

3.9 数据冲突:

RAW: ①②, ①③, ④⑤, ③⑥, ⑦⑧, ⑨⑩, ⑪⑫

WAW: (不考虑①③和④⑤)

WAR: ②③, ①④, ⑦⑧, ④⑩, ⑧⑩

1) 执行流程为: 周期1发射①, 周期5发射②, 周期16发射③, 周期17发射④,

→ 5

→ 16

→ 21

→ 21

周期21发射⑤, 周期22发射⑥, 周期25发射⑦, 周期26发射⑧

→ 24

→ 25

→ 27

→ 28

周期27发射⑨, 周期28发射⑩, 周期29发射⑪, 周期30发射⑫

→ 28

→ 29

→ 30

→ 32

至第33周期发射下一次循环指令①, 则迭代一次需要31个周期

2) 流水线1: 周期1发射①, 周期5发射③, 周期14发射⑤, 周期19发射⑦

→ 5

→ 16

→ 10

→ 14

→ 17

→ 19

→ 21

→ 21

流水线2: 周期5发射②, 周期10发射④, 周期16发射⑥, 周期19发射⑧

周期20发射⑨, 周期21发射⑪

→ 21

→ 21

→ 22

→ 24

迭代一次需要23个周期

周期20发射⑩, 周期22发射⑫

3) fld f2, 0(a0) ① addi a1, a1, 8 ⑦

fld f4, 0(a1) ② fmul.d f10, f8, f2 ⑧

fdiv.d f8, f0, f2 ③ fsd f10, 0(a0) ⑨

fadd.d f4, f0, f4 ④ addi a0, a0, 8 ⑩

fmul.d f2, f6, f2 ⑤ sub x20, x4, a0 ⑪

fsd f4, 0(a1) ⑥ bnez x20, Loop ⑫

周期1发射①, 周期5发射③, 周期6发射⑤, 周期10发射⑦, 周期11发射⑨

↳ 5 ↳ 5 ↳ 16 ↳ 8 ↳ 11 ↳ 10 ↳ 11 ↳ 19 ↳ 21 ↳ 20

周期1发射②, 周期5发射④, 周期8发射⑩, 周期16发射⑧, 周期19发射④

周期20发射⑪

↳ 21 ↳ 23

可将一次迭代需要的周期数减少至22

周期21发射⑫

3.10 fld f4, 0(a0) fld T9, 0(a0)

fmul.d f2, f0, f2 fmul.d T10, f0, f2

fdiv.d f8, f4, f2 fdiv.d T11, T9, T10

fld f4, 0(a1) ⇒ fld T12, 0(a1)

fadd.d f6, f0, f4 fadd.d T13, f0, T12

fsub.d f8, f8, f6 fsub.d T14, T11, T13

fsd f8, 0(a1) fsd T14, 0(a1)

3.11 显式重命名时物理上的寄存器数多于汇编代码中使用的寄存器数, 并通过映射表将两种寄存器的对应关系保存起来, 隐式重命名使用的物理寄存器个数与逻辑寄存器相同, 每个操作数既保存在ROB(重排序缓冲), 又保存在ISA寄存器中,

映射表记录操作数在 ROB 的位置, 以及 ROB 与 ISA 寄存器的值何为最新

优缺点: 显式重命名优点是可以完全解决 WAW 与 WAR 依赖, 提高了乱序执行的灵活度, 缺点是需大量物理寄存器, 提高硬件成本

隐式重命名优点是映射表简单, 利用 ROB 记录结果或中间值, 缺点是每个数据在两个位置中保存, 读取数据的复杂度和功耗更高

实现方式: 显式重命名使用 RAM、CAM 等结构作为映射表, 表项数等于物理/逻辑寄存器数目, 用 free_list 记录寄存器状态

隐式重命名使用 ROB 和 ISA 寄存器同时保存数据, 使用数据总线实现 ROB 与映射表通信, 映射表除记录对应关系外, 还要记录 ROB 和 ISA 寄存器中的最新数据