

## 第七章 指令流水线

### 2. 简单回答下列问题。(参考答案略)

- (1) 流水线方式下,一条指令的执行时间缩短了还是加长了? 程序的执行时间缩短了还是加长了? 为什么?
  - (2) 具有什么特征的指令集易于实现指令流水线?
  - (3) 流水线处理器中时钟周期如何确定? 单条流水线处理器的CPI为多少? 每个时钟周期一定有一条指令完成吗? 为什么?
  - (4) 流水线处理器的控制器实现方式更类似于单周期控制器还是多周期控制器?
  - (5) 为什么要在各流水段之间加寄存器? 各流水段寄存器的宽度是否都一样? 为什么?
  - (6) 你能列出哪几种流水线被阻塞的情况? 你知道硬件和软件是如何处理它们的吗?
  - (7) 超流水线和多发射流水线的主要区别是什么?
  - (8) 静态多发射流水线和动态多发射流水线的主要区别是什么?
  - (9) 为什么说Pentium 4是“CISC壳、RISC核”的体系结构?
3. 假定在一个五级流水线(如P.205图7.1所示)处理器中,各主要功能单元的操作时间为: 存储单元: 200ps; ALU和加法器: 150ps; 寄存器堆读口或写口: 50ps。若执行阶段EX所用的ALU操作时间缩短20%,则能否加快流水线执行速度? 如果能的话,能加快多少? 如果不能的话,为什么? 若ALU操作时间增加20%,对流水线的性能有何影响? 若ALU操作时间增加40%,对流水线的性能有何影响?

#### 参考答案:

- a. ALU操作时间缩短20%不能加快流水线指令速度。因为存储单元的时间为200ps, 所以流水线的时钟周期不会因为ALU操作时间的缩短而变短。
- b. ALU操作时间延长20%时, 变为180ps, 比200ps小, 对流水线性能没有影响;
- c. ALU操作时间延长40%时, 变为210ps, 比200ps大, 所以, 流水线的时钟周期将变为210, 其效率降低了 $(210-200)/200=5\%$ 。

4. 假定某计算机工程师想设计一个新CPU,一个典型程序的核心模块有一百万条指令,每条指令执行时间为100ps。
- (1) 在非流水线处理器上执行该程序需要花多长时间?
  - (2) 若新CPU是一个20级流水线处理器,执行上述同样的程序,理想情况下,它比非流水线处理器快多少?
  - (3) 实际流水线并不是理想的,流水段间数据传送会有额外开销。这些开销是否会影响指令执行时间(Instruction latency)和指令吞吐率(Instruction throughput)?

#### 参考答案:

- (1) 非流水线处理器上执行该程序的时间为:  $100\text{ps} \times 10^6 = 100\mu\text{s}$ .
- (2) 若在一个20级流水线的处理器上执行,理想情况下,每个时钟周期为:  $100/20=5\text{ps}$ ,

所以，程序执行时间为  $5 \times 10^6 = 5\mu s$ ，快  $100/5=20$  倍。

(3) 流水线段之间数据的传递产生的额外开销，使得一条指令的执行时间被延长，即影响 **Instruction latency**；同时也拉长了每个流水段的执行时间，即影响 **Instruction throughput**。

5. 假定最复杂的一条指令所用的组合逻辑分成 6 块，依次为 A~F，其延迟分别为 80ps、30ps、60ps、50ps、70ps、10ps。在这些组合逻辑块之间插入必要的流水段寄存器就可实现相应的指令流水线，寄存器延迟为 20ps。理想情况下，以下各种方式所得到的时钟周期、指令吞吐率和指令执行时间各是多少？应该在哪里插入流水线寄存器？

- (1) 插入一个流水段寄存器，得到一个两级流水线
- (2) 插入两个流水段寄存器，得到一个三级流水线
- (3) 插入三个流水段寄存器，得到一个四级流水线
- (4) 吞吐量最大的流水线

参考答案：

- (1) 两级流水线的平衡点在 C 和 D 之间，其前面一个流水段的组合逻辑延时为  $80+30+60=170ps$ ，后面一个流水段的组合逻辑延时为  $50+70+10=130ps$ 。这样每个流水段都以最长延时调整为  $170+20=190ps$ ，故时钟周期为  $190ps$ ，指令吞吐率为  $1/190ps=5.26GOPS$ ，每条指令的执行时间为  $2 \times 190=380ps$ 。
- (2) 两个流水段寄存器分别插在 B 和 C、D 和 E 之间，这样第一个流水段的组合逻辑延时为  $80+30=110ps$ ，中间第二段的时延为  $60+50=110ps$ ，最后一个段延时为  $70+10=80ps$ 。这样每个流水段都以最长延时调整为  $110+20=130ps$ ，故时钟周期为  $130ps$ ，指令吞吐率为  $1/130ps=7.69GOPS$ ，每条指令的执行时间为  $3 \times 130=390ps$ 。
- (3) 三个流水段寄存器分别插在 A 和 B、C 和 D、D 和 E 之间，这样第一个流水段的组合逻辑延时为  $80ps$ ，第二段时延为  $30+60=90ps$ ，第三段时延为  $50ps$ ，最后一段延时为  $70+10=80ps$ 。这样每个流水段都以最长延时调整为  $90+20=110ps$ ，故时钟周期为  $110ps$ ，指令吞吐率为  $1/110ps=9.09GOPS$ ，每条指令的执行时间为  $4 \times 110=440ps$ 。
- (4) 因为所有组合逻辑块中最长延时为  $80ps$ ，所以，达到最大可能吞吐率的划分应该是以一个流水段延时为  $80ps+20ps$  来进行，因此，至少按五段来划分，分别把流水段寄存器插在 A 和 B、B 和 C、C 和 D、D 和 E 之间，这样第一段的组合逻辑延时为  $80ps$ ，第二段为  $30ps$ ，第三段为  $60ps$ ，第四段为  $50ps$ ，最后一段为  $70+10=80ps$ 。这样每个流水段都以最长延时调整为  $80+20=100ps$ ，故时钟周期为  $100ps$ ，指令吞吐率为  $1/100ps=10GOPS$ ，每条指令的执行时间为  $5 \times 100=500ps$ 。

6. 以下指令序列中，哪些指令对发生数据相关？假定采用“取指、译码/取数、执行、访存、写回”五段流水线方式，那么不用“转发”技术的话，需要在发生数据相关的指令前加入

几条 `nop` 指令才能使这段程序避免数据冒险？如果采用“转发”是否可以完全解决数据冒险？不行的话，需要在发生数据相关的指令前加入几条 `nop` 指令才能使这段程序不发生数据冒险？

```
add $s3, $s1, $s0
sub $t2, $s0, $s3
lw $t1, 0($t2)
add $t1, $t1, $t2
```

参考答案：

发生数据相关的有：第 1 和 2 间关于 `$s3`、第 2 和 3 间关于 `$t2`、第 2 和 4 间关于 `$t2`、第 3 和 4 间关于 `$t1`。

不进行“转发”处理的话，需要分别在第 2、3、4 条指令前加三条 `nop` 指令才能避免数据冒险。而通过“转发”可以避免 1 和 2、2 和 3、2 和 4 间的数据相关；但第 3 和 4 间是 `load-use` 数据相关，所以无法用“转发”消除冒险，因此，需在第 4 条指令前加入一条 `nop` 指令。

7. 在一个采用“取指、译码/取数、执行、访存、写回”的五段流水线中，若检测结果是否为“零”的操作在执行阶段进行，则分支延迟损失时间片（即分支延迟槽）为多少？以下一段指令序列中，哪些指令执行时会发生流水线阻塞？各需要阻塞几个时钟周期？

```
loop: add $t1, $s3, $s3
      add $t1, $t1, $t1
      add $t1, $t1, $s6
      lw $t0, 0($t1)
      bne $t0, $s5, Exit
      add $s3, $s3, $s4
      j Loop
```

Exit:

参考答案：

若检测操作在执行阶段进行，则分支延迟损失时间片（即分支延迟槽）为 2。

`bne` 指令执行时会发生流水线阻塞，阻塞 2 个时钟周期。`j` 指令如果在译码阶段就根据译码结果计算跳转目标地址，那么 `j` 指令会阻塞 1 个时钟周期，若在执行阶段计算，则要阻塞 2 个时钟周期。

8. 假定有一个程序共 1000 条指令，其指令序列为“`lw, add, lw, add, ...`”。`add` 指令仅依赖它前面的 `lw` 指令，而 `lw` 指令也仅依赖它前面的 `add` 指令。

- (1) 在带转发的五段流水线中执行该程序，其 CPI 为多少？  
(2) 在不带转发的五段流水线中执行该程序，其 CPI 为多少？

参考答案：

(1) 因为 **lw** 指令和 **add** 指令之间存在一个 **load-use** 数据冒险，所以每个 **lw** 指令和 **add** 指令之间要有一次流水线阻塞。而 **add** 指令和 **lw** 指令之间的数据冒险可通过数据转发解决。即：CPI 为 1.5

(2) 如果没有转发，而寄存器写口和寄存器读口分别在一个时钟周期的前、后半个周期内工作，则在每条 **lw** 指令和 **add** 指令之间将会有两个阻塞，这样每条指令相当于都要有三个时钟才能完成。即：CPI 为 3

9. 假定在一个带转发功能的五段流水线中执行以下程序段，则可以怎样调整以下指令序列使其性能达到最好？

```
lw      $2, 100($6)
add    $2, $2, $3
lw      $3, 200($7)
add    $6, $4, $7
sub    $3, $4, $6
lw      $2, 300($8)
beq    $2, $8, Loop
```

参考答案：

```
lw      $2, 100($6)
add    $6, $4, $7
add    $2, $2, $3
lw      $3, 200($7)
lw      $2, 300($8)
sub    $3, $4, $6
beq    $2, $8, Loop
```

10. 假设数据通路中各主要功能单元的操作时间为：存储单元：200ps；ALU 和加法器：100ps；寄存器堆读口或写口：50ps。程序中指令的组成比例为：取数 25%、存数 10%、ALU52%、分支 11%、跳转 2%。假设时钟周期取存储器存取时间的一半，MUX、控制单元、PC、扩展器和传输线路等的延迟都忽略不计，则下面的实现方式中，哪个更快？快多少？

- (1) 单周期方式：每条指令在一个固定长度的时钟周期内完成；
- (2) 多周期方式：每类指令时钟数：取数-7，存数-6，ALU-5，分支-4，跳转-4；
- (3) 流水线方式：取指 1、取指 2、取数/译码、执行、存取 1、存取 2、写回 7 段流水线；没有结构冒险；数据冒险采用“转发”技术处理；**load** 指令与后续各指令之间存在依赖关系的概率分别  $1/2$ 、 $1/4$ 、 $1/8$ 、...；分支延迟损失时间片为 2，预测准确率为 75%；不考虑异常、中断和访问失效引起的流水线冒险。

参考答案：

单周期：存储器操作变为两个时钟周期后，其数据通路的时钟周期不变，为 600ps

多周期:  $CPI=0.25 \times 7 + 0.10 \times 6 + 0.52 \times 5 + 0.11 \times 4 + 0.02 \times 4 = 5.47$

存储器操作变为两个时钟周期后, 多周期数据通路的时钟周期为 100ps,

故一条指令的执行时间为  $100 \times 5.47 = 547 \text{ ps}$

流水线: 存储器操作变为两个时钟周期后, 其流水线包含了 7 个阶段.

对于 **beq**, 若预测正确, 则为 1 个周期, 若预测错误, 则为 3 个周期 (与原五段流水线相比, 多一个取指周期, 多阻塞了 1 个周期), 故  $CPI=1/4 \times 3 + 3/4 \times 1 = 1.5$

对于 **load**, 随后第一条则为 3 个 (阻塞 2 个) 周期; 随后第二条则为 2 个 (阻塞 1 个) 周期, 以后的指令都不需要阻塞, 故  $CPI=1/2 \times 3 + 1/2 \times 1/4 \times 2 + 3/8 \times 1 = 2.125$

对于 **ALU** 指令, 随后的数据相关指令都可通过转发解决, 故  $CPI=1$

对于 **Store** 指令, 不会发生数据冒险, 故  $CPI=1$

对于 **Jump** 指令, 总要等到译码结束才能确定转移地址, 故  $CPI=3$

平均 **CPI** 为:  $2.125 \times 25\% + 1 \times 10\% + 1 \times 52\% + 1.5 \times 11\% + 3 \times 2\% = 1.38$

所以,  $N$  条指令的执行时间为  $1.38 \times 100 \times N = 138N(\text{ps})$

流水线比单周期快  $600/138=4.38$  倍。