

Chapter 1 Introduction

1.1 What is an operating system

Operating system (Kernel)

- 作業系統為電腦的核心，它是由驅動電腦硬體工作之軟體及韌體程式所組成，它是使用者與電腦硬體的操作介面，並管理電腦系統的資源。
- We view an operating system as the **programs** implemented in either **software or firmware**, that acts as an intermediary between user and hardware to make hardware usable, and **manage the system's resources**.
- 燒錄在硬體（例如ROM、EPROM、...）之軟體為韌體。

Operating system (Kernel)

- 作業系統的主要目的是使得電腦能夠被方便使用，次要目的是讓電腦硬體發揮在最有效率的狀態。
- The primary goal is thus to make the computer system **convenient** to use.
- A secondary goal is to use the computer hardware in an **efficient** manner.

作業程式所包含的範圍

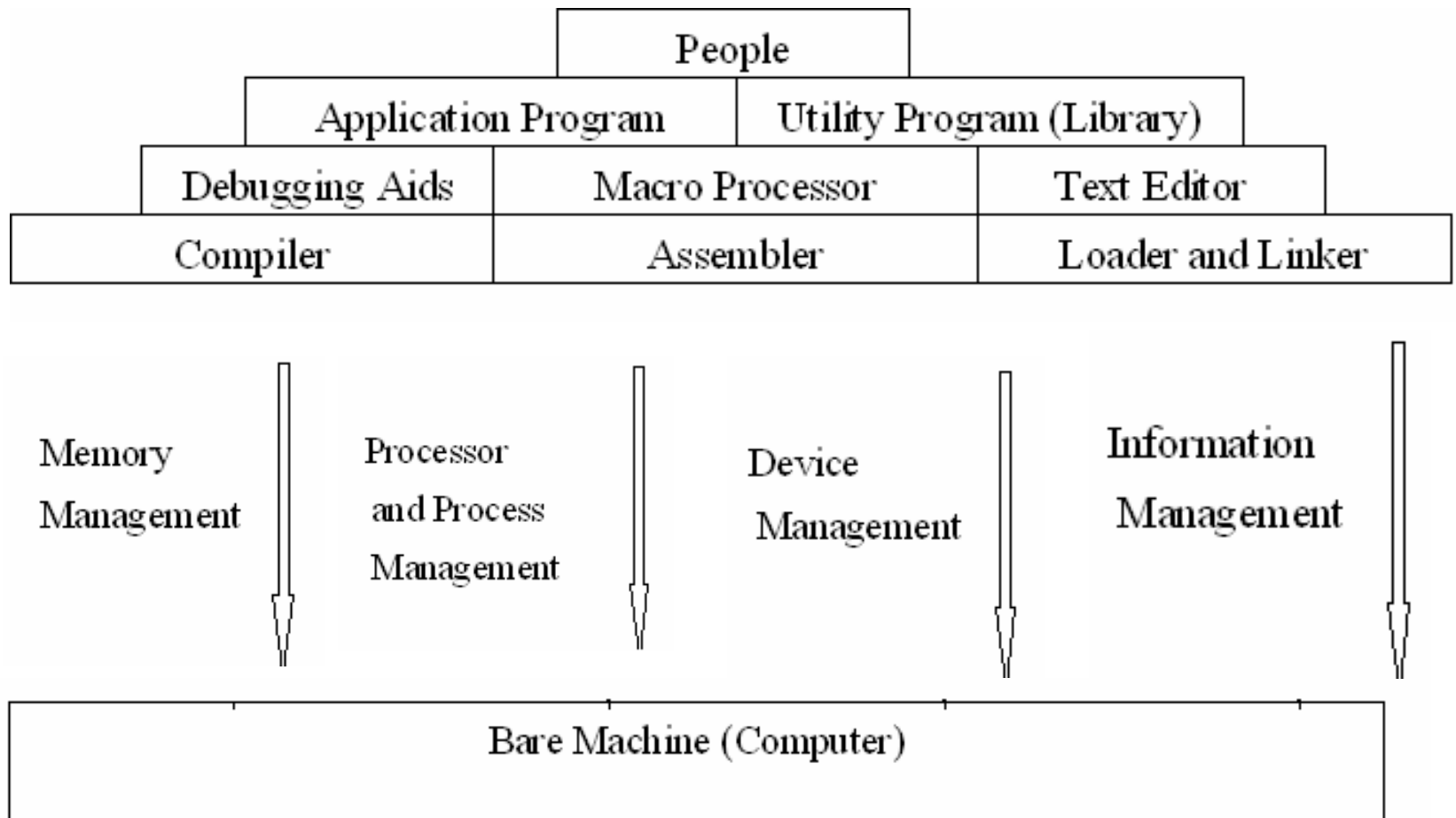


圖 1.1 系統程式與作業系統所包含的範圍

裸機(Bare Machine)

- 只有硬體而沒有加上軟體及韌體的電腦為裸機(Bare Machine)，裸機是不可能工作的，裸機必須由作業系統來管理、執行、操作、監督...。
- A computer without its software clothing.

作業系統的四大管理(1)

Resources Management

- 記憶體管理 (Memory Management)
 - 管理主記憶體，並決定那一支程式可以佔有主記憶體，那一支程式不能使用主記憶體，...。
- 處理器 / 處理元管理 (Processor / Process Management)
 - 因為大部份電腦之處理器僅有一個，而有許多處理元要執行，因此作業系統內必須管理那個處理元可以被處理器執行，那些處理元必須排隊等待，...等。

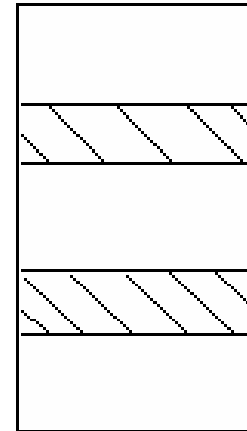
作業系統的四大管理(2)

- 設備管理 (Device Management)
 - 管理輸出輸入週邊設備之運作，尤其必須有設備驅動程式(Device Driver)來驅動設備工作。
 - Device driver: A special subroutine was written for each I/O device to perform I/O, it knows how the buffers, flags, registers, control bits, and status bits should be used for a particular device.
- 資訊管理 (Information Management)
 - 管理磁碟、磁片、光碟內之檔案結構及其內容。

Resource Manager (Operating System)

- Keep track of the resource.
- Enforce policy that determines **who gets what, when, how much.**
- Allocate resource
- Reclaim resource

- 記錄各個資源的狀態。
- 提供有效的政策（**Policy**）來決定誰可以得到資源？在什麼時候得到？得到多少？
- 提供資源配置軟體（**Resource Allocator**）來進行資源配置。
- 使用者完成工作後，進行資源回收工作。



作業系統服務的範圍

- Resource management
- Program execution
- Command interpreter
- I/O operations
- Error detection
- Accounting system
- System protection
- System call (BISO call, Dos call,...)
- :
- :

啓動載入器（Bootstrap Loader）（1）

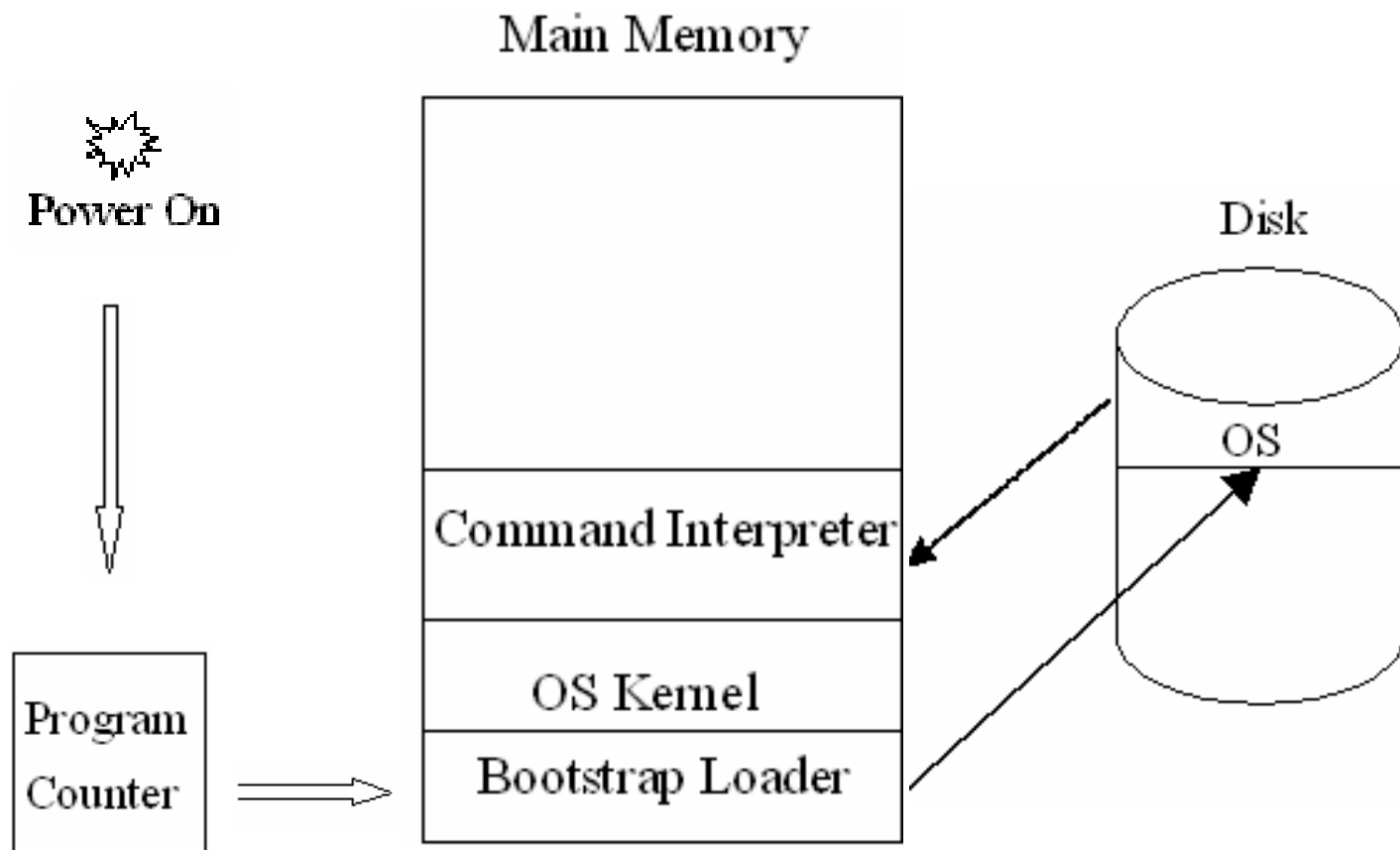


圖 1.11 啓動載入器

啓動載入器（Bootstrap Loader）（2）

- 用來將作業系統載入主記憶體的二體。
- 當我們打開電腦電源後，硬體在設計上便使得程式計數器（Program Counter）指向主記憶體存放啓動載入器的地方，這意味著一開機便執行啓動載入器。
- 啓動載入器會至硬碟內將作業系統的核心（Kernel）載入主記憶體內，然後將程式計數器設定指向作業系統核心，亦即將控制權交給作業系統核心，接著作業系統核心便隨時接受需要作業系統服務的工作。

啓動載入器（Bootstrap Loader）（3）

- 命令直譯器（Command Interpreter）便是作業系統核心內的一員，它扮演接受使用者命令的角色。
- 啓動載入器必須事先燒錄在主記憶體內，因此它是屬於韌體。
- 啓動載入器有時又稱爲初始程式載入器（Initial Program Loader ），簡稱IPL。

命令直譯器 (Command Interpreter) (1)

- 當你使用MS / DOS或UNIX作業系統時，你必須透過鍵盤輸入命令 (Command)，以命令作業系統來協助工作，而作業系統中用來解析命令的軟體，稱為命令直譯器。
- 當命令直譯器收到命令之後，會先進行解譯，以確定命令是否無誤，若沒有錯誤便呼叫對應的程式執行命令。
- 命令直譯器是使用者與作業系統最前端的使用者介面。
- 命令直譯器有時又被稱為控制卡直譯器 (Control Card Interpreter) 或命令列直譯器 (Command Line¹² Interpreter)。

命令直譯器（Command Interpreter）（2）

- 命令直譯器除了解譯命令外，還包含執行這個命令的軟體，因此有人又將它稱為外殼（Shell）。
- 內部命令（Internal Command）是隨著外殼一起在開機時便載入主記憶體之作業系統區域。
- 外部命令（External Command）則放在磁片、磁碟等輔助記憶體（Secondary Memory）內，當被使用到時，才載入主記憶體。
- 我們可以事先將一些命令整合在檔案內，等到要用時，再叫出這個檔案執行這些命令，稱為批次處理（Batch Processing）。

帳號管理 (Accounting)

- 爲保護電腦系統不被非法使用者侵入，故系統均要求使用者進行登錄之動作，以確保合法使用者之權益，這便是帳號管理。
- 一部電腦可同時供多人使用，爲確保每個人的資源不被他人破壞，並限定使用者使用資源的權限，以及資源分享的管理...等，這些也都是帳號管理的權責。

程式執行 (Program Execution)

- 由使用者下命令經由外殼執行程式。
- 事先在電腦內部設定某個時間、某個事件 (Event) 發生時執行程式。
- 執行程式的過程若有錯誤，例如：計算溢位 (Overflow)、硬體損壞、通訊錯誤...等；或非經授權的動作，例如：使用處理器太久、不合法存取記憶體...等，均將由作業系統來反應並通知使用者。

系統呼叫（System Call）（1）

- 系統呼叫是程式設計者所撰寫之程式與作業系統的介面。
- 作業系統必須對系統資源作某種程度的保護，**任何使用系統資源之程式必須透過系統呼叫**，由作業系統決定並安排使用資源。
- 使用者經由外殼命令或應用程式呼叫系統程式庫（System Library），再由系統呼叫要求作業系統核心執行，當工作是牽涉到週邊設備時，作業系統核心再行呼叫週邊設備驅動程式（Device Driver），並由驅動程式控制週邊設備進行輸出 / 輸入之工作。

系統呼叫 (System Call) (2)

Users		
Shell and System Library		
System Call Interface to the Kernel		
I/O	Device Driver	File System CPU Scheduling Virtual Memory Demand Paging (Operating System Kernel)
Kernel Interface to the Kernel		
Terminal Controller Terminals	Device Controllers Disks and Tapes	Memory Controllers Physical Memory

圖 1.12 UNIX 作業系統之分層結構

系統呼叫（System Call）（3）

- 輸出敘述被編譯成呼叫系統函數，而系統函數本身又被轉譯成系統呼叫。
- 系統呼叫與MS / DOS作業系統之DOS呼叫（DOS Call）及基本輸出 / 輸入呼叫（BIOS Call）是不同的。其使用方法大致上是相同的。

系統呼叫（System Call）（4）

MS / DOS之分層結構，應用程式可以直接呼叫基本輸出 / 輸入函數，而沒有透過作業系統之系統呼叫後，再經由作業系統核心出面安排處理工作；也因為MS / DOS作業系統的不明顯分層結構，而可以由應用程式直接控制系統資源，以致造成容易當機的問題發生。

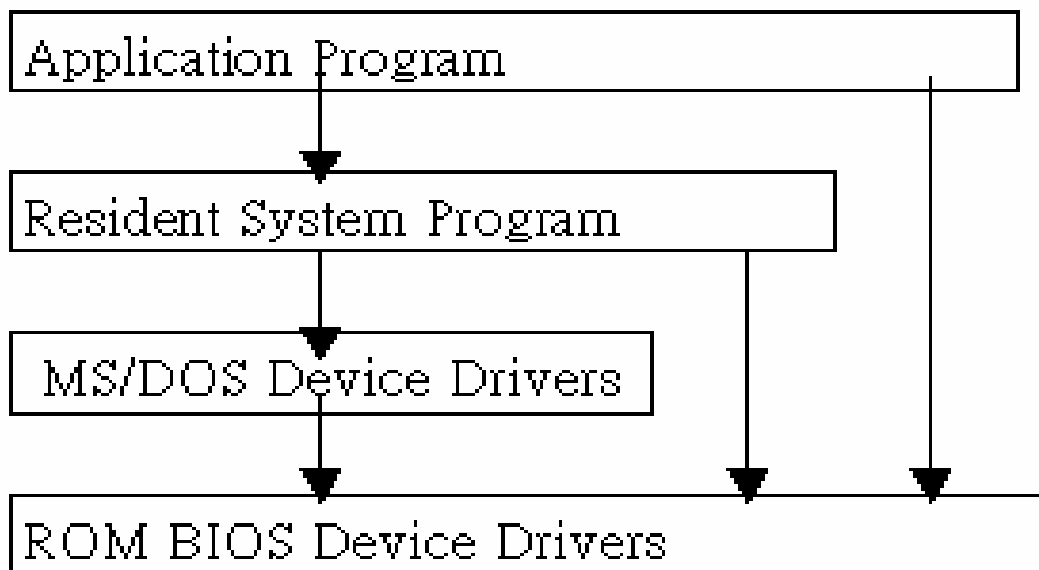


圖 1.13 MS / DOS 之分層結構

系統保護（System Protection）（1）

- 防止非經授權者使用系統資源，同時確保不正確的程式不會因此造成其他程式執行不正確。
- 雙模保護（Dual Mode Protection）
 - 使用者模式（User Mode）及監督者模式（Monitor Mode 或 Supervisor Mode）。
 - 一般使用者登錄後是在使用者模式進行作業，使用者模式下不能直接作影響系統核心的事。
 - 超級使用者（Super User）可以登錄或轉換為監督者模式，以進行系統設定、維護，轉換...等工作。

系統保護（System Protection）（2）

- 監督者模式有時又稱為系統模式（System Mode）、特權模式（Privileged Mode）或保護模式（Protect Mode）。
- 作業系統通常在監督者模式下執行，而在監督者模式內所使用之命令為特權指令（Privileged Instruction），例如：暫停系統（Halt System）、打開 / 關閉中斷（Turn Interrupt On / Off）、轉換模式（Change Mode）、以及所有的輸出/輸入指令...等均為特權指令。
- 程式在使用者模式下，進行任何動用系統資源之動作，均以系統呼叫藉由監督者模式代為處理。

我的故事(1)

1.2 Early systems

- HP2100 16K, 5M

- 畫卡(打卡)

Job Control Language (JCL)

- \$JOB
- \$START
- \$FTN
- PROGRAM
- \$LOAD
- \$RUN
- DATA
- \$END

- Batch

我的故事(2)

1.2 Early systems

- Real Time/Time Sharing
- Spooling

作業系統的種類

1.3 Terminology

- 批次處理(Batch Processing)
- 多重程式(Multiprogramming)
- 分時系統(Time Sharing)
- 週邊設備線上同時工作(Spooling)
- 即時系統(Real Time)
- 多處理器系統(Multiprocessing)
- 分散式系統(Distributed System)

批次系統（Batch Systems）（1）

- 其作業模式是批次處理（Batch Processing），使用JCL。
- 將要執行的工作先經過一段時間收集之後，再一起透過電腦去執行。
- Job were gathered in groups or batch until one period than run these jobs.
- 最大缺點便是缺乏使用者與程式之間的交談（Interaction）溝通，而且在時效上也較不經濟。
- 當某程式在進行輸出時，中央處理器卻被閒置（Idle），而無法發揮中央處理器的效能，往返時間（Turnaround Time）通常比較長。

批次系統（Batch Systems）（2）

- 電腦開機時，往往利用批次處理執行一些初始設定的工作。
- 批次處理可以利用寫程式來進行，例如計算全班學生的總成績，薪資的計算、財政部計算所得稅...等。
- 利用外殼命令事先撰寫在檔案內執行，例如MS/DOS內的 *.bat便是一種執行批次處理的批次檔案。
- 用來撰寫批次檔案內容的語言，我們稱為工作控制語言（Job Control Language），簡稱為JCL；或直接稱為命令語言（Command Language）。

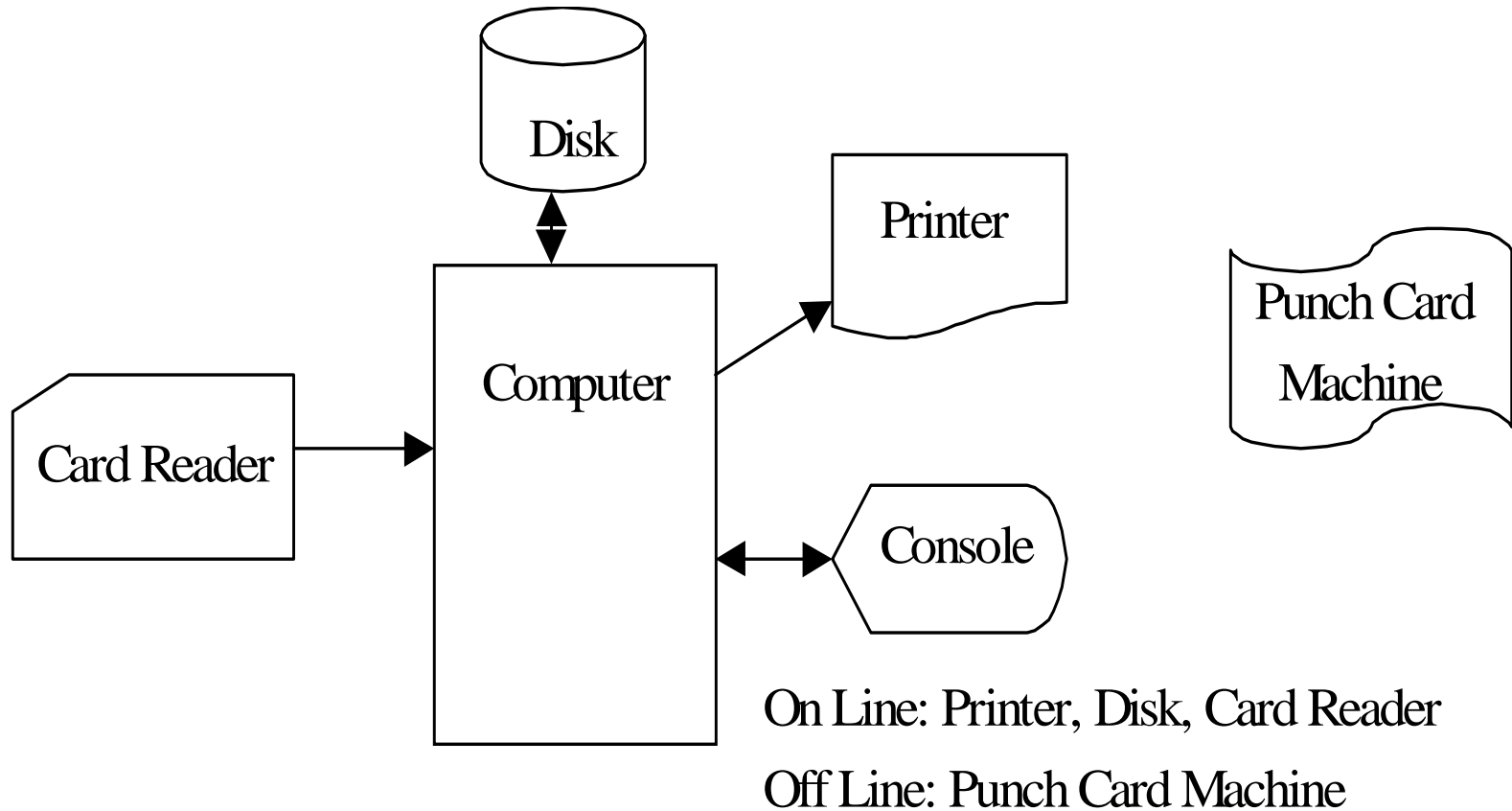
往返時間 (Turnaround Time)

- 就是指我們從開始委託 (Submit) 電腦工作到整個工作完成所需的時間，它包含中央處理器處理時間 (CPU Time)、輸入 / 輸出的時間、以及等待作業系統的時間。
- The **interval** between job submission and job completion.

離線（Off Line）/連線（On Line）（1）

- 設備離線（Off Line）
 - 設備不與電腦連接，不受中央處理器控制。
- 設備連線（On Line）
 - 設備與電腦連接，並直接受中央處理器控制。
- 列印時，你必須確認印表機的連線燈號是否有亮。

離線 (Off Line) / 連線 (On Line) (2)



多重程式 (Multiprogramming) (1)

- 同時有許多程式在主記憶體中，輪流快速的由中央處理器來執行，且當正在執行的程式等待資源時，中央處理器不需等待，它可以至主記憶體取另一個程式來執行。
- Several user programs are in main memory **at once (simultaneously)** and the processor is **switched rapidly between** the jobs.

多重程式 (Multiprogramming) (2)

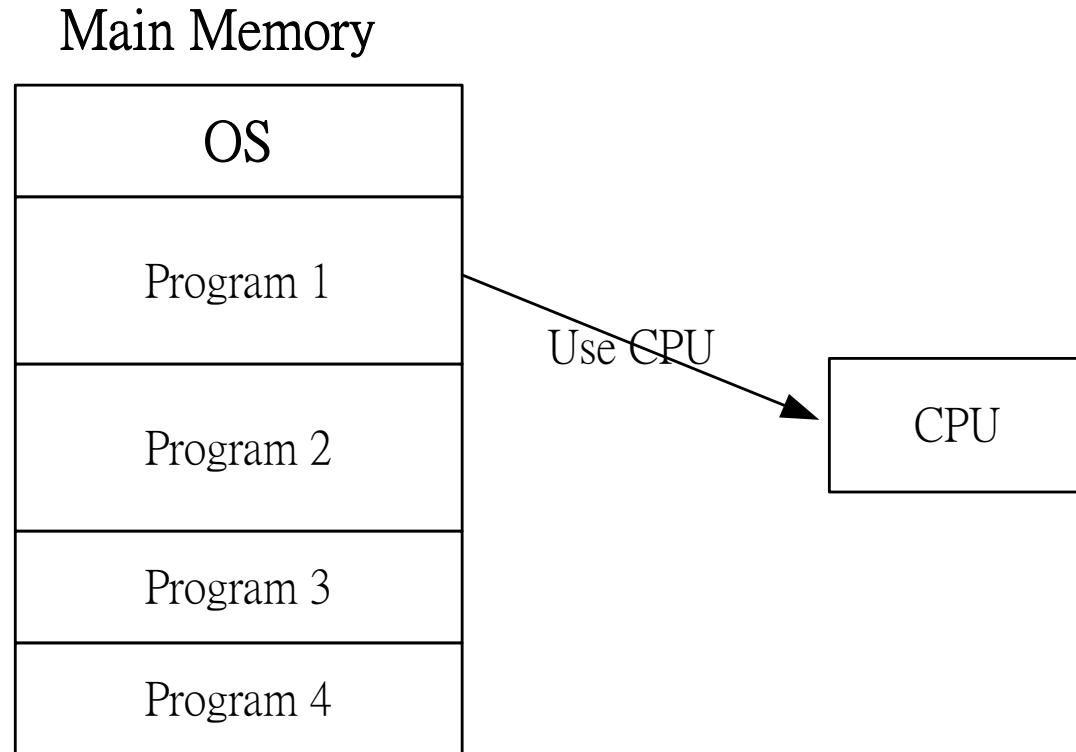


圖2.2 多重程式系統

多重程式 (Multiprogramming) (3)

- 當某一個程式在進行輸入 / 輸出而不用中央處理器時，就把中央處理器讓下一個程式使用。
- 多重程式使得中央處理器儘可能的忙碌，以發揮它的效能，同時也讓週邊設備充分的忙碌，它使得電腦的產量 (Throughput) 增加，也使得中央處理器使用率 (CPU Utilization) 增加。

產量 (Throughput)

- 單位時間的工作量，例如每單位時間完成多少個程式。
- Work processed per unit time.
- The number of processes that are completed per time unit.

分時系統（Time Sharing Systems）(1)

- 作業系統依中央處理器排程（CPU Scheduling），將中央處理器的時間切割為極小的時間片段（Time Slice），例如50ms。
- 任何程式佔有中央處理器時，若用罄一個時間片段，則輪由下一個程式佔有中央處理器，同樣的，若下一個程式用罄時間片段，再由另一個程式執行...
- Uses CPU scheduling and multiprogramming to provide each user with a small portion of CPU time.
- User may interact with each program while it is running.

分時系統 (Time Sharing Systems) (2)

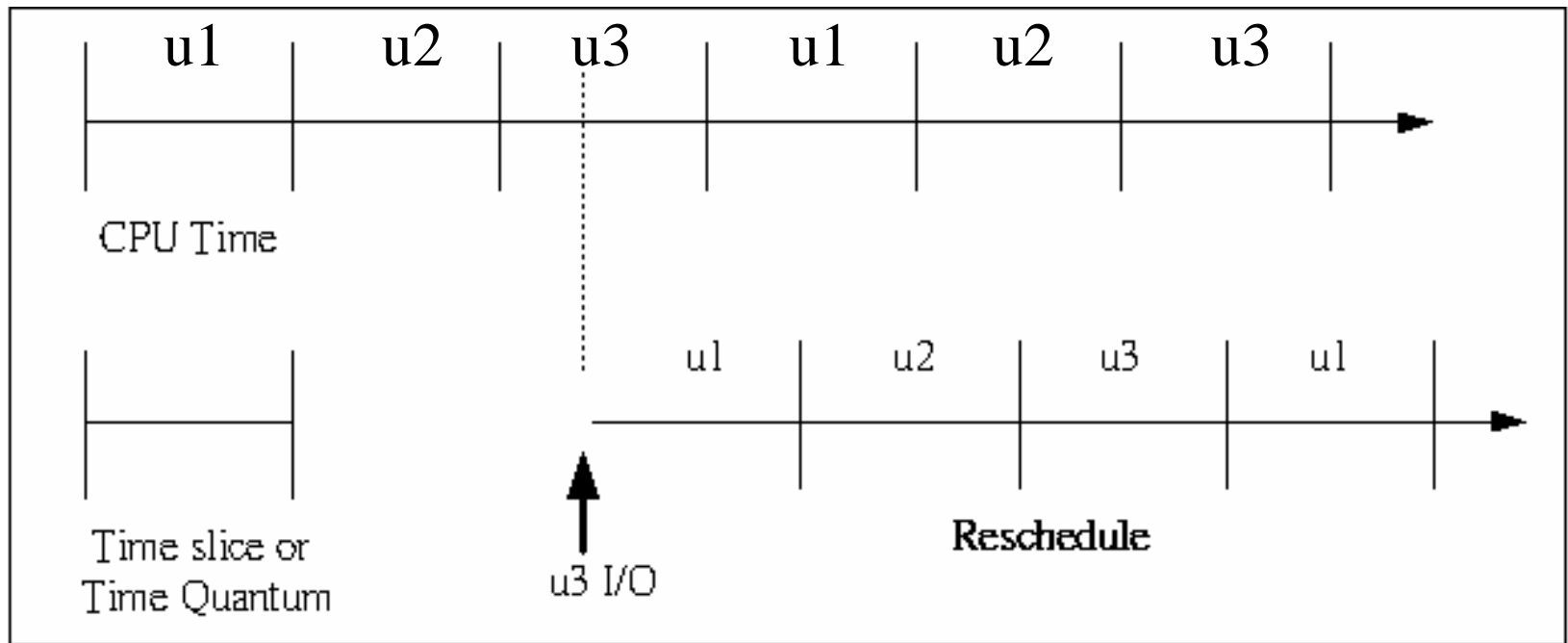


圖 2.3 分時系統

分時系統（Time Sharing Systems）(3)

- 程式要進行輸入 / 輸出或等待事件發生時，程式便到懸置佇列（Blocked Queue）去排隊，而中央處理器排程至等待佇列中，取下一個程式佔有中央處理器並執行工作，此時中央處理器排程亦會重新劃分時間片段，讓每個人享有相同之時間片段。
- 當在懸置佇列內的程式完成輸入 / 輸出或等待的事件發生，則進入等待佇列的最後面，並與等待佇列內的程式輪流享有中央處理器。

分時系統（Time Sharing Systems）（4）

- 爲達到快速輪流服務的目的，一個分時系統一定採用多重程式功能。
- 多重程式系統並不一定提供分時的功能。

週邊設備線上同時工作 (Spooling)(1)

- 當一個程式在使用專屬設備 (Dedicated Device) 時，在還沒有用完之前，別的程式是不能使用這個設備的。例如：印表機。

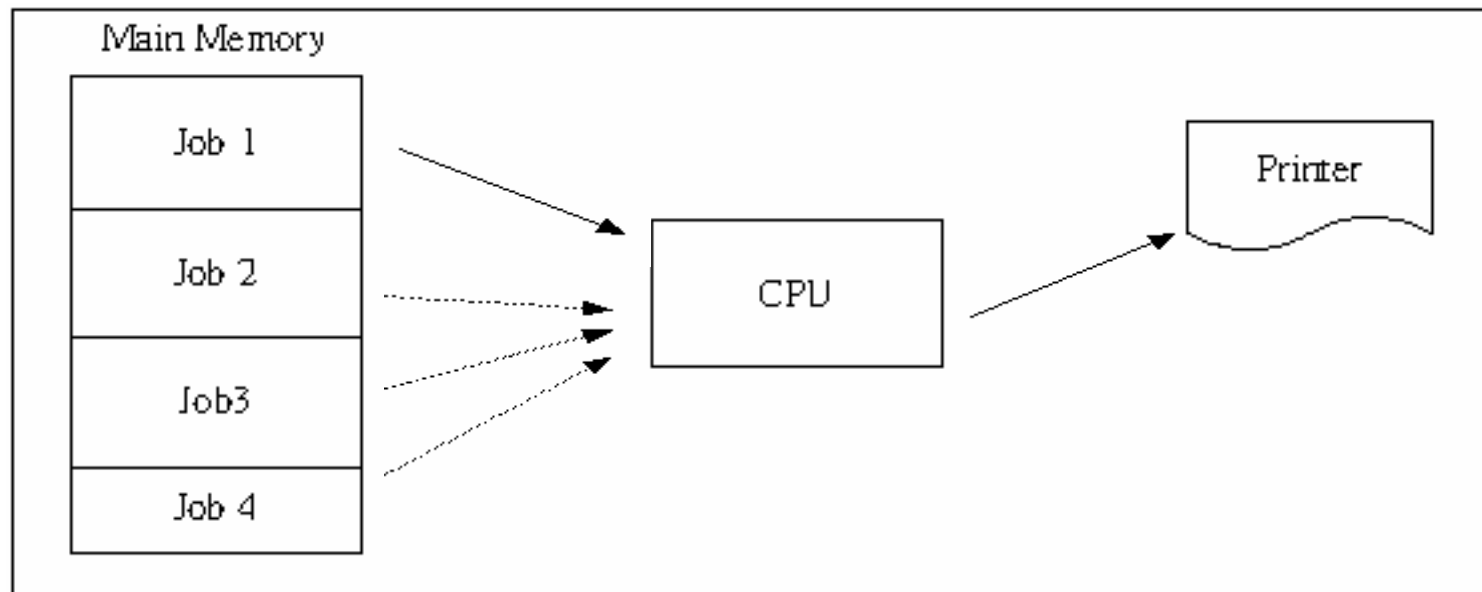


圖2.4 多個程式使用印表基的問題

週邊設備線上同時工作 (Spooling)(2)

- 週邊設備線上同時工作 (Simultaneous Peripheral Operation On Line)，簡稱Spooling，就是用來解決共用專屬設備的機制。
- 它使得專屬設備被模擬成分享設備 (Shared Device)，而且不會有你列印一行，我列印一行的問題發生。

週邊設備線上同時工作 (Spooling)(3)

- 作業系統內的週邊設備線上同時工作軟體 (Spooler) 會出面將所有輸出均暫時儲存在硬碟中，然後週邊設備線上同時工作軟體會依據印表機的使用狀況，依序由磁碟中，將欲輸出的資料由印表機輸出。
- Spooling essentially uses the disk as a very large buffer, for storing output files until output devices are able to accept them.

週邊設備線上同時工作 (Spooling)(4)

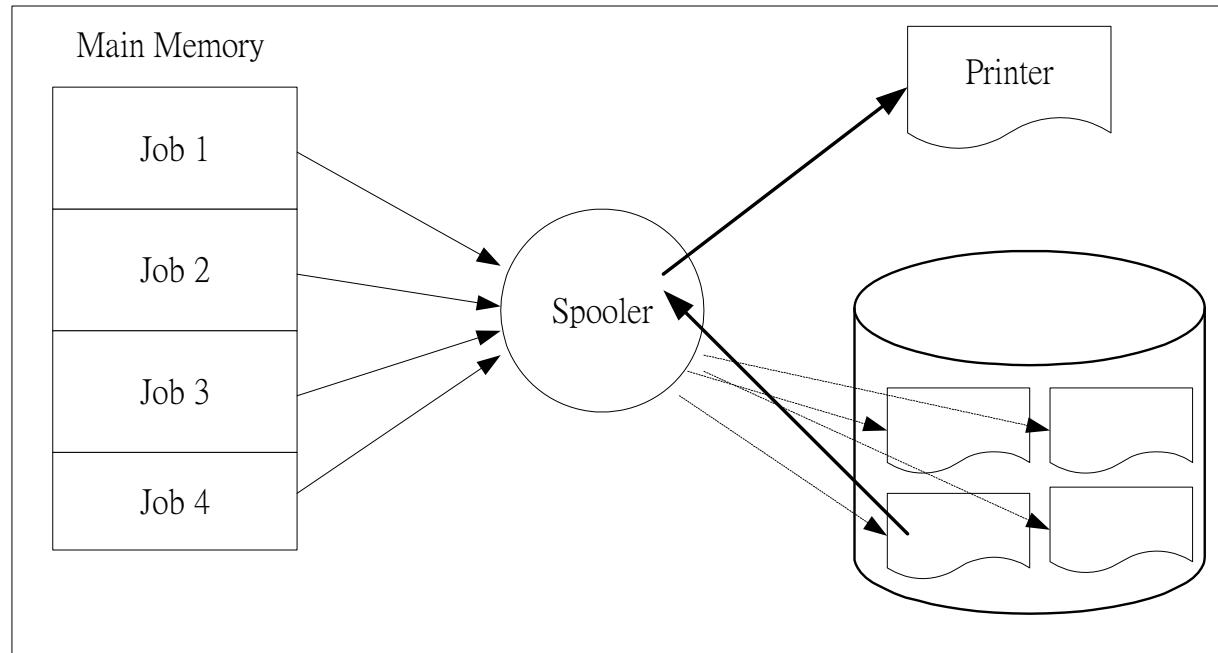


圖2.5 週邊設備同時線上工作

週邊設備線上同時工作 (Spooling)(5)

- 週邊設備線上同時工作可以使得一個程式的輸入 / 輸出與另一個程式的計算同時進行。
- Spooling overlaps the I/O of one job with the computation of other jobs.
- 週邊設備線上同時工作可以同時增加中央處理器及輸入 / 輸出設備之使用率。
- Improve both CPU and I/O devices utilization.

反應時間 (Response Time)

- 提出要求到電腦回應所需的時間。
- Time from the submission of a request until the first response is produced.

即時系統（Real Time Systems）（1）

- 立即針對問題反應，並在所規範的條件內自動執行相關軟體，且得到正確的結果。
- It is desirable to **response quickly** and returns the correct results within the **defined constraints**.
- 在即時系統中，反應時間（Response Time）非常重要。

即時系統（Real Time Systems）（2）

- 即時系統有時又被區分為硬性即時（Hard Real Time）及軟性即時（Soft Real Time）。
 - 硬性即時：必須在限制的時效內，即時反應並完成所有處理工作。
 - 軟性即時：在限制時效內即時反應，而且應變程式持續佔有中央處理器進行善後處理，所以它並沒有像硬性即時規定的那麼嚴格。

即時系統（Real Time Systems）（3）

- 由於即時系統必須立即反應，故它一定是連線的。
- 連線系統並不一定需要即時。

多處理器系統 (Multiprocessor Systems) (1)

- 在一部電腦內有許多顆中央處理器，它們同時執行工作，並共用電腦的匯流排 (Bus)、時脈 (Clock)、記憶體、週邊設備，且共用同一個作業系統，這便是多處理器系統 (Multiprocessor Systems)。
- 亦稱為多重處理 (Multiprocessing)。
- 它可以讓多顆中央處理器同時執行工作，故又可稱為平行系統 (Parallel System)。
- 由於共用一部電腦內的資源，所以它們是一種緊密性系統 (Tightly Coupled System)。Single OS and single memory.

多處理器系統 (Multiprocessor Systems) (2)

- 在一部電腦內有許多顆中央處理器，使得電腦的處理能力增加(產量及可靠度)。
- Several processors are used on a single computer system, and to increase the processing power of computer (Throughput and Reliability)
- 所謂優雅的衰減 (Graceful Degradation) 就是說，在一個多處理器系統中，壞掉一顆中央處理器，並不會讓電腦系統停止運作，而只是讓電腦的執行效果降低。
- 這種提供優雅的衰減功能的系統具有軟性失誤 (Fail Soft) 之特性。

多處理器系統（Multiprocessor Systems）(3)

- 在多處理器系統中執行程式，可以有兩種方式
 - 第一種是每個程式分配給一個中央處理器執行，若遇有程式與程式互相交換訊息的需求時，再由系統執行同步（Synchronous）的工作。
 - 另一種方式是一個程式可同時分配多個中央處理器執行。

多處理器系統 (Multiprocessor Systems) (4)

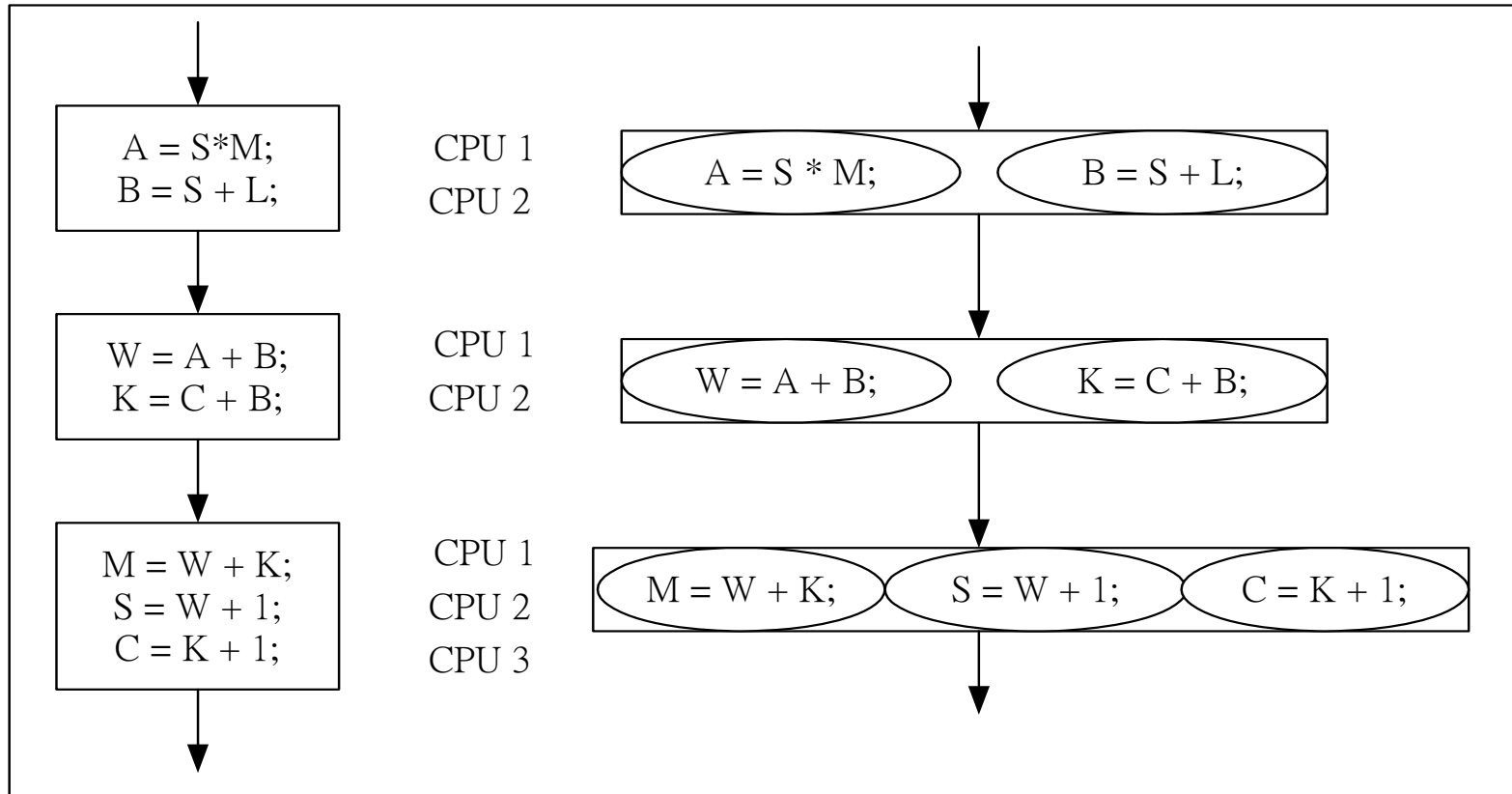


圖2.6 多處理器系統執行程式的情形

多處理器系統 (Multiprocessor Systems) (5)

- 多處理器系統之硬體結構會影響作業系統，通常多處理器作業系統可分為
 - 對稱性多重處理 (Symmetric Multiprocessing) 系統
 - 非對稱性多重處理 (Asymmetric Multiprocessing) 系統。

對稱性多重處理系統

- 所有中央處理器都是一致且同等地位的，沒有一個主要的中央處理器；作業系統依管理的需求，分派中央處理器工作，這些中央處理器透過匯流排互相通訊並彼此分工合作。
- Each processor runs an **identical** copy of the OS, and these copies communicate with one another as needed.

非對稱性多重處理系統

- 中央處理器當中選定一個主處理器（Master Processor），並且由主處理器執行作業系統；而其他的中央處理器則稱為從屬處理器（Slave Processor），從屬處理器接受主處理器的命令，進行特殊或事先規定的工作。
- Each processor is assigned a special task. A **master** processor control the system; the other processors either look to the master for instruction or have predefined tasks.
- 非對稱性多重處理結構有時又稱為主從式（Master/Slave）架構。

多處理器系統 (Multiprocessor Systems) (6)

- Front end processor: I/O interface內有CPU.
Handles terminal I/O.
- Back end processor: 在 Disk controller (Interface)內有CPU處理。

多處理器系統 (Multiprocessor Systems) (7)

- 多處理器系統因為有許多顆中央處理器，所以在實際用途上，可以有許多的應用變化，茲列舉它的應用如下：
 - 雙工系統 (Duplex System)
 - 雙套系統 (Dual System)
 - 雙工/雙套混合型

雙工系統 (Duplex System) (1)

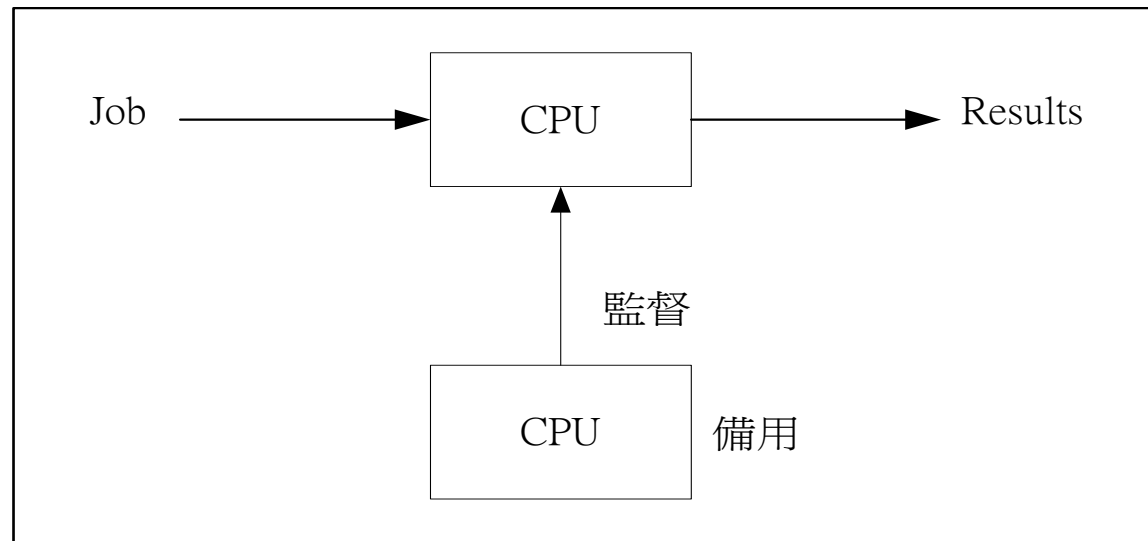


圖2.7 雙工系統結構

雙工系統（Duplex System）（2）

- 平時使用一套硬體進行工作，另一套為備用，當運作的硬體出了問題時，系統自動轉換到備用系統執行。
- 在不必關機的情況下，進行設備、介面卡抽換工作，也就是所謂的熱抽換（Hot Swap）。
- 由於可以進行熱抽換及轉用備用系統，所以它是一個容錯（Fault Tolerance）非常高的系統；這類系統通常都具有永不停止（Non - Stop ）的優點。

雙套系統（Dual System）（1）

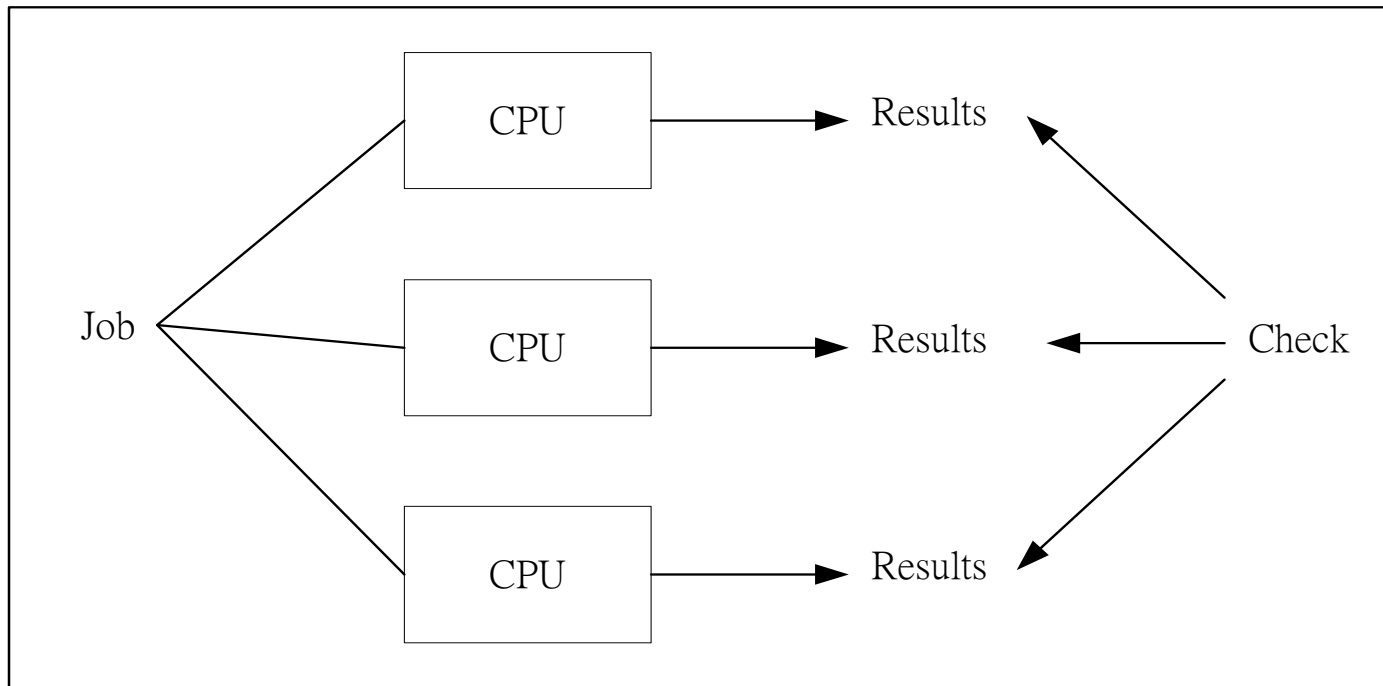


圖2.8 雙套系統結構

雙套系統（Dual System）（2）

- 爲了確保資料的正確性，可以讓多套系統同時進行相同的計算，然後依據每套系統計算結果進行比較，以確定正確性。
- 當有系統計算結果不一致的問題發生時，可以利用少數服從多數或重新計算...等方法，來解決問題。
- 所謂雙套系統並不一定只有兩套，原則上同時具有多套系統執行運作便稱爲雙套系統。

雙工/雙套混合型

- 平時以雙套系統結構進行工作，當有電腦系統壞掉時，再啓動備用系統，加入雙套系統內執行。
- 這類系統主要是要求不停止且必須計算精確。

分散式系統（Distributed Systems）（1）

- 將電腦經由通訊線路（Communication Lines）連接，使得彼此可以分工合作、交換訊息、及共用資源。
- User gain access to networks of geographically dispersed computers through various types of communication lines.

分散式系統 (Distributed Systems) (2)

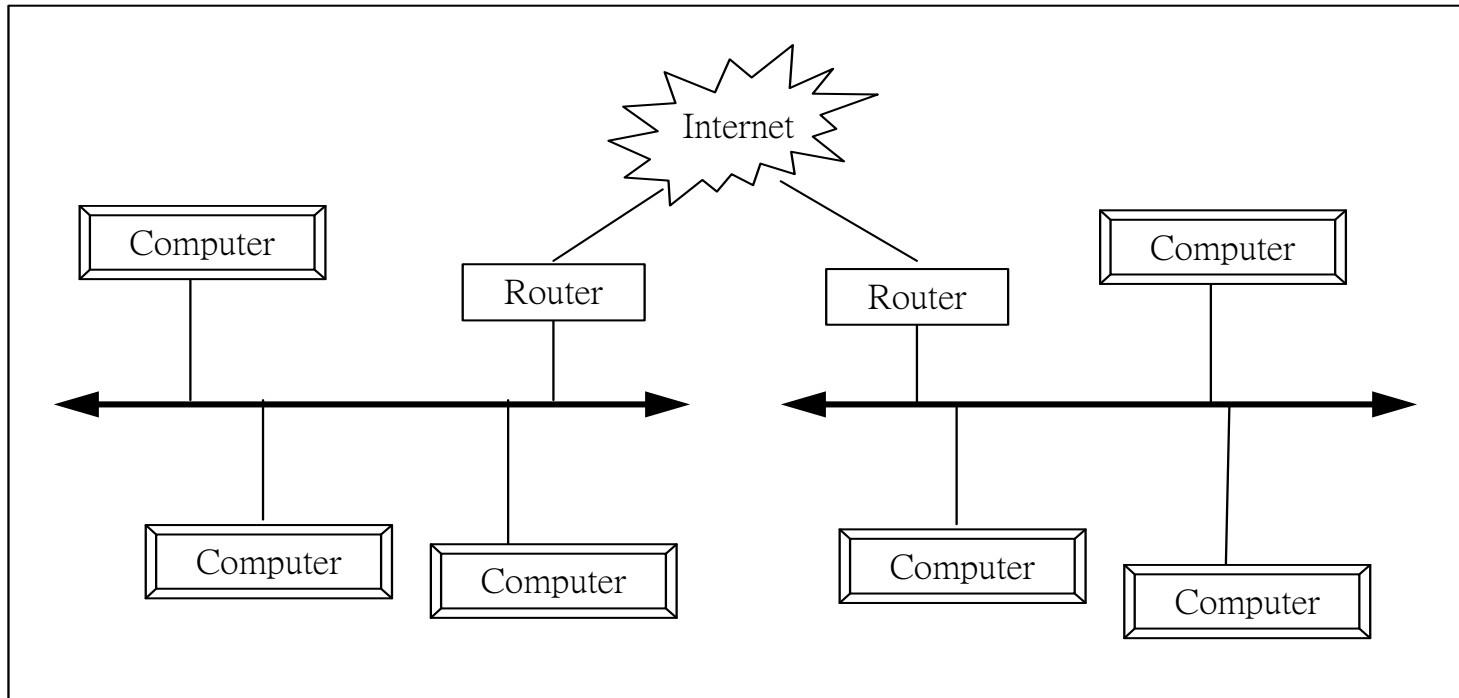


圖2.9 分散式系統

分散式系統 (Distributed Systems) (3)

- 分散式系統是一種鬆散性系統(Loosely Coupled System)。Each system has its own OS and memory.
- 多處理器系統是一種緊密性系統(Tightly Coupled System)。

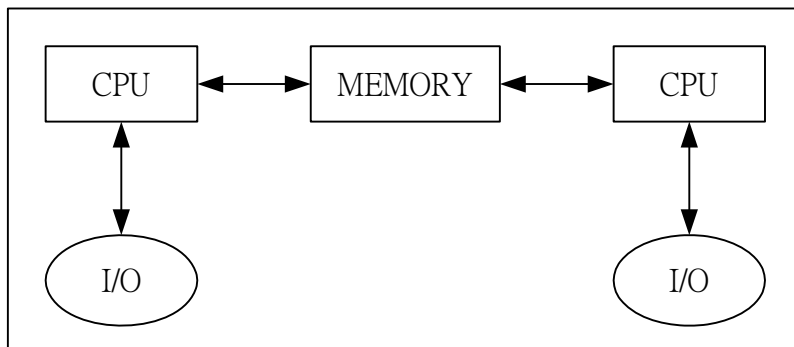


圖2.10 緊密性系統

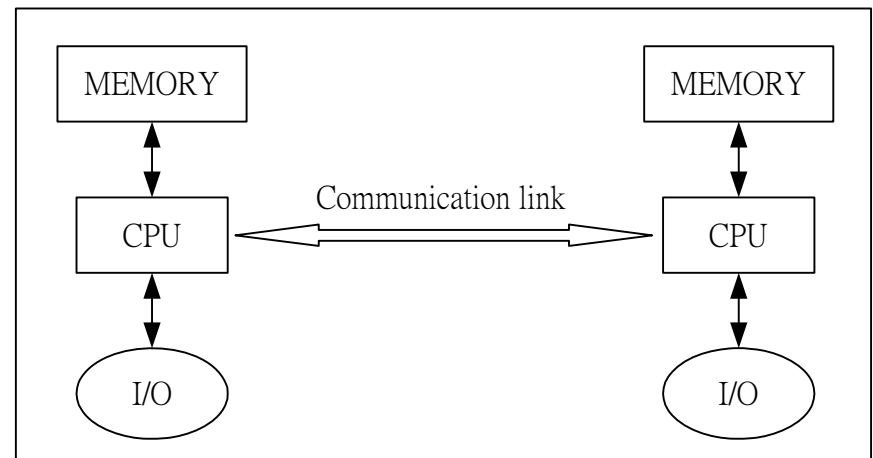


圖2.11 鬆散性系統

分散式系統的優點

- 資源共享（Resources Sharing）
 - Increased data availability.
- 增快計算速度（Computation Speedup）
 - Parallel Virtual Machine (PVM)
 - Load sharing: 將工作較重的CPU的工作分一些給較輕的CPU
- 可靠度（Reliability）
 - 壞了一部還可以工作。
- 通訊（Communication）
 - e-mail, FTP,...

分散式系統（Distributed Systems）（4）

- 分散式系統環境之作業系統分爲以下兩類：
 - 網路作業系統（Network Operating Systems）
 - 分散式作業系統（Distributed Operating Systems）

網路作業系統（Network Operating Systems）

- 連接在一起的電腦，可以使用不同的作業系統。
- 這些電腦彼此共用資源時，必須使用遠端登錄或直接以檔案傳輸方式傳輸檔案使用。
- 網路作業系統，簡稱為NOS。

分散式作業系統 (Distributed Operating Systems)

- 連接在一起的電腦，使用相同的作業系統。
- 任何遠端的資源，都被作業系統視為本身的資源，而可以直接存取，（當然了，它有時必須有密碼...等保護）。
- 分散式作業系統簡稱為DOS。

現代化的作業系統

- 目前在市面上普遍的作業系統特色：
 - 多人使用
 - 架構在分散式系統環境
 - 透過網路共用資源及通訊
 - 具備分時系統功能
 - 週邊設備線上同時工作
 - 即時反應，優先等級機制
 - 批次處理