

NAMA : VICTORYAN ARIFIN

KELAS : V

NIM :19104243

TUGAS : ARSITEKTUR KOMPUTER

1)1.1.Cache memory adalah memori sesaat yang memiliki kecepatan tinggi dengan ukuran yang kecil. Memori ini bertugas untuk menyimpan salinan perintah / instruksi yang selalu diakses oleh CPU. Tugas Cache memory adalah sebagai penghubung perbedaan kecepatan antara CPU dan Memori Utama. Dalam implementasinya jenis memori yang digunakan untuk cache adalah statik RAM (SRAM).

Level Cache Memory

Cache memori ada tiga Jenis Atau level yaitu L1,L2 dan L3.

Cache memori level 1 (L1) adalah cache memori yang terletak dalam prosesor (cache internal). Cache ini memiliki kecepatan akses paling tinggi dan harganya paling mahal. Ukuran memori berkembang mulai dari 8Kb, 64Kb dan 128Kb.Cache level 2 (L2) memiliki kapasitas yang lebih besar yaitu berkisar antara 256Kb sampai dengan 2Mb. Namun cache L2 ini memiliki kecepatan yang lebih rendah dari cache L1.

Cache L2 terletak terpisah dengan prosesor atau disebut dengan cache eksternal.

Cache level 3 hanya dimiliki oleh prosesor yang memiliki unit lebih dari satu misalnya dualcore dan quadcore. Fungsinya adalah untuk mengontrol data yang masuk dari cache L2 dari masing-masing inti prosesor.

L1 CACHE, L2 CACHE, L3 CACHE

L1 dan L2 Cache adalah memori sementara pada processor. Jadi ketika komputer dimatikan, maka ingatan yang ada pada processor pun akan hilang. L1 dan L2 mempunyai fungsi dan perbedaan, diantaranya adalah.

Fungsi Cache L1, L2, dan L3

Cache L1: Sejumlah kecil SRAM memori yang digunakan sebagai cache yang terintegrasi atau satu paket di dalam modul yang sama pada prosesor. L1 cache ini dikunci pada kecepatan yang sama pada prosesor. Berguna untuk menyimpan secara sementara instruksi dan data, dan memastikan bahwa prosesor memiliki supply data yang stabil untuk diproses sementara memori mengambil dan menyimpan data baru.

Cache L2: Fungsinya sama dengan L1 Cache, L2 Cache dikenal juga dengan nama secondary cache, adalah memory yang memiliki urutan kecepatan kedua (tipe memori yang paling cepat adalah L1 Cache) yang disediakan untuk mikroprosesor.

Cache L3: L3 cache memori khusus yang bekerja tangan-di-tangan dengan L1 dan L2 cache untuk meningkatkan kinerja komputer. L1, L2 dan L3 cache yang pemrosesan komputer unit (CPU) cache, ayat-ayat jenis lain dalam sistem cache seperti hard disk cache

Operasi Cache

Jika prosesor membutuhkan suatu data, pertama prosesor akan mencarinya pada cache. Jika data ditemukan, prosesor akan langsung membacanya dengan delay yang sangat kecil. Tetapi jika data yang dicari tidak ditemukan, prosesor akan mencarinya pada RAM yang kecepatannya lebih rendah.

Pada umumnya, cache dapat menyediakan data yang dibutuhkan oleh prosesor sehingga pengaruh kerja RAM yang lambat dapat dikurangi. Dengan cara ini maka memory bandwidth akan naik dan kerja prosesor menjadi lebih efisien. Selain itu kapasitas memori cache yang semakin besar juga akan meningkatkan kecepatan kerja komputer secara keseluruhan.

Dua jenis cache yang sering digunakan dalam dunia komputer adalah memory caching dan disk caching. Implementasinya dapat berupa sebuah bagian khusus dari memori utama komputer atau sebuah media penyimpanan data khusus yang berkecepatan tinggi

Karakteristik Cache Memory

Kapasitas relatif lebih kecil dari main memory, tetapi memiliki kecepatan yang relatif lebih tinggi dibanding main memory;

Cache memory merupakan suatu memori buffer (salinan data) bagi memori utama

Meskipun cache menggunakan informasi yang tersimpan dalam memori utama, tetapi ia tidak berhadapan secara langsung dengan memori utama

Word yang disimpan didalam cache memory adalah word yang diambil dari main memory, yang dikerjakan sesuai perintah CPU.

Fungsi dan Manfaat Cache Memory

Cache memory berfungsi sebagai tempat penyimpanan sementara untuk data atau instruksi yang diperlukan oleh processor. Secara gampangnya, cache berfungsi untuk mempercepat akses data pada komputer karena cache menyimpan data/informasi yang telah diakses oleh suatu buffer, sehingga meringankan kerja processor.

Manfaat lain dari cache memory adalah bahwa CPU tidak harus menggunakan sistem bus motherboard untuk mentransfer data. Setiap kali data harus melewati bus sistem, kecepatan transfer data memperlambat kemampuan motherboard. CPU dapat memproses data lebih cepat dengan menghindari hambatan yang diciptakan oleh sistem bus.

2.1 Mapping (Pemetaan)

Saluran cache lebih sedikit dibandingkan dengan blok memori utama sehingga diperlukan algoritma untuk pemetaan blok-blok memori utama ke dalam saluran cache. Selain itu, diperlukan juga alat untuk menentukan blok memori utama mana yang sedang memakai saluran cache. Pemilihan fungsi pemetaan akan menentukan bentuk organisasi cache. Terdapat tiga metode yang digunakan yaitu :

1. Pemetaan Langsung (Direct Mapping)

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).

Berikut penjelasan lebih detail :

§ Setiap blok pada main memory dipetakan dengan line tertentu pada cache. $i = j \text{ modulo } C$ di mana i adalah nomor line pada cache yang digunakan untuk meletakkan blok main memory ke- j .

§ Jika $M = 64$ dan $C = 4$, maka pemetaan antara line dengan blok menjadi seperti berikut :

Line 0 can hold blocks 0, 4, 8, 12, ...

Line 1 can hold blocks 1, 5, 9, 13, ...

Line 2 can hold blocks 2, 6, 10, 14, ...

Line 3 can hold blocks 3, 7, 11, 15, ...

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

- o Tag identifier.
- o Line number identifier
- o Word identifier (offset)

§ Word identifier berisi informasi tentang lokasi word atau unit addressable lainnya dalam line tertentu pada cache.

§ Line identifier berisi informasi tentang nomor fisik (bukan logika) line pada chace

§ Tag identifier disimpan pada cache bersama dengan blok pada line.

o Untuk setiap alamat memory yang dibuat oleh CPU, line tertentu yang menyimpan copy alamat tsb ditentukan, jika blok tempat lokasi data tersebut sudah dikopi dari main memory ke cache.

o Tag yang ada pada line akan dicek untuk melihat apakah benar blok yang dimaksud ada line tsb.

Organisasi Direct Mapping di cache memory

Gambar 2.1 : Gambar Organisasi Direct Mapping.

Keuntungan Menggunakan Direct Mapping antara lain :

§ Mudah dan Murah diimplementasikan

§ Mudah untuk menentukan letak salinan data main memory pada chace.

Kerugian menggunakan Direct Mapping antara lain :

§ Setiap blok main memory hanya dipetakan pada 1 line saja.

§ Terkait dengan sifat lokal pada main memory, sangat mungkin mengakses blok yang dipetakan pada line yang sama pada cache. Blok seperti ini akan menyebabkan seringnya sapu masuk dan keluar data ke/dari cache, sehingga hit ratio mengecil. Hit ratio adalah perbandingan antara jumlah ditemukannya data pada cache dengan jumlah usaha mengakses cache.

Ringkasan direct mapping nampak pada tabel berikut:

Item

Keterangan

Panjang alamat

$(s+w)$ bits

Jumlah unit yang dapat dialamati

$2^s + w$ words or bytes

Ukuran Bloks sama dengan ukuran Line

2^w words or bytes

Jumlah blok memori utama

$2^s + w/2^w = 2^s$

Jumlah line di chace

$M = 2^r$

Besarnya tag

$(s - r)$ bits

2. Pemetaan Asosiatif (Associative Mapping)

Pemetaan asosiatif mengatasi kekurangan pemetaan langsung dengan cara mengizinkan setiap blok memori utama untuk dimuatkan ke sembarang saluran cache. Dengan pemetaan asosiatif, terdapat fleksibilitas penggantian blok ketika blok baru dibaca ke dalam cache. Kekurangan pemetaan asosiatif yang utama adalah kompleksitas rangkaian yang diperlukan untuk menguji tag seluruh saluran cache secara parallel, sehingga pencarian data di cache menjadi lama.

§ Memungkinkan blok diletakkan di sebarang line yang sedang tidak terpakai.

§ Diharapkan akan mengatasi kelemahan utama Direct Mapping.

§ Harus menguji setiap cache untuk menemukan blok yang diinginkan.

o Mengecek setiap tag pada line

o Sangat lambat untuk cache berukuran besar.

§ Nomor line menjadi tidak berarti. Address main memory dibagi menjadi 2 field saja, yaitu tag dan word offset.

§ Melakukan pencarian ke semua tag untuk menemukan blok.

§ Cache dibagi menjadi 2 bagian :

- o lines dalam SRAM
- o tag dalam associative memory

Keuntungan Associative Mapping : Cepat dan fleksibel.

Kerugian Associative Mapping : Biaya Implementasi, misalnya untuk cache ukuran 8 kbyte dibutuhkan 1024 x 17 bit associative memory untuk menyimpan tag identifier.

Ringkasan Associative Mapping nampak pada tabel berikut:

Item

Keterangan

Panjang alamat

$(s+w)$ bits

Jumlah unit yang dapat dialamati

2^{s+w} words or bytes

Ukuran Bloks sama dengan ukuran Line

2^w words or bytes

Jumlah blok memori utama

$2^{s+w} / 2^w = 2^s$

Jumlah line di chace

Undetermined

Besarnya tag

s bits

3. Pemetaan Asosiatif Set (Set Associative Mapping)

Pada pemetaan ini, cache dibagi dalam sejumlah sets. Setiap set berisi sejumlah line. Pemetaan asosiatif set memanfaatkan kelebihan-kelebihan pendekatan pemetaan langsung dan pemetaan asosiatif.

§ Merupakan kompromi antara Direct dengan Full Associative Mapping.

§ Membagi cache menjadi sejumlah set (v) yang masing-masing memiliki sejumlah line (k)

§ Setiap blok dapat diletakkan di sebarang line dengan nomor set: nomor set = j modulo v

Organisasi K-Way Set Associative Mapping

Gambar 2.5 : Gambar Organisasi K-Way Set Associative Mapping.

§ Jika sebuah set dapat menampung X line, maka cache disebut memiliki X way set associative cache.

§ Hampir semua cache yang digunakan saat ini menggunakan organisasi 2 atau 4-way set associative mapping.

Keuntungan menggunakan Set Associative Mapping antara lain:

Setiap blok memori dapat menempati lebih dari satu kemungkinan nomor line

(dapat menggunakan line yang kosong), sehingga thrashing dapat diperkecil

Jumlah tag lebih sedikit (dibanding model associative), sehingga jalur untuk melakukan perbandingan tag lebih sederhana.

Ringkasan Set Associative Mapping nampak pada tabel berikut:

Item

Keterangan

Panjang alamat

$(s+w)$ bits

Jumlah unit yang dapat dialamati

2^{s+w} words or bytes

Ukuran Bloks sama dengan ukuran Line

2^w words or bytes

Jumlah blok memori utama

2^d

Jumlah line dalam set

k

Jumlah set

$V=2^d$

Jumlah line di chace

$K_v = k \cdot 2^d$

Besarnya tag

$(s - d)$ bits