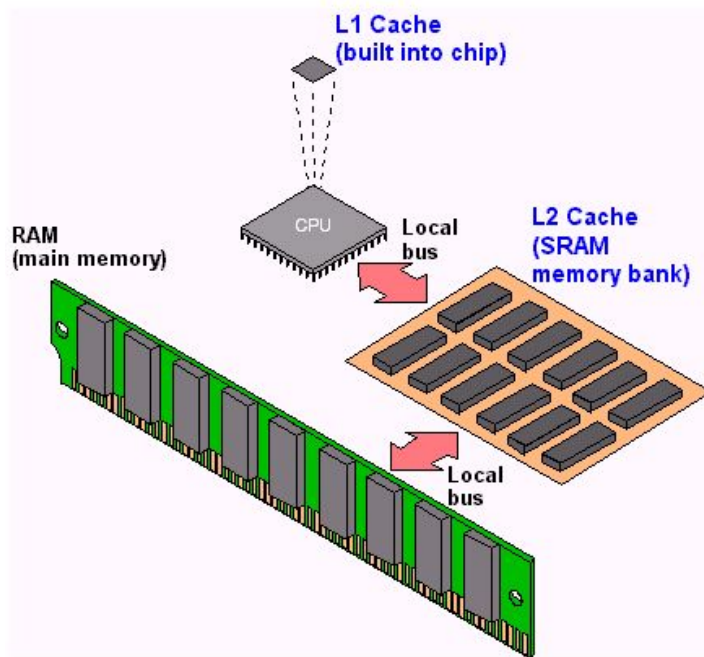


Cache Memory

Prinsip Cache Memory

From Computer Desktop Encyclopedia
© 1999 The Computer Language Co., Inc.



fungsi cache memory yaitu:

- ☐ Mempercepat Akses data pada komputer
- ☐ Meringankan kerja prosessor
- ☐ Menjembatani perbedaan kecepatan antara cpu dan memory utama.

lanjutan

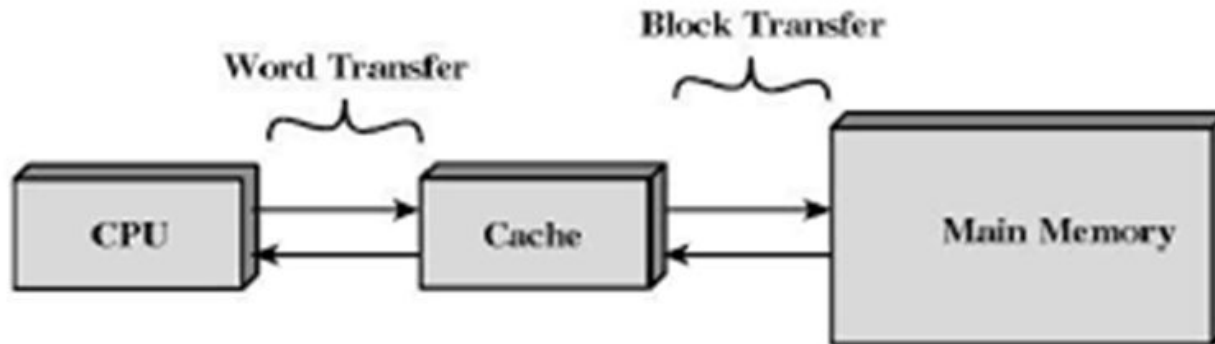


Figure 1.16 Cache and Main Memory
(p. 32)

Pemetaan (Mapping) Cache Memori

1. *Pemetaan Direct / langsung*
2. *Pemetaan asosiatif*
3. *Pemetaan asosiatif set*

Pemetaan Direct / langsung

Pemetaan langsung adalah teknik yang paling sederhana, yaitu teknik ini memetakan blok memori utama hanya ke sebuah saluran cache saja. Jika suatu block ada di cache, maka tempatnya sudah tertentu.

Keuntungan dari direct mapping adalah sederhana dan murah. Sedangkan kerugian dari direct mapping adalah suatu blok memiliki lokasi yang tetap (jika program mengakses 2 blok yang di map ke line yang sama secara berulang-ulang, maka cache-miss sangat tinggi)

Pada cara ini, *address* pada main memory dibagi 3 *field* atau bagian, yaitu:

1. *Word identifier* berisi informasi tentang lokasi word atau unit *addressable* lainnya dalam line tertentu pada cache.
2. *Line identifier* berisi informasi tentang nomor fisik (bukan logika) line pada chace
3. *Tag identifier* disimpan pada cache bersama dengan blok pada *line*.

Contoh :

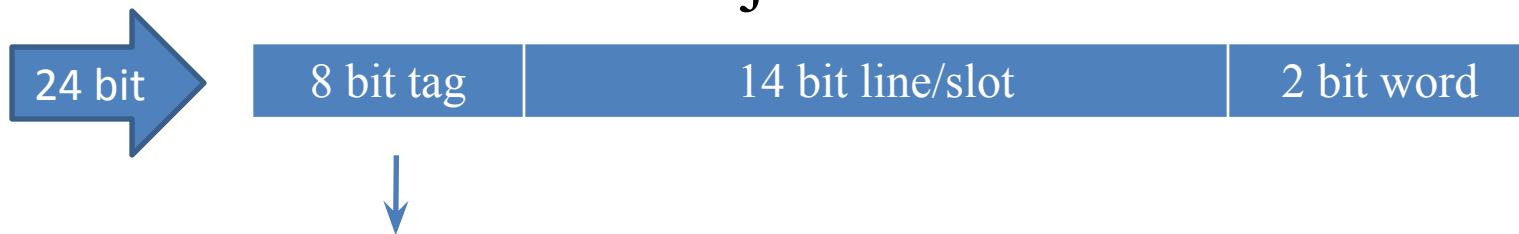
Suatu sistem komputer memiliki memori utama sebesar 16 Mbyte, memiliki cache sebesar 64 Kbyte. Transfer data antara memori utama dengan cache dalam ukuran blok besarnya 4 byte. Jika digunakan pemetaan langsung (*direct mapping*) tunjukkan masing-masing berapa bit untuk tag, slot/line dan word.

Penyelesaian:

- Memori utama : $16 \text{ MB} = 2^{20} + 2^4 = 2^{24}$ \square lebar alamat = 24 bit
- Size blok : 4 B = 2^2 \square lebar word = 2 bit
- Cache : $64 \text{ KB} = \frac{64 \text{ KB}}{4 \text{ B}} = 16 \text{ K}$

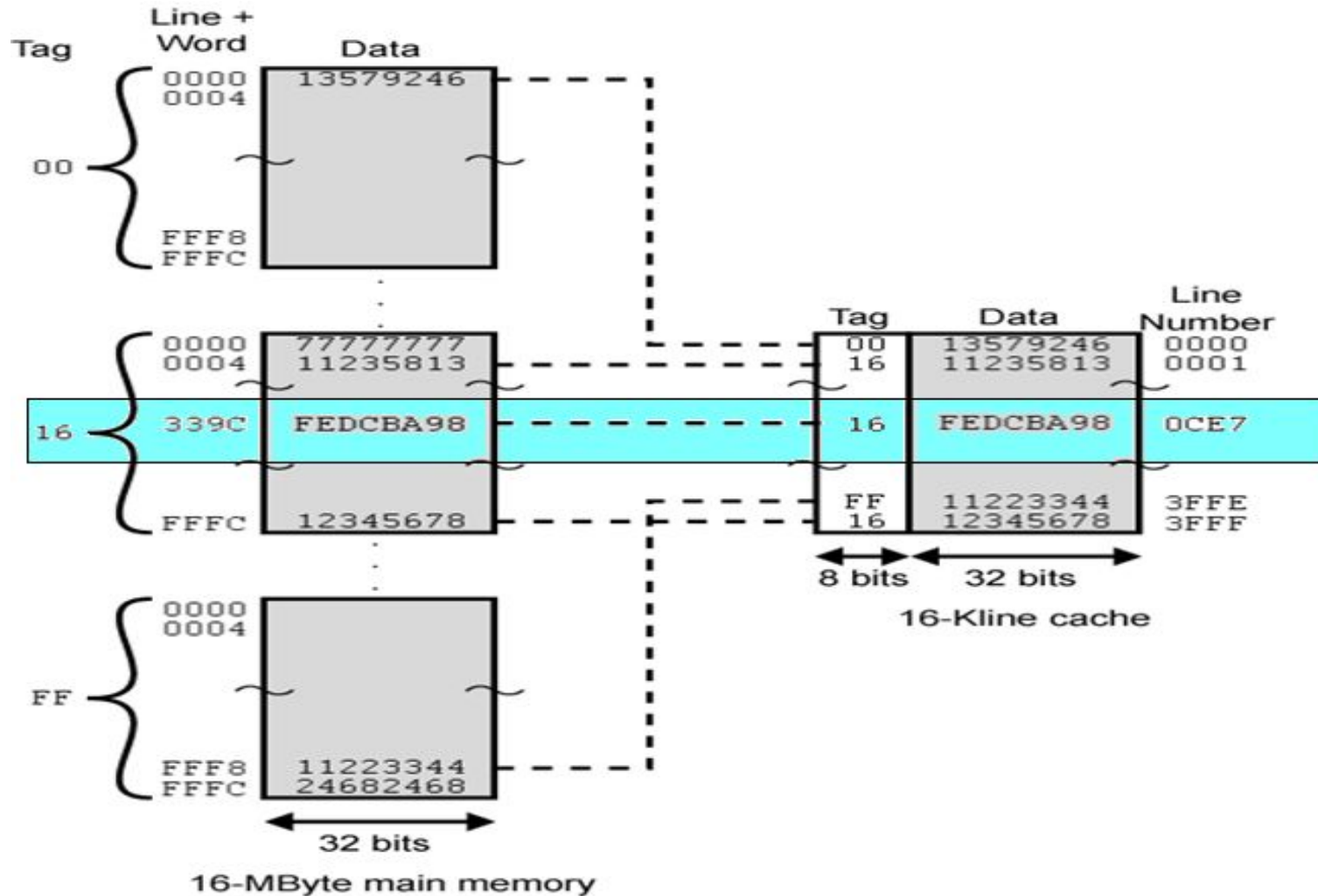
$$16 \text{ K} = 2^{10} + 2^4 = 2^{14} \quad \square \text{ lebar Line/slot} = 14 \text{ bit}$$

Jadi direct mapping untuk main memori 16MB, cache 64KB dan ukuran blok=4B menjadi

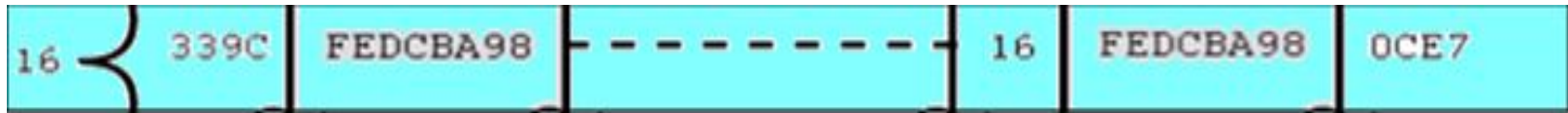


$$\begin{aligned} \text{Tag} &= 24 - 14 - 2 \\ &= 8 \text{ bit} \end{aligned}$$

2. Pemetaan langsung/ direct mapping :



	Tag	Line	Word
Main memory address =	8	14	2



Main memori

cache memori

Tag

data

data

16339C

Tag	data
	FE
	DC
	BA
	98

tag

0

1

2

3



0	1	2	3
16	FE	DC	BA
			98

Bagaimana Penyelesaiannya?????

Step Penyelesaian:

1. Cari pemetaan alamatnya (jika menggunakan direct mapping) cari: berapa tag, line dan word
2. Alamat hexa 16339C di konversi ke biner
3. Ambil LSB untuk bit word.
4. Sisa digit biner diambil sebesar line dan dihitung ulang.
5. Sisa digit dari word dan line adalah tag.

Penyelesaian :

8 bit tag	14 bit line/slot	2 bit word
-----------	------------------	------------

16339C dihexakan

1	6	3	3	9	c
<hr/>					
0001	0110	0011	0011	1001	1100

tag = 16 line = 0CE7 word = 0

Jadi masukan data 16339C dengan direct mapping dihasilkan alamat tag = 16, line/slot = 0CE7 dan word = 0

LATIHAN

1. Konversikan alamat memori dibawah ini
 - a. 256 MB =.....
 - b. 512 MB =
 - c. 512 KB =.....

2. Suatu sistem komputer memiliki memori utama sebesar 512 Mbyte, memiliki cache sebesar 512 Kbyte. Transfer data antara memori utama dengan cache dalam ukuran blok besarnya 32 byte. Jika digunakan pemetaan langsung (*direct mapping*) tunjukkan masing-masing berapa bit untuk tag, slot/line dan word.