

Tecniche di input e output

Esistono diversi tipi di tecniche I/O :

- Bus di sistema
- Input/output - Periferiche
- Periferiche mappate in memoria
- Controllo di memoria
- Banda passante e latenza
- Input-Output gestito direttamente dal programma
- Input-Output con interruzioni
- Input-Output con DMA (Direct Memory Access)
- Periferiche in SPI

Definizione | L'I/O è l'insieme di architetture e dispositivi per il trasferimento di informazioni da e verso l'elaboratore.
Si diversificano tra loro per:

- Velocità di trasferimento
- Latenze
- Sincronizzazione
- Modalità di interazione

Bus di sistema

Definizione | il bus di sistema è un collegamento I/O con il processore.
È composto da:

- Bus di dati → trasferimento dati e istruzioni con dispositivi.
- Bus di controllo → trasporto info per definire le operazioni da compiere e sincronizzazione tra dispositivi.
- Bus degli indirizzi → CPU trasmette indirizzi di memoria o di periferica per la scrittura/lettura di memoria/periferiche.

OSS Tutte le unità dell'elaboratore sono connesse al bus.

Vantaggi ~ Flessibilità, semplicità, basso costo.

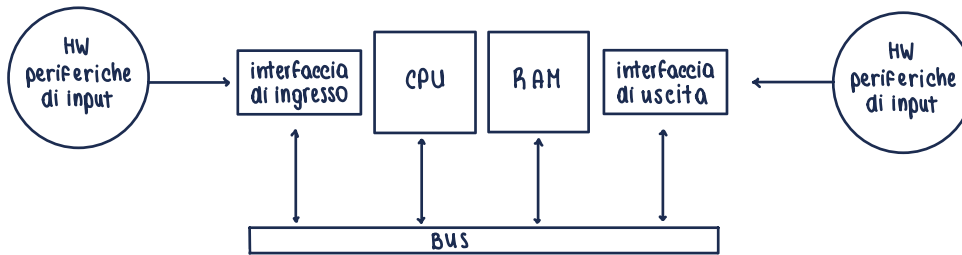
Svantaggi ~ gestione complessa del canale condiviso

Periferiche

Definizione | Le periferiche sono dispositivi per l'I/O di informazioni, collegati alla CPU tramite bus di sistema e/o interfacce.

Definizione | Le interfacce fanno da ponte tra CPU e periferiche.

Struttura → componente HW (controllore della periferica) + componente SW (driver → RAM)



Registri dell'interfaccia

- Registro di stato → indica lo stato della periferica letto dalla CPU
- Registro dei dati → rappresenta i dati di input/output (dipende dalla periferica).

Periferiche mappate in memoria

Definizione | tecnica nel quale ogni periferica ha uno spazio dedicato nella memoria identificato ciascuno da un indirizzo unico ed è riservato per la comunicazione.
I registri dell'interfaccia della periferica sono mappati in memoria ed è possibile l'accesso unicamente tramite SO (simile all'accesso in memoria tramite lw e sw).

Osservazione | In MIPS 32 le periferiche non sono mappate in memoria, in quanto compie solo 3 tipi di operazioni: logico-aritmetiche, Lw/Sw, salto.

Tecniche di gestione I/O

	No Interrupts no hardware	use of Interrupts si hardware
Transfer I/O → memory tramite processore	Programmed I/O	Interrupt-driven I/O
Transfer I/O → memory diretto		Direct Access Memory (DMA)

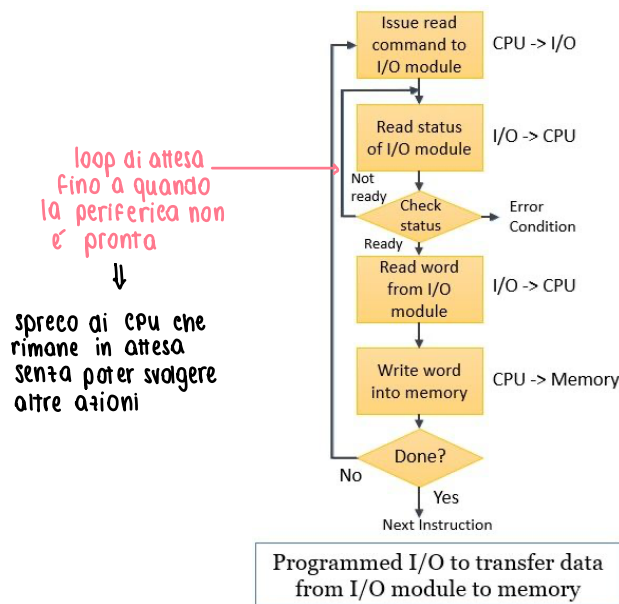
Programmed I/O

Caratteristiche

- I/O gestito dal controllo di programma
- CPU si occupa sia del controllo sia del trasferimento dati
 - ↳ si ferma finché la periferica non è pronta (reg. di stato - ready bit)

Vantaggio → risposta veloce al ready bit

Svantaggio → CPU bloccata in stato busy waiting → spreco di velocità



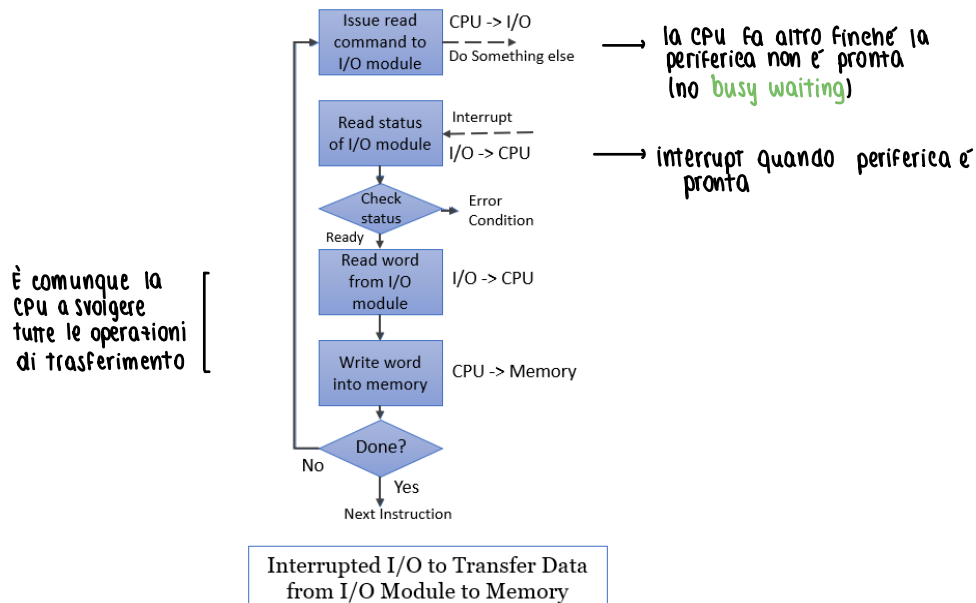
Interrupt-driven I/O

Caratteristiche

- La periferica manda un segnale **interrupt request** tramite bus di controllo alla CPU in caso di trasferimento.
- Il processore quando, in una fase di fetch, se ne accorgerà, informa la periferica con un segnale di **interrupt acknowledge** e interrompe l'esecuzione del programma corrente (ricordare eccezioni). Esegue la procedura di risposta all'interrupt al cui termine verrà ripresa l'esecuzione del programma interrotto.

Vantaggio → no CPU busy waiting

Svantaggio → CPU gestisce comunque operazioni di trasferimento

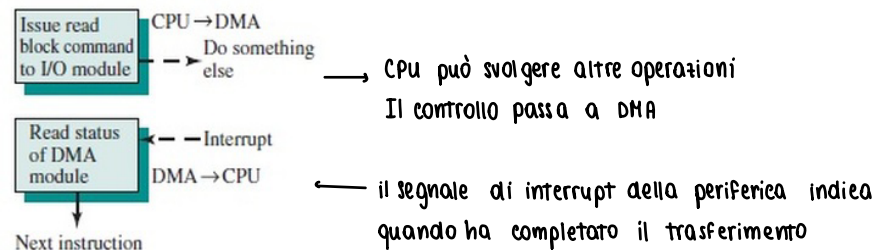


DMA (accesso diretto alla memoria)

Questo protocollo di trasferimento è stato introdotto per evitare l'intervento della CPU nella fase di trasferimento dei dati. Viene usato in trasferimenti di grandi quantità di dati.

Caratteristiche

- La periferica gestisce i trasferimenti; la CPU non interviene nel trasferimento dei dati.
- Vengono usati 2 registri in più per ogni periferica (mappati in memoria):
 - 1) un registro per l'indirizzo di memoria dal/sul quale trasferire i dati
 - 2) un registro per indicare la quantità dei dati da trasferire.
- Alla fine la periferica invia un interrupt alla CPU per segnalare il completamento del trasferimento.



OSS Dovendo CPU e DMA svolgere contemporaneamente azioni diverse, ci sono due modi per usare il bus:

- I Trasferimento dati quando CPU non usa il bus
- II DMA ruba cicli di clock alla CPU

Valutazione prestazioni delle gestioni I/O

Esistono misure che permettono di valutare l'efficienza della gestione delle operazioni I/O

Banda passante Quantità dei dati che si può trasferire per unità di tempo → $\frac{\text{byte}}{\text{secondo}}$

Latenza Tempo che intercorre tra l'istante in cui una periferica è pronta per il trasferimento e l'istante in cui il dato viene trasferito.



• Programmed I/O

Banda passante → tendenzialmente alta (dipende da periferica)
trasferimento veloce dei dati + poche istruzioni per gestione periferica
Latenza → minima (CPU busy waiting finché periferica non è pronta)

• Interrupt-driven I/O

Banda passante → minore del Programmed I/O
trasferimento dati più lungo
Latenza → maggiore del Programmed I/O (+ operazioni da eseguire)

- **DMA**

Banda passante → massima

Latenza → minima

CPU non fa nessuna istruzione per il Trasferimento