



图 4-59 转发优先级的说明。在周期 4 中, $\%rdx$ 的值既可以从执行阶段也可以从访存阶段得到。转发逻辑应该选择执行阶段中的值, 因为它代表最近产生的该寄存器的值

练习题 4.33 假设 d_valA 的 HCL 代码中第五和第六种情况(来自写回阶段的两个转发源)的顺序是反过来的。写出一个会运行错误的 Y86-64 程序。请描述错误是如何发生的, 以及它对程序行为的影响。

练习题 4.34 根据提供到流水线寄存器 E 的源操作数 $valB$ 的值, 写出信号 d_valB 的 HCL 代码。

写回阶段的一小部分是保持不变的。如图 4-52 所示, 整个处理器的状态 $Stat$ 是一个块根据流水线寄存器 W 中的状态值计算出来的。回想一下 4.1.1 节, 状态码应该指明是正常操作(AOK), 还是三种异常条件中的一种。由于流水线寄存器 W 保存着最近完成的指令的状态, 很自然地要用这个值来表示整个处理器状态。唯一要考虑的特殊情况是当写回阶段有气泡时。这是正常操作的一部分, 因此对于这种情况, 我们也希望状态码是 AOK:

```
word Stat = [
    W_stat == SBUB : SAOK;
    1 : W_stat;
];
```

3. 执行阶段

图 4-60 展现的是 PIPE 执行阶段的逻辑。这些硬件单元和逻辑块同 SEQ 中的相同, 使用的信号做适当的重命名。我们可以看到信号 e_valE 和 e_dstE 作为转发源, 指向译码阶段。一个区别是标为“Set CC”的逻辑以信号 m_stat 和 W_stat 作为输入, 这个逻辑决定了是否要更新条件码。这些信号被用来检查一条导致异常的指令正在通过后面的流水线阶段的情况, 因此, 任何对条件码的更新都会被禁止。这部分设计在 4.5.8 节中讨论。