

CO-P5课上测试攻略

#CPU

#Verilog

ALU计算类 Analysis

这种指令按照经验而言，扩展位置主要在：

1. ALU内部
2. Controller信号
3. 考虑到阻塞的Tnew和Tuse

应该不会动数据通路，也就是说转发的SRCMUX和DSTMUX不用管，维持即可

WorkFlow

1. 规定新指令信号编码，填写控制信号表
2. 根据当前指令的行为，确定Tnew和Tuse，计算类指令大概率和add/sub的指令T一致
3. 根据控制信号表，在Controller中进行解码，扩展，指定控制信号和Tnew_E/Tuse
4. 在 **head.v** 中加入宏
5. 对 **ALU.v** 中加入宏对应操作的实现

B/J + Link型指令

Analysis

指令分类：

1. 条件跳转 + 无条件链接
2. 条件跳转 + 条件链接
3. 条件跳转 + 条件（无条件）链接 + 条件不成立清空延时槽

扩展点：

1. 对CMP元件进行扩展
2. 对于条件链接，扩展流水级寄存器接口，需要将CMP结果进行流水
3. 对于条件链接，扩展Controller单元接口，对于RegWrite信号和特殊的那条指令需要增加判断

```
input wire CMP_res;
output RegWrite;

assign RegWrite = (newInstr & CMP_res) | add
| sub;
```

4. 清空延时槽，需要对D级流水扩展flush信号

首先，在D级我们一定可以通过CMP得知当前到底条件不成立，那么在D级，我们就需要D_CU给我们生成出一个flush信号，代表当前到底请不请空延时槽，同时，还注意到在stall的时候，我们不管flush，因此在D级流水内部，需要考虑这点

```

// Controller.v -- D_CU
Controller(
    // ...
    input check;
);
assign flush = newInstr & ~CMP_res;

// D_REG
// assign real_flush = flush & ~stall;
// 或者刻画优先级即可
always @(posedge clk) begin
    if(reset)
        // reset to all Zero
    else if(stall)
        // reg <= reg
    else if(flush)
        // same as reset
    else
        // fetch F_info
end

```

WorkFlow

1. 对指令分类并进行规划
2. 确定编码，加入 **head.v**

3. 修改CMP行为
 4. 按照常规情况生成控制信号表
 5. 分析指令Tnew_E和Tuse, 扩展Controller.v
 6. 考虑条件链接: 先扩展流水线寄存器加入check接口, 之后在Controller.v内部进行扩展, 在mips.v顶层执行接线
 7. 考虑清空流水槽: 根据CMP结果, 反馈到Controller.v -- D_CU, 并判断生成flush信号, 扩展D级流水接口
-

条件存取类

Analysis

题型分类:

1. DM中取出, 判断是否满足Condition, if → 写regA, else → 写regB
2. DM中取出, 判断是否满足Condition, if → 写regA, else → 不写 (你可以认为是写0号寄存器--统一化)
3. 从DM取出, 写入位置取决于MemData

由于在M阶段, 我们才能知道到底是往哪里写回

需要修改stall信号为“但凡可能要写回的rs/rt”就得阻塞流水

因此我们需要扩展MFSCU元件, 一方面扩展接口 (E_Instr、M_Instr) 判断是否是新指令, 使用Check_M和Check_E标记

```
// 可能的地址为A1_E, A2_E
// 阻塞行为改为:
STALL_RS_E = (Check_M ? (rs_D == A1_E | rs_D
== A2_E) : rs_D == RegAddr_E) & // other...
```

同时，在流水线M级，我们必须确定当前到底是往哪里写，相当在M级先来一个

修改选择信号，需要将condition回传给Controller（其他级condition信号置为1'b0）

修改转发逻辑

该情况下**必须全部改为条件转发**，暴力转发会出现问题
所以相当于我们由于不知道到底往哪里转（换言之，**必须得等写回数据产生了才知道往哪里转**），这一点其实与条件转发逻辑一样，那么干脆直接条件转发

注意一下这里，我们的转发条件是

1. 是新指令，注意是对Opcode和funct进行解析
2. 如果所有**可能写**的构成**阻塞条件**，但是因为不知道到底写谁，所以**不转发**！
3. 必须修改为**条件转发**类型，转发必须保证流水级的**Tnew = 0**