

泡。有了这个对流水线寄存器设计的小扩展，我们就能用组合逻辑、时钟寄存器和随机访问存储器这样的基本构建块，来实现一个完整的、包括所有控制的流水线。

条件	流水线寄存器				
	F	D	E	M	W
处理 ret	暂停	气泡	正常	正常	正常
加载/使用冒险	暂停	暂停	气泡	正常	正常
预测错误的分支	正常	气泡	气泡	正常	正常

图 4-66 流水线控制逻辑的动作。不同的条件需要改变流水线流，或者会暂停流水线，或者会取消部分已执行的指令

4. 控制条件的组合

到目前为止，在我们对特殊流水线控制条件的讨论中，假设在任意一个时钟周期内，最多只能出现一个特殊情况。在设计系统时，一个常见的缺陷是不能处理同时出现多个特殊情况的情形。现在来分析这些可能性。我们不需要担心多个程序异常的组合情况，因为已经很小心地设计了异常处理机制，它能够考虑流水线中其他指令的情况。图 4-67 画出了导致其他三种特殊控制条件的流水线状态。图中所示的是译码、执行和访存阶段的块。暗色的方框代表要出现这种条件必须要满足的特别限制。加载/使用冒险要求执行阶段中的指令将一个值从内存读到寄存器中，同时译码阶段中的指令要以该寄存器作为源操作数。预测错误的分支要求执行阶段中的指令是一个跳转指令。对 ret 来说有三种可能的情况——指令可以处在译码、执行或访存阶段。当 ret 指令通过流水线时，前面的流水线阶段都是气泡。

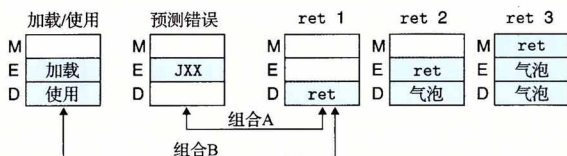



图 4-67 特殊控制条件的流水线状态。图中标明的两对情况可能同时出现

从这些图中我们可以看出，大多数控制条件是互斥的。例如，不可能同时既有加载/使用冒险又有预测错误的分支，因为加载/使用冒险要求执行阶段中是加载指令（`mrmovq` 或 `popq`），而预测错误的分支要求执行阶段中是一条跳转指令。类似地，第二个和第三个 ret 组合也不可能与加载/使用冒险或预测错误的分支同时出现。只有用箭头标明的两种组合可能同时出现。

组合 A 中执行阶段中有一条不选择分支的跳转指令，而译码阶段中有一条 ret 指令。出现这种组合要求 ret 位于不选择分支的目标处。流水线控制逻辑应该发现分支预测错误，因此要取消 ret 指令。

 **练习题 4.37** 写一个 Y86-64 汇编语言程序，它能导致出现组合 A 的情况，并判断控制逻辑是否处理正确。

合并组合 A 条件的控制动作（图 4-66），我们得到以下流水线控制动作（假设气泡或暂停会覆盖正常的情况）：