

6. 解: ①命中率高, 利用空间、时间局部性, 中间位为组索引将相邻地址放
到一个组, 而相邻地址有极大可能访问。

②节省命中时间?

7. 解: 便于标签对比和组索引同时进行, 减少命中时间。

8. 解: ①. $(1-3\%) \times 1 + 3\% \times 110 = 0.97 + 3.3 = 4.27$

② $1GB = 2^{30} B$.

$\eta = \frac{84 \times 2^{10}}{2^{30}} = \frac{2^{16}}{2^{30}} = \frac{1}{2^{14}} \times 100\%$

$\Rightarrow \frac{1}{2^{14}} \times 1 + (1 - \frac{1}{2^{14}}) \times 110 = 110$

③. ①中局部性原理降低访问时间降低延迟; ②中 $L1$ 缓存命中率低, 体现的是局部性低, 对延迟影响可以忽略不计。

④. $\eta \times 1 + (1-\eta) \times 110 \leq 105 \Rightarrow \eta \geq \frac{5}{109} = 4.6\%$

9. 解:

| | 地址位数 Bit | 缓存大小 KB 10^3 | 块大小 Byte 2^6 | 相联度 $\frac{2^{16}}{2^{10} \times 2^6}$ | 组数 2^5 | 组索引位数 Bit | 标签位数 Bit | 偏移位数 Bit |
|---|-------------|-------------------|-------------------|---|----------------|--------------|-------------|-------------|
| 1 | 32 | 4 | 64 | 2 | 32 | 5 | 21 | 6 |
| 2 | | 4 | 64 | 8 | 8 | 3 | 23 | 6 |
| 3 | | 4 | 64 | 全 | 1 | 0 | 26 | 6 |
| 4 | | 16 | 64 | 1 | 64 | 6 | 18 | 6 |
| 5 | | 16 | 128 | 2 | 64 | 6 | 19 | 7 |
| 6 | | 64 | 64 | 4 | 2 ⁸ | 8 | 18 | 6 |
| 7 | | 64 | 64 | 16 | 64 | 6 | 20 | 6 |
| 8 | | 64 | 128 | 16 | 32 | 5 | 20 | 7 |

$\log_2 \times$



$$10. \text{解: } ①: (0.22 + 100p_1) < (0.52 + 100p_2)$$

$$\Rightarrow p_1 - p_2 < 3 \times 10^{-3}$$

$$②: (0.22 + 0.22kp_1) < 0.52 + 0.52kp_2$$

$$\Rightarrow (0.22kp_1 - 0.52p_2) < 0.3$$

R.1-VR

$$11. \text{解: } ①: 0x1001, 0x1021, 0x1045, 0x1305, 0xffff$$

$$②: 0x1001, 0x1005, 0xffff$$

$$③: 0x1001, 0x1005, 0xffff$$

$$④: 0xffff$$

12. 解

13. 两个 for 循环换位?

$$14. \text{解: } ① \text{ 前: } 2^6 \times 2^7 = 2^{13}$$

$$\text{后: } \frac{2^6}{8} \times 2^7 = 2^{10}$$

$$② \text{ 前: } \frac{2^6}{8} \times 2^7 = 2^{10}$$

$$\text{后: } 2^{10}$$

$$③ \text{ 前: } 2^{10} \Rightarrow 2^5 \times 2^{10} = 2^{15} = 2^5 \text{ KB}$$

$$\text{后: } 4 \text{ KB}$$

