Chapter 5. Cache Memory

Problem 5.1

\[ \begin{array}{cccc}
\text{tag} & \text{index} & W/L & B/W \\
A_{23} \ldots A_{15} & A_{14} \ldots A_6 & A_5 \ldots A_3 & A_2 \ldots A_0 \\
\end{array} \]

a. Address bits unaffected by translations
   \( A_{14} - A_0 \)
b. Bits to address the cache directories
   \( A_{14} - A_6 \)
c. Address bits compared to entries in the cache directory
   \( A_{23} - A_{15} \)
d. Address bits appended to (b) to address cache array
   \( A_5 - A_3 \)

Problem 5.3

The effective miss rate for cache in 5.1:

\[
\text{DTMR} = .007 \ (128\text{KB}, 64\text{B}/\text{L}, \text{Fully Assoc}, \text{Table A.1}) \\
\text{Adjustment factor} = 1.05 \ (64\text{B}/\text{L}, 4\text{-way}, \text{Table A.4}) \\
\text{Miss rate} = .007 \times 1.05 = .00735 = .735\%
\]

Problem 5.6

Assume the cache of problem 5.1 with 16 \( B/L \).

a. \( Q = 20,000 \)
   Miss rate from Figure A.9: \( .01 = .0947 \)
   Miss rate = .0508

b. The optimal cache size is the smallest size with a low miss rate. From Figure 5.26, the \( Q = 20,000 \) line flattens out around 32KB. From Table A.9 (which tabulates the same data), we see that the miss rate is essentially unchanged for caches of either size 32KB or 64KB and above.

Problem 5.7

a. L1 cache: 8 KB, 4-way, 16B/L
   DTMR: from Table A.1, \( .075 = 7.5\% \)
   Adjustment for 4W associativity: from Table A.4, 1.04
   Miss rate = \( .075 \times 1.04 = .078 = 7.8\% \)
L2 cache: 64KB, direct mapped, 64B/L
DTMR: from Table A.1, .011 = 1.1%
Adjustment for direct mapped cache: from Table A.4, 1.57
Miss rate = .011 \times 1.57 = .0173 = 1.73%

b. Expected CPI loss due to cache misses

\[
\begin{align*}
\text{Refs/L} & = 1.5 \\
\text{I-Reads/L} & = 1.0 \\
\text{D-Reads/L} & = .31 \\
\text{D-Writes/L} & = .19 \\
\text{Miss penalty (MP) for L1} & = 3 \text{ cycles} \\
\text{MP for L1 + MP for L2} & = 10 \text{ cycles} \\
\text{MP for L2} & = 10 - 3 = 7 \text{ cycles}
\end{align*}
\]

We make the assumption of statistical inclusion, that is, if we miss in L2 we also miss in L1, so we can use the DTMR (solo) as global miss rates. We can make this assumption because L2 is significantly larger than L1.

Note that since the L1 cache is WTNWA, we need only consider read misses. Assume that writes can be considered L1 cache hits when calculating miss penalties if they miss in L1 but hit in L2.

Assume 30% of lines in L2 integrated cache are dirty (from Section 5.6).

Expected CPI loss due to cache miss:

\[
\begin{align*}
\text{CPI loss} = & \left( \text{I-reads} + \text{D-reads} \right) / I \times MP_{L1} + \\
& \left( \text{I-reads} + \text{D-reads} + \text{D-writes} \right) / I \times MP_{L2} \times (1 + w) \\
= & (1.31) \times .078 \times 3 + (1.5) \times (.0173) \times 7 \times 1.3 = .54 \text{ CPI}
\end{align*}
\]

c. Will all lines in L1 always reside in L2?

No, because L1 is 4-way associative and L2 is direct mapped. Considering the criteria from 5.12.1:

(i) Number of L2 sets \geq Number of L1 sets
\[
\begin{align*}
\text{Number of L2 sets} & = \frac{64K \times B}{4} = 1024 \text{ sets} \\
\text{Number of L1 sets} & = \frac{8K \times B}{4 \times 16 \times 64} = 128 \text{ sets} \\
1024 \geq 128, \text{ so this criterion holds.}
\end{align*}
\]

(ii) L2 assoc \geq L1 assoc
\[
\begin{align*}
\text{L2 assoc} & = 1 \text{ < L1 Assoc = 4} \\
\text{This condition does not hold, so logical inclusion does not apply.}
\end{align*}
\]
Problem 5.8

L1: 4KB direct-mapped, WTNWA
L2: 8KB direct-mapped, CBWA
16BL for both caches

a. Yes
   L2 sets (512) > L1 sets (256)
   L2 associativity (1) = L1 associativity (1)
   Also, since L1 is write through, L2 has the updated data of L1.

b. No
   L2 sets (128) < L1 sets (256)

c. No
   L2 associativity < L1 associativity
   Also, since L1 is copyback, L2 may not have the updated data of L1.

Problem 5.9

Assume there is statistical inclusion.

$MR_{L1} = 10\%$

$MR_{L2} = 2\%$

$MP_{L1} = 3$ cycles

$MP_{L1+L2} = 10$ cycles

$MP_{L2} = 10 - 3 = 7$ cycles

1 ref/I

Excess CPI due to cache misses $= \frac{1}{1} \times (MR_{L1} \times MP_{L1} + MR_{L2} \times MP_{L2})$

$= .1 \times 3 + .02 \times 7 = .44$ CPI

Problem 5.13

a. 12KB, 3-way set associative

$32R = 2^5B$ line size. There are 5 bits for line address.
There are $\frac{12KB}{\frac{32M}{2^5}} = 128 = 2^7$ sets. There are 7 bits for index.
There are $2^6 - 7 - 5 = 14$ bits for tag.

Address: $A_{25} \ldots A_{12} A_{11} \ldots A_5 A_4 \ldots A_0$

Directory entry: 14 bits 14 bits 14 bits
b. DTMR for 8KB: 0.05 (Table A.1)

DTMR for 16KB: 0.035

Interpolate: DTMR for 12KB (fully associative): \( \frac{0.05 + 0.035}{2} = 0.0425 \)

Adjustment for 2-way set associativity (8KB): 1.13 (Table A.4)

Adjustment for 2-way set associativity (16KB): 1.13

Adjustment for 4-way set associativity (8KB): 1.035

Adjustment for 4-way set associativity (16KB): 1.035

Interpolate:

Adjustment for 2-way set associativity (12KB): 1.13

Adjustment for 4-way set associativity (12KB): 1.035

Adjustment for 3-way set associativity (12KB): \( \frac{1.13 + 1.035}{2} = 1.0825 \)

DTMR for 3-way set associative cache (12KB): \( 1.0825 \times 0.0425 = 4.6\% \)

c. Since 12KB is not a power of 2, the usual way of dividing the address bits to access the cache will not work.

Observe that a 8KB direct-mapped cache uses the least significant 13 bits of address to access the cache and a 16KB direct-mapped cache uses the least significant 14 bits. For a 12KB direct-mapped cache, when bits [13:12] are 00, 01 or 10, this maps well into the cache, but when bits [13:12] are 11, we will have to decide where to put the data.

One method is to leave all addresses with bits [13:12] = 11 out of the cache. Another method is to have these addresses map to the same lines as addresses with bits [13:12] = 00, 01 or 10.

For either method, there are 5 bits for line address and 14 - 5 = 9 bits for index (where the most significant two bits of the index have special significance). If we want to map addresses with bits [13:12] = 11 to the same cache lines as addresses with bits [13:12] = 01, bit 13 needs to be in the tag as well as the index. In this case, there are 26 - 13 = 13 tag bits. If the addresses with bits [13:12] = 11 are left out of the cache, there are 26 - 14 = 12 tag bits.

\[
\text{Address for } 11\rightarrow01: \overbrace{A_{13}}^{\text{tag}} \ldots \overbrace{A_{14}}^{\text{tag/index}} \overbrace{A_{12}}^{\text{index}} \ldots A_4 \overbrace{A_3}^{\text{line}}
\]

Directory entry for 11→01: 13 bits

d. From (b), DTMR for 12KB fully associative cache: 0.0425

Adjustment for direct-mapped (8KB): 1.35

Adjustment for direct-mapped (16KB): 1.38

Adjustment for direct-mapped (12KB): \( \frac{1.35 + 1.38}{2} = 1.365 \)

DTMR for 12KB direct-mapped cache: \( 1.365 \times 0.0425 = 5.8\% \)

The actual miss rate will probably be worse than this, since this assumes that the addresses are evenly distributed throughout the 12KB, but due to the implementation limitations, this even distribution cannot be achieved.

---

**Problem 5.14**

8 KB integrated level 1 cache (direct mapped, 16 B lines)

128 KB integrated level 2 cache (2 way, 16 B lines)
Solo Miss Rate for L2 cache:
The solo miss rate for L2 cache is same as the global miss rate.

From Table A.1 and A.4, \(.02 \times 1.17 = .0234\).

Local Miss Rate for L2 cache:
The miss rate of an 8 KB level 1 cache is \(.075 \times 1.32 = .099\) from Tables A.1 and A.4, assuming an R/M machine. The number of memory access/I for R/M architecture in scientific environment is:

\[.73 \text{ (instruction)} + .34 \text{ (data read)} + .21 \text{ (data write)} = 1.23 \text{ (pages 31-5)}.\]

Missed memory access/I for L1 = \(1.23 \times .099 = .1218\).

From solo miss rate, we know that the miss rate for L2 cache is \(.0234\).

Missed memory access/I for L2 = \(1.23 \times .0234 = .0288\).

So, L2 local miss rate = \(\frac{.0288}{.1218} = .291\).

Problem 5.18

a. CPI lost due to cache misses

User-only, R/M environment

I-Cache:
8KB, direct-mapped
64B lines
DTMR: 2.4\% (Table A.3)
Adjustment for direct-mapped: 1.46 (Table A.4)
I-cache MR = \(.024 \times 1.46 = .035\), 3.5\%
I-cache MP = 5 + 1 cycle/4B \times 64B = 21 cycles

\[
\text{CPI lost due to I-misses} = \frac{\text{I-Refs}/I \times \text{MR}_{\text{I-cache}} \times \text{MP}_{\text{I-cache}}}{21}
\]
\[= 1.0 \times .035 \times 21\]
\[= .735\]

D-cache:
4KB, direct-mapped
64B lines
CBWA
\(w = \%\) dirty = 50\%
DTMR: 5.5\% (Table A.2)
Adjustment for direct-mapped: 1.45 (it says .45 in Table A.4, should be 1.45)
D-Cache MR = \(.055 \times 1.45 = .0798\), 7.98\%
D-cache MP = 5 + 1 \(/4B \times 64B = 21\) cycles

\[
\text{CPI lost due to D-misses} = \frac{\text{D-refs}/I \times \text{MR}_{\text{D-cache}} \times (1 + \%\) dirty) \times \text{MP}_{\text{D-cache}}}{21}
\]
\[= .5 \times .0798 \times 1.5 \times 21\]
\[= 1.26\]

Total CPI loss = .735 + 1.26 = 2.0 (optional)
b. Find the number of I and D-directory bits and corresponding \textbf{rbe} (area) for both directories.

\textbf{I-Cache}:
Tag size = $26\log_2(8KB) = 13b$
Control bits = valid bit = 1b
Directory bits = $14 \times (8KB/64B) = 14 \times 128 = 1792b \times .6 = 1075.2$ rbe

\textbf{D-Cache}:
Tag size = $26\log_2(4KB) = 14b$
Control bits = valid bit + dirty bit = 2b
Directory bits = $16 \times (4KB/64B) = 1024b \times .6 = 614.4$ rbe

c. Find the number of color bits in each cache
Since both caches are direct-mapped, we are going to have to worry about the 8KB I-cache, which is larger than the page size.
Page offset = $\log_5 4096 = 12$

\textbf{I-Cache}:
\[
\text{Cache index + block offset} = \log_5(8192/64) + \log_2(64) \\
= 7 + 6 = \log_5 8192 = 13 \text{ bits}
\]
Color bits for I-cache = $13b$ (index + offset) $- 12b$ (page offset)
= 1 bit

\textbf{D-cache}:
\[
\text{Cache index + block offset} = \log_5(4096/64) + \log_2(64) \\
= 6 + 6 = \log_5 4096 = 12 \text{ bits}
\]
Color bits for D-cache = $12b$ (index + offset) $- 12b$ (page offset)
= 0 bits

$\rightarrow$ No page coloring is necessary for D-cache.

The operating system must be able to ensure that text (i.e., code) pages match $V = R$ in the first bit of the page address. This requires $2^1 = 2$ free page lists.