

# Cache

La velocità dei processori supera di molto la velocità di accesso alla memoria principale (decine di nanosecondi), perciò si introducono livelli intermedi di memoria che sono meno capienti e più costosi, ma anche più veloci (SRAM vicina al processore).

I processori moderni tipicamente hanno a disposizione tre livelli di cache; ciascun core ha una cache L1 (accesso in pochi cicli, idealmente 1) ad uso esclusivo, a cui si aggiungono una cache L2 che può essere condivisa tra gruppi di core e una cache L3 utilizzata da tutti.

È necessario gestire alcune problematiche legate alle cache:

- gestione dei miss: se il processore accede ad un indirizzo non in cache questo deve essere caricato dal livello inferiore (o dalla memoria principale per l'L3);
- le cache sono trasparenti ai programmi, che accedono alla memoria utilizzando gli indirizzi delle memoria principale; questi devono essere tradotti in indirizzi della cache;
- inserimento di dati in una cache piena: sostituire i dati che non verranno usati a breve (e.g. politica LRU);
- overhead: un miss ha un costo maggiore di un accesso diretto alla memoria in assenza di cache, che è vantaggiosa solo se si riesce a garantire un miss rate sotto il 10%;
- se core diversi operano sulla stessa area di memoria i contenuti delle cache potrebbero non rispecchiare lo stato effettivo della memoria – protocolli di cache coherence (e.g. snoopy bus).

Il processore interagisce con la memoria tramite la MMU, che fa passare tutti gli accessi (a memoria effettiva, non I/O) dalla cache L1: non c'è modo di bypassare le cache.

$$\text{AMAT} = t_1 + \text{MR}_1(t_2 + \text{MR}_2(t_3 + \text{MR}_3 t_M))$$