KIẾN TRÚC MÁY TÍNH

Chương 4

Bộ nhớ Cache

Chương 4. Bộ nhớ cache

4.1 Tổng quan về bộ nhớ máy tính
Phân loại bộ nhớ máy tính

Vị trí	**Hiệu suất**
Bên trong (vd: thanh ghi, cache, bộ nhớ chính	Thời gian truy cập
Bên ngoài (vd: đĩa quang, đĩa từ, băng từ)	Chu kỳ xung nhịp
Dung lượng	Tốc độ truyền tải
Số lượng từ	**Loại vật lý**
Số lượng byte	Bán dẫn
Đơn vị truyền	Từ
Từ	Quang học
Khối	Quang từ
Phương pháp truy cập	**Tính chất vật lý**
Tuần tự	Điện động/điện tĩnh (Dữ liệu có bị mất khi mất điện)
Trực tiếp	Có thể xóa/không xóa được
Ngẫu nhiên	**Tổ chức**
Kết hợp	Module bộ nhớ

Phân loại bộ nhớ

a. **Vị trí**
- Bộ nhớ có thể ở trong và ngoài máy tính
- Bộ nhớ chính là bộ nhớ trong
- Bộ xử lý cần có bộ nhớ cục bộ riêng của nó: thanh ghi
- Cache là một dạng khác của bộ nhớ trong
- Bộ nhớ ngoài bao gồm các thiết bị lưu trữ ngoại vi có thể truy cập vào bộ xử lý thông qua bộ điều khiển I/O

b. **Dung lượng**
- Bộ nhớ thường được biểu diễn dưới dạng byte

c. **Đơn vị truyền**
- Đối với bộ nhớ trong, đơn vị truyền bằng số lượng đường điện đi vào và ra khỏi module bộ nhớ

Phân loại bộ nhớ (tiếp)

d. Phương pháp truy cập các khối dữ liệu

Truy cập tuần tự	Truy cập trực tiếp	Truy cập ngẫu nhiên	Kết hợp
• Bộ nhớ được tổ chức thành các đơn vị dữ liệu được gọi là bản ghi (record) • Truy cập được thực hiện tuần tự • Thời gian truy cập biến đổi	• Có một cơ chế đọc-ghi chia sẻ • Mỗi khối hoặc bản ghi có một địa chỉ duy nhất dựa trên vị trí vật lý • Thời gian truy cập biến đổi	• Mỗi vị trí trong bộ nhớ có một cơ chế định địa chỉ riêng • Thời gian truy cập vào một vị trí nhất định không đổi và phụ thuộc vào chuỗi các truy cập trước đó • Một vị trí bất kỳ có thể được chọn ngẫu nhiên, định địa chỉ và truy cập trực tiếp • Bộ nhớ chính và một số bộ nhớ cache là truy cập ngẫu nhiên	• Một word được truy xuất dựa trên một phần nội dung thay vì địa chỉ của nó • Mỗi vị trí có cơ chế định địa chỉ riêng. Thời gian truy xuất là không đổi, phụ thuộc vào vị trí hoặc các truy cập trước đó • Bộ nhớ Cache có thể sử dụng truy cập kết hợp

e. Hiệu năng

Hai đặc điểm quan trọng nhất của bộ nhớ: dung lượng và hiệu năng

Ba tham số hiệu năng được sử dụng:

Thời gian truy cập (độ trễ)
- Đối với bộ nhớ truy cập ngẫu nhiên, nó là thời gian cần để thực hiện 1 thao tác đọc hoặc ghi
- Đối với bộ nhớ truy cập không ngẫu nhiên, nó là thời gian cần để đặt cơ chế đọc-ghi vào vị trí mong muốn

Chu kỳ bộ nhớ
- Thời gian truy cập cộng với thời gian cần trước khi truy cập thứ hai có thể bắt đầu
- Có thể cần thêm thời gian để các transients chết trên đường tín hiệu hoặc để khôi phục lại dữ liệu bị hỏng
- Liên quan đến hệ thống bus, không liên quan bộ xử lý

Tốc độ truyền tải
- Tốc độ truyền dữ liệu vào hoặc ra khỏi bộ nhớ
- Đối với bộ nhớ truy cập ngẫu nhiên, tốc độ truyền tải bằng 1/(chu kỳ)

f. Vật lý

- Các dạng phổ biến nhất là: Bộ nhớ bán dẫn, Bộ nhớ bề mặt từ, Bộ nhớ quang, Bộ nhớ quang từ
- Một số đặc điểm vật lý quan trọng của lưu trữ dữ liệu:
  - Bộ nhớ điện động (Volatile memory)
    - Thông tin bị suy yếu hoặc bị mất khi nguồn điện tắt
  - Bộ nhớ điện tĩnh (Non-volatile memory)
    - Thông tin một khi đã được ghi thì sẽ không bị hư hỏng cho đến khi được cố tình thay đổi
    - Không cần cấp điện để giữ lại thông tin
  - Bộ nhớ bề mặt từ (Magnetic-surface memories)
    - Bộ nhớ điện tĩnh
  - Bộ nhớ bán dẫn (Semiconductor memory)
    - Bộ nhớ điện động hoặc điện tĩnh
  - Bộ nhớ không xoá được (Nonerasable memory)
    - Không thể thay đổi, trừ khi phá hủy các khối lưu trữ
    - Bộ nhớ chỉ đọc (ROM) là bộ nhớ bán dẫn thuộc loại này
- - Với bộ nhớ truy cập ngẫu nhiên, vấn đề quan trọng khi thiết kế là tổ chức hay sự sắp xếp vật lý của các bit để tạo thành các từ word

g. Tổ chức bộ nhớ: mô hình phân cấp bộ nhớ

- Thiết kế bộ nhớ của máy tính cần trả lời ba câu hỏi:
  - How much? How fast? How expensive?
- Cần có sự cân đối giữa dung lượng, thời gian truy cập và chi phí
  - Thời gian truy cập nhanh hơn, chi phí lớn hơn cho mỗi bit
  - Dung lượng lớn hơn, chi phí nhỏ hơn cho mỗi bit
  - Dung lượng lớn hơn, thời gian truy cập chậm hơn
- Giải pháp cho tình trạng khó xử khi thiết kế bộ nhớ:
  - Không dựa hoàn toàn vào một thành phần hoặc công nghệ bộ nhớ
  - Sử dụng một hệ thống phân cấp bộ nhớ

Bộ nhớ phân cấp - Sơ đồ

- **Chi phí trên bit giảm**
- **Dung lượng tăng**
- **Thời gian truy cập tăng**
- **Tần suất truy cập bộ nhớ của VXL giảm**

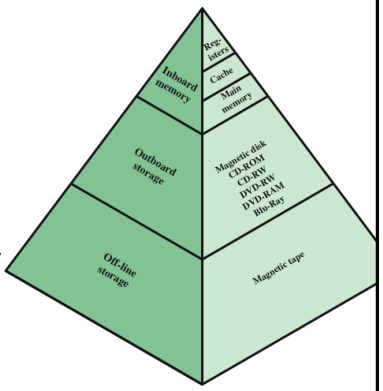
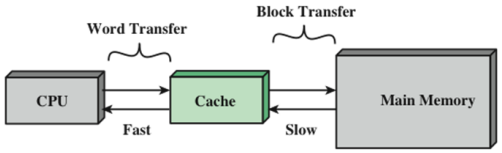


Figure 4.1 The Memory Hierarchy

4.2. Nguyên lý bộ nhớ cache

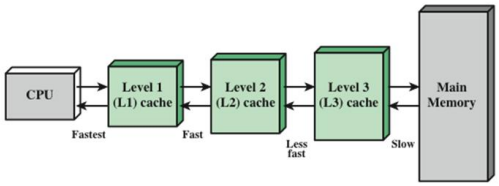
Bộ nhớ cache và bộ nhớ chính

- Cache đơn



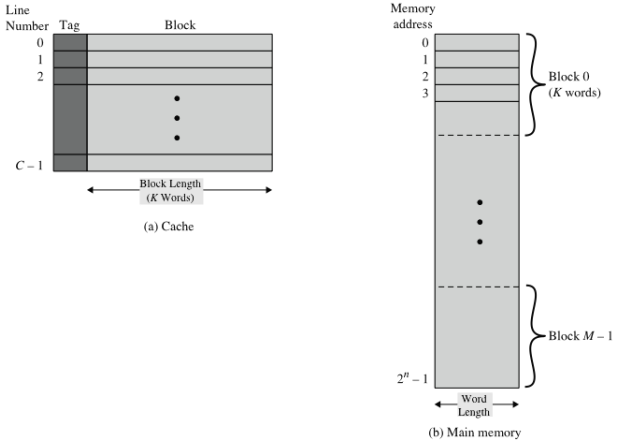
(a) Single cache

- Tổ chức cache 3 level

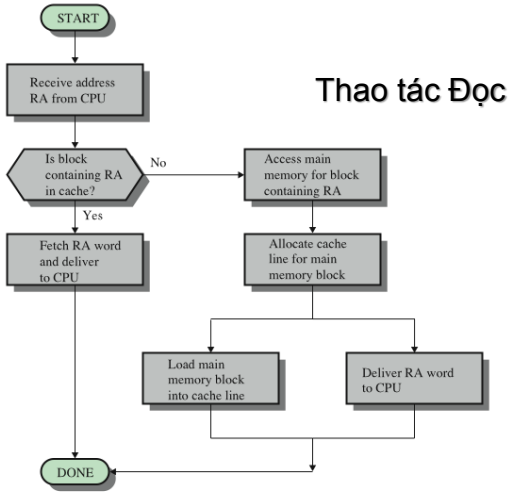


(b) Three-level cache organization

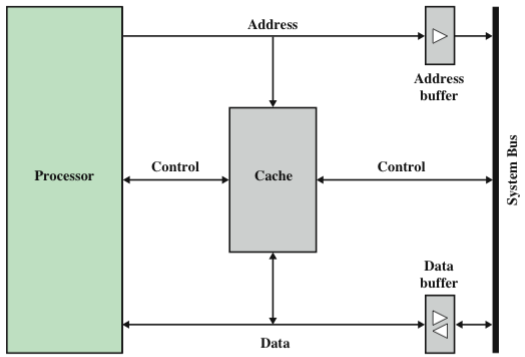
Cấu trúc bộ nhớ chính/cache



Thao tác Đọc Cache



Tổ chức bộ nhớ cache điển hình



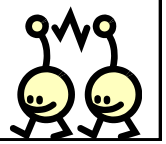
4.3. Các yếu tố khi thiết kế Cache

a. Địa chỉ bộ nhớ cache	e. Chính sách ghi
Logic	Ghi xuôi
Vật lý	Ghi ngược
b. Kích thước bộ nhớ cache	f. Kích thước line
c. Ánh xạ bộ nhớ	g. Cache nhiều cấp
Trực tiếp	Một hoặc hai cấp
Kết hợp	Thống nhất hoặc tách riêng
Tập kết hợp	
d. Thuật toán thay thế	
Least recently used (LRU)	
First in first out (FIFO)	
Least frequently used (LFU)	
Random	

a. Địa chỉ bộ nhớ cache

Bộ nhớ ảo

- Bộ nhớ ảo
  - Cho phép các chương trình định địa chỉ bộ nhớ theo quan điểm logic, không liên quan đến số lượng bộ nhớ chính có sẵn
  - Khi được sử dụng, các trường địa chỉ trong lệnh chứa các địa chỉ ảo
  - Để đọc ra và ghi vào bộ nhớ chính, một khối quản lý bộ nhớ (MMU – Memory Management Unit) sẽ dịch từng địa chỉ ảo sang địa chỉ vật lý trong bộ nhớ chính



Cache vật lý và cache logic

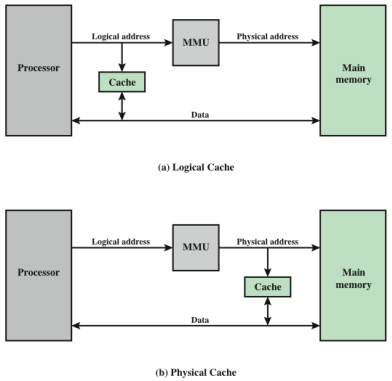


Figure 4.7 Logical and Physical Caches

Processor	Type	Year of Introduction	L1 Cache[a]	L2 cache	L3 Cache
IBM 360/85	Mainframe	1968	16 to 32 kB	—	—
PDP-11/70	Minicomputer	1975	1 kB	—	—
VAX 11/780	Minicomputer	1978	16 kB	—	—
IBM 3033	Mainframe	1978	64 kB	—	—
IBM 3090	Mainframe	1985	128 to 256 kB	—	—
Intel 80486	PC	1989	8 kB	—	—
Pentium	PC	1993	8 kB/8 kB	256 to 512 KB	—
PowerPC 601	PC	1993	32 kB	—	—
PowerPC 620	PC	1996	32 kB/32 kB	—	—
PowerPC G4	PC/server	1999	32 kB/32 kB	256 KB to 1 MB	2 MB
IBM S/390 G6	Mainframe	1999	256 kB	8 MB	—
Pentium 4	PC/server	2000	8 kB/8 kB	256 KB	—
IBM SP	High-end server/ supercomputer	2000	64 kB/32 kB	8 MB	—
CRAY MTA[b]	Supercomputer	2000	8 kB	2 MB	—
Itanium	PC/server	2001	16 kB/16 kB	96 KB	4 MB
Itanium 2	PC/server	2002	32 kB	256 KB	6 MB
IBM POWER5	High-end server	2003	64 kB	1.9 MB	36 MB
CRAY XD-1	Supercomputer	2004	64 kB/64 kB	1MB	—
IBM POWER6	PC/server	2007	64 kB/64 kB	4 MB	32 MB
IBM z10	Mainframe	2008	64 kB/128 kB	3 MB	24-48 MB
Intel Core i7 EE 990	Workstaton/ server	2011	6 × 32 kB/32 kB	1.5 MB	12 MB
IBM zEnterprise 196	Mainframe/ Server	2011	24 × 64 kB/ 128 kB	24 × 1.5 MB	24 MB L3 192 MB L4

b. Kích thước cache trong một số bộ xử lý

a, Hai giá trị cách nhau bằng dấu / là cache chỉ thị và cache dữ liệu.
b, Cả hai cache đều là cache chỉ thị; Không có cache dữ liệu.

c. Ánh xạ bộ nhớ

- Bởi vì số đường cache ít hơn số khối bộ nhớ chính, cần có một thuật toán ánh xạ các khối bộ nhớ chính vào các đường bộ nhớ cache
- Ba kỹ thuật có thể được sử dụng:

Trực tiếp
- Đơn giản nhất
- Ánh xạ mỗi khối của bộ nhớ chính vào một đường cache có thể

Kết hợp
- Cho phép một khối nhớ chính được nạp vào bất kỳ đường cache nào
- Logic điều khiển cache diễn giải địa chỉ bộ nhớ bằng một trường Tag và trường Word
- Để xác định một khối có ở trong một cache không, logic điều khiển cache phải cùng lúc kiểm tra Tag của tất cả các đường

Set Associative
- Thể hiện ưu điểm của cả phương pháp trực tiếp và kết hợp, đồng thời giảm nhược điểm

Ánh xạ trực tiếp

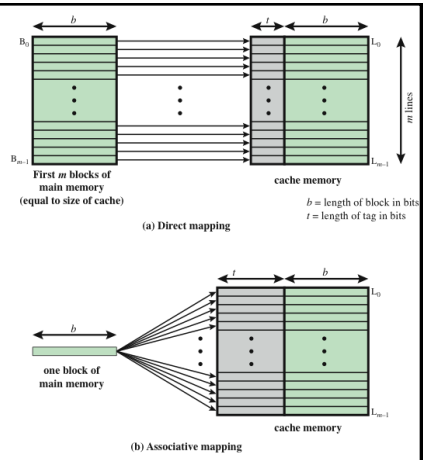
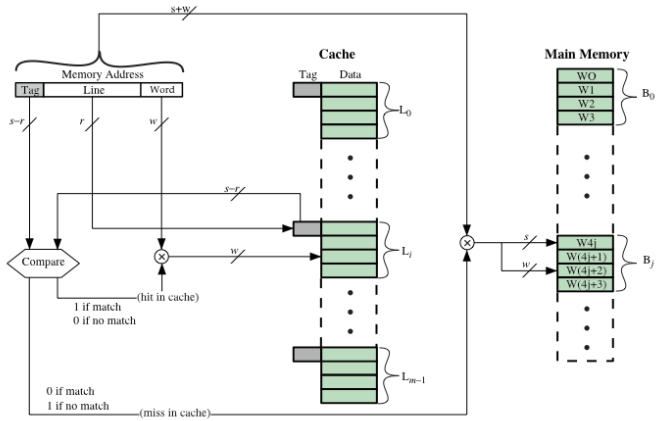
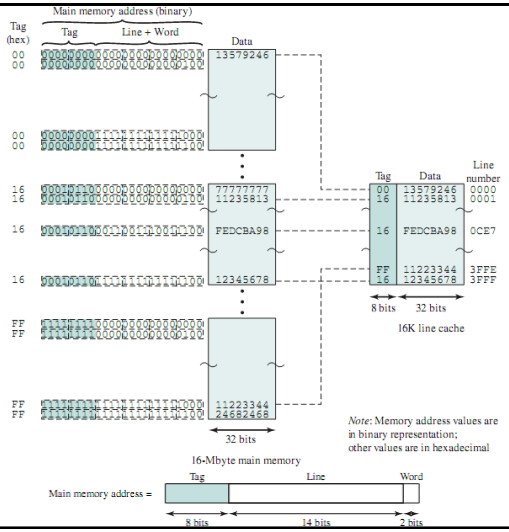


Figure 4.8 Mapping From Main Memory to Cache: Direct and Associative

Tổ chức cache ánh xạ trực tiếp



Ví dụ ánh xạ trực tiếp



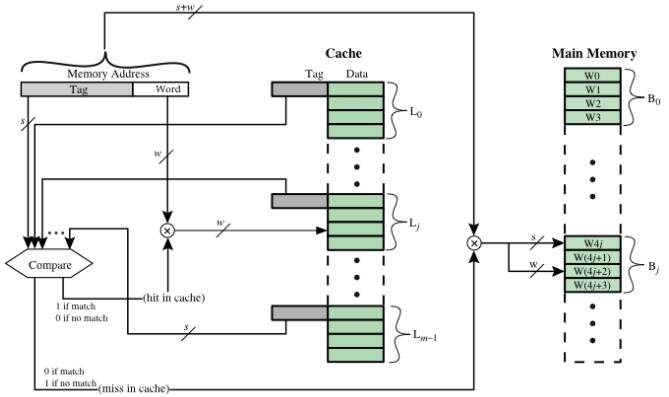
Tổng kết ánh xạ trực tiếp

- Độ dài địa chỉ= (s + w) bits
- Number of addressable units = 2^{s+w} words or bytes
- Kích thước khối= kích thước line = 2w words or bytes
- Số khối trong bộ nhớ chính = $2^{s+w}/2^{w} = 2^{s}$
- Số line trong bộ nhớ cache = m = 2r
- Kích thước của tag = (s – r) bits

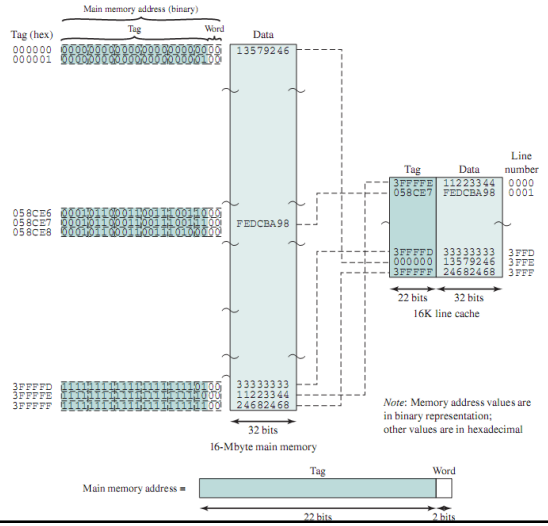
Victim Cache

- Ban đầu được đề xuất để giảm các trễ xung đột trong các cache ánh xạ trực tiếp mà không ảnh hưởng đến thời gian truy cập nhanh của nó
- Bộ nhớ cache kết hợp hoàn toàn
- Kích thước điển hình là 4 đến 16 đường cache
- Nằm giữa bộ nhớ cache L1 được ánh xạ trực tiếp và cấp độ bộ nhớ tiếp theo

Tổ chức Cache kết hợp hoàn toàn



Ví dụ ánh xạ kết hợp



Tổng hợp ánh xạ kết hợp

- Address length = (s + w) bits
- Number of addressable units = 2^{s+w} words or bytes
- Block size = line size = 2w words or bytes
- Number of blocks in main memory = $2^{s+w}/2^{w} = 2^{s}$
- Number of lines in cache = undetermined
- Size of tag = s bits

Set Associative Mapping

- Thể hiện ưu điểm của cả phương pháp trực tiếp và kết hợp đồng thời giảm nhược điểm
- Cache bao gồm một số set
- Mỗi set chứa một số line
- Một khối sẽ được ánh xạ vào một line bất kỳ trong một set nhất định
- Ví dụ: 1 set có 2 line
  - Ánh xạ kết hợp 2 chiều
  - Một khối có thể nằm trong 1 trong 2 line trong một set

Ánh xạ từ
bộ nhớ chính
đến bộ nhớ Cache:

k-Way
Set Associative

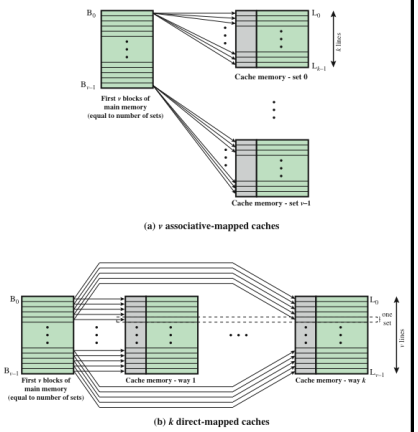
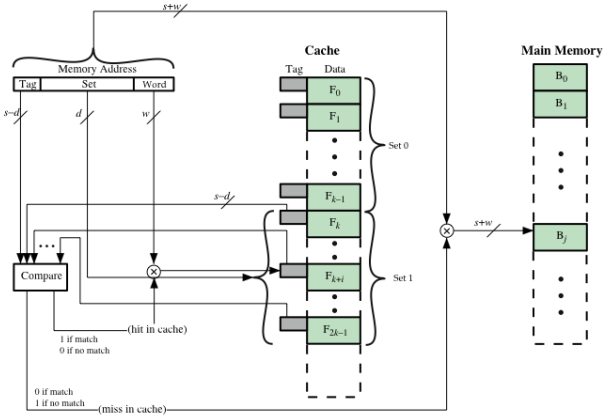


Figure 4.13 Mapping From Main Memory to Cache: *k*-way Set Associative

Tổ chức cache *k*-Way Set Associative



Tổng kết ánh xạ Set Associative

- Address length = (s + w) bits
- Number of addressable units = 2^{s+w} words or bytes
- Block size = line size = 2^{w} words or bytes
- Number of blocks in main memory = $2^{s+w}/2^{w}=2^{s}$
- Number of lines in set = k
- Number of sets = v = 2^{d}
- Number of lines in cache = m=kv = k * 2^{d}
- Size of cache = *k* * 2^{d+w} words or bytes
- Size of tag = (s – d) bits

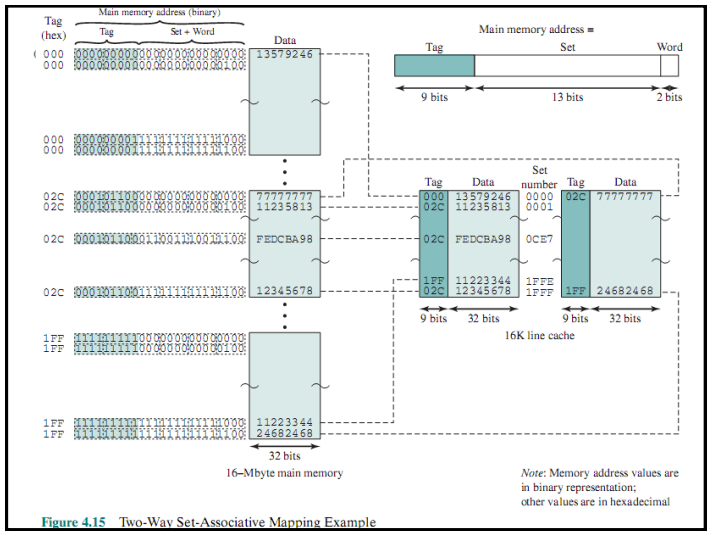
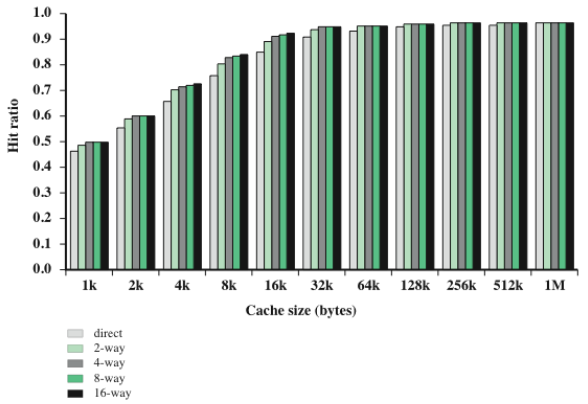
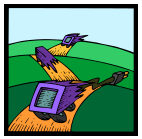


Figure 4.15 Two-Way Set-Associative Mapping Example

Sự kết hợp thay đổi theo kích thước cache



d. Thuật toán Thay thế



- Khi bộ nhớ cache đã đầy, nếu một khối mới được đưa vào cache, một trong những khối hiện có phải được thay thế
- Đối với ax trực tiếp: chỉ có một line có thể cho một khối bất kỳ và không có sự lựa chọn nào khác
- Đối với các kỹ thuật kết hợp và set-associative, cần có một thuật toán thay thế
- Để đạt được tốc độ cao, thuật toán phải được thực hiện trong phần cứng

4 thuật toán thay thế phổ biến nhất

- Least recently used (LRU)
  - Hiệu quả nhất
  - Thay thế khối nằm trong cache lâu nhất mà không có tham chiếu đến nó
  - Do triển khai đơn giản, LRU là thuật toán thay thế phổ biến nhất
- First-in-first-out (FIFO)
  - Thay thế khối đã nằm trong cache lâu nhất
  - Dễ dàng thực hiện như một kỹ thuật vòng đệm hoặc round-robin
- Least frequently used (LFU)
  - Thay thế khối có ít tham chiếu đến nó nhất
  - Có thể thực hiện bằng cách kết hợp một bộ đếm với mỗi line

e. Chính sách ghi

Khi một khối trong cache được thay thế, có 2 trường hợp cần xem xét:

Nếu khối cũ trong cache không thay đổi, có thể ghi đè khối mới lên mà không cần ghi khối cũ ra trước

Nếu ít nhất 1 thao tác ghi đã được thực hiện trên 1 word trong line của cache thì bộ nhớ chính phải được cập nhật bằng cách ghi line của cache ra khối của bộ nhớ trước khi đưa khối mới vào

Có hai vấn đề phải đối mặt:

Nhiều thiết bị có thể có quyền truy cập vào bộ nhớ chính

Một vấn đề phức tạp hơn xảy ra khi nhiều bộ xử lý được gắn vào cùng một bus và mỗi bộ xử lý lại có cache cục bộ riêng - nếu một word bị thay đổi trong một cache, nó có thể làm mất hiệu lực một word trong các cache khác

Write Through và Write Back

- Write through
  - Kỹ thuật đơn giản nhất
  - Tất cả các thao tác ghi được thực hiện cho bộ nhớ chính cũng như cache
  - Nhược điểm: tạo ra lưu lượng bộ nhớ đáng kể và có thể tạo ra nút cổ chai
- Write back
  - Giảm bộ nhớ ghi
  - Chỉ cập nhật trong bộ nhớ cache
  - Các bộ nhớ chính không có hiệu lực. Do đó truy cập bằng các mô-đun I / O chỉ có thể được cho phép thông qua cache
  - Nhược điểm: mạch phức tạp và khả năng có nút cổ chai

f. Kích thước Line

- Khi 1 khối dữ liệu được lấy ra và đặt trong cache, sẽ thu được không chỉ word mong muốn mà còn 1 số word liền kề
- Khi kích thước khối tăng, ban đầu tỷ lệ truy cập sẽ tăng do nguyên tắc cục bộ
- Khi kích thước khối tăng, dữ liệu hữu ích hơn được đưa vào cache
- Tỷ lệ truy cập bắt đầu giảm khi khối lớn dần và xác suất sử dụng thông tin mới tìm được sẽ thấp hơn xác suất tái sử dụng thông tin đã thay thế
- 2 tác động:
- 1, Các khối lớn hơn làm giảm số lượng khối trong 1 cache
- 2, Khi 1 khối lớn lên, mỗi word thêm vào lại càng khác word yêu cầu

g. Cache nhiều cấp

- Khi mật độ logic tăng lên, cache có thể nằm trên cùng chip với bộ xử lý
- Cache trên chip làm giảm hoạt động bus ngoài của bộ xử lý; tăng tốc thời gian xử lý và tăng hiệu năng toàn hệ thống
  - Khi chỉ thị yêu cầu hoặc dữ liệu được tìm thấy trong cache trên chip, truy cập bus được loại bỏ
  - During this period the bus is free to support other transfers
  - Truy cập bộ nhớ cache trên chip sẽ nhanh hơn đáng kể so với các chu trình bus trạng thái zero-wait
  - Trong giai đoạn này, bus tự do hỗ trợ các lượt truyền khác
- Cache 2 cấp: Cache bên trong là cấp1 (L1), Cache bên ngoài là cấp 2 (L2)
- Tiết kiệm tiềm năng do sử dụng cache L2 phụ thuộc vào tỷ lệ truy cập vào cả cache L1 và L2
- Việc sử dụng cache nhiều cấp làm cho các vấn đề thiết kế liên quan đến cache phức tạp hơn, gồm kích thước, thuật toán thay thế, chính sách ghi

Tỉ lệ truy cập (L1 & L2) cho 8 Kbyte và 16 Kbyte L1

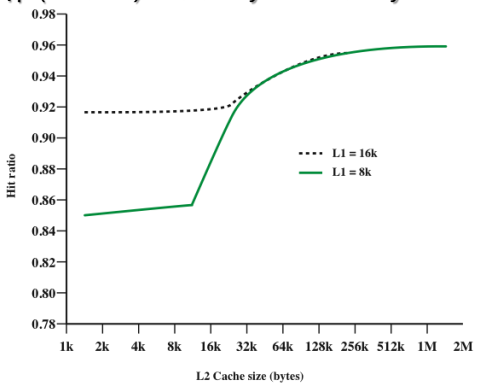


Figure 4.17 Total Hit Ratio (L1 and L2) for 8 Kbyte and 16 Kbyte L1

Cache thống nhất / cache phân chia

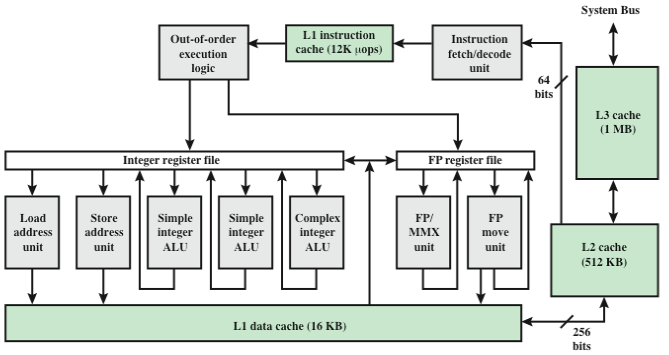
- Phân chia cache trở nên phổ biến:
  - 1 dành cho chỉ thị / 1 cho dữ liệu
  - Tồn tại ngang hàng, thường là 2 cache L1
- Ưu điểm của cache thống nhất:
  - Tốc độ truy cập cao hơn
    - Cân bằng tải của chỉ thị và dữ liệu được nạp tự động
    - Chỉ cần thiết kế và thực hiện một bộ nhớ cache
  - Xu hướng: cache phân chia ở L1 và cache thống nhất ở các cấp cao hơn
- Ưu điểm của cache phân chia:
  - Loại bỏ sự cạnh tranh cache giữa khối tìm nạp / giải mã lệnh và khối thực hiện
    - Quan trọng trong pipelining

Problem	Solution	Processor on which Feature First Appears
External memory slower than the system bus.	Add external cache using faster memory technology.	386
Increased processor speed results in external bus becoming a bottleneck for cache access.	Move external cache on-chip, operating at the same speed as the processor.	486
Internal cache is rather small, due to limited space on chip	Add external L2 cache using faster technology than main memory	486
Contention occurs when both the Instruction Prefetcher and the Execution Unit simultaneously require access to the cache. In that case, the Prefetcher is stalled while the Execution Unit's data access takes place.	Create separate data and instruction caches.	Pentium
Increased processor speed results in external bus becoming a bottleneck for L2 cache access.	Create separate back-side bus that runs at higher speed than the main (front-side) external bus. The BSB is dedicated to the L2 cache.	Pentium Pro
	Move L2 cache on to the processor chip.	Pentium II
Some applications deal with massive databases and must have rapid access to large amounts of data. The on-chip caches are too small.	Add external L3 cache.	Pentium III
	Move L3 cache on-chip.	Pentium 4

Cache Pentium 4

Bảng 4.4 Intel Cache Evolution

Sơ đồ khối Pentium 4



Các chế độ hoạt động Cache Pentium 4

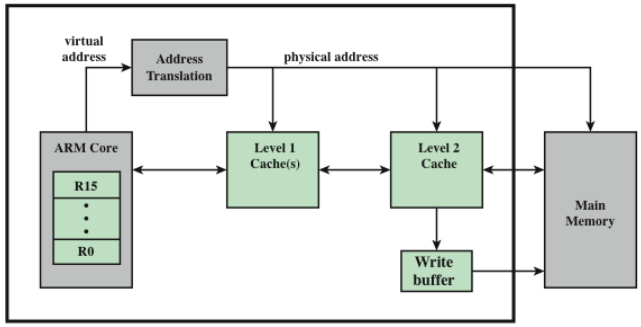
Control Bits	Operating Mode		
NW	Cache Fills	Write Throughs	Invalidates
0	Enabled	Enabled	Enabled
0	Disabled	Enabled	Enabled
1	Disabled	Disabled	Disabled

Note: CD = 0; NW = 1 là kết hợp không hợp lệ.

Đặc tính Cache ARM

Core	Cache Type	Cache Size (kB)	Cache Line Size (words)	Associativity	Location	Write Buffer Size (words)
ARM720T	Unified	8	4	4-way	Logical	8
ARM920T	Split	16/16 D/I	8	64-way	Logical	16
ARM926EJ-S	Split	4-128/4-128 D/I	8	4-way	Logical	16
ARM1022E	Split	16/16 D/I	8	64-way	Logical	16
ARM1026EJ-S	Split	4-128/4-128 D/I	8	4-way	Logical	8
Intel StrongARM	Split	16/16 D/I	4	32-way	Logical	32
Intel Xscale	Split	32/32 D/I	8	32-way	Logical	32
ARM1136-JF-S	Split	4-64/4-64 D/I	8	4-way	Physical	32

ARM Cache và tổ chức bộ đệm ghi



Tổng kết

Chương 4

- Đặc điểm của hệ thống bộ nhớ
  - Vị trí
  - Dung lượng
  - Đơn vị truyền
- Bộ nhớ phân cấp
  - How much?
  - How fast?
  - How expensive?
- Nguyên lý bộ nhớ Cache

Bộ nhớ Cache

- Các yếu tố trong thiết kế cache
  - Địa chỉ bộ nhớ cache
  - Kích thước bộ nhớ cache
  - Ánh xạ bộ nhớ
  - Thuật toán thay thế
  - Chính sách ghi
  - Kích thước line
  - Cache nhiều cấp
- Tổ chức cache Pentium 4
- Tổ chức cache ARM