

入气泡的平均数。每种处罚都是由该种原因引起的插入气泡的总数(C_b 的一部分)除以执行指令的总数(C_i)。

为了估计每种处罚,我们需要知道相关指令(加载、条件转移和返回)的出现频率,以及对每种指令特殊情况出现的频率。对 CPI 的计算,我们使用下面这组频率(等同于[44]和[46]中报告的测量值):


- 加载指令(mrmovq 和 popq) 占有所有执行指令的 25%。其中 20% 会导致加载/使用冒险。
- 条件分支指令占有所有执行指令的 20%。其中 60% 会选择分支,而 40% 不选择分支。
- 返回指令占有所有执行指令的 2%。


因此,我们可以估计每种处罚,它是指令类型频率、条件出现频率和当条件出现时插入气泡数的乘积:

原因	名称	指令频率	条件频率	气泡	乘积
加载/使用	<i>lp</i>	0.25	0.20	1	0.05
预测错误	<i>mp</i>	0.20	0.40	2	0.16
返回	<i>rp</i>	0.02	1.00	3	0.06
总处罚					0.27

三种处罚的总和是 0.27, 所以得到 CPI 为 1.27。

我们的目标是设计一个每个周期发射一条指令的流水线, 也就是 CPI 为 1.0。虽然没有完全达到目标, 但是整体性能已经很不错了。我们还能看到, 要想进一步降低 CPI, 就应该集中注意力预测错误的分支。它们占到了整个处罚 0.27 中的 0.16, 因为条件转移非常常见, 我们的预测策略又经常出错, 而每次预测错误都要取消两条指令。

 **练习题 4.43** 假设我们使用了一种成功率可以达到 65% 的分支预测策略, 例如后向分支选择、前向分支就不选择(BTFNT), 如 4.5.4 节中描述的那样。那么对 CPI 有什么样的影响呢? 假设其他所有频率都不变。

 **练习题 4.44** 让我们来分析你为练习题 4.4 和练习题 4.5 写的程序中使用条件数据传送和条件控制转移的相对性能。假设用这些程序计算一个非常长的数组的绝对值的和, 所以整体性能主要是由内循环所需要的周期数决定的。假设跳转指令预测为选择分支, 而大约 50% 的数组值为正。

- 平均来说, 这两个程序的内循环中执行了多少条指令?
- 平均来说, 这两个程序的内循环中插入了多少个气泡?
- 对这两个程序来说, 每个数组元素平均需要多少个时钟周期?

4.5.10 未完成的工作

我们已经创建了 PIPE 流水线化的微处理器结构, 设计了控制逻辑块, 并实现了处理普通流水线流不足以处理的特殊情况的流水线控制逻辑。不过, PIPE 还是缺乏一些实际微处理器设计中所必需的关键特性。我们会强调其中一些, 并讨论要增加这些特性需要些什么。

1. 多周期指令

Y86-64 指令集中的所有指令都包括一些简单的操作, 例如数字加法。这些操作可以在执行阶段中一个周期内处理完。在一个更完整的指令集中, 我们还将实现一些需要更为复杂操作的指令, 例如, 整数乘法和除法, 以及浮点运算。在一个像 PIPE 这样性能中等