Homework Assignment 4 of Computer Architecture

College of Computer Science, Zhejiang University

1. 1. (18 points) Consider the following description of a memory hierarchy.

Virtual address wide = 45 bits, Memory physical address wide = 38 bits, Page size = 4KB.

Cache capacity =8KB. It is a write-back 2-way associative cache with 32Bytes block size.

a) How many bits are there in the fields of tag, index and block offset of the physical memory address.

(b) Draw a graph to show a cache line (including tag, data, and some other control bits) in the cache.

(c) Draw a graph to show if it is implemented in the way of virtually indexed and physically tagged cache.

(d) Please describe the access procedure to the memory hierarchy in (c) that when a CPU address (virtual address) is given to access the cache.

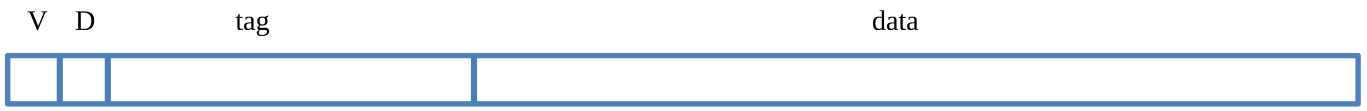
a) (3 points) block: 5bits, index: 7 bits； tag: 26bits；

b) (4 points) 1 bit: valid bit;

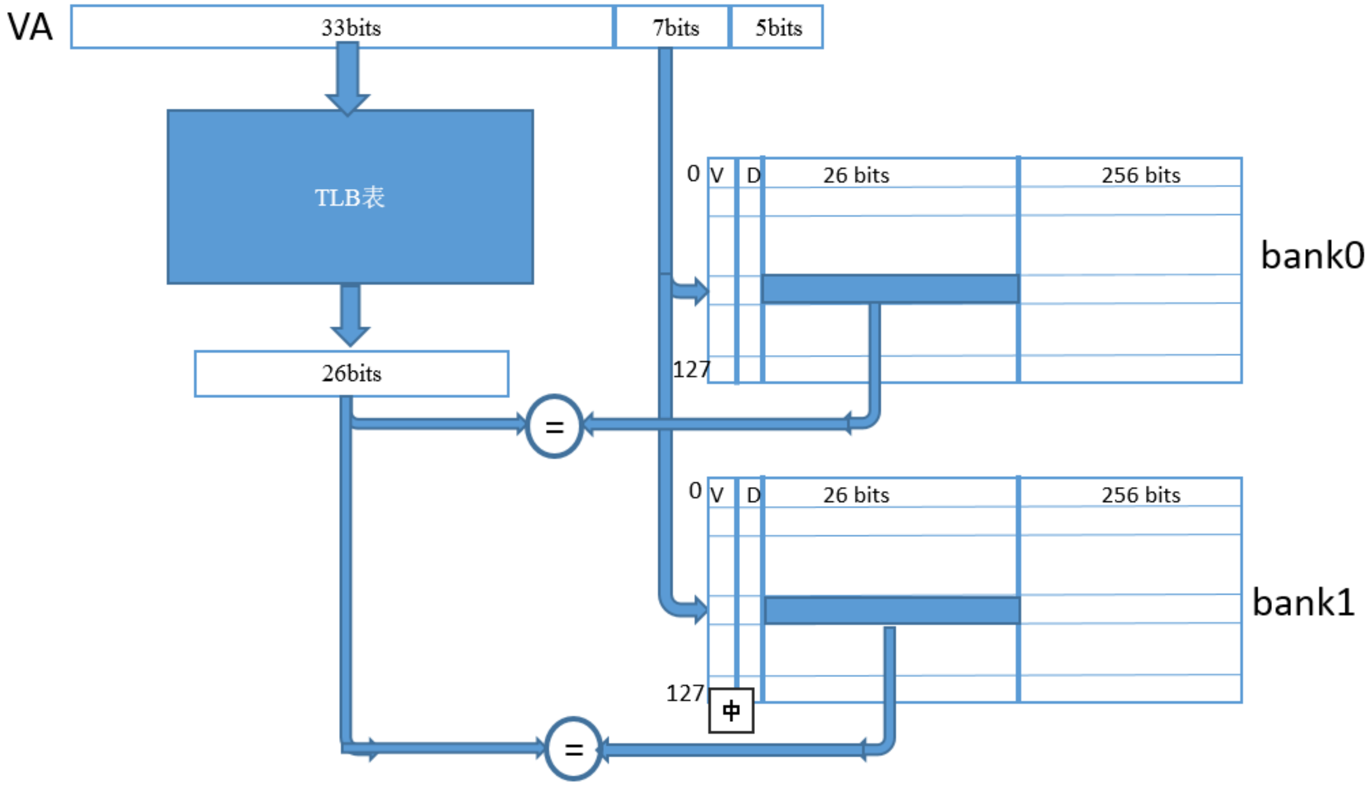
1 bit: dirty bit

26bits: tag bit

256 bits: data block



c) (6 points)



d) (5 points)

CPU use 33 bits VPN (in Virtual address) to access the TLB table, then translate it into 26bits PPN, at the same time use the 7 bits index in 12 bits page offset to access the cache, get two 26bits tags , compare the the two tags with PPN, if one of the tag matches, then hit the cache, get the right data from the matched data line; if none of the tags matches, then miss the cache.

2. (20 points) Assume that we have two machines A and B. The only difference between A and B lies in their cache hierarchies:

Machine A: 64 KB level-one data cache with a *8 ns* access time and a miss rate of 8%

Machine B: 8 KB level-one data cache with a *2 ns* access time and a miss rate of 15%, and a 1 MB level-two cache with a *20 ns* access time and a miss rate of 10%.

Assume that both machines have an I-cache miss rate of 0%, a main memory access time of *50 ns*, and all the bus transfer time could be ignored. Which machine will have a better performance in memory access（AMAT）? Why?

Macheine A AMAT = HitTime + MR * MP = 8 + 8%*50= 12ns

Macheine B AMAT = HitTime + MR * MP = 2 + 15%*(20 + 10% *50) = 5.75ns

3. (10 points) There is a split cache with 16KB data cache and 16KB instruction Cache. Both cache are write back cache with 16 Byte block size using no-write-allocate policy without write buffer. It is known that when the memory is accessed to fetch the miss block, it takes 5 clock cycles to send the access address to memory, 25 clock cycles to get the first word and with a bus transfer bandwidth of 4 bytes/5 clock cycles. Cache hit time= 1 clock cycle.

If the experiments are obtained as following:

Memory Access type	total number	instruction access	data access	data read access	data write access
access time	1000002	757341	242661	159631	83030
Miss	22587	20311	4619	2276	2343

Fetched word from memory: 90348 (4B per word) （=20311+2276）*4 ）

words copied back to memory: 10972

Please calculate the the AMAT.

total = 1000002* 1

+ (20311 + 2276) * (5+ 25 + 4 * 5) (Load the request block when instruction miss and data read miss.Miss Penalty = time for transfer the whole block.)

+ 10972/4 *(5+ 25 + 4 * 5) (Penalty to write back the dirty block)

+ 2343 * (5+25+**5**) (Only the request **word** instead of the request block will write around into memory when write miss happen.)

= 1000002 + 1129350 + 137150+ 82005

= 2348507

AMAT = 2348507/1000002 = 2.35

4. (50points)You are building a system around a processor with in-order execution that runs at 1.1 GHz an has a CPI of 0.7 excluding memory accesses. The only instructions the read or write data from memory are loads (20% of all instructions) and stores (5% of all instructions).

The memory system for this computer is composed of a split L1 cache that impose no penalty on hits. Both the I-cache and D-cache are direct mapped and hold 32KB each. The I-cache has a 2% miss rate and 32-byte blocks, and the D-cache is write through with a 5% mis rate and 16-byte blocks. There is a write buffer on the D-cache that eliminates stalls for 95% of all writes.

The 512KB write-back, unified L2 cache has 64-byte blocks and an access time of 15ns. It is connected to the L1 cache by a 128-bit data bus that runs at 266MHz and can transfer on 128-bit word per bus cycle. Of all memory references sent to the L2 cache in this system, 80% are satisfied without going to main memory. Also 50% of all blocks replaced are dirty.

The 128-bit-wide main memory has an access latency of 60ns, after which any number of bus words may be transferred at the rate of one per cycle on the 128-bit-wide 133 MHz main memory bus.

a) What is the average memory access time for instruction accesses ?

b) What is the average memory access time for data reads ?

c) What is the average memory access time for data writes ?

d) What is the overall CPI, including memory accesses ?

e) You are considering replacing the 1.1GHz CPU with one that runs at 2.1GHz, but is otherwise identical. How much faster doe the system run with a faster processor ? Assume the L1 cache still has no hit penalty, and that the speed of the L2 cache, main memory, and buses remains the same in absolute terms (e.g. the L2 cache has a 15n access time and a 266MHz bus connecting it to the CPU and L1 cahce.

Assumption:

CR = 1.1 GHz cycle = 1/(1.1*10^9) = 0.9 (ns)

CPI excluding memory = 0.7 (cycle)

Load% = 20%

Store% = 5%

L1: HitTime = 0

I-cache: direct mapped, Capacity = 32KB, MR = 2%, 32B/block;

D-cache: direct mapped, wirte through, Capacity=32KB, MR=5%, 16B/block;

Write buffer eliminates 95% stall.

L2: Capacity=512KB, write back, 64B/block, Accesstime=15ns, 128bit/data bus, 64*8/128=4(times transformation for one block)

TR= 266MHz, 128bit/transfer cycle,

one transfer cycle time(TT_{L2}) = 1/(266*10^6) = 3.76ns

MR= 20%, DirtyRate = 50%

MM: 128bit wide, access latency=60ns, 128bit/transfer cycle,

one transfer cycle time(TT_{mm}) = 1/(133*10^6) = 7.52ns

load / write block latency = 60ns + 4*7.52 = 90ns

Solution:

a. AMAT of Instruction Access

AMATinstr = HitTime_I-cache + MRi-cache* (HT-L2 + Time to fetch miss block from L2 + Time to fetch from MM + Time to write back the dirty block when miss)

= 0+ 2% * (15 + (32*8/128) * TT_{L2}+ 20% * ((60 + (64*8/128) * TTmm + 20%*50%* (60 + 64*8/128*TTmm))

= 2% * (15 + 2*3.76 + 20%*(60+4*7.52) + 20%*50%*(60+4*7.52)

= 2% * (15 + 7.52 + 20% * 90.08 + 20% * 50% *90.08) = 0.99 (ns)

b. AMAT of Data Read

AMAT = HitTime_D-cache + MRd-cache* (HT-L2 + Time to fetch miss block from L2 + Time to fetch from MM + Time to write back the dirty block when miss)

= 0 + 5% * (15 + (16*8/128) * TT_{L2} + 20% * ((60 + (64*8 /128) *TTmm + 20%*50%* (60 + 64*8/128*TTmm))

= 5% * (15 + 3.76 + 20% *1.5* (60+4*7.52)) = 5% * 45.78=2.29(ns)

c. AMAT of Data Write (Every write will go to the write buffer. Each time for one word.)

AMAT = Hit time of L1 + (1-95%)* （100% x (Hit time of L2 + write a word into L2 if hit + Miss rate of L2 x write a word around into memory) ）

= 0 + (1-95%) * (15 + 80%*1 * TT_{L2} + 20% * (60 + 1* TT_{mm}))

= 5% * (15 + 0.8*3.76 + 20% * (60 + 7.52 + ~~50%*90~~)) (NO WRITE ALLOCATE)

= 5% *(15 + 3.008 + 13.504) = 1.576 (ns)

*   ~~50%*90~~)):   No replacement will happen due to NO WRITE ALLOCATE .

How about write allocate ?

= Hit time of L1 + (1-95%)* （Hit time of L2 + Miss rate of L2 * 50% * write dirty block to memory + fetch miss block to into L2 + write a word into L2）

= 0+ (1-95%) * (15 + 20%* 50% * (60 + 4* TTmm)+ 20% * (60 + 4* TTmm) + 1*3.76)

= 5% * (15 + 9 + 18 + 3.76)

= 2.29

* 50%*90: time latency for write back the dirty block.

d. Over all of CPI

CPI = CPI of org + stalls for instruction reference + stalls of Data read per instruction + stalls of Data write per instruction

= 0.7 + 0.99/0.9 + 20%*2.29/0.9 + 5%*1/0.9

= 0.7 + 1.1 +0.182 + 0.56 = 2.54

e.

CR = 2.1 GHz   cycle = 1/(2.1*10^9) = 0.48 (ns)

CPI = 0.7 + 0.99/0.48 + 20%*2.29/0.48 + 5%*2.29/0.48 = 0.7 + 2.1 + 1.2 = 4.0

$$\text{Speedup} = \frac{\text{IC} * 2.54 * 0.9}{\text{IC} * 4.0 * 0.48} = 1.19$$

So the fast processor is 1.19 times faster.