SSD

A differenza dell’hard disk tradizionale, un hard disk a stato solido è basato sulla tecnologia delle memorie flash. La velocità di accesso al disco è superiore rispetto a quella dell’hard disk ma il disco diventa inaffidabile dopo un certo numero di accessi.

Scheda video e scheda audio

Sono moduli hardware dedicati all’input-output dei segnali video e audio. Ciascuno di essi contiene una memoria interna e un processore dedicato, grazie ai quali le schede possono gestire i trasferimenti indipendentemente dalla CPU.

Parallelismo

Tassonomia di Flynn  
I computer con parallelismo a livello di processore si possono distinguere in base alla combinazione tra dati e istruzioni con cui funzionano. La distinzione prevede: computer Single Instruction Multiple Data che operano su diversi insiemi di dati svolgendo sempre la stessa operazione; computer Single Instruction Single Data, in cui una sola operazione viene eseguita su un solo set di dati; computer Multiple Instruction Single Data, mai realizzati; computer Multiple Instruction Multiple Data che corrispondono alle architetture multiprocessore più diffuse, in cui su diversi insiemi di dati vengono svolte più operazioni. I computer MIMD possono avere una memoria condivisa o una memoria dedicata; nel primo caso si parla di Symmetric Multi-processing, nel secondo di Multicomputer.  
  
Pipeline  
Il parallelismo si può implementare anche a livello di istruzione mediante la struttura di esecuzione a pipeline. La pipeline rende contemporanee le fasi di esecuzione di istruzioni diverse. Nel caso di salto condizionato le prestazioni della pipeline si riducono notevolmente perché le istruzioni che sono entrate nella pipeline prima dell’istruzione che genera il salto attraversano le prime fasi della pipeline inutilmente.  
Si possono gestire i salti in più modi: prelevando entrambe le destinazioni del salto e immettendole entrambe nel flusso di esecuzione, creando due flussi paralleli; il problema può essere la presenza di altre istruzioni di salto in uno o entrambi i rami. Un’altra soluzione è prelevare anticipatamente la destinazione del salto; in questo modo se il salto avviene la destinazione si trova già in cache o nei registri del processore, a seconda dell’implementazione. Alternativamente si possono mantenere in un buffer circolare tutte le ultime destinazioni dei salti; se l’istruzione genera un salto la destinazione viene prima cercata in questo registro. Oppure si può tentare di prevedere il comportamento dell’istruzione di salto in modo statico, ad esempio in base al tipo di istruzione, o dinamico, usando una tabella, in cui per ogni istruzione di salto viene memorizzato il suo comportamento recente, o un bit associato ad ogni istruzione di salto che segnala se l’ultima esecuzione ha provocato o meno un salto. In ultima alternativa è possibile ritardare l’esecuzione dell’istruzione di salto eseguendo prima le istruzioni successive che non dipendono dal risultato di essa.

Multiprocessori  
I multiprocessori possono avere configurazione master-slave o SMP. Nel primo caso il sistema operativo viene eseguito su un solo processore che fa da master e gestisce l’esecuzione dei processi sugli altri processori. I problemi derivano dalla dipendenza dell’intero sistema su un solo processore che se fallisce o rallenta causa il rallentamento o il blocco dell’intero sistema. Nel secondo caso ogni processore esegue una propria copia del sistema operativo e quindi gestisce l’esecuzione in modo indipendente. Il problema in questo caso sta nella sincronizzazione tra i processori, che non devono violare le aree di memoria non condivise con gli altri processori, non compromettere le strutture del sistema operativo comuni e non schedulare lo stesso processo già schedulato da un altro processore.