

I/O

Dispositivi di input/output

Corso di Calcolatori Elettronici A

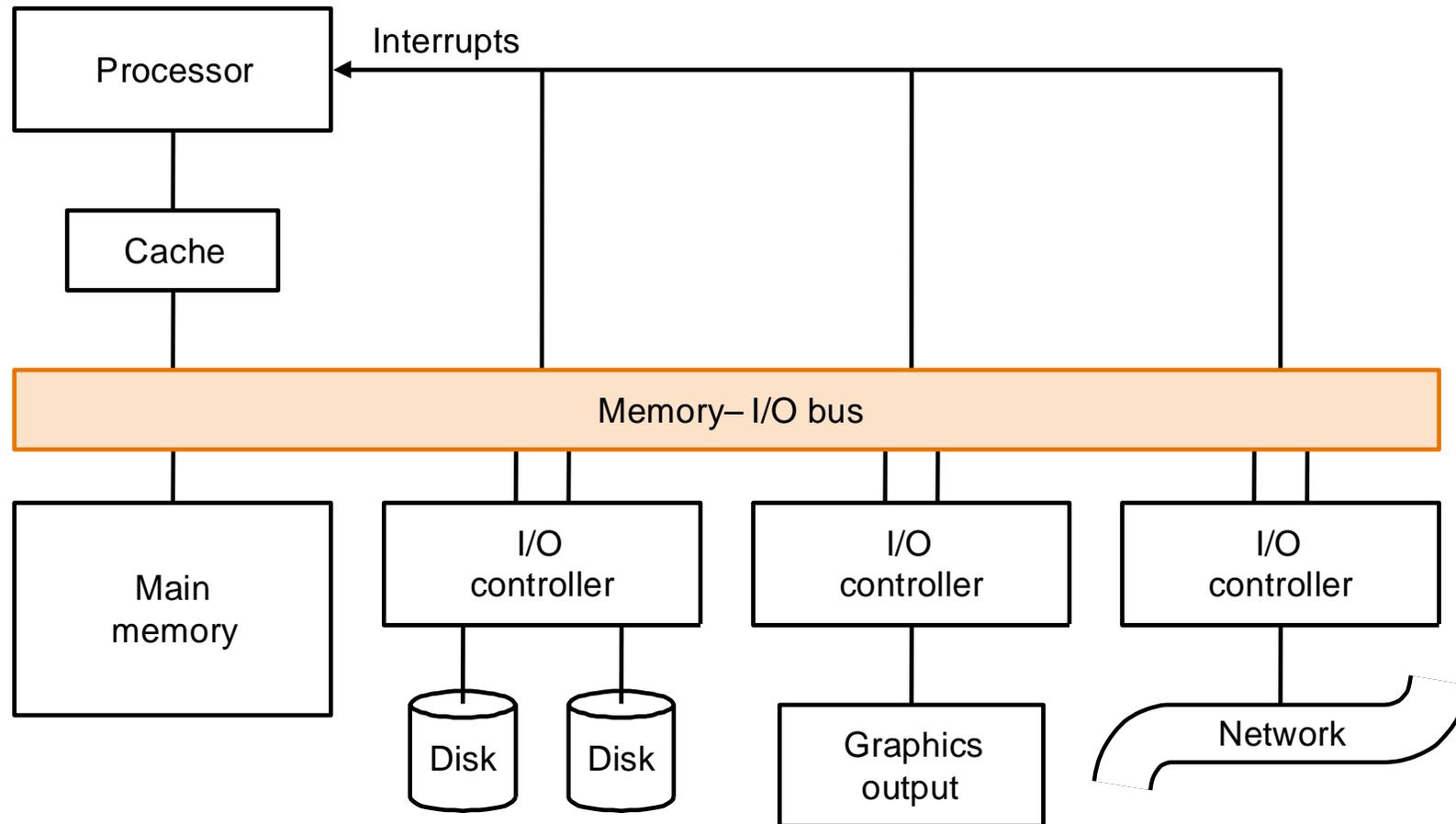
2007/2008

Sito Web: <http://prometeo.ing.unibs.it/quarella>

Prof. G. Quarella

prof@quarella.net

Dispositivi di I/O



Prestazioni

Le prestazioni di un sistema di I/O sono complesse da valutare, in quanto, in base al tipo di dispositivo, bisogna considerare parametri differenti e sono influenzate da molti fattori come le modalità di connessione, la gerarchia delle memorie, il sistema operativo, ecc..

In generale alcuni parametri fondamentali da considerare per valutare le prestazioni sono:

- **Latenza di accesso**
- **Throughput**

Le prestazioni delle periferiche influenzano le prestazioni del calcolatore (legge di Amdahl).

Prestazioni

Il **tempo di risposta**, ovvero il tempo impiegato da un dispositivo per completare una determinata attività, è il parametro da ottimizzare in molte applicazioni. Tale tempo dipende sia dalla latenza di accesso, sia dall'ampiezza di banda. Se le operazioni di I/O sono frequenti e comportano il trasferimento di pochi dati, è più importante l'ottimizzazione della latenza di accesso, diversamente se le operazioni di lettura/scrittura sono pesanti, una maggiore ampiezza di banda abbasserà il tempo di risposta.

Per il progetto di un sistema di I/O bisogna trovare il giusto compromesso fra tempo di risposta e throughput, dato che massimizzare il throughput farà in certi casi aumentare il tempo di risposta.

Misura delle prestazioni dell'I/O

La velocità di trasferimento, che dipende dalla frequenza di clock espressa solitamente in MHz, viene espressa in MB/s. Nei sistemi di I/O però si utilizza la base 10 anziché la base 2 come per le memorie, quindi:

1 KB = $10^3 = 1.000$ byte invece che $2^{10} = 1.024$ byte

1 MB = $10^6 = 1.000.000$ byte invece che $2^{20} = 1.048.576$ byte

1 GB = $10^9 = 1.000.000.000$ byte invece che $2^{30} = 1.073.741.824$ byte

Es.: il tempo per trasferire 100MB di dati su un bus a 100 MB/s è pari a $1048576/1000000=1,048576$ s, non considerando la differenza di basi, l'errore che si introduce è abbastanza piccolo. Diventa più grande confondendo 10^9 con 2^{30} .

Per evitare questa confusione si usa talvolta il minuscolo (k, m, g) per indicare l'utilizzo della base 10.

Tipi di dispositivi di I/O

I dispositivi di I/O possono essere classificati in base alle seguenti caratteristiche:

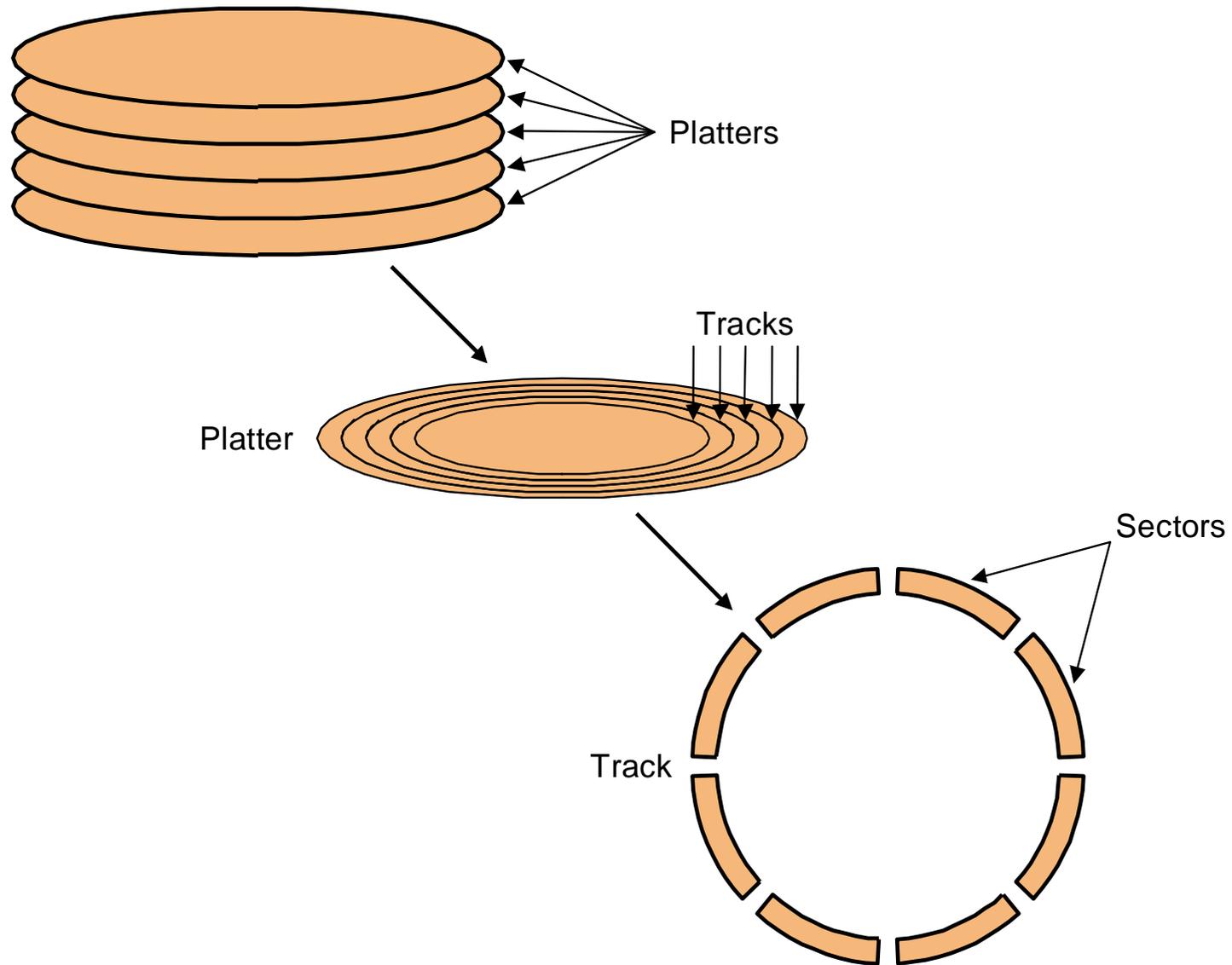
- **Comportamento:** input, output o memorizzazione (possono essere riletti e di solito riscritti).
- **Partner:** il dispositivo può mettere in comunicazione il calcolatore con l'uomo o con un'altra macchina.
- **Frequenza di trasferimento dati:** la frequenza di picco alla quale i dati possono essere trasferiti tra il dispositivo di I/O e la memoria principale o il processore.

Tipi di dispositivi di I/O

Il processore è il soggetto dell'attività di input-lettura o output-scrittura, quindi abbiamo per esempio:

- **Dispositivi di input:** tastiera, mouse, ecc.
- **Dispositivi di output:** video, stampante, ecc.
- **Dispositivi di input o output:** modem, rete, ecc.
- **Dispositivi di memorizzazione:** floppy disk, hard disk, ecc.

Il disco fisso



Prestazioni del disco fisso

Per valutare le prestazioni sono necessari i seguenti dati:

- **Velocità di rotazione (revolutions per minute RPM)** (es.: 7200 RPM)
- **Tempo di seek.** I produttori forniscono i tempi di seek medi (8-20 ms), ma a causa della località degli accessi i valori medi reali possono essere solo il 25-33% dei valori dichiarati.
- **Latenza o ritardo di rotazione:** la latenza media corrisponde a mezza rotazione ($1/2 \times 1/(RPM/60) = 30/RPM$ s).
- **Tempo di trasferimento:** riferito al trasferimento di un blocco di bit (di solito un settore). La presenza di una cache integrata può rendere difficile la valutazione di tale tempo.
- **Tempo del controllore**

Es.: tempo medio di accesso di un disco fisso

Disco fisso con velocità di rotazione pari a 5400 RPM, tempo medio di seek indicato dal costruttore 12 ms, tempo di trasferimento 5 MB/s e tempo del controllore pari a 2 ms. Calcolare il tempo medio di accesso ad un settore di 512 byte, supposto che il disco non sia impegnato in altre operazioni.

T_m = tempo medio di seek + ritardo di rotazione medio + tempo di trasferimento + tempo di controllore

$$T_m = 12 + 30/5400 \cdot 10^3 + 0,5 \text{ KB}/(5 \text{ MB/s} \cdot 10^3) + 2 = 19,7 \text{ ms}$$

Se il tempo medio di seek fosse il 25% di quello dichiarato

$$T_m = 10,7 \text{ ms}$$

Affidabilità - Disponibilità

L'**Affidabilità** (Reliability) è la probabilità che un sistema funzioni senza guastarsi in un intervallo di tempo assegnato, date le condizioni operative ed ambientali.

Il **Mean Time To Failure** (MTTF espresso in ore) è una misura dell'affidabilità

La **Disponibilità** (Availability) è la probabilità che un sistema funzioni anche in presenza di guasti.

L'interruzione del servizio è misurata dal **Mean Time To Repair** (MTTR)

$$A = \frac{MTTF}{(MTTF + MTTR)}$$

RAID

Redundant Arrays of Inexpensive Disks (Patterson 1988)

Inexpensive → Independent

Obiettivo: **fault tolerance** – prestazioni – capacità

In generale l'affidabilità peggiora (MTTF inversamente proporzionale al numero dei dischi)

La **ridondanza** migliora la **disponibilità**.

Più dischi solitamente identici visti come uno solo

Varie tipologie di RAID chiamate **livelli**

Usato per lo più su server

Implementazione via hardware (controller – costosa, ma prestazioni migliori) o via software (SO)

Funzionalità aggiuntive: **hot swapping** – **hot spare**

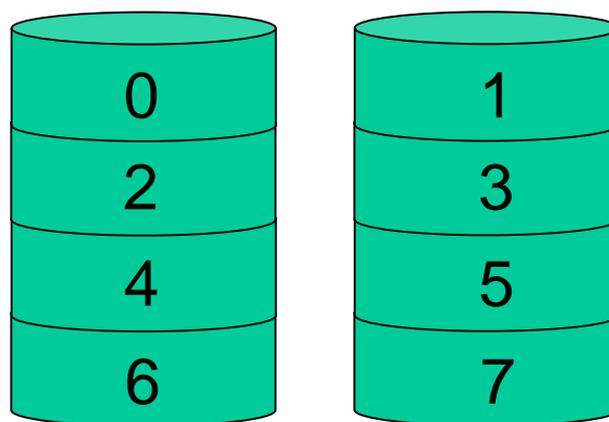
RAID 0

Striping – Striped Set (blocchi di dimensione pari ad un multiplo della dimensione di un settore)

Non è propriamente un livello RAID

- Nessuna ridondanza (parità, dischi di controllo)
- Nessuno spreco di capacità
- Scarsa affidabilità
- Maggiori prestazioni (applicazioni di Video-editing)

Lo striping, insieme alla ridondanza, è usato anche in altri livelli di RAID



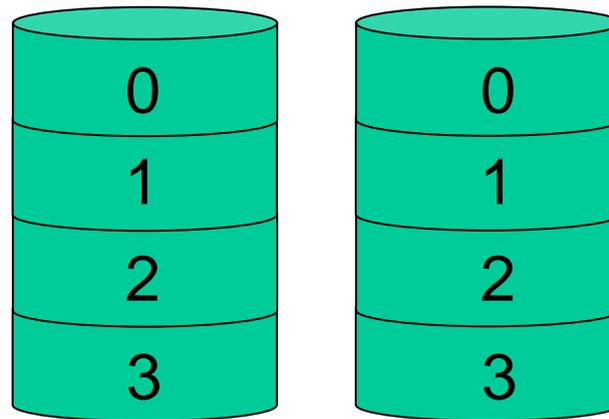
RAID 1

Mirroring – Mirror set – Shadowing

Si crea una copia di ogni disco – Spreco di capacità 50%

L'affidabilità raddoppia (con una copia sola)

Prestazioni in lettura raddoppiate (Database)



Parità

Informazione ridondante che permette di rilevare la presenza di errori in un parola di bit.

Parità pari (schema in cui la parità deve essere pari)

Si aggiunge un bit in modo che il conteggio dei bit 1 sia pari

0101111 → 0101111**1** (0 xor 1 xor 0 xor 1 xor 1 xor 1 xor 1 = 1)

0000000 → 0000000**0**

Parità dispari (schema in cui la parità deve essere dispari)

Si aggiunge un bit in modo che il conteggio dei bit 1 sia dispari

0101111 → 0101111**0**

0000000 → 0000000**1**

Un singolo bit di parità permette di rilevare l'alterazione di un numero dispari di bit, ma non quella di un numero pari di bit.

RAID 3

Bit-Interleaved Parity

Almeno 3 dischi, 1 dedicato alla parità.

1/N capacità persa

Se un disco si guasta l'informazione mancante può essere ricostruita a scapito delle prestazioni, in quanto deve soddisfare la parità pari. Sostituire il disco guasto il prima possibile. Un secondo disco guasto non è tollerato.

Ogni lettura comporta l'accesso a tutti di dischi dati.

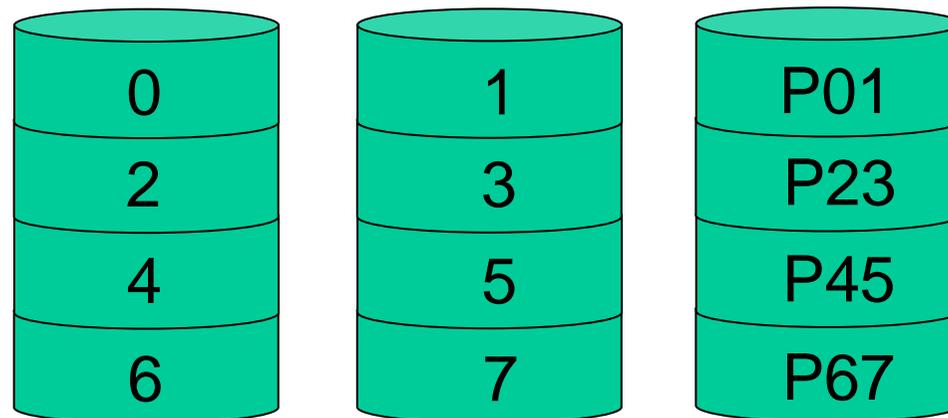
Ogni scrittura comporta l'accesso anche al disco di parità.

RAID 4

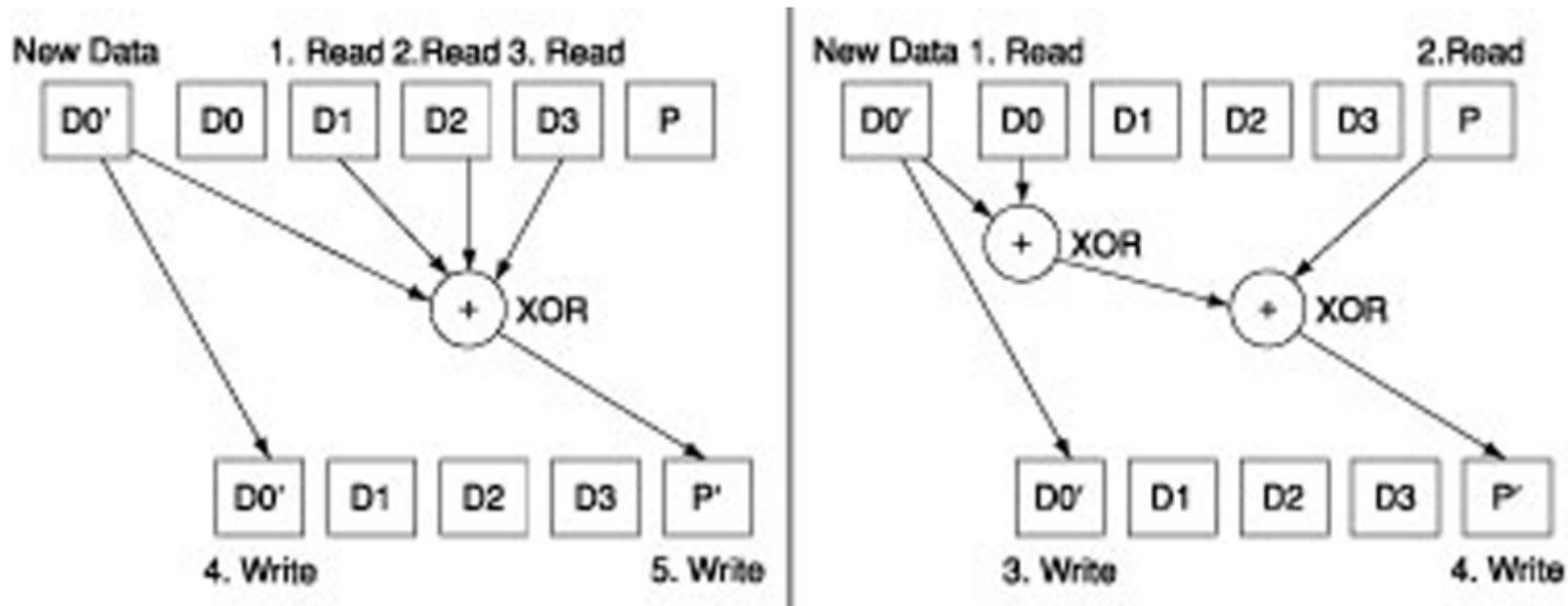
Block-Interleaved Parity

Come RAID 3 eccetto che l'unità di stripe è un blocco invece che un bit.

Ogni lettura piccola (inferiore al blocco) comporta l'accesso ad un singolo disco; quindi per piccole scritture si può ottimizzare l'aggiornamento della parità rispetto al RAID 3.



RAID 4



Invece che leggere tutti i blocchi per ricalcolare la parità

$$P' = D0' \text{ xor } D1 \text{ xor } D2 \text{ xor } D3$$

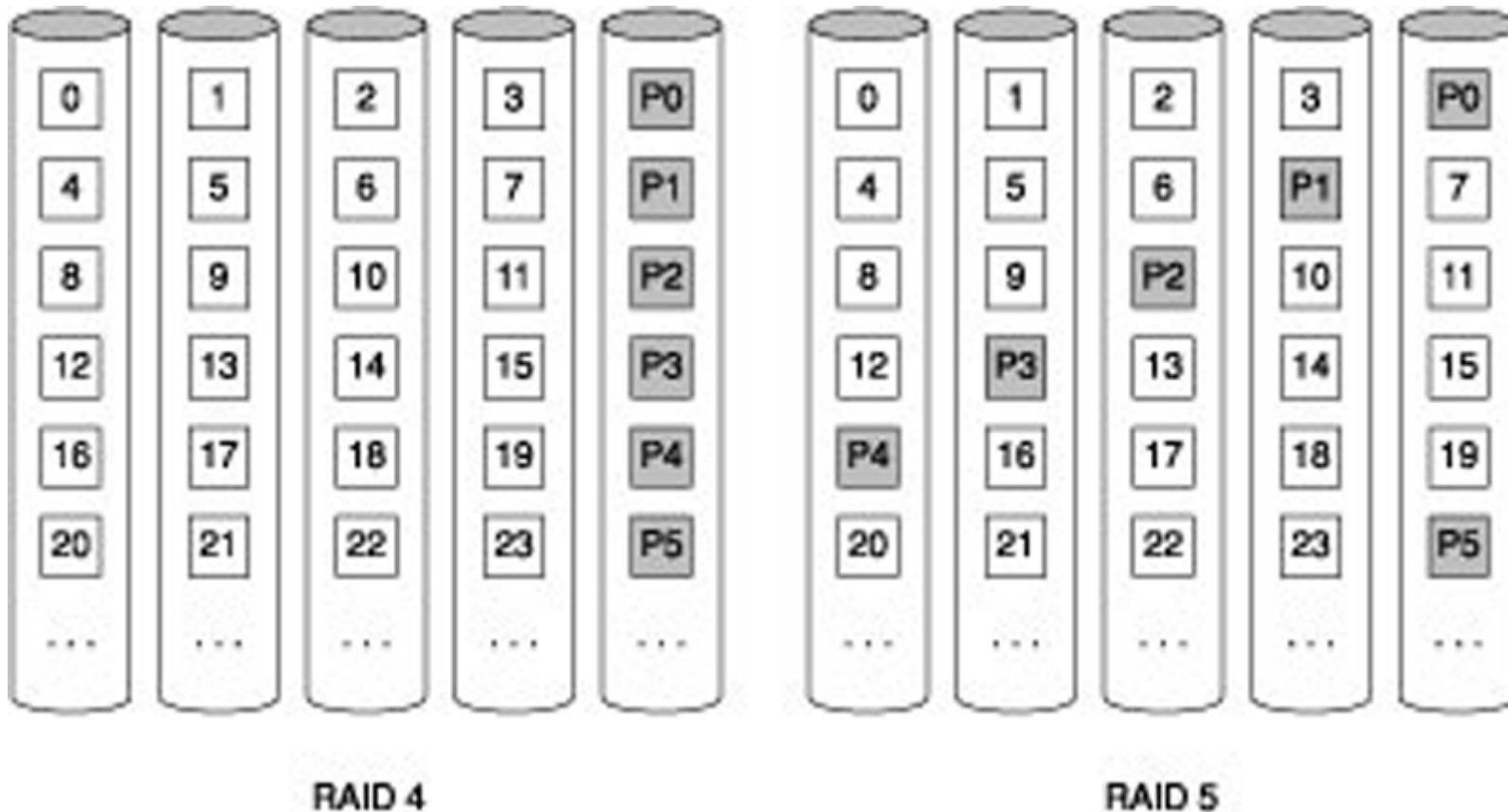
Si può effettuare il seguente calcolo

$$P' = D0' \text{ xor } D0 \text{ xor } P = D0' \text{ xor } D0 \text{ xor } D0 \text{ xor } D1 \text{ xor } D2 \text{ xor } D3$$

Si risparmia in termini di letture in proporzione al numero dei dischi nel gruppo di protezione.

RAID 5

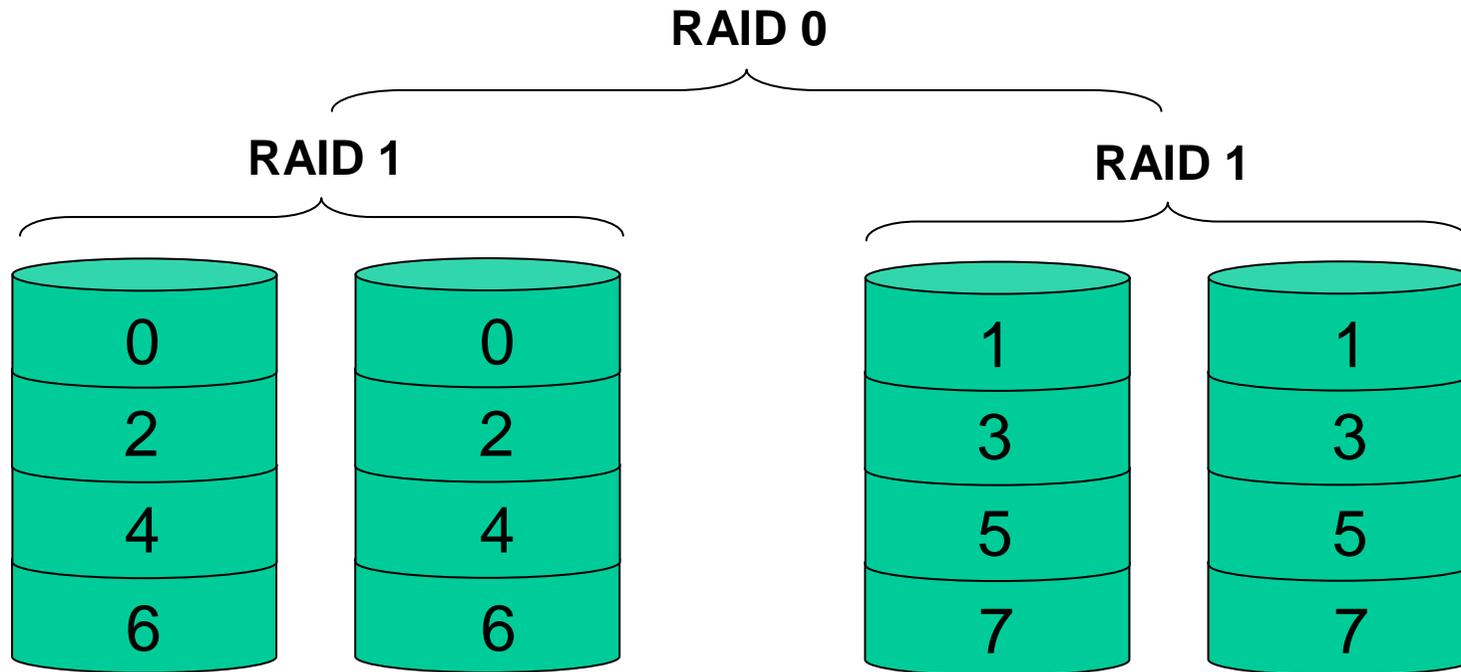
Distributed Block-Interleaved Parity



No c'è più un disco dedicato alla parità, che quindi non è più il collo di bottiglia.

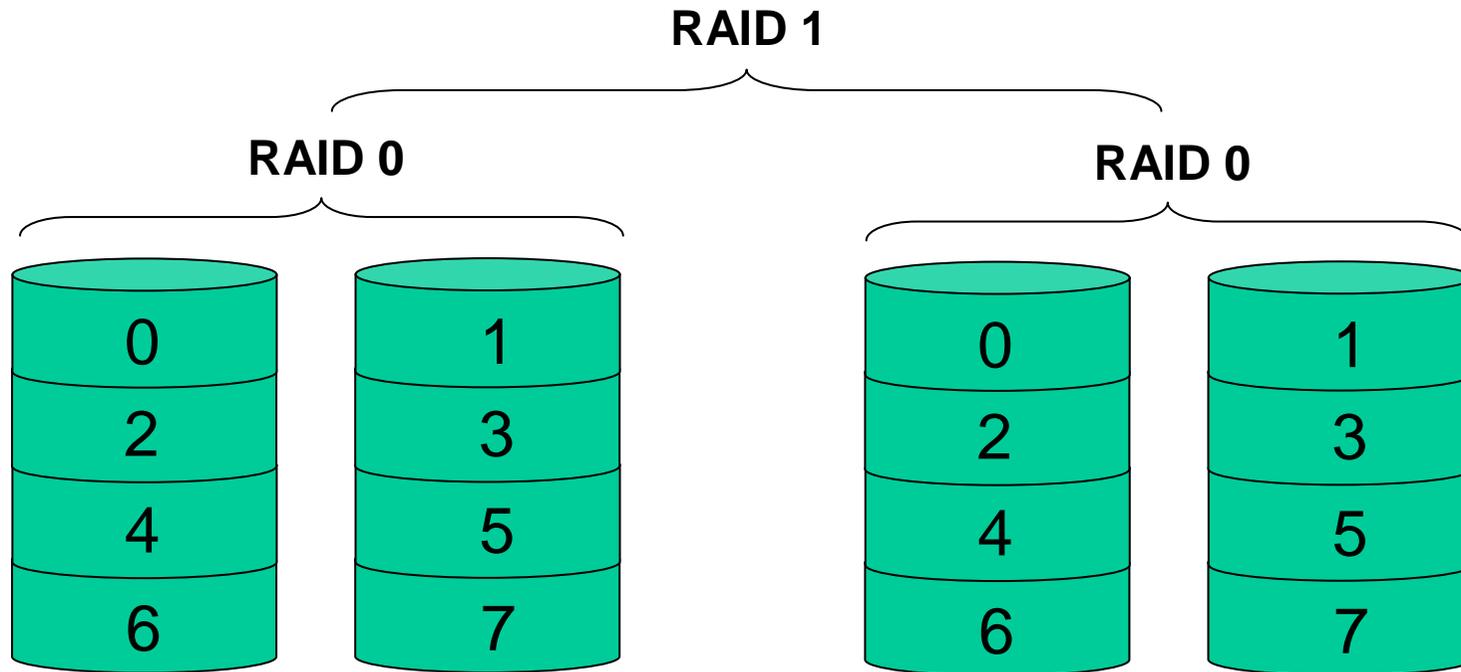
RAID 10

Striped mirrors – RAID 1+0



RAID 01

Mirrored stripes – RAID 0+1



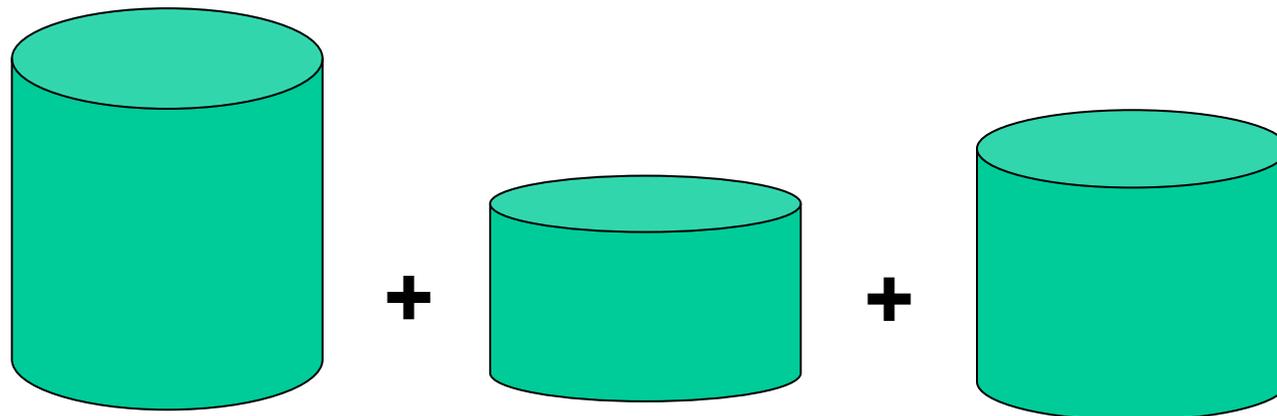
JBOD

Just a Bunch Of Disks

Concatenazione dello spazio di più dischi.

Nessuna ridondanza come RAID 0, ma migliore in caso di guasti.

Nessun incremento delle prestazioni



Il sistema operativo e i sistemi di I/O

Il sistema operativo è l'interfaccia fra l'hardware e i programmi che effettuano richieste di I/O.

Nella gestione dei dispositivi di I/O deve considerare le seguenti caratteristiche:

- **Il sistema di I/O è condiviso fra i programmi.**
- **I dispositivi di I/O spesso usano le interruzioni per comunicare informazioni.**
- **Il controllo di basso livello di un dispositivo di I/O è estremamente complesso.**

Il sistema operativo e i sistemi di I/O

Funzionalità offerte

- **Gestione dei permessi di accesso ad un dispositivo.**
- **Procedure specifiche per gestire le operazioni di basso livello sui dispositivi (astrazioni).**
- **Gestione interruzioni.**
- **Gestione dell'equità nell'accesso alle risorse di I/O condivise ottimizzando il throughput.**

Il sistema operativo e i sistemi di I/O

Tipi di comunicazione

- 1. Invio da parte del S.O. di comandi di lettura, scrittura ed altri comandi dipendenti dal dispositivo.**
- 2. Invio da parte del dispositivo di informazioni sulle operazioni svolte e di stato (errori).**
- 3. Trasferimento dati fra memoria e dispositivo di I/O.**

Indirizzamento dispositivi di I/O

Fornire comandi ai dispositivi di I/O

- **I/O mappato in memoria**
 - Porzione dello spazio di indirizzamento riservato all'I/O
 - Protezione: indirizzi riservati al SO
 - L'indirizzo codifica l'identità e il tipo di trasmissione
- **Istruzioni speciali di I/O**
 - Specificano il dispositivo e la parola di comando (o la sua posizione in memoria)
 - Istruzioni utilizzabili solo in modalità kernel

Comunicazione CPU – I/O

- **Interrogazione o polling (*sondaggio*) o I/O programmato**
- **Interruzioni**
- **DMA (Direct Memory Access)**

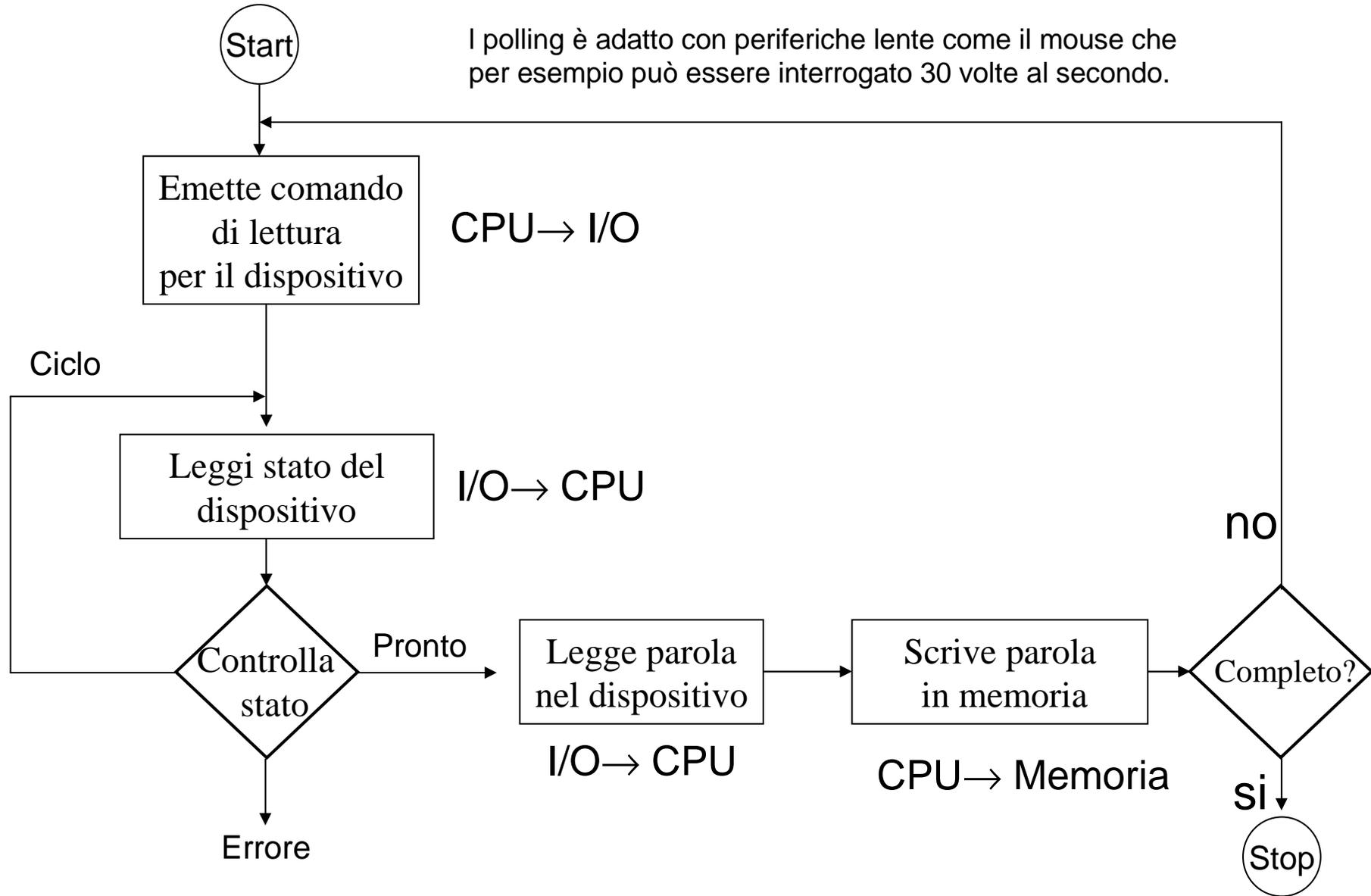
Polling – I/O programmato

Processo di verifica dei bit di un registro di stato

- Il dispositivo scrive in questo registro informazioni sul suo stato: completamento operazioni di I/O, condizioni di errore, ecc.
- Ad intervalli regolari la CPU esegue una routine di polling ritornando poi al programma utente
- Si spreca tempo di CPU se la frequenza di polling è elevata; le periferiche
 - sono spesso più lente
 - possono essere inattive
- Inadatto per dispositivi veloci

Polling – I/O programmato: es. di lettura

Il polling è adatto con periferiche lente come il mouse che per esempio può essere interrogato 30 volte al secondo.



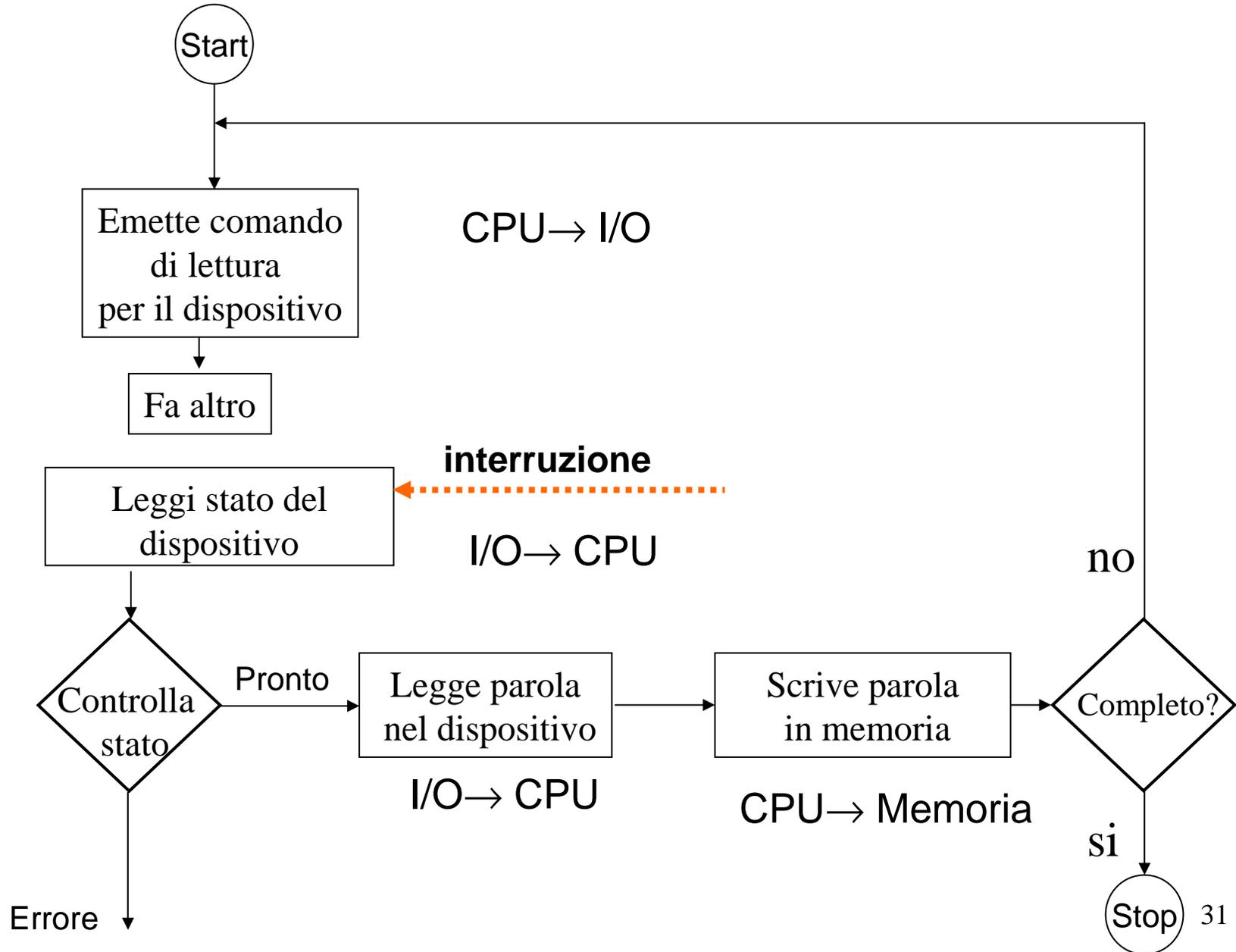
Interruzioni di I/O

Le interruzioni di I/O indicano al processore che un dispositivo richiede attenzione, evitando il costo associato al meccanismo di polling. Rispetto alle eccezioni (interne) si hanno le seguenti differenze:

- Un'interruzione di I/O è asincrona rispetto all'esecuzione delle istruzioni.
- È necessario fornire informazioni riguardo al dispositivo che richiede l'intervento del sistema operativo.

L'identità del dispositivo viene comunicata al processore tramite le **interruzioni vettorizzate** o il **registro Causa** delle eccezioni.

Interruzioni di I/O: es. di lettura



Direct Memory Access

Polling e interruzioni di I/O

- Il trasferimento dati è gestito dal processore
- Il numero di byte per operazione è limitato

Per dispositivi con elevata ampiezza di banda, come gli hard disk, la frequenza delle interruzioni durante il trasferimento dati sarebbe tale da occupare una significativa percentuale del tempo del processore; serve quindi un meccanismo che tolga lavoro al processore e nel contempo permetta di trasferire centinaia di KB.

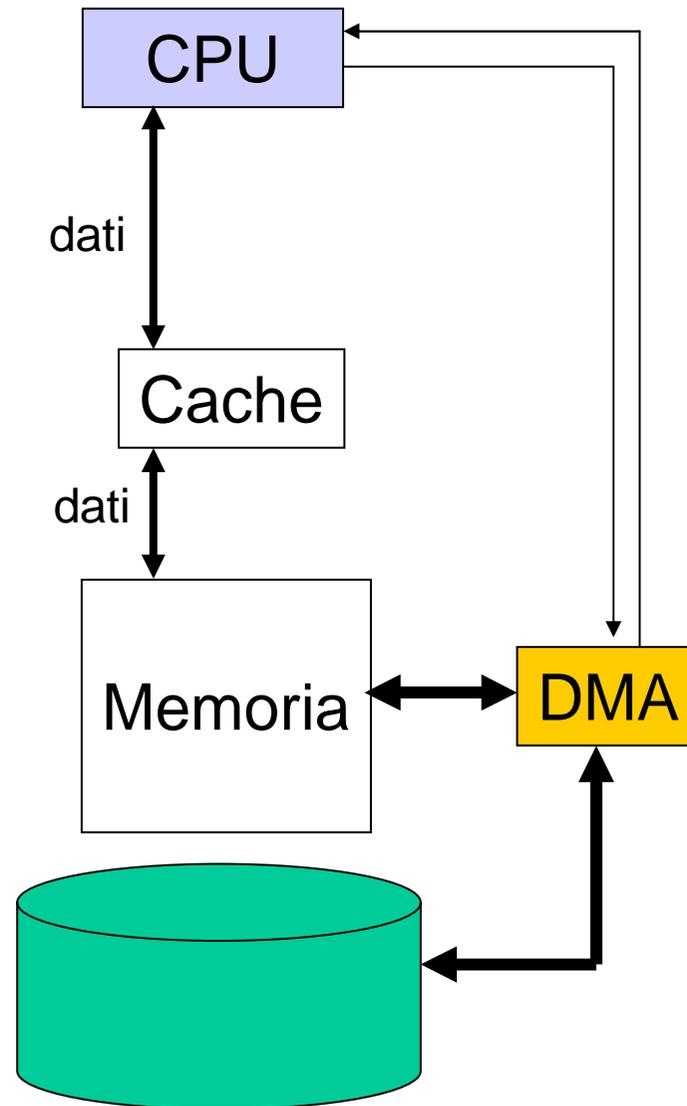
Il meccanismo di **accesso diretto alla memoria** (DMA) assolve a tale compito utilizzando ancora le interruzioni, ma solo per comunicare il completamento del trasferimento dati ed eventualmente il verificarsi di errori.

DMA: implementazione

Un controllore apposito trasferisce i dati tra un dispositivo di I/O e la memoria indipendentemente dal processore. Il **controllore di DMA** comanda le operazioni di lettura e scrittura tra se stesso e la memoria utilizzando i seguenti passi:

1. Il processore programma l'operazione di DMA fornendo l'identità del dispositivo, il tipo di operazione, l'indirizzo di memoria da cui iniziare a leggere o scrivere e il numero di byte da trasferire.
2. Il controllore di DMA fa partire l'operazione sul dispositivo senza impegnare la CPU.
3. A trasferimento effettuato il controllore segnala al processore con un'interruzione il completamento dell'operazione.

DMA e cache



DMA e cache

Nello schema precedente si vede come il controllore DMA non utilizza la gerarchia della cache, pertanto dati trasferiti tramite DMA tra la memoria principale e l'hard disk possono creare problemi di **coerenza** rispetto ai dati contenuti nella cache. Si hanno le seguenti possibili soluzioni:

- Far passare l'I/O attraverso la cache. Tecnica costosa che penalizza le prestazioni.
- In caso di lettura di I/O: invalidare in maniera selettiva i dati nella cache
- In caso di scrittura di I/O: forzare l'esecuzione del write-back – Scaricamento – Flush della cache. Effettuabile via software (SO) o hardware.

Riferimenti

Computer Organization and Design

The Hardware/Software Interface 3rd Edition

David A. Patterson, John L. Hennessy

Capitolo 8

Versione italiana:

Struttura e Progetto dei Calcolatori

L'Interfaccia Hardware-Software

2^a edizione Zanichelli

<http://en.wikipedia.org/> o <http://it.wikipedia.org/>