

第六章作业

191220008 陈南瞳

3、

(1)

$512\text{MB}/(64\text{M}\times 8\text{bit})=8$ 个DRAM

(2)

$2\text{GB}/512\text{MB}=4$ 个内存条

(3)

因按字节编址，故主存地址有32位，记为0~31位

因内存条容量为512MB，有8个芯片，故：

芯片内地址为3~28位

行地址为3~15位

列地址为16~28位

选择芯片为0~2位

5、

磁盘旋转一圈： $1\text{s}/(7200\text{转}/\text{分}/60)\times 1000\approx 8.33\text{ms}$

平均旋转等待时间： $8.33\text{ms}/2\approx 4.17\text{ms}$

一个数据块的传输时间： $4\text{KB}/40\text{MB}/\text{s}=0.1024\text{ms}$

数据块平均读取/写回时间： $2\text{ms}+10\text{ms}+4.17\text{ms}+0.1024\text{ms}\approx 16.27\text{ms}$

数据块的处理时间： $20000/500\text{MHZ}\times 1000=0.04\text{ms}$

完成一次“读出-处理-写回”的时间： $2\times 16.27\text{ms}+0.04\text{ms}=32.58\text{ms}$

每秒可以完成： $\lfloor 1\text{s}\times 1000/32.58 \rfloor=30$ 次

8、

(1)

主存地址空间大小为1GB，故主存地址有30位，记为0~29位

每个主存块大小为128字节，且按字节编址，故块内地址有7位，为0~6位

cache有64KB/128B=512行，且为直接映射，故cache行号（行索引）有9位，为7~15位

标记则有 $30-7-9=14$ 位，为16~29位

(2)

cache总容量=512行× (1bit+14bit+128×8bit) =531968(bit)=519.5(Kbit)=64.9375(KB)

12、

(1)

空间局部性：数组x和y都按顺序存放，访问方便，故空间局部性较好

时间局部性：每个元素只访问了一次，故没有时间局部性

命中率：因为命中率的高低与映射方式、cache容量、块大小等相关，但这些信息均未知，故无法推断命中率高低

(2)

直接映射，块大小为16B：

cache行数：32B/16B=2行

每块有：16B/4B=4个数组元素

因数组x和y连续存放，故0x8049040后的连续四个主存块为：x[0]~x[3]、x[4]~x[7]、y[0]~y[3]、y[4]~y[7]

所以，x[0]~x[3]和y[0]~y[3]被映射到cache的同一行，x[4]~x[7]和y[4]~y[7]被映射到cache的同一行

因此，每次for循环访问x[i]和y[i]时，出现颠簸现象，均发生缺失，需要分别重新装入cache，故命中率为0

(3)

2路组相联，块大小为8B：

cache行数：32B/8B=4行

组数：4行/2=2组

每组有：2行

因数组x和y连续存放，故0x8049040后的连续八个主存块为：x[0]~x[1]、x[2]~x[3]、x[4]~x[5]、x[6]~x[7]、y[0]~y[1]、y[2]~y[3]、y[4]~y[5]、y[6]~y[7]

所以，x[0]~x[1]、x[4]~x[5]、y[0]~y[1]、y[4]~y[5]被映射到cache的同一组，x[2]~x[3]、x[6]~x[7]、y[2]~y[3]、y[6]~y[7]被映射到cache的同一组

因此，每次for循环访问x[i]和y[i]时，可以将x[i]和y[i]放在同一组的不同行中；每装入一块的两个数组元素，其中的第二个数组元素总是命中，故命中率为50%

(4)

x[12],直接映射，块大小为16B：

cache行数：32B/16B=2行

每块有：16B/4B=4个数组元素

因数组x和y连续存放，故0x8049040后的连续五个主存块为：x[0]~x[3]、x[4]~x[7]、x[8]~x[11]、y[0]~y[3]、y[4]~y[7]

所以，x[0]~x[3]、x[8]~x[11]、y[4]~y[7]被映射到cache的同一行，x[4]~x[7]、y[0]~y[3]被映射到cache的同一行

因此，每次for循环访问x[i]和y[i]时，x[i]和y[i]不被映射到cache的同一行中；每装入一块的四个数组元素，其中第一个数组元素总是不命中，剩下三个均命中，故命中率为75%

23、（汪老师群里发的第一版）

(1)

指令1: `addl(%edx, %ecx, 4), %eax`

存储器操作数的寻址方式：基址+比例变址+偏移量

(2)

指令的线性地址：0x0+0x8049c08=0x8049c08

操作数的线性地址：0x804d000+50×4=0x804d0c8

(3)

```
movl $0, %ecx
.LOOP
cmpl %ebx, %ecx
jge .EXIT
addl (%edx, $ecx, 4), %eax
incl %ecx
jmp .LOOP
.EXIT
```

(4)

执行到程序P时，已经是保护模式，采用分页虚拟管理模式

因此，CR0中PE=1，PG=1

(5)

第一次执行指令1时，有可能发生缺页异常：

①取指令时，不会发生缺页异常。因为指令1不在一个页的起始处，当指令1前面的指令被调用发生缺页时，会将指令1一起调入内存。

②取操作数时，有可能发生缺页。因为a[0]的地址为0x804d000，正好位于一个页的起始处，若它所在页此前未被访问过，则可能发生缺页异常。此时，页故障线性地址为0x804d000，保存在控制寄存器CR2中。

(6)

指令1的线性地址是：0x8049c08

指令1所在页的虚页号是：0000 1000 0000 0100 1001

页目录索引：0000 1000 00

页表索引：00 0100 1001

页内偏移量：1100 0000 1000

P: 1

R/W: 0

U/S: 1

A: 1

D: 0

(7)

第一次执行指令1时，有可能发生TLB缺失：

①取指令时，不会发生TLB缺失。因为指令1不在一个页的起始处，当指令1前面的指令发生TLB缺失时，会将指令1所在页装入TLB。

②取操作数时，有可能发生TLB缺失。因为若a[0]所在页是第一次被访问，则对应页表项不在TLB中。

20位虚拟页号中，高18位为TLB标记，低2位为TLB组索引。

因为组索引为01，标记为0000 1000 0000 0100 10 (02012H)；发现第一组中，有与该标记相同的，取出对应的页框号028B0H，与页内偏移量c08H拼接，得到主存地址为0x28b0c08。

(8)

cache共有：8KB/32B=256行

组数：256行/2=128组

因此，高20位为标记，中间7位为组索引，低5位为块内地址。

指令1的线性地址为0x8049c08，因页大小为4KB，故低12位为页内偏移量1100 0000 1000，组索引为1100 000，块内地址为0 1000。可知，指令1不在一个主存块的起始位置，故在第一次执行指令1时，不会发生cache缺失，因为执行前面的指令时，已经将指令1装入了指令cache。由组索引可知，指令1在1100000组，即96组。

(9)

当N=2000时，数组a占用了8000个字节

因为数组a的起始线性地址为0x0+0x804d000=0x804d000，可知这是一个页的起始地址，

故数组a所占页个数= $\lceil 8000\text{B}/4\text{KB} \rceil = 2$ 个

虚页号分别为0000 1000 0000 0100 1101和0000 1000 0000 0100 1110

因 $4 \times 1200 = 4800 > 4\text{KB}$ ，所以a[1200]在第二个页中