

# *Translation Lookaside Buffer*

Il ***translation lookaside buffer*** (in acronimo **TLB**) è una memoria tampone che l'MMU usa per velocizzare la traduzione degli indirizzi virtuali. Il TLB possiede un numero fisso di elementi della tabella delle pagine, la quale viene usata per mappare gli indirizzi virtuali in indirizzi fisici. La memoria virtuale è lo spazio visto da un processo che può essere più grande della memoria fisica

reale. Questo spazio è catalogato in pagine di dimensioni prefissate. Generalmente solo alcune pagine vengono caricate nella memoria fisica in zone dipendenti dalla politica di rimpiazzo pagine. La tabella delle pagine (generalmente caricata in memoria) tiene traccia di dove le pagine virtuali sono caricate nella memoria fisica. Il TLB è una cache della tabella delle pagine, cioè solamente un sottoinsieme del suo contenuto viene memorizzato.

## Funzionamento

---

La memoria tampone è tipicamente una memoria associativa (CAM), nel quale la

chiave di ricerca è l'indirizzo virtuale e il risultato della ricerca è un indirizzo reale o fisico (che spesso non sono la stessa cosa). Se la ricerca produce un risultato, il valore tradotto è presente nella CAM e il risultato viene utilizzato. Se non viene prodotto nessun risultato, la ricerca procede nella tabella delle pagine, che richiede molti più cicli per essere completata, specialmente quando la tabella è mantenuta in una memoria esterna.

## Varianti e ottimizzazioni

---

Gli elementi contenuti nel TLB fanno riferimento agli indirizzi di memoria

fisica. Il TLB può risiedere tra la CPU e la *cache* o tra la *cache* e la memoria primaria. Nel primo caso, la *cache* è indirizzata utilizzando l'indirizzamento fisico, nel secondo caso l'indirizzamento virtuale. Se la *cache* è indirizzata virtualmente, le richieste sono inviate direttamente dalla CPU alla *cache*, la quale accede al TLB se necessario. Se la *cache* è fisicamente indirizzata, la CPU effettua una ricerca nel TLB per ogni operazione in memoria, e il risultante indirizzo fisico è inviato alla *cache*.

Entrambe le soluzioni possiedono pregi e difetti. La *cache* indirizzata virtuale non richiede l'accesso al TLB ma deve essere, in ambiente multiprocessore,

invalidata ad ogni commutazione di contesto.

Un'ottimizzazione comune per la *cache* a indirizzamento fisico consiste nell'effettuare la ricerca nel TLB in parallelo con l'accesso alla *cache*. I bit meno significativi di ogni indirizzo virtuale (per esempio, in un sistema a memoria virtuale con pagine di 4 KB, i 12 bit meno significativi di un indirizzo virtuale) non cambiano nella traduzione dell'indirizzo virtuale-fisico. Per ogni accesso alla *cache*, vengono effettuate due operazioni: l'inserimento nella *cache* dei dati, e successivamente un confronto del *tag* della linea individuata nella *cache*. Se

la *cache* è strutturata in modo da poter essere indicizzata usando solo i bit che non cambiano durante la traduzione, può effettuare la sua operazione di indicizzazione mentre il TLB traduce i bit più significativi dell'indirizzo. A questo punto, l'indirizzo tradotto passa dal TLB alla *cache*, che confronta il *tag* per determinare se l'accesso è stato effettuato (*hit*) o è stato mancato (*miss*).

Nel secondo caso, possono essere utilizzati due differenti schemi nelle architetture di calcolatori moderne. Con la gestione *hardware* del TLB, la stessa CPU scorre le tabelle delle pagine per verificare se è presente una voce per il

rispettivo indirizzo virtuale di memoria. Se presente, la voce viene memorizzata all'interno del TLB e viene rieffettuato l'accesso al TLB (questo può diventare complicato a seconda di quante volte viene restituito un *miss* del TLB). Se non c'è nessun riscontro nella tabella delle pagine, occorre un'eccezione di *page fault*, e il sistema operativo preleva il dato richiesto dalla memoria virtuale e lo trasferisce nella cache. Con la gestione software del TLB, un *TLB miss* genera un'eccezione e il sistema operativo deve effettuare la ricerca nelle tabelle delle pagine e poi la traduzione via *software*. Il sistema operativo carica la traduzione nel TLB e riavvia il programma

dall'istruzione che aveva causato il *TLB miss*.

## Bibliografia

---

- (EN) John L. Hennessy e David A. Patterson, *Computer Architecture: A Quantitative Approach*, 5ª ed., Elsevier, 7 ottobre 2011, ISBN 9780123838735.

## Voci correlate

---

- Paginazione
- Memoria virtuale

### Controllo

di

autorità

GND (DE) 4620645-0 (<https://d-nb.info/gnd/4620645-0>).

 **Portale Informatica**: accedi alle voci di Wikipedia che trattano di informatica

Estratto da "[https://it.wikipedia.org/w/index.php?title=Translation\\_Lookaside\\_Buffer&oldid=113908041](https://it.wikipedia.org/w/index.php?title=Translation_Lookaside_Buffer&oldid=113908041)"

---

WIKIPEDIA

Questa pagina è stata modificata per l'ultima volta il 25 giu 2020 alle 15:44. •

Il contenuto è disponibile in base alla licenza [CC BY-SA 3.0](#) , se non diversamente specificato.