

Lab6 综合设计

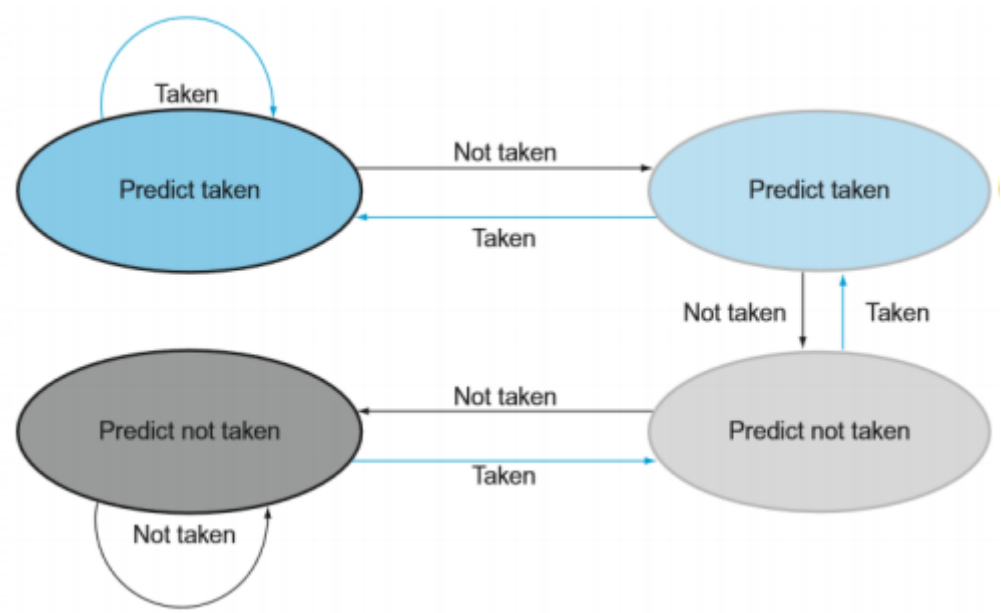
梁峻滔 PB19051175

2021 年 6 月 10 日

动态分支预测

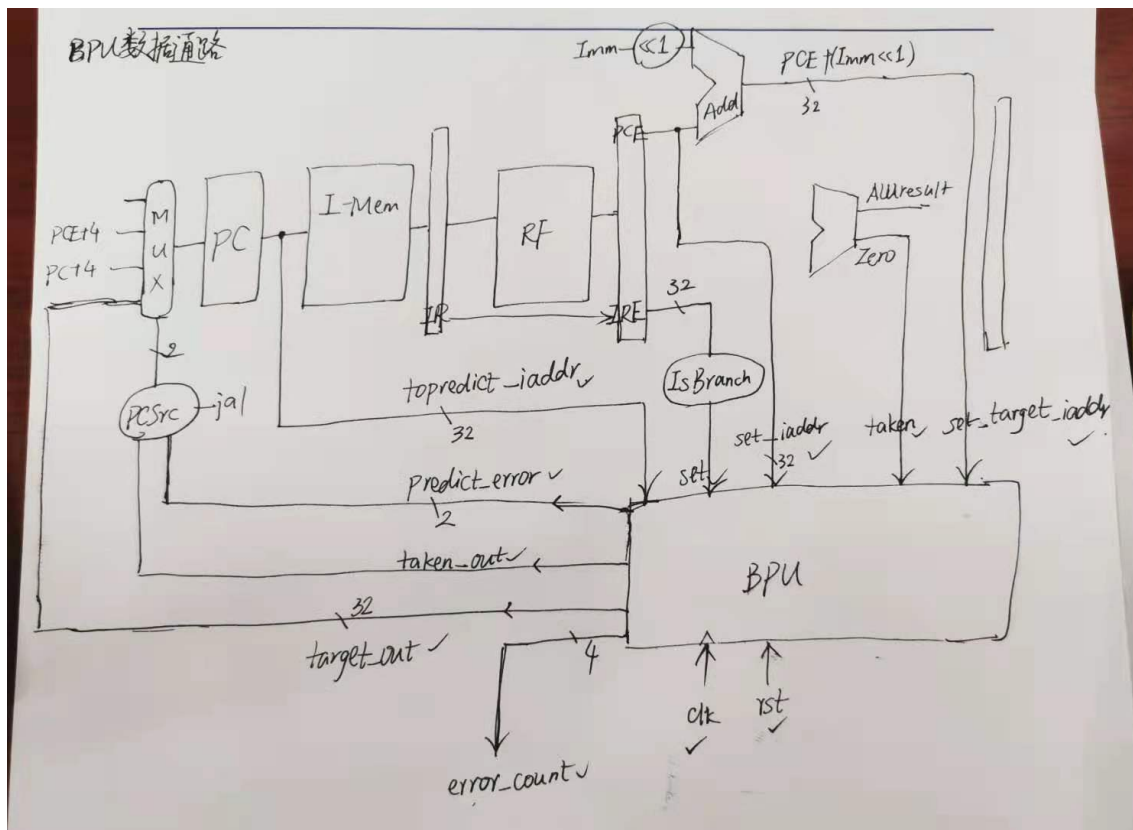
1. 基本思路

分支预测单元(Branch Prediction Unit, BPU)模块内建两个缓存，一个缓存 `reg [1:0]` `counter[(1<<BTB_WIDTH)-1:0]` 用于为每条分支指令设置一个两位的计数器来记录该分支指令的预测状态(Strongly_Taken, Weakly_taken, Weakly_Not_Taken, Strongly_Not_Taken)，一个缓存 `reg [PC_WIDTH-1:0]` `btb[(1<<BTB_WIDTH)-1:0]` 用于为每条分支指令存一个预测的目标地址。两个缓存均用分支指令地址的低10位寻址。



Prediction State	localparam
STRONGLY_TAKEN	2'b11
WEAKLY_TAKEN	2'b10
WEAKLY_NOT_TAKEN	2'b01
STRONGLY-NOT_TAKEN	2'b00

2. BPU数据通路



BPU各端口声明:

Input:

- **topredict_iaddr**: 要预测的指令地址, 来自PC寄存器。
- **set_iaddr**: EX段的指令, 检测它是否是 beq 指令, 如果是则根据它的实际跳转情况(即 taken)设置BPU里的目标地址缓存和预测状态。
- **set**: 如果当前再EX段的指令是beq指令则置1, 否则清0, 需要增加一个段间寄存器IRE和一个判断单元IsBranch。
- **taken**: 指示该分支指令实际是否跳转。
- **set_target_iaddr**: 要存进缓存的目标地址, 来自PCE + (Imm<<1)。

Output:

- **taken_out**: 预测 topredict_iaddr 指令是否跳转。
- **target_out**: 给 topredict_iaddr 指令的预测目标地址。
- **predict_error**: 指示当前在EX段的分支指令是否预测错误和错误类型。

错误类型	predict_error
在IF段预测跳转但实际不跳转	2'b11
在IF段预测不跳转但实际跳转	2'b10
预测正确	2'b00

- **error_count**: 记录程序运行过程中预测错误的总次数。

3. 测试

- 测试程序

```

1  .org 0x0
2  .global _start
3  _start:

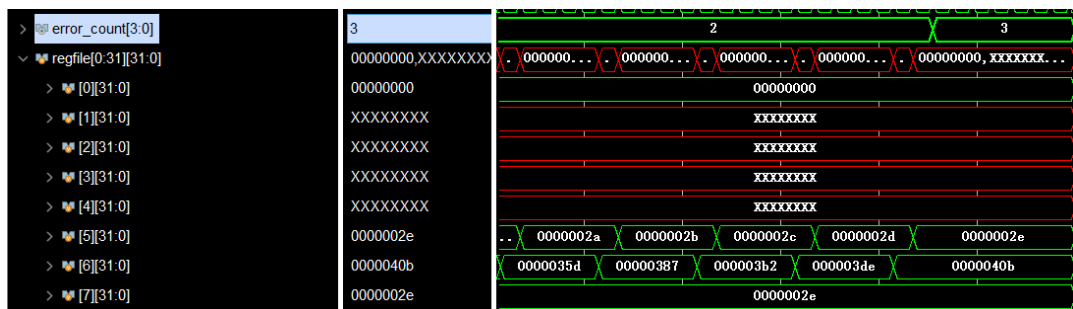
```

```

4      addi t0, zero, 0      #3000 t0 = 0
5      addi t1, zero, 0      #3004 t1 = 0
6      addi t2, zero, 46     #3008 t2 = 46
7      for:
8          beq t0, t2, final   #300c
9          add t1, t1, t0      #3010 t1 = 0+1+2+...+45
10         addi t0, t0, 1      #3014
11         beq zero, zero, for #3018
12     final:
13         addi zero, zero, 0   #301c
14         jal zero, final      #3020
15     # t1即x6, 最后结果应为为t1 = 1035(0x40b), t0 = 0x2e
16     # 预测错误次数总共为3, 300c指令一直预测不跳转, 最后一次预测错误
17     # 3018指令实际上一直跳转, 前两次预测不跳转, 有两次预测错误

```

仿真结果

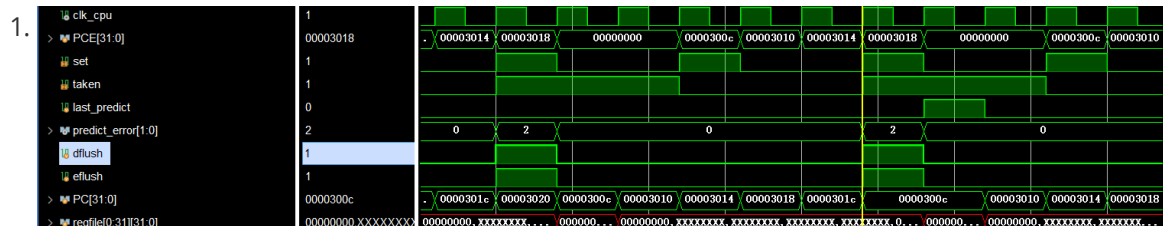


仿真结果显示, 相应的寄存器值正确, 预测错误次数正确, 表示该BPU能正确进行动态分支预测。

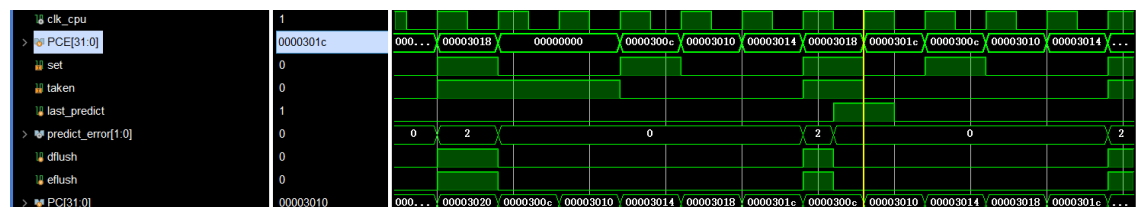
上板结果: 已通过助教检查。

4. 核心代码: 见附件。

总结: 本次实验所踩的坑



这样显然是不对的, 从第三次开始3018号指令 beq zero, zero, for 应该不会预测错误了才对, 但是因为 last_predict 延迟了一个周期, 导致该指令预测错误。



把 last_predict 改成时钟下降沿更新后就正确预测了。

2. 一些小而不易发现的错误, 例如

```

1      reg last_predict;
2      always @(negedge clk) // 注意这里需要用negedge
3          last_predict <= counter[set_tb_entry][1]; //这里少加了[1]导致找了好久的
          bug。。。

```

以及某些reg变量在多个always语句块中赋值。