计算机组成小测4答案

1. By convention, a cache is named according to the amount of data it contains (i.e., a 4 KB cache can hold 4 KB of data); however, caches also require SRAM to store metadata such as **tags**, **valid bits**, **dirty bits**. Assume that the caches are byte addressable, and that addresses and words are 32 bits. Calculate the total number of bits required to implement a 16 KB direct-mapped cache with 4-word blocks.

   Answer:
   Offset: 4bits Index: 10bits Tag: 32 - 4 - 10 = 18bits
   Total bits: (18 + 1 + 1)×1024 + 1024×16×8 = 151552 bits

2. Assume we have a direct-mapped byte-addressed cache with capacity 32B and block size of 8B.
   Q1: Of the 32 bits in each address, which bits do we use to find the tag, index, and offset of the cache?
   Q2: Classify each of the following byte memory accesses as a cache hit, cache miss, or cache miss with replacement.

   Answer:
   Offset: [2: 0] Index:[4: 3] tag:[31: 5]

Address	Tag & Index & Offset (Decimal)	Hit / Miss / Replace
0x00000004	Tag: 0 Index: 0 Offset: 4	Miss
0x00000005	Tag: 0 Index: 0 Offset: 5	Hit
0x00000068	Tag: 3 Index: 1 Offset: 0	Miss
0x000000c8	Tag: 6 Index: 1 Offset: 0	Replace
0x00000068	Tag: 3 Index: 1 Offset: 0	Replace
0x000000dd	Tag: 6 Index: 3 Offset: 5	Miss
0x00000045	Tag: 2 Index: 0 Offset: 5	Replace
0x000000cf	Tag: 6 Index: 1 Offset: 7	Replace
0x000000f3	Tag: 7 Index: 2 Offset: 3	Miss
0x000000db	Tag: 6 Index: 3 Offset: 3	Hit
0x0000009c	Tag: 4 Index: 3 Offset: 4	Replace
0x00000157	Tag: 10 Index: 3 Offset: 7	Replace
0x00000fe9	Tag: 127 Index: 1 Offset: 1	Replace

3. It assumes that the address of main memory is 32-bit and is addressed by byte. 8- way set-associative mapping is used between instruction cache and data cache and main memory, and **write through** strategy is used. The data capacity of the cache is 32KB and the block size is 64B
   Q1: How many bits is the tag of each cache line? Does it contain dirty bits?
   Q2: Here is a C code:

```
for (k = 0; k < 1024; k++)
    s[k] = 2 * s[k];
```

   If both array s and variable k are int, int accounts for 4B, variable k is allocated in registers, and the start address of array s in main memory is 0x008000C0, what are the number of data cache misses accessing array s during the execution of this program segment?

Q3: If the first access operation of the CPU is to read the instruction in the main memory unit 0x0001003, briefly explain the process of accessing the instruction from the cache, including the cache miss handling process.

Answer：

Q1：

Offset: 6bits index: 32KB÷64B÷8 = 26 -> 6bits

Tag: 32 - 6 - 6 = 20bits

因为采用 write-through 策略，不需要在 cache 中设置脏位

Q2:

1 次 miss, 15 次 hit，那么 1024 次共有 64 次 miss

Q3:

根据地址算出对应的 index，再由 index 访问对应的 cache line，若 valid 位无效或 valid 有效但 tag 与 cache 中的不相等则发生 read miss，需要到内存中取出对应地址所在 block 的所有数据并写回内存

4. Assume:

1) Instruction cache miss rate is 2%; 2) Data cache miss rate is 4%;

3) CPI without any memory stalls is 2; 4) Miss penalty is 100 cycles;

5) The frequency of all loads and stores in gcc is 36%

Q: How faster a processor would run with a perfect cache?

Answer:

Instruction miss cycles = I×2%×100 =2.00I

Data miss cycles = I×36%×4%×100 =1.44I

Total memory-stall cycles= 2.00I+ 1.44I =3.44 I

CPI with stall = CPI with perfect cache + total memory-stalls = (2 + 3.44)I = 5.44I