

Memorias cache asociativa por conjuntos

Algoritmo

Memorias cache asociativa por conjuntos

Algoritmo

Asociativa por conjuntos

- Hay más de un conjunto
- Cada conjunto tiene más de un bloque
- En que conjunto hay que buscar se determina por la dirección (como en CD).
- Dentro de cada conjunto se realiza una búsqueda secuencial (como en TA).

RAM: memoria RAM

CACHE: memoria cache

DIR: dirección de memoria que queremos consultar.

NB: número de bloques de cada conjunto

INICIO:

[etiqueta, índice, palabra] = SegmentarDirección (DIR)

PARA (i=0; i<NB; i++)

SI (CACHE.conjunto[índice].bloque[i].bit_validez == 0)

CACHE.conjunto[índice].bloque[i].bit_validez = 1

FALLO

break

SINO SI (CACHE.conjunto[índice].bloque[i].etiqueta == etiqueta)

ACIERTO

break

FINSI

FIN PARA

SI (i == NB) // Se ha llegado al final y no se ha encontrado

FALLO CON REEMPLAZO

FINSI

FIN

ACIERTO

Devolver `CACHE.conjunto[índice].bloque[i].datos[palabra]`

FALLO

Ir a la memoria RAM y traer el bloque completo

Copiar el bloque en `CACHE.conjunto[índice].bloque[i].datos`

Devolver `CACHE.conjunto[índice].bloque[i].datos[palabra]`

FALLO CON REEMPLAZO (con política de reemplazo LRU)

Ir a la memoria RAM y traer el bloque completo

Copiar el bloque en `CACHE.conjunto[índice].bloque[j].datos`

(j es el bloque que hace más tiempo que ha sido usado)

Devolver `CACHE.conjunto[índice].bloque[i].datos[palabra]`