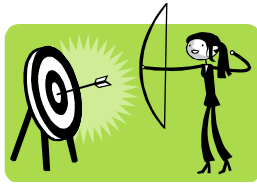


Gerarchie di memoria

Hit e miss



Un dato richiesto dalla CPU può essere presente in cache (**hit**) oppure mancante (**miss**)

- Un **hit**, *successo*, deve essere molto probabile (>90%) se si vuole guadagnare efficienza prestazionale
- Un **miss**, *fallimento*, richiede l'avvio di una procedura di scambio dati (**swap**) con il livello inferiore

Gerarchie di memoria

Tempo medio di accesso



T_a : Tempo medio di accesso ad un dato in memoria

$$T_a = T_h \times P_h + T_m \times (1 - P_h)$$

T_h = tempo di accesso ad un dato **presente** in cache

T_m = tempo medio di accesso ad un dato **non** in cache
(funzione della dimensione del blocco)

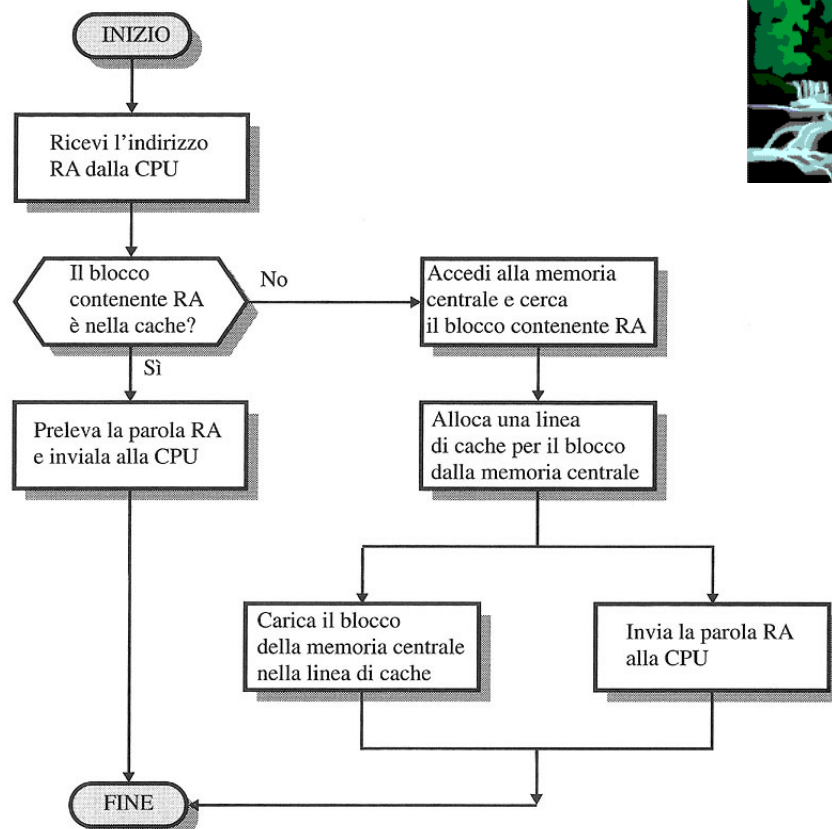
P_h = probabilità di **hit**

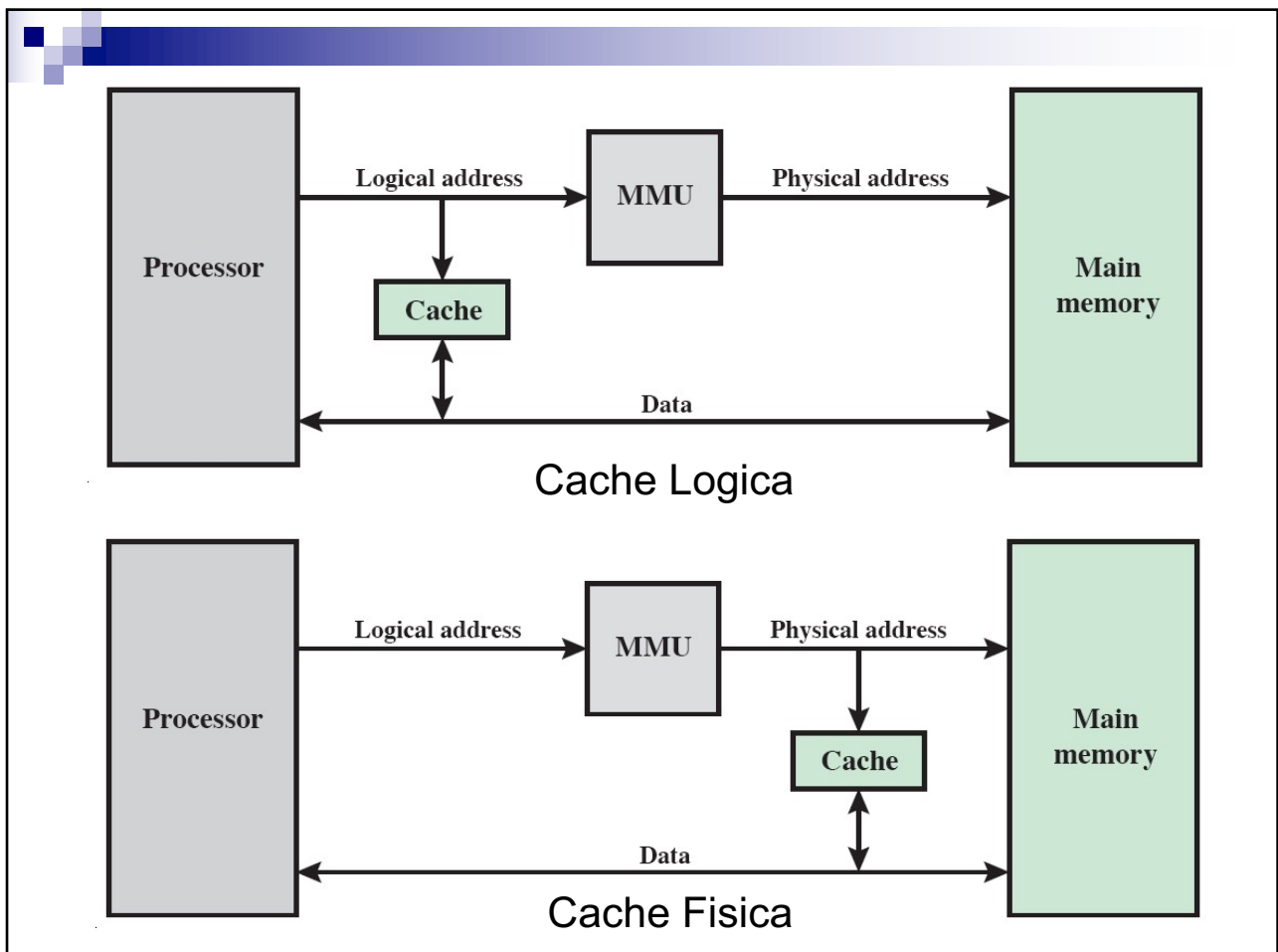
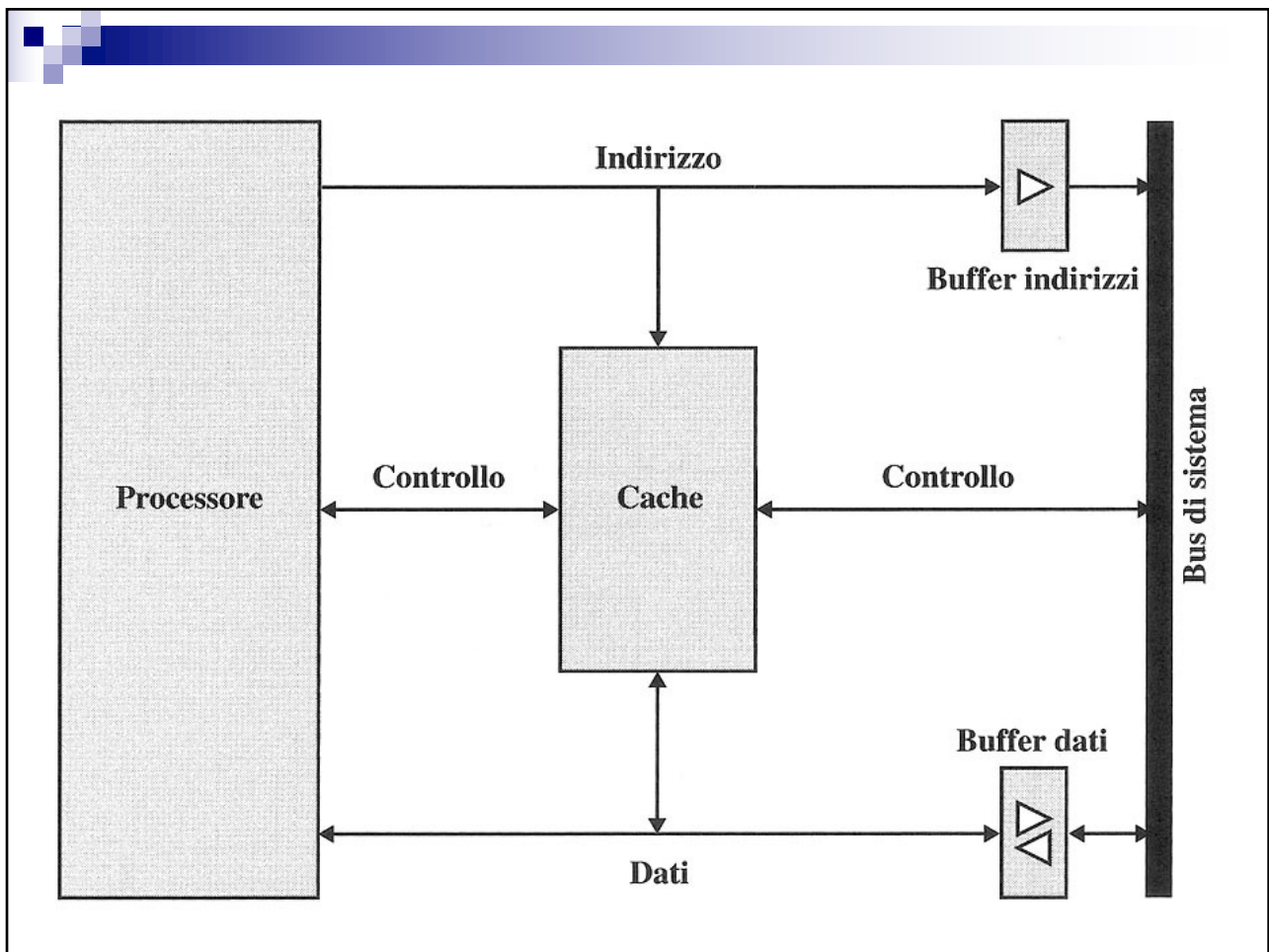
(funzione della dimensione del blocco e della politica di gestione)

Gerarchie di memoria

Tecnica generale

- Suddivisione della memoria centrale in blocchi **logici**
- Dimensionamento della cache in **multiplo** di blocchi
- Per ogni indirizzo emesso dalla CPU
 - **Hit** ⇒ Il dato richiesto viene fornito **immediatamente** alla CPU
 - **Miss** ⇒ La cache richiede il dato al livello inferiore
Il blocco contenente il dato viene posto in cache
Il dato richiesto viene fornito alla CPU





Gerarchie di memoria

Problematiche



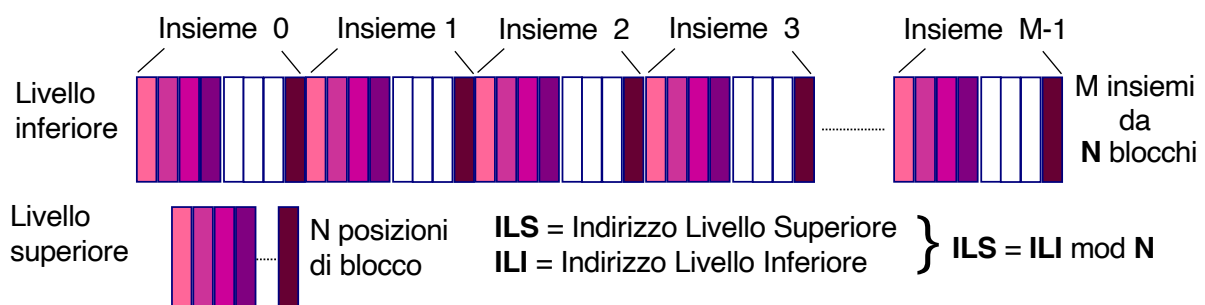
- Organizzazione della cache e tecniche di allocazione
- Individuazione di hit o miss
- Politica di rimpiazzo dei blocchi
- Congruenza dei blocchi

Gerarchie di memoria

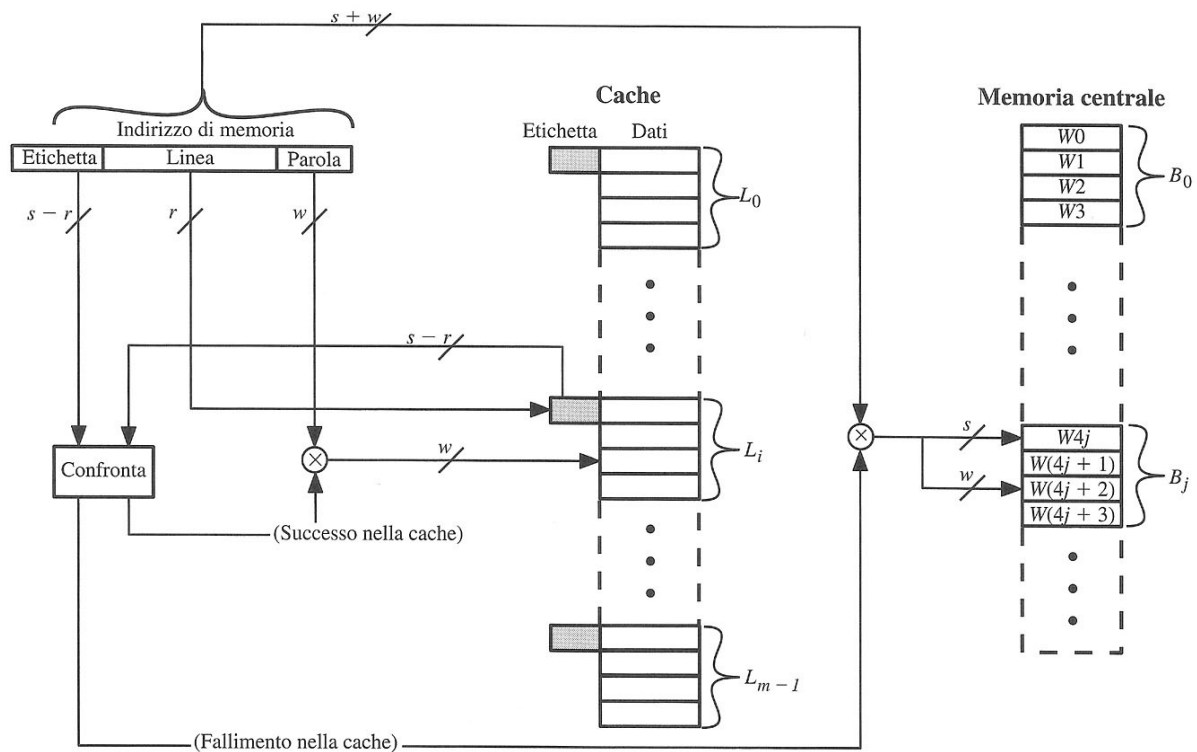
Associazione diretta

Tecnica nota come *direct mapping*

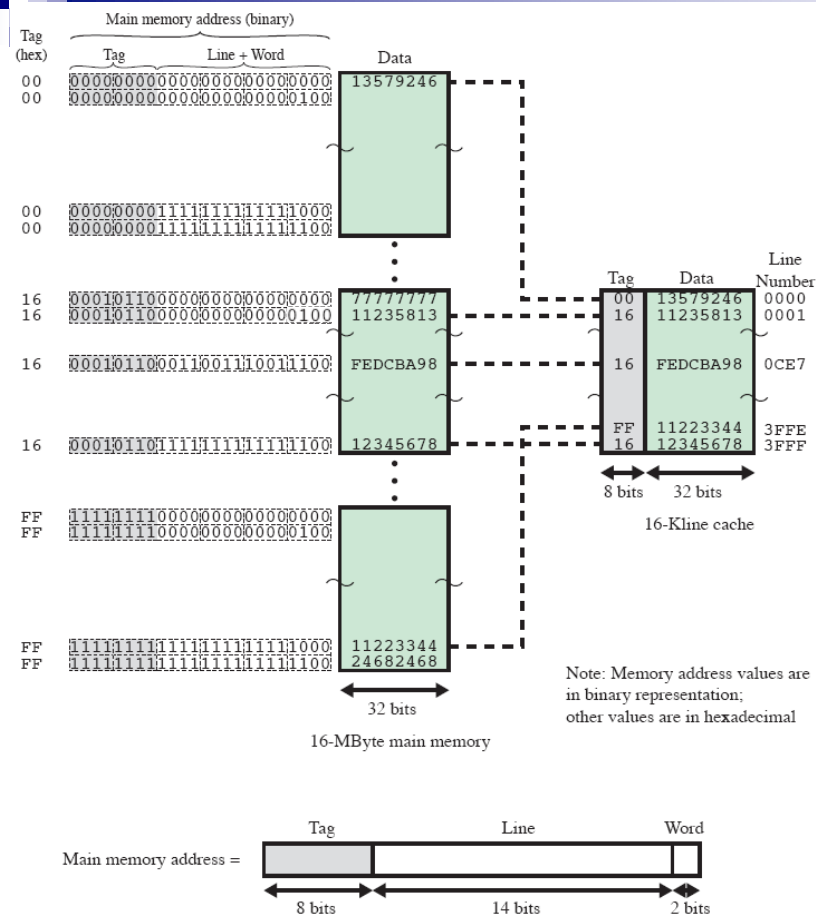
- Ogni blocco del livello inferiore può essere allocato *solo* in una specifica posizione (detta *linea* o *slot*) del livello superiore



Associazione diretta



Esempio di associazione diretta



Gerarchie di memoria

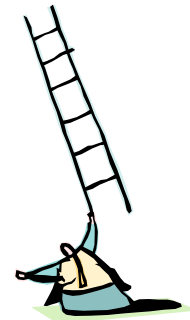
Associazione diretta

Vantaggi

- Semplicità di traduzione da indirizzo ILI (memoria) ad indirizzo ILS (cache)
- Determinazione veloce di hit o miss

Svantaggi

- Necessità di contraddistinguere il blocco presente in ILS (introduzione di un'etichetta, '**tag**')
- Swap frequenti per accesso a dati di blocchi adiacenti

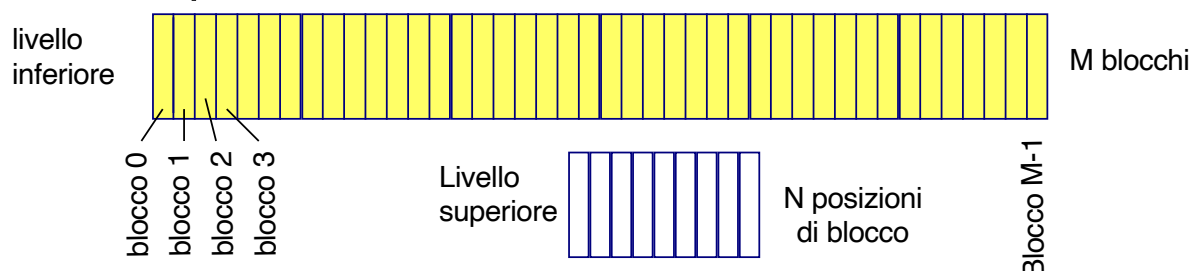


Gerarchie di memoria

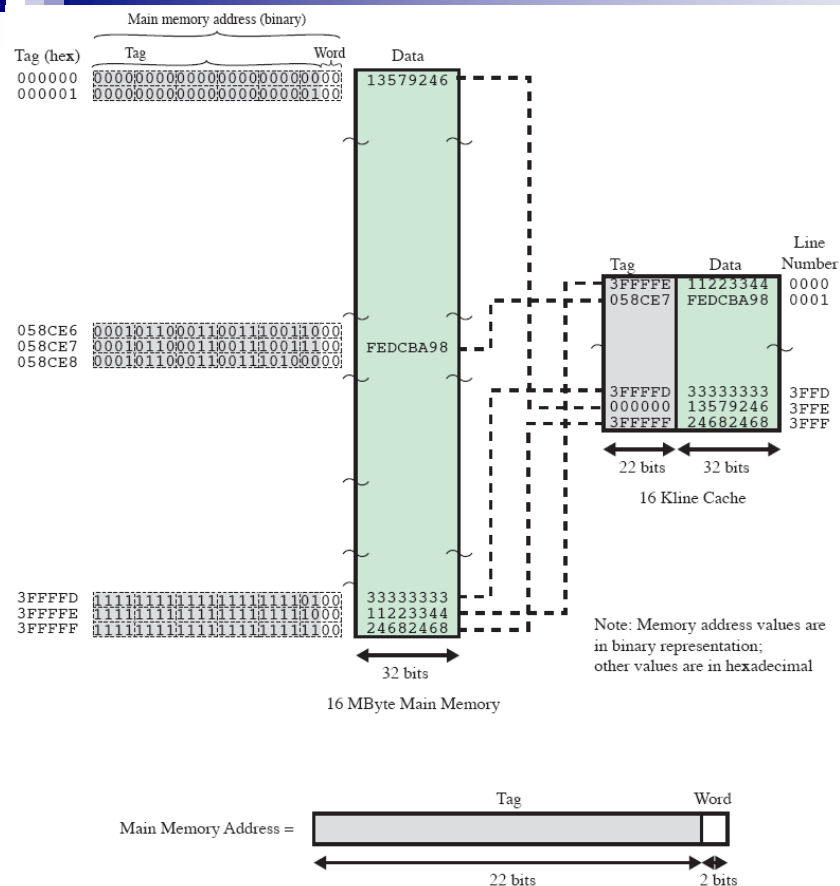
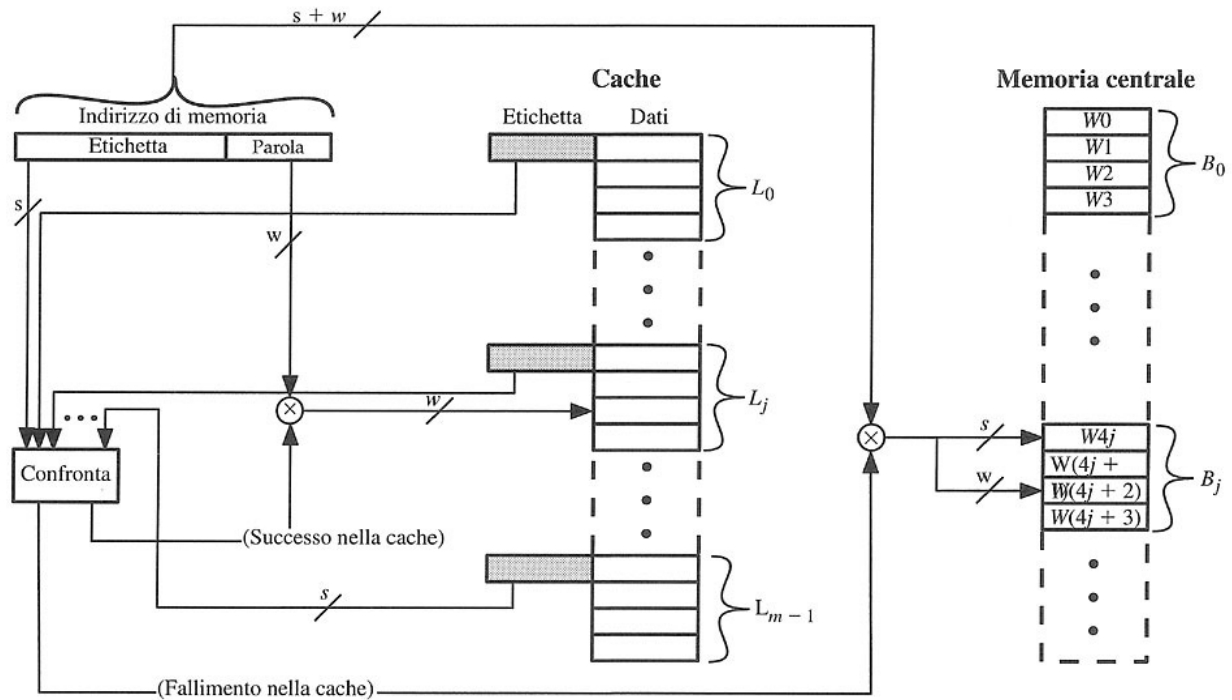
Associazione completa

Tecnica nota come *fully associative*

- Ogni blocco del livello inferiore può essere posto in *qualunque* posizione del livello superiore



Associazione completa



Esempio di associazione completa

Gerarchie di memoria

Associazione completa

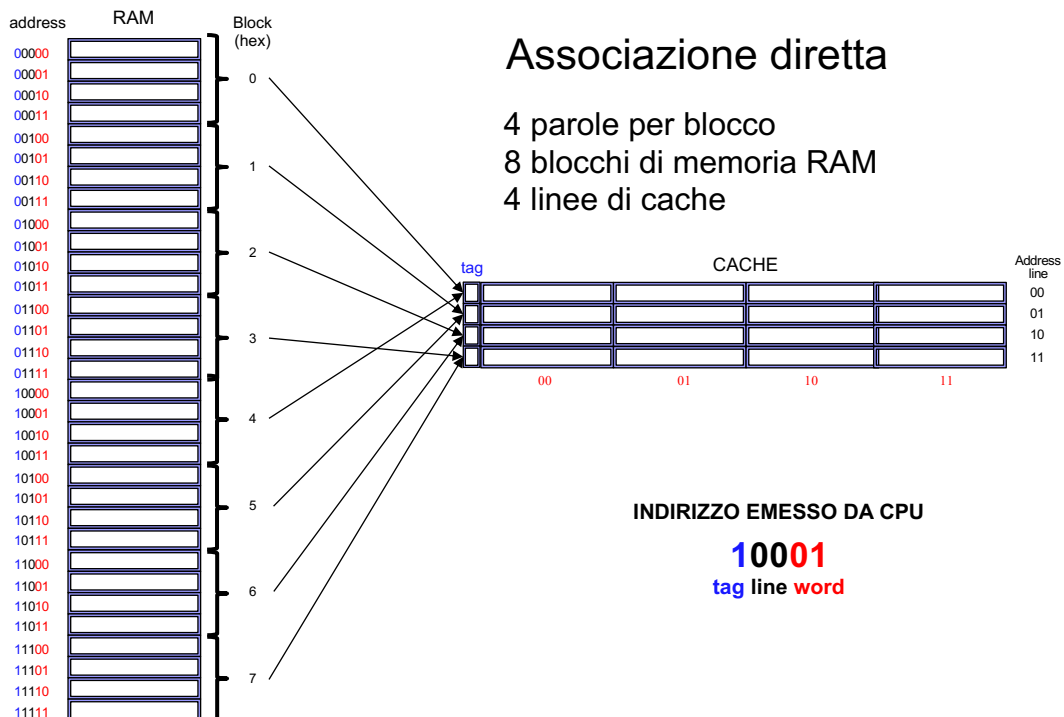
Alla cache capace di N blocchi viene associata una tabella di N posizioni, contenenti il numero di blocco effettivo (**tag**) in essa contenuto.

Vantaggi

- Massima efficienza di allocazione

Svantaggi

- Determinazione onerosa della corrispondenza ILS-ILI e della verifica di hit/miss

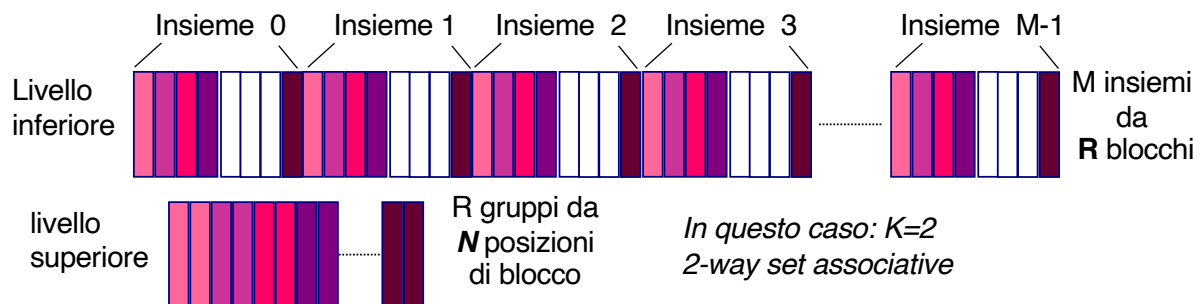


Gerarchie di memoria

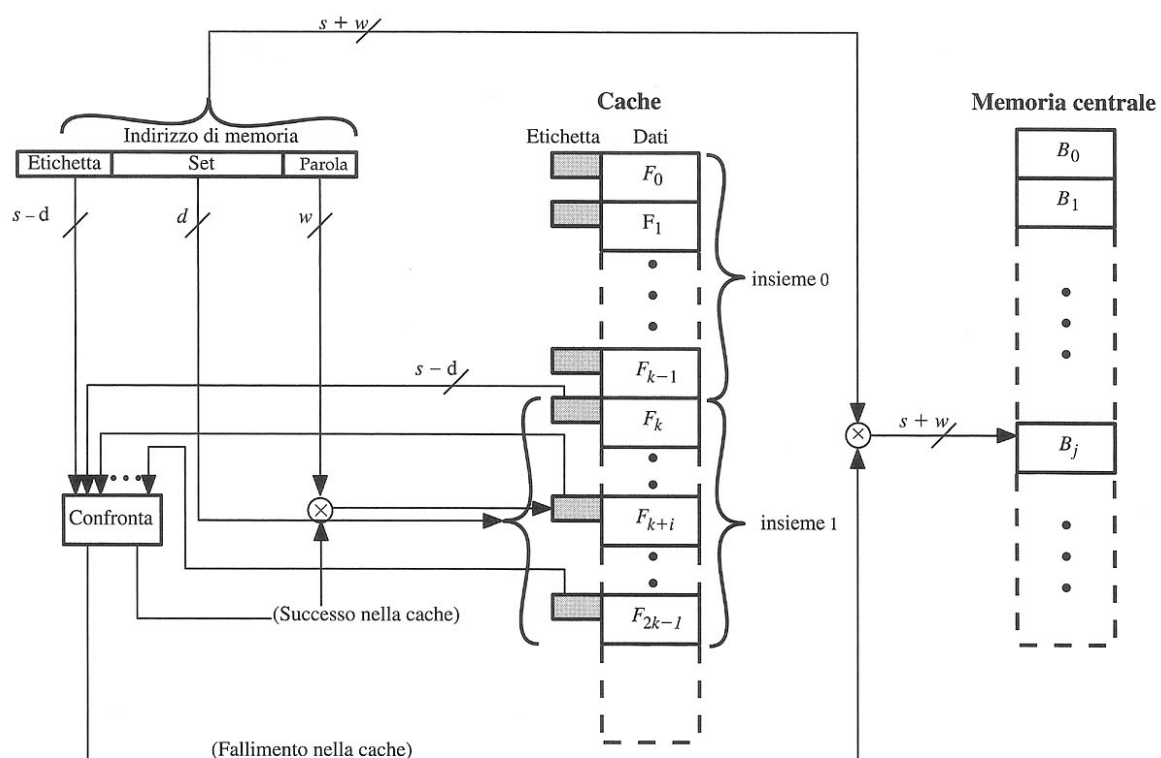
Associazione a gruppi

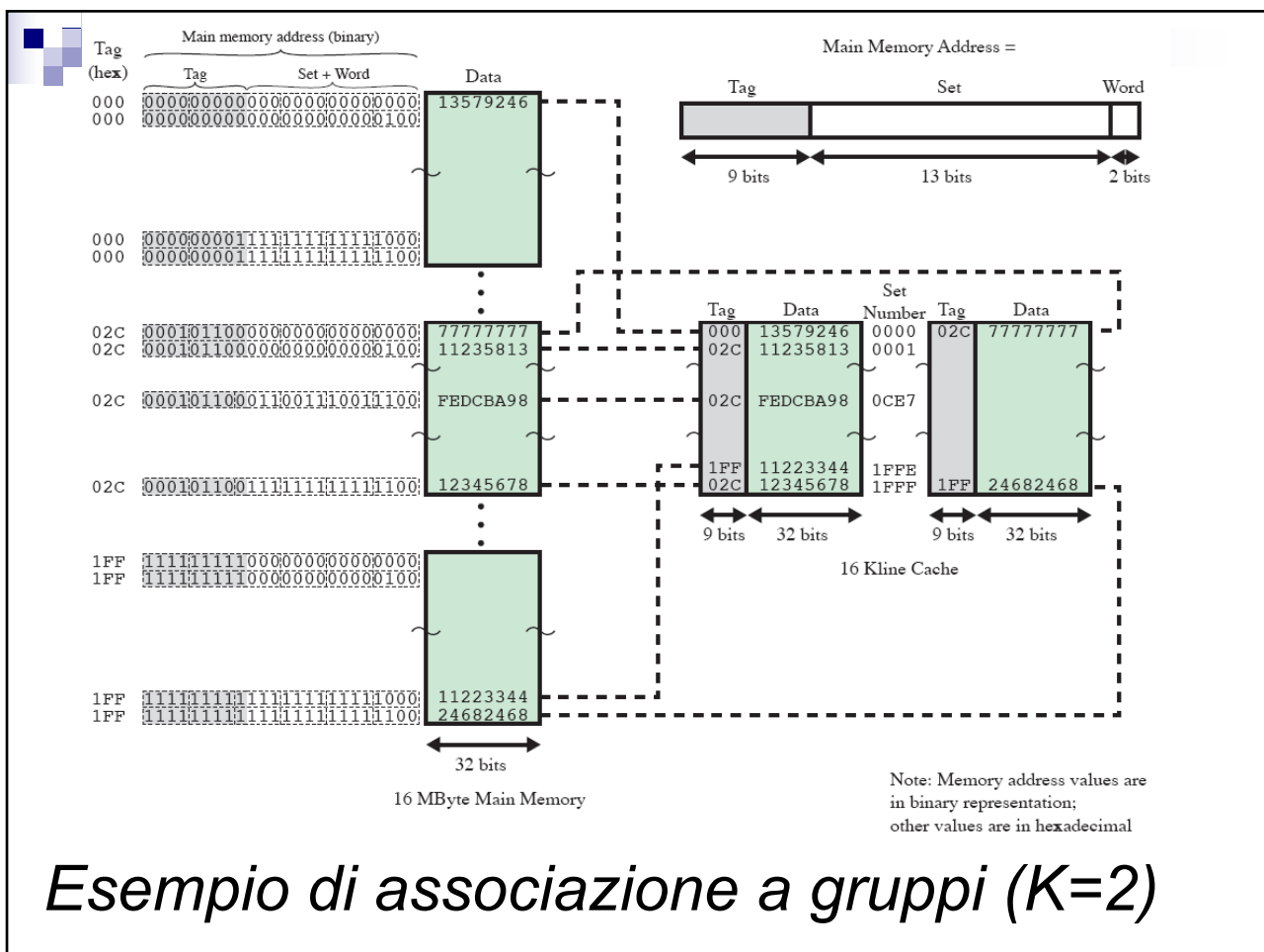
Tecnica nota come *K-way set associative*

- Ogni blocco di un certo insieme di blocchi del livello inferiore può essere allocato *liberamente* in uno *specifico* gruppo di blocchi del livello superiore



Associazione a gruppi





Gerarchie di memoria

Associazione a gruppi



- Alla cache, composta da R gruppi di K posizioni di blocco ciascuno, si affiancano R tabelle di K elementi, contenenti le etichette (**tag**) che designano i blocchi effettivi posti nelle posizioni corrispondenti
 - **Valutazione:** buona efficienza di allocazione a fronte di una sopportabile complessità di ricerca

