分支地址计算模块说明

|  |  |  |
| --- | --- | --- |
| exmem\_branch(1) | final\_target |  |
| exmem\_condition(3) |  |
| exmem\_target(32) | (32)final\_target |
| exmem\_pc\_4(32) | (1)branch\_state |
| exmem\_if(1) |  |
| exmem\_zf(1) |  |

Final\_target框架图接口说明

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 序号 | 接口名 | 宽度(bit) | 输入/输出 | 作 用 |
| 1 | exmem\_branch | 1 | 输入 | 指示是否为branch系列或者jr指令(跳转指令) |
| 2 | exmem\_condition | 3 | 输入 | 说明该代码branch的跳转条件 |
| 3 | exmem\_target | 32 | 输入 | branch系列指令/jr指令跳转地址 |
| 4 | exmem\_pc\_4 | 32 | 输入 | MEM阶段执行的指令pc+4 |
| 5 | exmem\_if | 1 | 输入 | 由EX阶段ALU输出的Less flag，用于检测跳转条件是否满足 |
| 6 | exmem\_zf | 1 | 输入 | 由EX阶段ALU输出的Zero flag，用于检测跳转条件是否满足 |
| 7 | final\_target | 32 | 输出 | 最终得出的分支跳转地址（pc+4或pc+8或者分支跳转地址）,该分支跳转地址包括branch与jr的跳转地址 |
| 8 | branch\_state | 1 | 输出 | branch系列指令/jr指令是否跳转 |

功能说明：

用于得到真正的分支跳转地址（不跳转或者跳转到的地址），以及判断branch系列指令、jr指令是否满足跳转条件，并输出判断信息输出到Control Unit，Control Unit对判断信息进行处理，并选择此次执行新的PC，判断逻辑如下所示：

if(exmem\_branch == false) begin

branch\_state = false //跳转不成立

final\_target = pc+4

end

if(condition == true) begin

branch\_state = true //branch跳转成立

final\_target = exmem\_target

end

else if(condition == 3’b000) begin // 是jr指令

branch\_state = true //jr跳转成立

final\_target = exmem\_target

end

else begin

branch\_state = false //跳转不成立

final\_target = pc+8

end