



Computer Organization and Architecture

课后作业讲解

School of Computer Science (National Pilot Software Engineering School)

AO XIONG (熊翱)

xiongao@bupt.edu.cn





第二章



Assignment- 1

2.16: 考虑第 2.5 节中计算平均 CPI 和 MIPS 速率的示例，其结果为 $CPI = 2.24$ 和 MIPS 速率 = 178。现在假设程序可以在 8 个并行任务或线程中执行，每个任务中的指令条数大概相等。该程序在 8 核系统上执行，每个核（处理器）的性能与最初使用的单处理器相同。部件之间的协调和同步为每个任务增加了额外的 25,000 条指令执行。假设每个任务的指令混合与示例中的相同，但由于内存争用，在缓存未命中时存储器访问的 CPI 增加到 12 个周期。

- 计算平均CPI
- 计算相应的MIPS
- 计算加速比因子
- 将实际加速比与阿曼达定律决定的理论加速比进行比较并分析



Assignment- 1

- 计算平均CPI

根据题意，可以得到如下的指令类型、CPI和比例的表

指令类型	CPI	指令占比
算术和逻辑运算	1	60%
Cache命中的取数/存数	2	18%
分支	3	12%
Cache失效的存储器访问	12	10%

依据上表，可以计算得到CPI：

$$CPI = 0.6 + (2 * 0.18) + (4 * 0.12) + (12 * 0.1) = 2.64$$

由于内存访问时间的增加，cpi有所增加



Assignment- 1

- 计算相应的MIPS

$$MIPS = \frac{400}{2.64} = 152,$$

之前的MIPS是178。所以MIPS速率也相应下降



Assignment- 1

• 计算加速比因子

- 加速系数是执行时间的比率。使用方程式2.2，我们计算执行时间如 $T = I_c / (MIPS * 10^6)$
- 对于单处理器情况是 $T_1 = (2 * 10^6) / (178 * 10^6) = 11.24ms$
- 对于8个处理器，有额外的开销。每个处理器执行200万条指令中的1/8加上25000条开销指令。在这种情况下，8个处理器中的每个处理器的执行时间为：

$$- T_s = \frac{\frac{2*10^6}{8} + 0.025*10^6}{152*10^6} = 1.8ms$$

- 因此

$$Speedup = \frac{\text{time to excute program on a single processor}}{\text{time to execute program on } N \text{ parallel processors}} = \frac{11.24}{1.8} \\ = 6.24$$



Assignment- 1

- **将实际加速比与阿曼达定律决定的理论加速比进行比较并分析**

分析如下：理论上并行系统的加速比跟可以并行的指令比例有关。但实际上并行系统有两个低效之处。

首先，添加了额外的指令来协调线程。

第二，内存访问存在竞争。

假设代码中可并行的部分是 $f = 1$ ，那么在这种情况下，阿姆达尔定律简化为加速比 = $N = 8$ 。但是，在实际上存在一个调度开销，第二个内存访问的竞争，所以实际的加速比肯定达不到理论值。本例中，实际的加速比是理论加速比的75%左右。



第四章



Assignment- 1

4.7: Intel80486有一个统一的片内cache, 其容量为8KB, 采用四路组相联结构, 每块包含4个32位字。cache组织成128组, 每行有一个“行有效位”与另外3个位B0,B1,B2(LRU) 位。在cache未命中时, 80846在一个总线存储器读周期从主存读取一个16字节的行。画出简化的cache框图, 并说明地址的不同字段是如何转换的。

Intel80486有32位地址线, 按字节寻址

Cache总共128组, 组号字段用 $\log_2 128 = 7$ 位表示

Cache每行包含 $4 * 4 = 16$ 个Byte

字字段用 $\log_2 16 = 4$ 位表示

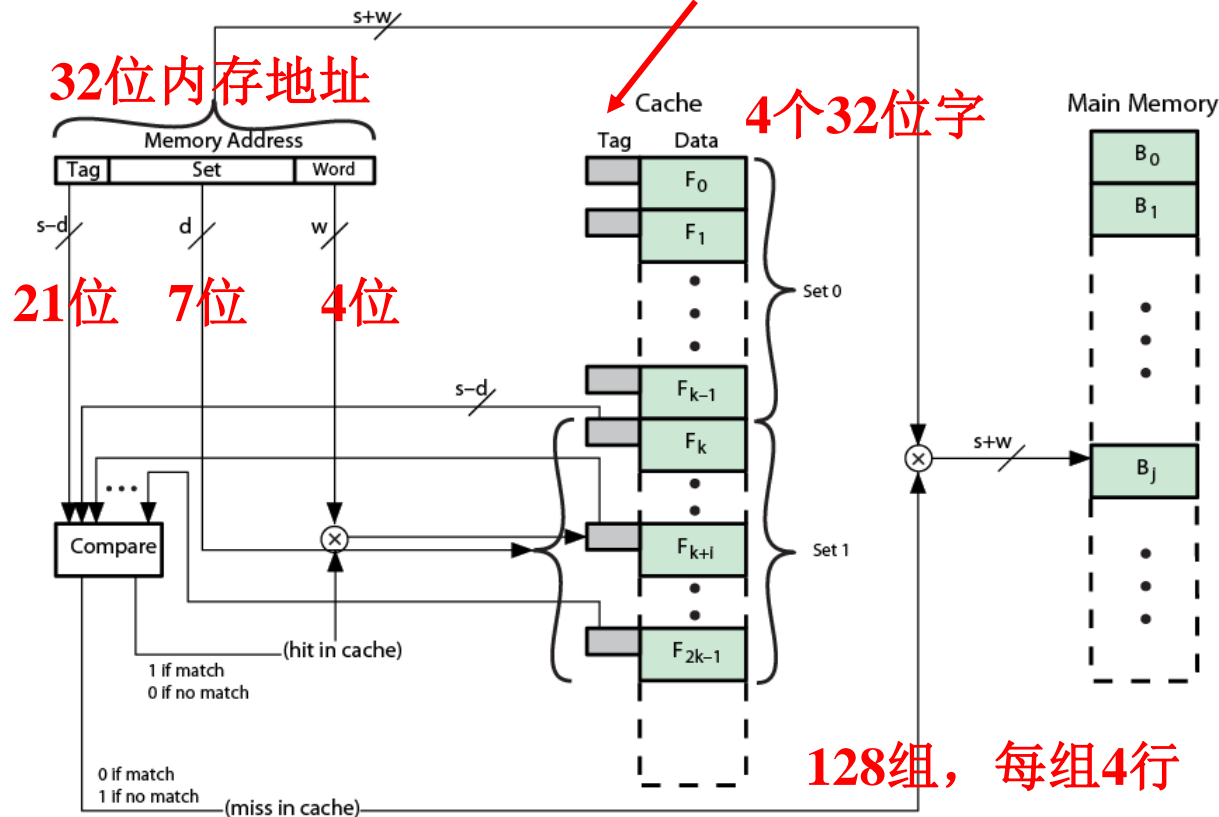
标签字段用 $32 - 4 - 7 = 21$ 位表示



Assignment- 1

32位地址由21位标签字段、7位组号字段和4位字字段组成。高速缓存中的每组包括3个LRU位和4行。每行由4个32位字、一个有效位和一个21位标签组成

行有效位和3个LRU位，21个主存标记位





Assignment- 2

4.10: 一个组相联cache, 每块大小为4个16位字, 组大小为2, cache总共能容纳4096个字。可缓存的主存容量为64K*32位。设计一种cache的结构, 并说明如何转换处理器的地址。

Cache的组数为 $\frac{4096}{4 * 2} = 512$ 组

组号寻址位数为 $\log_2 512 = 9$ 位

每行包括 $4 * 2 = 8$ 个Byte

行内寻址位数为 $\log_2 8 = 3$ 位

- Cache, 4096个字, 每行4个16位字, 每组2行

内存64K*32位

Cache行号	行内字号
0	0, 1, 2, 3
1	
2	
3	
4	
.....	
1022	
1023	

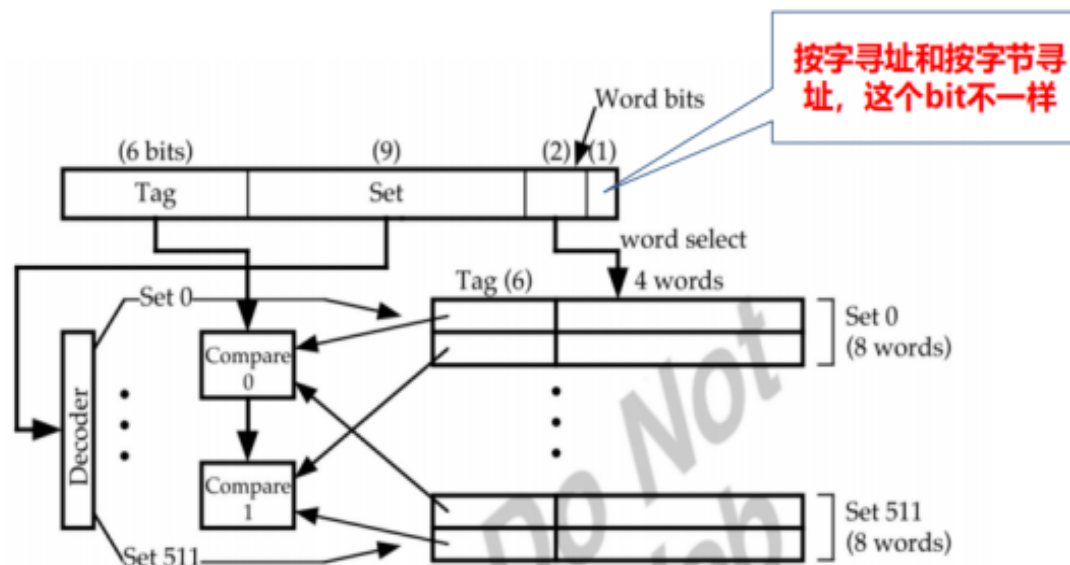
- 总共1024行
- 每行4个字, 每个字是16个bit
- Cache总共512个组

内存	行内字号
0	0, 1
1	
2	
3	
4	
.....	
65534	
65535	

- 内存块对应cache行, 所以内存总共有32k个块
- 如果按字寻址, 总共有128k字, 需要17位地址
- 如果按字节寻址, 总共有256k字节, 需要18位地址

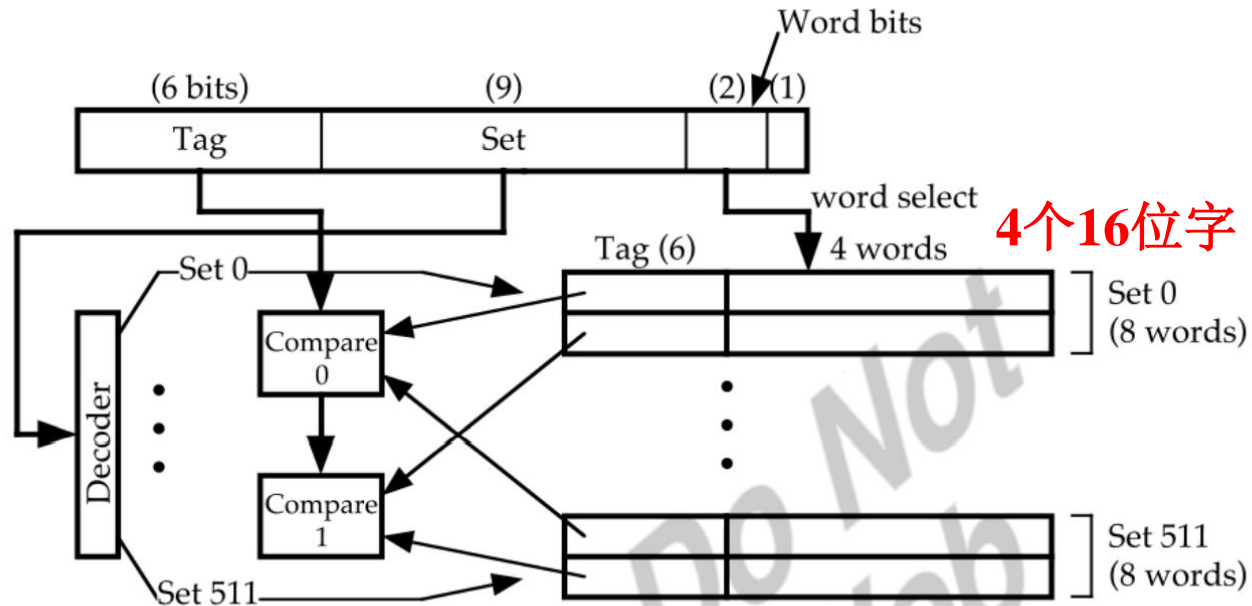
- 如果内存按字寻址, 也就是以字位单位, 那么内存总地址位: 17位地址。
- 块内地址为2位。内存总共32k块, cache组为512组, 所以cache的组号需要9位
- 标记位为: $17 - 2 - 9 = 6$ 位

- 如果内存按字节寻址, 那么内存总地址位: 18位地址。
- 一个块内有8个字节, 块内地址为3位。内存总共32k块, cache组为512组, 所以cache的组号需要9位
- 标记位为: $18 - 3 - 9 = 6$ 位





Assignment- 2



4个16位字

512组，每组2行

Cache的行大小 = 4个16位字 = 2个32位双字

cache大小为4086个16位字，行数为 $\frac{4086}{4} = 1024$ 行； $K = 2$ ，组数为 $\frac{1024}{2} = 512$ 组，需要9位组地址

主存按字节寻址，总容量是64K*32位，也就是256K字节，地址位数18位

Cache中一行是8个字节，所以行内地址需要3位。地址的tag为需要6位。



Assignment- 3

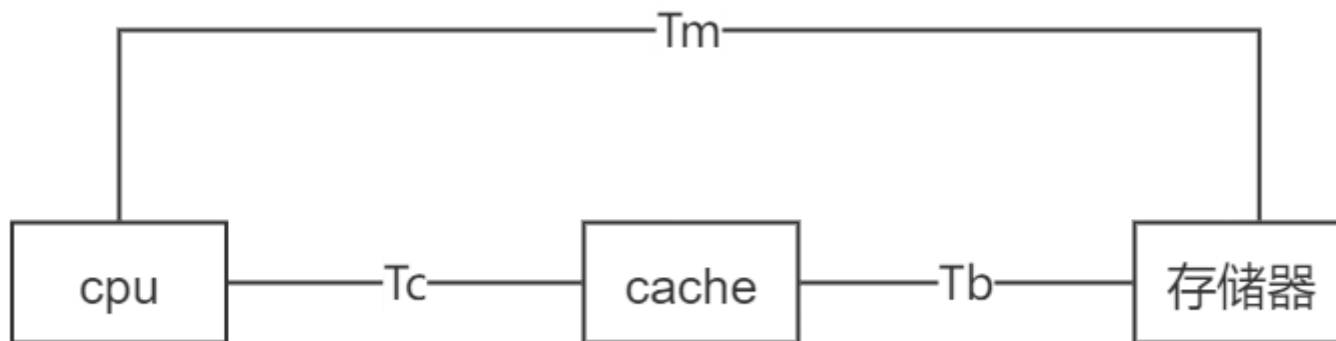
4.26: 一个单级cache系统的读操作性能可用下式来表征:

$$T_a = T_c + (1 - H)T_m$$

其中, T_a 是平均存取时间, T_c 是cache存取时间, T_m 是存储器存取时间(主存到处理器寄存器), H 是命中率。为简单起见, 假定字是并行装入到cache和处理器寄存器的。上式是与等式4.2相同的格式。

(a) 定义: T_b =在cache和主存之间传送一行的时间, W =在所有访问中, 写访问占的比率。试使用写直达策略, 改写上式使之既考虑到读也考虑到写。

(b) 定义 Wb 为cache中的一行已被修改的概率, 试为写回策略提供一个 T_a 等式。





Assignment- 3

(a) W 为写访问占的比率。

1. 对于读的情况，对于不命中的时候，需要同时将数据读到寄存器和cache。而寄存器的速度更快，所以，读的性能是：

$$T_r = HT_c + (1-H) (T_c + T_b) = T_c + (1-H) T_b$$

2. 对于写的情况，cache命中时，写访问的时间为 T_m ，未命中时写访问的时间为 $T_m + T_b$ ，也就是需要先写内存，然后再读到cache中来。

$$\text{因此 } T_a = W * [H * T_m + (1 - H) * (T_m + T_b)] + (1 - W) * [T_c + (1 - H)T_m]$$

$$= W * [(1 - H) * T_b + T_m] + (1 - W) * [T_c + (1 - H)T_b]$$

$$= T_c + (1-H) T_b + W (T_m - T_c)$$



Assignment- 3

(b) 在写回策略的 T_a 计算时，可以这样考虑。如果下一个读的数据在cache的时候，那么它的访问是 T_c 。如果没有命中的时候，需要将cache中的一个块替换出去，然后将需要查询的块替换到cache中。替换出去的块，需要根据是否修改进行写存储器。替换cache之后，还需要一个 T_c ，到CPU中

所以：

$$\begin{aligned} T_a &= H T_c + (1 - H)(W_b T_b + T_b + T_c) \\ &= T_c + (1 - H)(1 + W_b) T_b \end{aligned}$$



第五章



Assignment- 1

5.8 使用容量为64*1位的SRAM芯片设计一个总容量为8192位的16位存储器。要求给出芯片在存储器板上的阵列配置，画出为存储器分配最低地址空间所要求的输入和输出信号，并且该设计应既能满足以字节存取，又能满足以16位字存取。

$$\text{需要SRAM} \frac{8192}{64} = 128 \text{块}$$

因为允许16位访问，将SRAM阵列设置为16列， $\frac{128}{16} = 8$ 行

每个SRAM64位，需要 $\log_2 64 = 6$ 位做位选择

因为允许按字节访问，需要1位做16位数据中的字节选取

寻址空间为 $8192/8=1024$ ，需要10个地址位

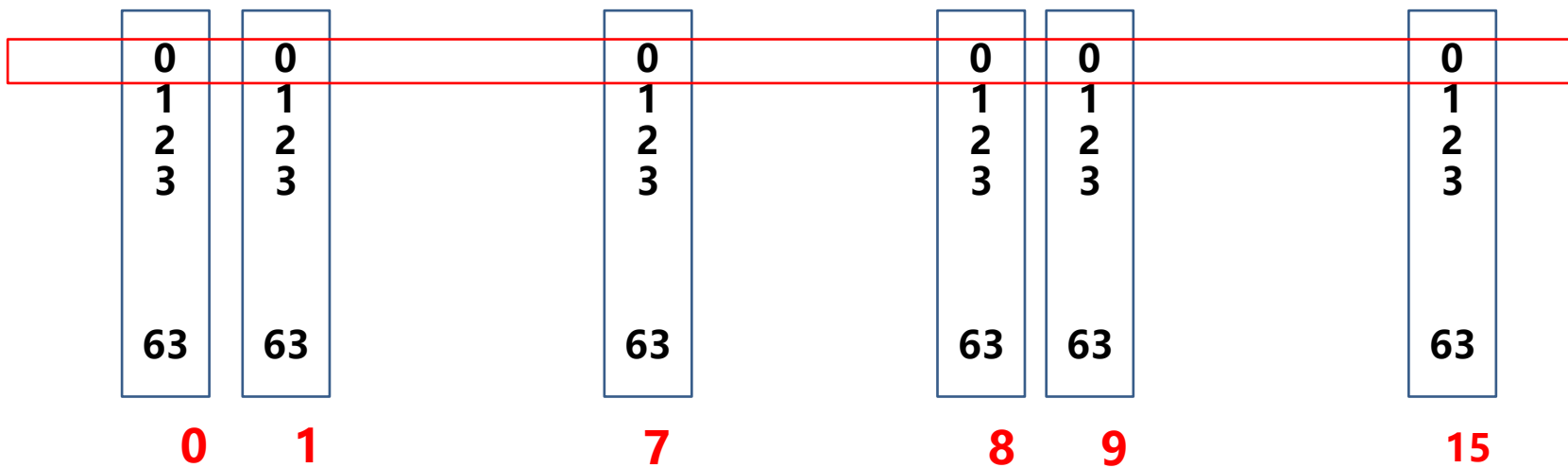


Assignment- 1

第1组

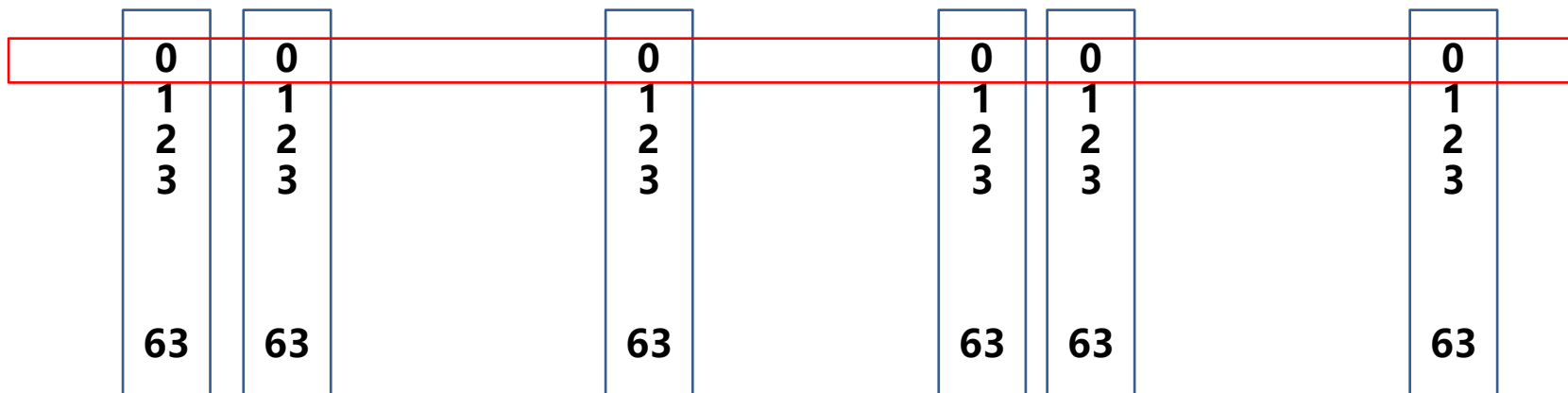
16位字的低位

16位字的高位



第8组

16位字的低位







Assignment- 1

5.11 假设存放在存储器中的一个8位数据字是11000010，请使用汉明码算法确定需要哪些校验位与此数据字一起存放在存储器中，并说明解题步骤。

Position	12	11	10	9	8	7	6	5	4	3	2	1
Bits	D8	D7	D6	D5	C8	D4	D3	D2	C4	D1	C2	C1
Block	1	1	0	0		0	0	1		0		
Codes	1100	1011						0101				



Assignment- 1

校验位为8, 4, 2, 1

第8位由第12, 11, 10, 9位异或决定

第4位由第12, 7, 6, 5位异或决定

第2位由第11, 10, 7, 6, 3位异或决定

第1位由第11, 9, 7, 5, 3位异或决定

校验码为0, 0, 1, 0

3	0011	0
5	0101	1
6	0110	0
7	0111	0
9	1001	0
10	1010	0
11	1011	1
12	1100	1



第六章



Assignment- 1

6.4 考虑一个有8个面的磁盘驱动器，每面有512个磁道，每道上有64个扇区，扇区大小为1KB。平均寻道时间是8ms,道间移动时间是1.5ms,磁盘转速为3600rpm。可以读取同一柱面上的连续磁道而磁头不需要移动。

(a)磁盘容量是多少？

(b)平均存取时间是多少？假设某文件被存储在连续柱面的连续扇区和连续磁道上，起始位置为柱面i上第0道的第0号扇区。

(c)估计传送5MB大小的文件所需要的时间。

(b)突发传送率是多少？



Assignment- 1

(a) 磁盘容量为

$$8 * 512 * 64 * 1KB = 256MB$$

(b)

平均寻道时间 $8ms$

$$\text{旋转延迟时间} \frac{60}{3600 * 2} * 1000 = 8.3ms$$

$$\text{平均存取时间为} 8 + 8.3 = 16.3ms$$



Assignment- 1

(c) 5MB需要 $\frac{5MB}{64KB} = 80$ 个磁道

由于总共有8个盘，所以总共是 $\frac{80}{8} = 10$ 个柱面

传送文件所需时间主要包括寻道时间，旋转延迟，读取时间，道间移动时间

读取时间为每个柱面 $\frac{60}{3600} * 8 = 133.3ms$

传送5MB大小文件所需时间为

$$8 + (8.3 + 133.3) + 9 * (8.3 + 133.3 + 1.5) = 1437.5ms$$

(d) 突发传送率为

$$\frac{3600}{60} * 64 * 1KB = 3.84MB/S$$



第七~八章



Assignment- 1

7.13考虑一个系统,其总线周期为500ns。无论是从处理器到DMA模块,还是从DMA模块到处理器,总线控制的传递都用250ns。一个I/O设备使用DMA方式,其数据传输率是50KB/s。数据每次传送一个字节。

(a) 若使用突发式DMA,即数据块传送之前DMA模块获得总线控制权,并一直维持对总线的控制,直到整个数据块传输完毕。当传送128字节的数据块时,设备占用总线多长的时间?

(b) 若使用周期窃取式DMA,重复上问。



Assignment- 1

(a) 数据传输需要 $\frac{128B}{50KB/S} = 2.56ms$

传输开始和结束时分别需要250ns进行总线控制的传递

250ns相对于2.56ms可以忽略

因此使用突发式DMA设备占用总线约2.56ms

(b) 使用周期窃取DMA时传输一个字节需要占用的总线是： $250ns + 250ns + 500ns = 1\mu s$

传送128字节的数据块时设备占用总线约 $128\mu s$



Assignment- 1

8.6假设处理器当前正在执行的进程的页表如下,所有数据是十进制,用数字表示每个事情均从0开始,所有地址都是存储器的字节地址,一个页的大小为1024B。

(a)准确描述CPU生成的虚拟地址如何转换成主存的物理地址。

(b)虚拟地址:(I)1052,(II)2221,(III)5499对应的物理地址是什么(不考虑页故障)?

Virtual page number	Valid bit	Reference bit	Modify bit	Page frame number
0	1	1	0	4
1	1	1	1	7
2	0	0	0	—
3	1	0	0	2
4	0	0	0	—
5	1	0	1	0



Assignment- 1

(a) 虚拟地址由虚页号和页内偏移组成，确定虚页号后通过页表将虚页号映射为页帧号，将页帧号与页内偏移组合得到物理地址。

(b) 虚拟地址和物理地址的转换

- I. 因为一个页的大小为1024B， $1024 < 1052 < 2048$ ，可知1052的虚页号为1，页内偏移为 $1052 - 1024 = 28$ ，通过页表映射，页帧号为7，物理地址为 $1024 * 7 + 28 = 7196$
- II. 同理，2221的虚页号为2，查页表可知，页缺失。
- III. 5499的虚页号为5，页内偏移为 $5499 - 1024 * 5 = 379$ ，查页表可知映射后的页帧号为0，物理地址为 $0 * 1024 + 379 = 379$



第九章



Assignment- 1

9.23 以IEEE32位浮点格式表示如下的数：

- (a) -5
- (b) -6
- (c) -1.5
- (d) 384
- (e) $1/16$
- (f) $-1/32$

IEEE32位浮点格式包括1位符号位S,8位阶码E,23位有效数字M

(a). 以(a)为例，因为-5是负数，符号位S为1；5的二进制表示为101，可以表示为 $1.01 * 2^2$ ，阶码E为 $2+127=129$ ，有效数字M为1.01省略小数点前的1。因此-5以IEEE32位浮点格式表示为

1 10000001 010000000000000000000000

(b). 1 10000001 100000000000000000000000



Assignment- 1

(c). 1.5可以表示为 $1.1 * 2^0$,符号位S为1, 阶码E为 $0+127=127$, 有效数字M为1.1省略小数点前的1。因此-1.5的IEEE32位浮点格式表示为

1 01111111 100000000000000000000000

(d). 384可以表示为 $1.1 * 2^8$, IEEE32位浮点格式表示为

0 10000111 100000000000000000000000

(e). $1/16$ 可以表示为 $1.0 * 2^{-4}$, IEEE32位浮点格式表示为

0 01111011 000000000000000000000000

(f). $1/32$ 可以表示为 $1.0 * 2^{-5}$, IEEE32位浮点格式表示为

0 01111010 000000000000000000000000



第十一章



Assignment- 1

11.2. 若存于程序计数器中的地址标记为 $X1$ ，存于 $X1$ 中的指令的地址部分(操作数引用)是 $X2$ ，执行此指令所需的操作数存于地址为 $X3$ 的存储器字中。变址寄存器有值 $X4$ 。若此指令的寻址方式是

- (a) 直接寻址
- (b) 间接寻址
- (c) PC相对寻址
- (d) 变址寻址

那么 $X1$ ， $X2$ ， $X3$ ， $X4$ 应该如何组合，从而得到需要的地址？



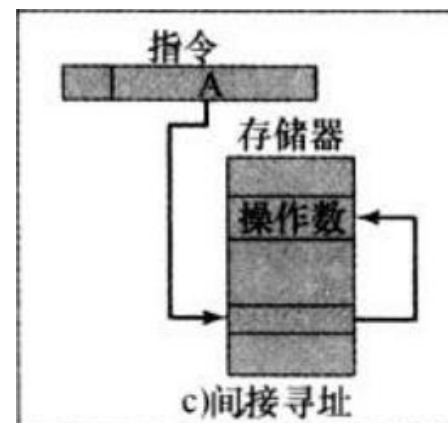
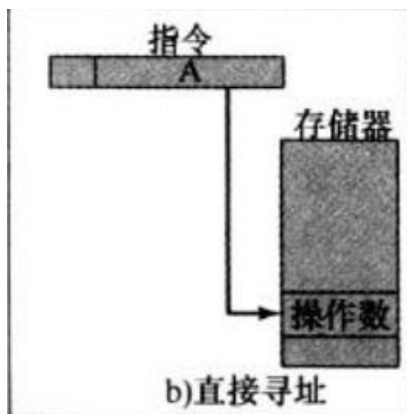
Assignment- 1

PC=X1

指令为: Opcode X2

(a)对于直接寻址, 指令的地址码部分就是操作数在存储器中的地址, 因此 $X3 = X2$ 。

(b)对于间接寻址, 指令的地址码部分的地址不是操作数的实际地址, 它里面的数才是操作数的物理地址。所以 $X3 = (X2)$ 。 ($X2$) 表示的是 $X2$ 存储单元的值。

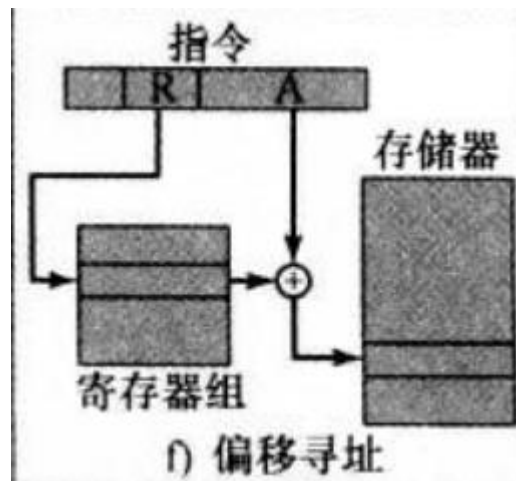




Assignment- 1

(c) PC相对寻址是基址寻址的变种。将基址寄存器改为程序计数器PC，指令的地址码部分给出偏移量，因此 $X3 = X1 + X2 + 1$ 。加1是因为这个指令执行的时候，PC已经+1了。

(d) 在变址寻址中，变址寄存器的内容可以改变（作为偏移量），而指令的地址码部分保持不变（作为基地址），因此 $X3 = X2 + X4$ 。





Assignment- 1

11. 13假定有一个面向栈的处理器，包括有PUSH和POP栈操作。算术运算自动涉及栈顶的1或2个元素。开始时栈为空。下述指令执行后栈中保留下来的栈元素是什么？

PUSH 4

PUSH 7

PUSH 8

ADD

PUSH 10

SUB

MUL



第十二章



Assignment- 1

12.10.某时钟频率为2.5GHz的非流水式处理器，其平均CP1 (每指令周期数)是4。此处理器的升级版本引入了5段流水。然而，由于如锁存延迟这样的流水线内部延迟，使新版处理器的时钟频率必须降低到2GHz。

(a)对典型程序，新版处理器所实现的加速比是多少？

(b)新、旧两版处理器的MIPS速率是多少？



Assignment- 1

(a) 令执行指令数为 n ，指令流水的段数 $k=5$

假设两个处理器的主频相同，那么不采用流水线的处理器，总执行的指令段数为 $5n$ ，花费的时间为 $5n$ 个时间单位，每个时间单位为执行1个阶段的时间。

采用流水线的处理器，执行 n 个指令需要花费的总时间为 $n+5-1$ 个时间单位。时间单位和上面的相同。

所以如果主频相同，加速比 $=5n/(n-4)$

但是因为主频降低了，所以加速比 $=(2/2.5)*5n/(n-4)$



Assignment- 1

(b). 旧版处理器的MIPS速率为 $\frac{2.5G}{4} = 625\text{MIPS}$

新版处理器的情况下，当指令足够多时，每一个时钟周期结束将有一个指令被完成，因此MIPS速率为2000MIPS