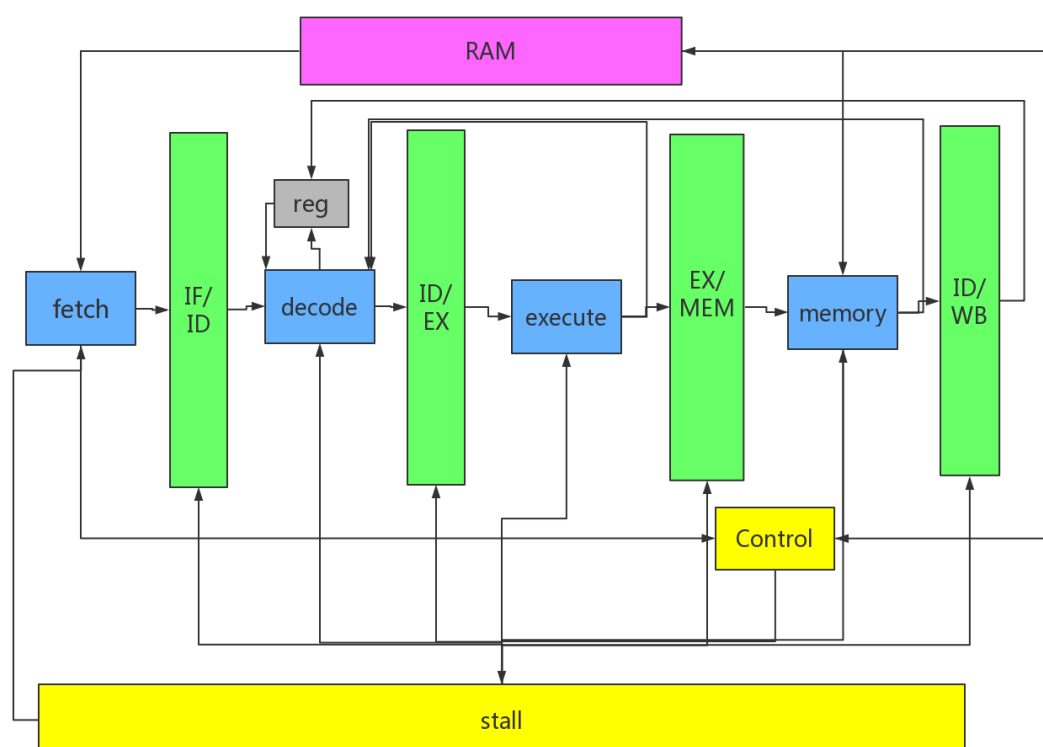


任务描述

使用 Verilog 完成一个能运行 [RV32I](#) 指令集中 Base Integer Instruction Set V2.0 中 2.1-2.7 指令的 CPU，并烧录在板子（BASYS 3）上。

基本架构



基本架构的理论是依靠 Computer Organization and Design 一书第四章对流水线的讲解，实现方面参考了《自己动手写 CPU》一书 OpenMIPS 的实现进行模改，需要注意的是，由于使用 block_ram 所以需要注意 mem 和 fetch 同时对于 memory 的访问。

构建了 Forwarding 来避免除了 load 指令以外的 data hazard，如果有 load 的 RAW hazard，则令 decoder 和 fetch 暂停。

对于 control hazard，这里采用了静态预测，即预测永不跳转。由于 Branch 仅需要简单的比较操作所以可以提前到 Decoder 阶段，来缩短延迟。

模块介绍

- regfile 32 个寄存器，使用了转发，可以同时进行两个寄存器的读操作和一个寄存器的写操作。

- if_id, id_ex, ex_mem, ex_mem 是流水线寄存器保存上个阶段的结果，在下一个时钟传递到下个阶段
- if, id, ex, mem 是流水线中间模块
- stall 用来控制流水线的暂停
- control 当 fetch 和 mem 同时对内存访问时，通过 stall 来暂停流水线，由于暂停 mem 的话就会一直 fetch 下去，所以这里暂停 fetch。

感谢和感悟

感谢陈林淇、万梓煜、邓伟信同学对我的帮助。

感谢两位助教不厌其烦的解答。

建议学习路线应该是《COD》然后《自己动手写 CPU》和《CAAQA》，这样理解的会更加清晰。

由于对于语言不够熟悉，写起来花了很多时间，未能加上 Cache 等优化，以后应尽早动工。

在读 Patterson 的书时感觉思想十分精妙，特抄录两句书中的话：

- There are a thousand hacking at the branches of evil to one who is striking at the root. -- Henry David Thoreau 《walden》
- What do you mean, why's it got to be built? It's a bypass. You've got to build bypasses.—Douglas Adams 《The Hitchhiker's guide to the galaxy》

第一句是说 Control Hazard 所带来的的不可避免的速度下降

第二句是说为了加速必须使用 Forwarding

参考资料

RISC-V 官方文档 <https://riscv.org/specifications/>

《Computer Organization and Design》 David A. Patterson

《Computer Architecture A Quantitative approach》 David A. Patterson

《自己动手写 CPU》 雷思磊