

体系结构_Lab1

PB19000015 贾欣宇

体系结构_Lab1

1. 描述执行一条 XOR 指令的过程（数据通路、控制信号等）。
2. 描述执行一条 BEQ 指令的过程（数据通路、控制信号等）。
3. 描述执行一条 LHU 指令的过程（数据通路、控制信号等）。
4. 如果要实现 CSR 指令 (csrrw, csrrs, csrrc, csrrwi, csrrsi, csrrci)，设计图中还需要增加什么部件和数据通路？给出详细说明。
5. Verilog 如何实现立即数的扩展？
6. 如何实现 Data Memory 的非字对齐的 Load 和 Store？
7. ALU 模块中，默认 wire 变量是有符号数还是无符号数？
8. 简述 BranchE 信号的作用。
9. NPC Generator 中对于不同跳转 target 的选择有没有优先级？
10. Harzard 模块中，有哪几类冲突需要插入气泡，分别使流水线停顿几个周期？
11. Harzard 模块中采用静态分支预测器，即默认不跳转，遇到 branch 指令时，如何控制 flush 和 stall 信号？
12. 0 号寄存器值始终为 0，是否会对 forward 的处理产生影响？

- 由于原图中各信号线存在名称冲突的问题，将原图中部分信号名称进行了一定修改。

1. 描述执行一条 XOR 指令的过程（数据通路、控制信号等）。

IF: 根据 PC，从 Instruction Cache 中取出指令 XOR rd rs1 rs2。

ID: 寄存器堆 Addr1, Addr2 地址分别为 rs1, rs2，读出待异或的值 Reg1OutD 和 Reg2OutD；rd 传给 RegDstD；指令 Inst 传给 Controller Decoder，把 ALUFuncD（选异或），ALUSrc1D（选 Reg1Out），ALUSrc2D（选 Reg2Out），RegWriteD（使能），WBSelectD（在 MEM 段选 ALUResult）传入段寄存器。

EX: ALU 执行异或计算，结果写入 ALUOutE，ResultE 选择 ALUOutE；RegDstE，RegWriteE 写入段寄存器。

MEM: WBDataM 选择 ResultM；RegDstM，RegWriteM 写入段寄存器。

WB: RegWriteW 使能，将结果写回寄存器堆。

2. 描述执行一条 BEQ 指令的过程（数据通路、控制信号等）。

IF: 根据 PC，从 Instruction Cache 中取出指令 BEQ rs1 rs2 Imm。

ID: 寄存器堆 Addr1, Addr2 地址分别为 rs1, rs2，读出待比较的值 Reg1OutD 和 Reg2OutD；指令 Inst 传给 Controller Decoder，把 BRTyped（选 BEQ）传入段寄存器，同时输出 ImmTypeD（选 BEQ 类型处理方式）；Immediate Generate 生成立即数并传入 ImmD，左移一位后与 PCD 相加得到 BRTargetD，传入段寄存器。

EX: Branch Module 计算 Reg1 - Reg2，若结果为 0，BRE 使能，Hazard Module 输出相应控制信号，否则继续执行；将 BRTargetE 传回 IF 段。

不需要用到 MEM 和 WB 段。

3. 描述执行一条 LHU 指令的过程（数据通路、控制信号等）。

IF: 根据 PC, 从 Instruction Cache 中取出指令 `LHU rd rs1 Imm`。

ID: 寄存器堆 Addr1 地址为 rs1, 读出待计算的值 `Reg1OutD`; 指令 `Inst` 传给 Controller Decoder, 把 `ALUFuncD` (选加法), `AluSrc1D` (选 `Reg1Out`), `AluSrc2D` (选 `Imm`), `RegWrted` (使能), `MemLoadD` (使能), `WBSe1sctD` (在 MEM 段选 `MemOut`), `LoadTypeD` (选读半字) 传入段寄存器, 并输出 `ImmTypeD` (选 LHU 类型处理方式); Immediate Generate 生成立即数并传入 `ImmD`。

EX: ALU 计算 `Reg1OutE + ImmE`, 存入 `AluOutE`, `ResultE` 选择 `AluOutE`。

MEM: 从 Data Cache 和 Data Extension 读取地址为 `{ResultM[31:2], 2'b00}` 的值, 无符号扩展后结果写入 `WBdataM`。

WB: 将 `WBdataW` 写回寄存器堆。

4. 如果要实现 CSR 指令 (`csrrw`, `csrrs`, `csrrc`, `csrrwi`, `csrrsi`, `csrrci`), 设计图中还需要增加什么部件和数据通路? 给出详细说明。

`csrrw`: 读后写控制状态寄存器, `t = CSRs[csr]; CSRs[csr] = x[rs1]; x[rd] = t`, 故需要额外的寄存器用于保存 `t`。

`csrrwi`: 立即数读后写控制状态寄存器, `x[rd] = CSRs[csr]; CSRs[csr] = zimm`, 故需要立即数处理单元实现 5 位立即数的扩展。

`csrrs`: 读后置位控制状态寄存器, `t = CSRs[csr]; CSRs[csr] = t | x[rs1]; x[rd] = t`

`csrrsi`: 立即数读后置位控制状态寄存器, `t = CSRs[csr]; CSRs[csr] = t | zimm; x[rd] = t`

`csrrc`: 读后清除控制状态寄存器, `t = CSRs[csr]; CSRs[csr] = t & ~x[rs1]; x[rd] = t`

`csrrci`: 立即数读后清除控制状态寄存器, `t = CSRs[csr]; CSRs[csr] = t & ~zimm; x[rd] = t`

综上, 需要增加额外的寄存器作为状态寄存器; 在 IDEX 段寄存器的输入端需要加入多选器, 确定读的数是来自 CSR 还是原来的寄存器堆; 增加新的立即数处理功能和逻辑控制, 以支持 CSR 指令与原有指令不同的立即数扩展; 在 Controller Decoder 中也要加入 CSR 指令相关的控制逻辑和多选器信号等。

5. Verilog 如何实现立即数的扩展?

无符号扩展: 只需在立即数的剩余高位补零。

有符号扩展: 只需重复立即数的最高位进行扩展。

6. 如何实现 Data Memory 的非字对齐的 Load 和 Store?

Load: 把访存的地址最后两位置为 00, 把整个字读取出来, 根据原地址后两位选择需要在 WB 阶段写回的部分, 并进行无符号扩展。

Store: 把访存的地址最后两位置为 00, 把整个字读取出来, 根据原地址后两位选择需要修改的部分, 再写回 Data Memory。

7. ALU 模块中，默认 wire 变量是有符号数还是无符号数？

无符号数。

8. 简述 BranchE 信号的作用。

若 BranchE 信号为 1 则说明条件转移指令的条件为真，应该发生转移，下条指令地址为 BRTargetE；Hazard Module 也需要进行相应控制。

9. NPC Generator 中对于不同跳转 target 的选择有没有优先级？

JALRTarget 和 BRTarget 的优先级高于 JALTarget。因为若 JALD 使能的同时 JALRE 或 BRE 使能，说明 JALR 指令或 BR 类指令先于 JAL 指令发生，故应优先处理前二者。而 JALRTarget 和 BRTarget 的优先级可以相同，因为二者的使能 JALRE 和 BRE 不可能同时有效。

10. Harzard 模块中，有哪几类冲突需要插入气泡，分别使流水线停顿几个周期？

RAW 类冲突如 Load 和 ALU 指令冲突，在 EX 段产生一个 stall，需要停顿 1 个周期。

控制相关的冲突如跳转，在 ID 段产生一个 flush，需要停顿 1 个周期；条件转移时，在 ID 和 EX 段产生一个 flush，需要停顿 2 个周期。

11. Harzard 模块中采用静态分支预测器，即默认不跳转，遇到 branch 指令时，如何控制 flush 和 stall 信号？

发生转移即 BRE 信号为 1 时，对 IFID，IDEX 段寄存器输出 flush。

12. 0 号寄存器值始终为 0，是否会对 forward 的处理产生影响？

会。否则按照设计，可能使 forward 的 0 号寄存器非零。所以在 forward 中需要加入判断是否为 0 号寄存器，控制多选器的输出，若为 0 号寄存器，则不应进行 forward。