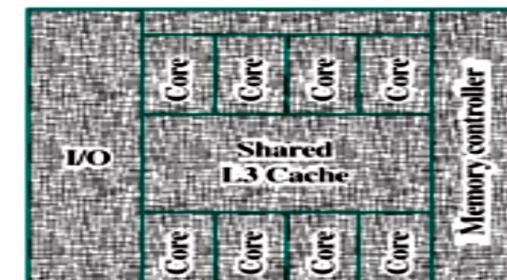
Multiprocessors



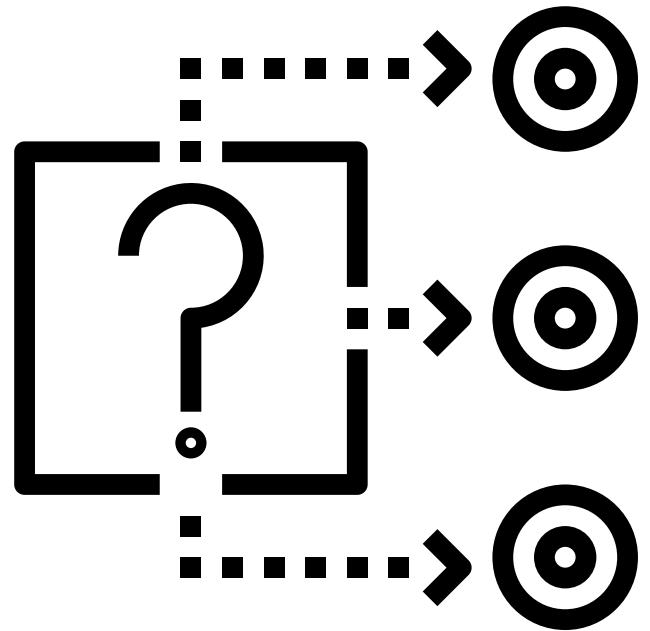
Multicores



(a) Block diagram



(b) Physical layout on chip



Operating Systems



Introduction

- Program that controls the execution of application programs
- Acts as an intermediary between the user and the computer and between the application programs and system hardware
- Interface between applications and the computer hardware

Objectives

- Convenience
- Efficiency
- Ability to evolve

Windows

Android

iOS

MacOS

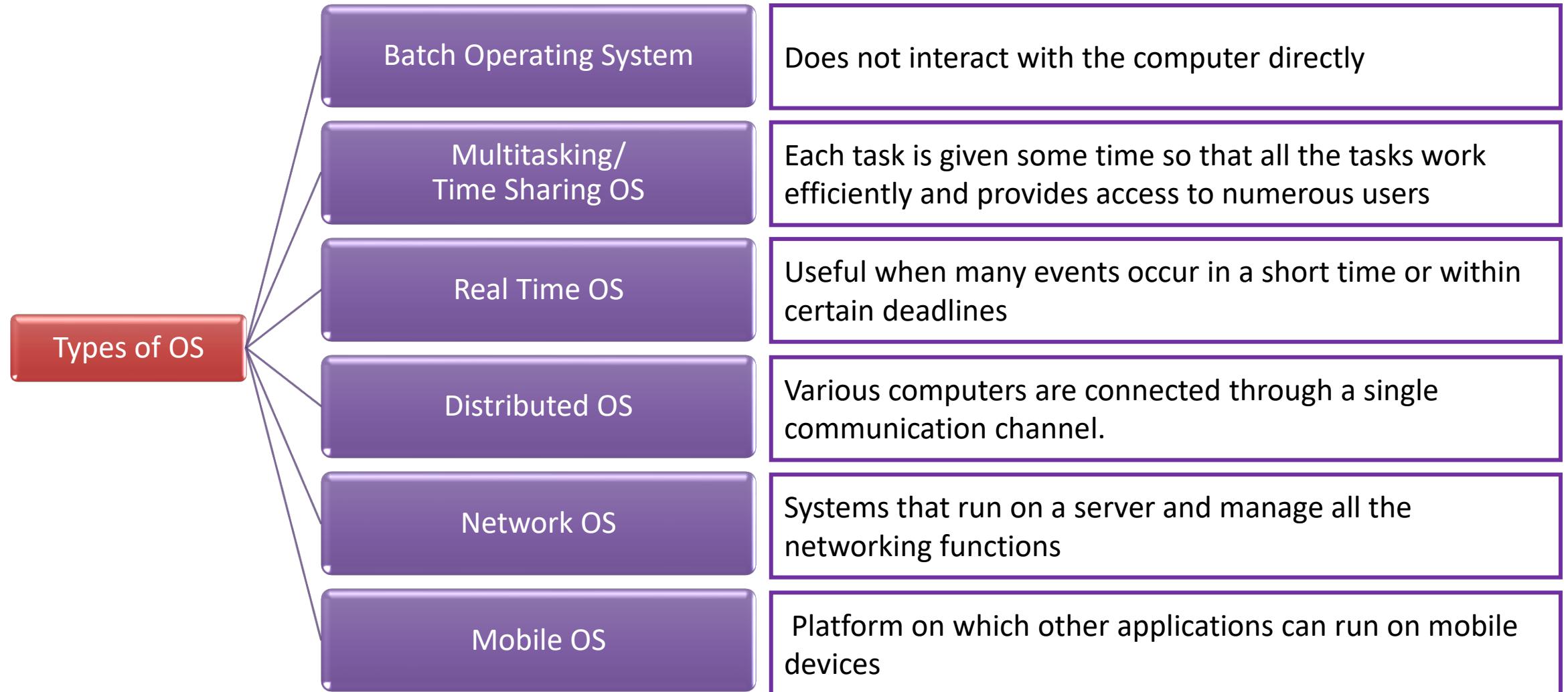
Linux

Chrome OS

Windows
Phone OS



Introduction





Introduction

Features of Operating System

Protected and supervisor mode

Allows disk access and file systems

Device drivers

Networking Security

Program Execution

Memory management
Virtual Memory
Multitasking

Handling I/O operations

Manipulation of the file system

Error Detection and handling

Resource allocation

Information and Resource Protection



Introduction



Advantages of OS

- Hide details of hardware by creating an abstraction
- Easy to use with a GUI
- Offers an environment for users to execute programs/applications.
- Convenient to use the computer system
- Provides the computer system resources with easy-to-use format

Disadvantages of OS

- May lose all the contents which have been stored in the system
- Quite expensive
- Never entirely secure as a threat can occur at any time





Functions of OS

Process management

Memory management

File management

Device Management

I/O System Management

Secondary-Storage Management

Security

Command interpretation

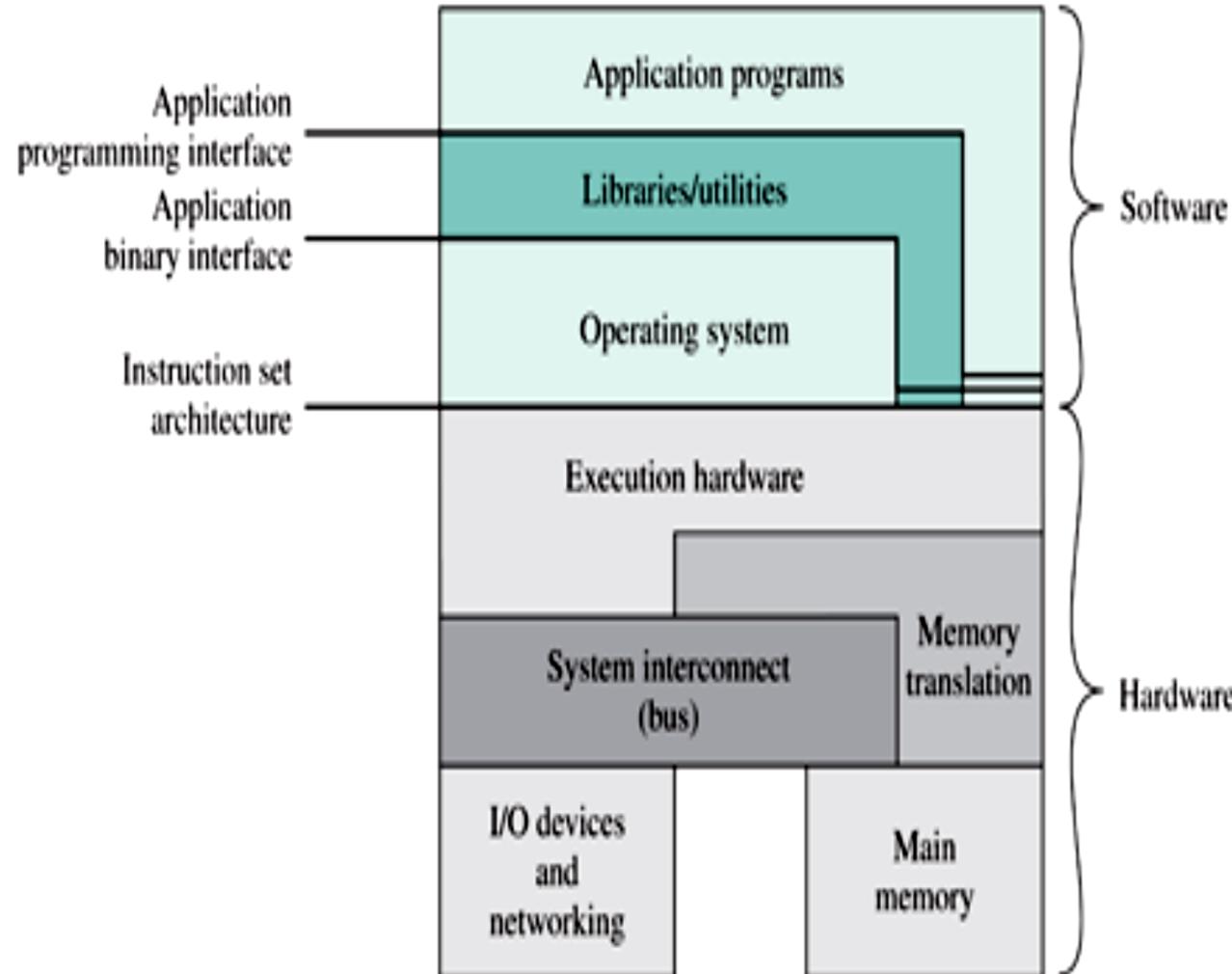
Networking

Job accounting

Communication management



User / Computer Interface 用户/电脑界面



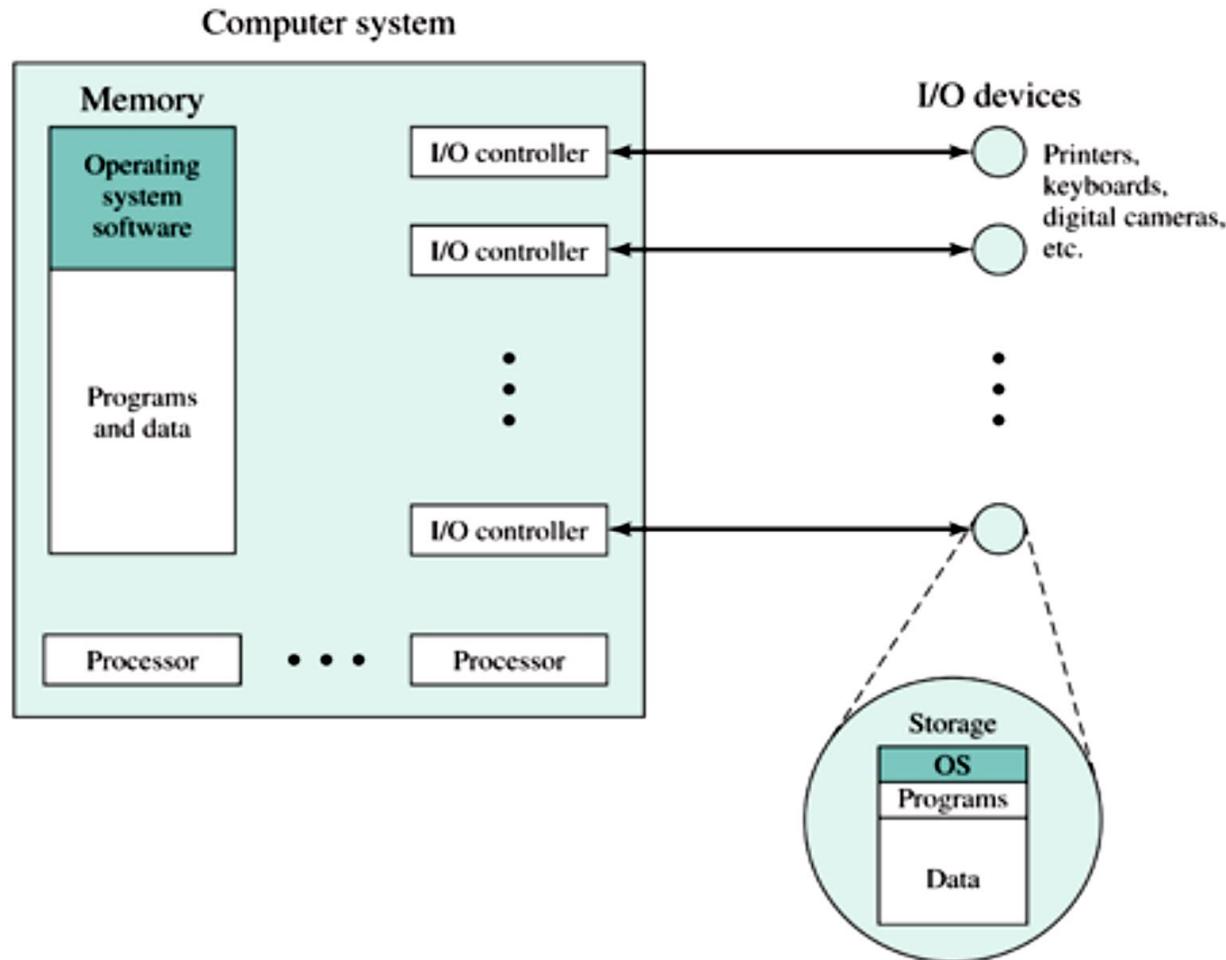
- ❑ Part of an OS that permits an operator to get the information.
- ❑ A user interface based on text displays the text as well as its commands which are typed over a command line with the help of a keyboard.

Services

- Program development
- Program execution
- Access to I/O devices
- Controlled access to files
- System access
- Error detection and response
- Accounting



Resource Manager



Heart of OS

Kernel / Nucleus

- To establish communication between user level application and hardware.
- To decide state of incoming processes.
- To control disk management.
- To control memory management.
- To control task management.



Resource Manager

Type of Kernels	Description	Advantage
Monolithic	<ul style="list-style-type: none">All operating system services operate in kernel spaceIt has dependencies between systems components	<ul style="list-style-type: none">Good performance
Micro	<ul style="list-style-type: none">It has virtual memory and thread scheduling.It is more stable with less services in kernel space.	<ul style="list-style-type: none">More stable
Hybrid	<ul style="list-style-type: none">Combination of both monolithic kernel and microkernel	<ul style="list-style-type: none">Speed and stability
Exo	<ul style="list-style-type: none">Follows end-to-end principle.It has fewest hardware abstractions as possible	<ul style="list-style-type: none">Fewer hardware abstraction
Nano	<ul style="list-style-type: none">Offers hardware abstraction but without system services	<ul style="list-style-type: none">Offers hardware abstractions without system services



Fault Tolerance

容错

Fault

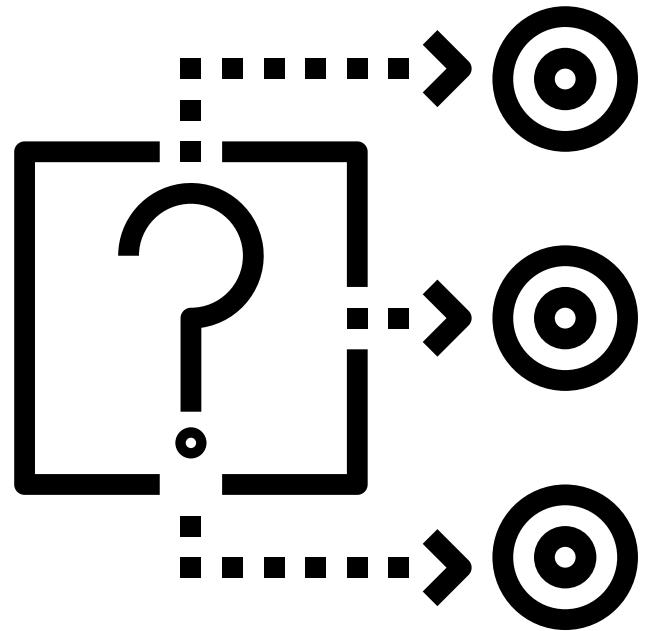
- An erroneous hardware or software state resulting from :
 - Component failure
 - Operator error
 - Physical interference from the environment
 - Design error
 - Program error
 - Data structure error

Fault Tolerance

Ability of a system or component to continue normal operation despite the presence of hardware or software faults

Fault

- Permanent
- Temporary
- Transient
- Intermittent



Process Description and Control



Process

- 执行**
- ❑ The execution of a program that performs the actions specified in that program.
 - ❑ OS helps to create, schedule, and terminates the processes.
 - ❑ Child process
 - A process created by the main process

Architecture

Stack	Stores temporary data
Heap	Allocates memory
Data	Contains the variable
Text	Includes the current activity

Stack

Heap

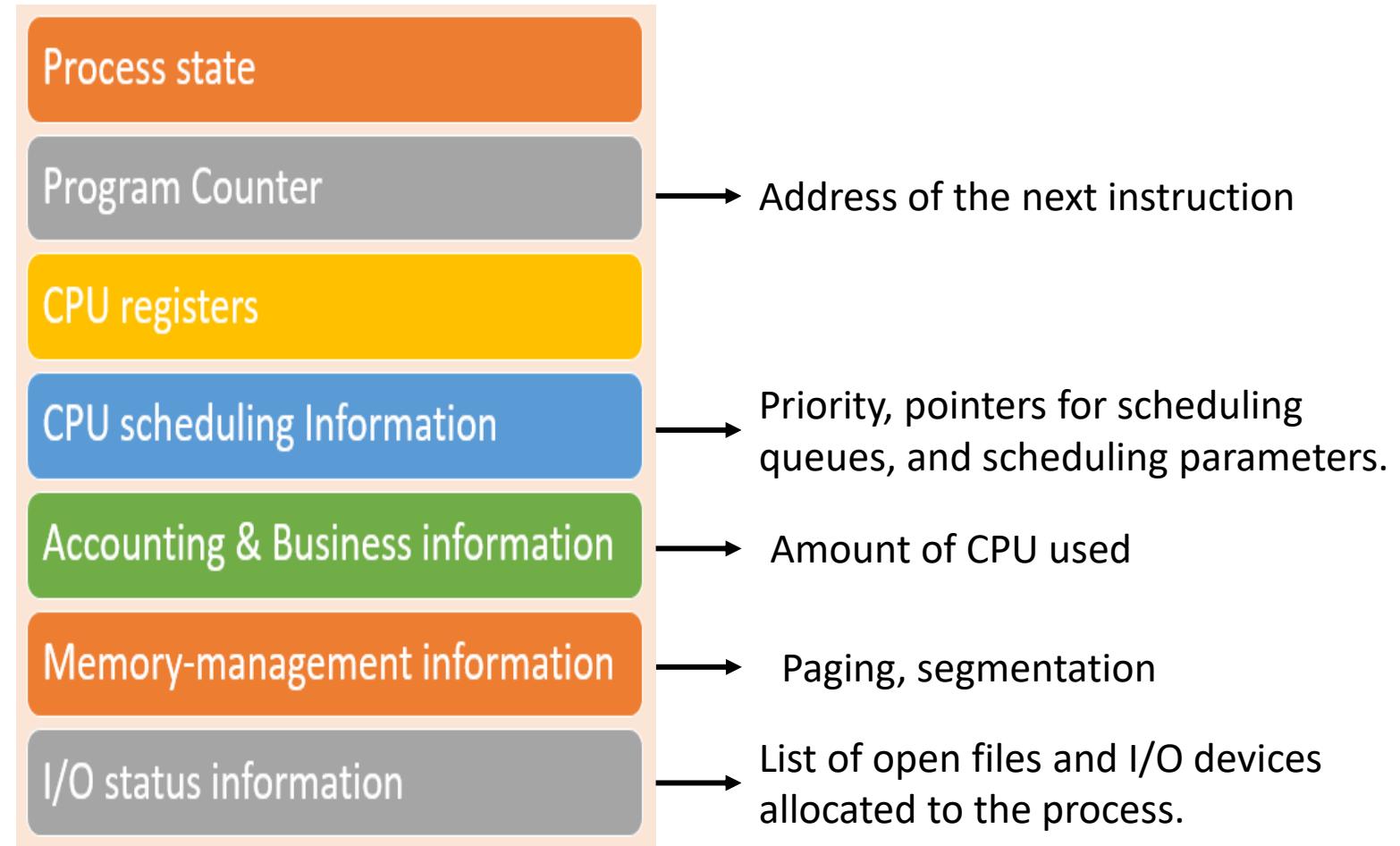
Data

Text



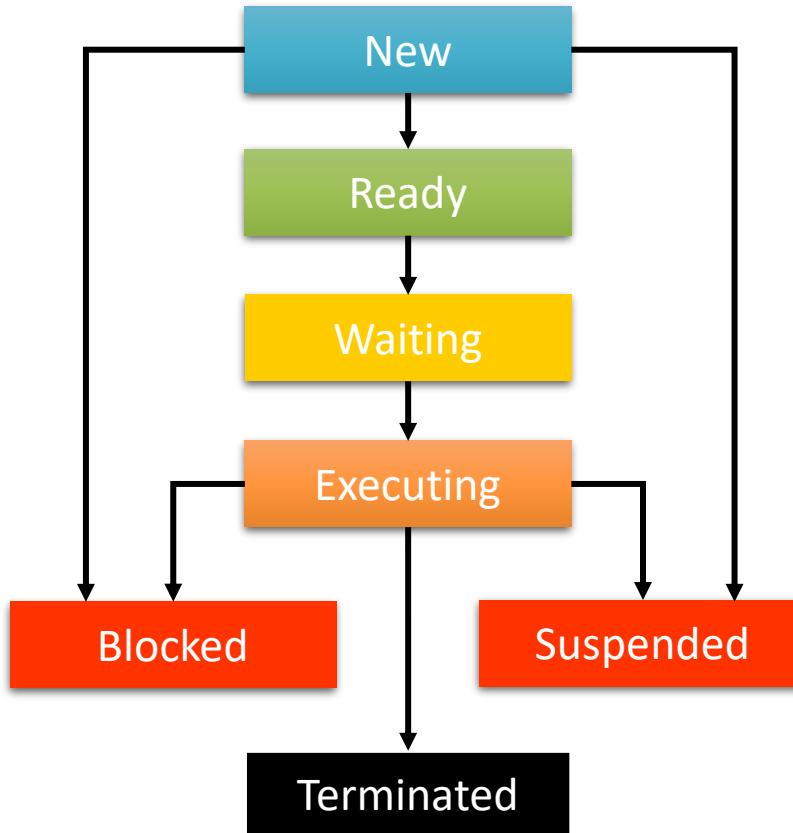
Process Control Block (PCB)

- ❑ Data structure maintained by the Operating System for every process
- ❑ Keeps all the information needed to keep track of a process
- ❑ Control the process management
- ❑ Identified by an integer process ID (PID)





Process States

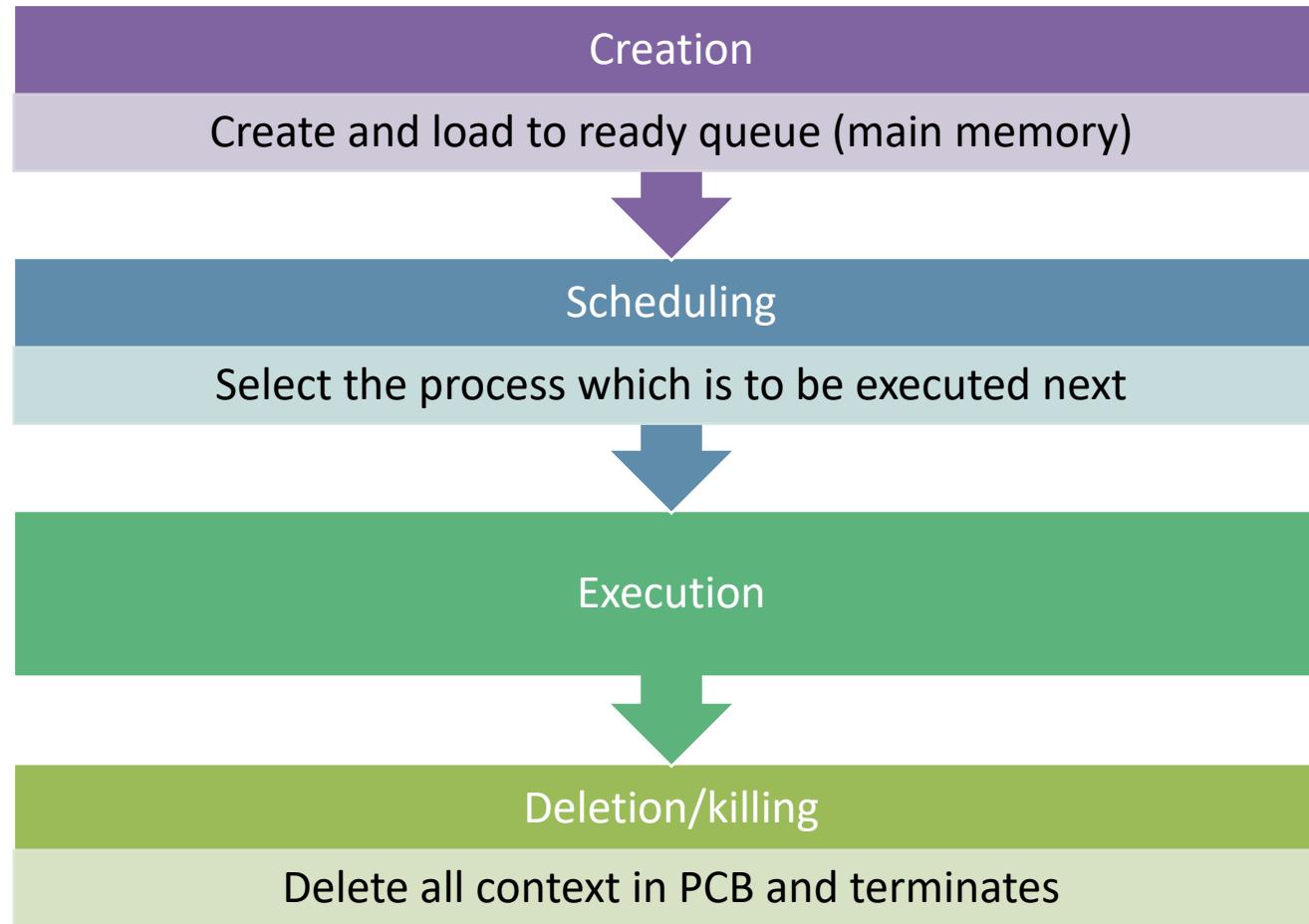


After completing every step, all the resources are used by a process, and memory becomes free.

State	Description
New	The new process is created when a specific program calls from secondary memory/ hard disk to primary memory/ RAM
Ready	The process should be loaded into the primary memory, which is ready for execution.
Waiting	The process is waiting for the allocation of CPU time and other resources for execution.
Executing	The process is an execution state.
Blocked	Time interval when a process is waiting for an event like I/O operations to complete.
Suspended	The time when a process is ready for execution but has not been placed in the ready queue by OS.
Terminated	The time when a process is terminated (finish execution)



Process Operations / Management

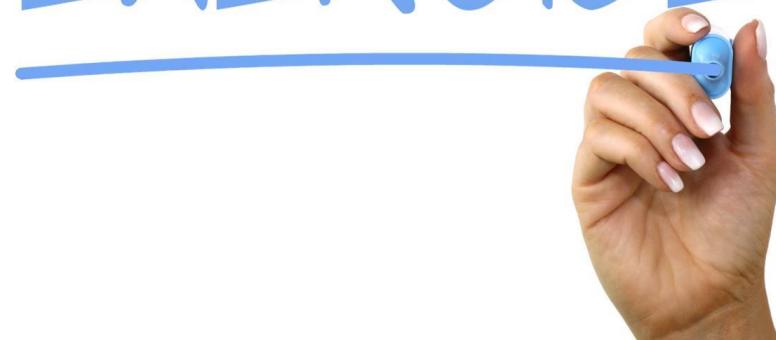




Revision Questions

1. Differentiate between a multiprocessor and a multicore system
2. State the examples of operating system
3. Illustrate and explain the process states.

EXERCISE



MULTICORE

VERSUS

MULTIPROCESSOR

MULTICORE

A single CPU or processor with two or more independent processing units called cores that are capable of reading and executing program instructions

Executes a single program faster

Not as reliable as a multiprocessor

Have less traffic

MULTIPROCESSOR

A system with two or more CPUs that allows simultaneous processing of programs

Executes multiple programs faster

More reliable since failure in one CPU will not affect the other

Have more traffic

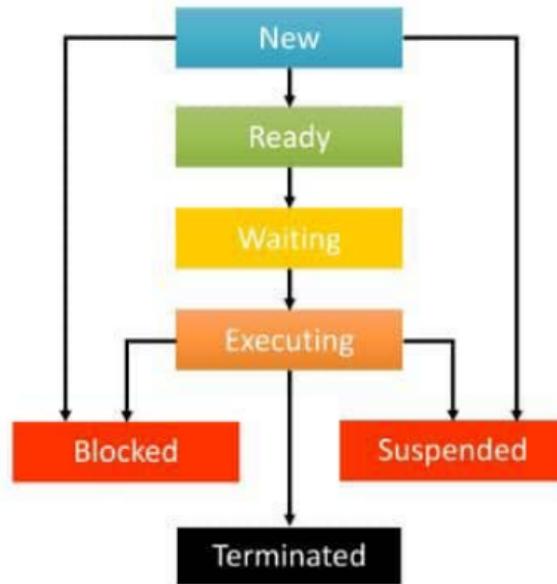


Introduction

Types of OS	Batch Operating System	Does not interact with the computer directly
	Multitasking/ Time Sharing OS	Each task is given some time so that all the tasks work efficiently and provides access to numerous users
	Real Time OS	Useful when many events occur in a short time or within certain deadlines
	Distributed OS	Various computers are connected through a single communication channel.
	Network OS	Systems that run on a server and manage all the networking functions
	Mobile OS	Platform on which other applications can run on mobile devices



Process States



After completing every step, all the resources are used by a process, and memory becomes free.

State	Description
New	The new process is created when a specific program calls from secondary memory/ hard disk to primary memory/ RAM
Ready	The process should be loaded into the primary memory, which is ready for execution.
Waiting	The process is waiting for the allocation of CPU time and other resources for execution.
Executing	The process is an execution state.
Blocked	Time interval when a process is waiting for an event like I/O operations to complete.
Suspended	The time when a process is ready for execution but has not been placed in the ready queue by OS.
Terminated	The time when a process is terminated

Chapter 1: Computer System

a) Basic Elements

→ processor 处理器

- controls the operation 操作
- data processing

→ main/real/primary memory

- stores data and programs

→ I/O modules

- move data between the computer and its external environment

→ system bus

- communication among processors, main memory and I/O modules

b) Registers 暫存器

→ memory address register (MAR)

- specifies 指定 the address in memory for the next read or write

→ memory buffer 缓冲 register (MBR)

- contains the data to be written into memory

→ I/O address register (I/OAR)

- specifies a particular I/O device

→ I/O buffer register (I/OBR)

- exchange of data between an I/O module and the processor

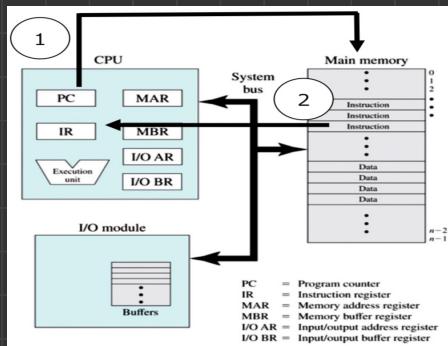
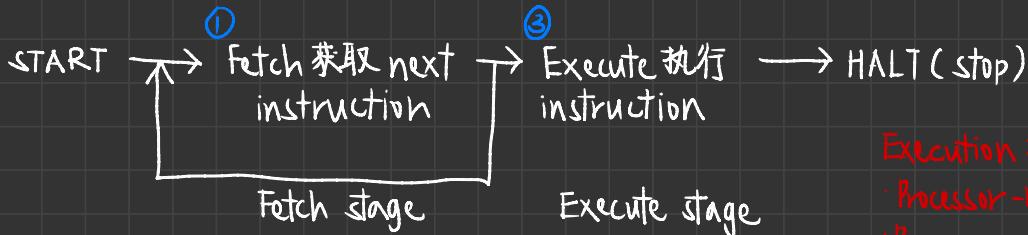
→ program counter

- hold the address of the next instruction to be executed 执行

→ instruction register

- hold the instruction that is currently being executed

c) Instruction Cycles

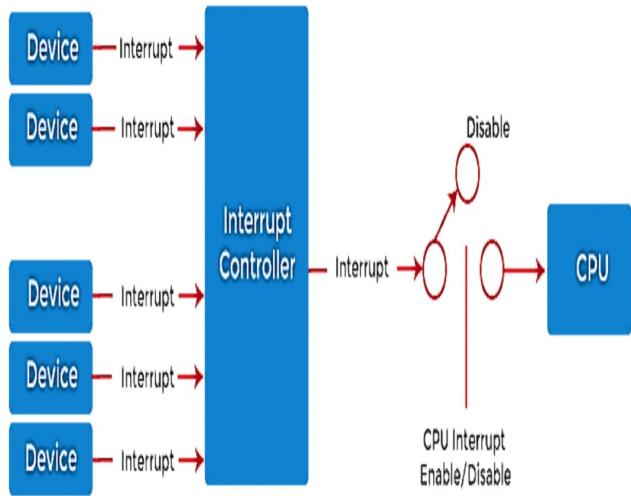


Execution:

- Processor-memory
- Processor-I/O
- Data processing
- Control

d) Interrupts

- 概念**
- A signal emitted by hardware or software when a process or an event needs immediate attention
 - 提醒** Alerts the processor to a **high-priority process** requiring interruption of the current working process
 - To improve processor utilization
 - In the operating system, **interrupts** are **essential**
 - It give a reliable technique for the **OS** to communicate and react to their surroundings.



Classes of Interrupts	Description
Program	<p>Result of an instruction execution • Arithmetic overflow <small>执行 溢出</small> • Division by zero <small>非法</small> • Attempt to execute an illegal machine instruction • Reference outside a user's allowed memory space.</p>
Timer	<p>Timer within the processor • Perform certain functions on a regular basis <small>执行 定期</small></p>
I/O	<ul style="list-style-type: none"> To signal normal completion of an operation To signal a variety of error conditions <small>各种 错误</small>
Hardware failure	<ul style="list-style-type: none"> Power failure <small>奇偶</small> Memory parity error

e) Interrupts and Instruction Cycle