RiscV-CPU Project Report

Yujie Lu

Shanghai Jiao Tong University ACM class 18

November 29, 2019

Contents

1 Introduction

In this project, I write a CPU with **five stage pipeline** and **instruction cache** with verilog hard ware design language. It is robust and could pass all the tests. It's also runnable on an FPGA board.

2 The Architecture

2.1 Five stages

The five stages are the same with the classic five stage architecture.

2.2 Memory Controller

2.3 Stall Controller

3 Implementation Details

The dataflow of my CPU is as follows:

4 Some Suggestions

- The instructions on how to install RiscV tool chain and the bash script to build test cases are **obsolete** . It wasted me a lot of time to make it work. And although I opened a pull request, the TA didn't merge it or modified something on master branch.
- There should be more instructions on README.md since we even didn't know what does `hci.v` do.
- The data bus is too narrow that we can only send 1 byte per cycle. The should be more choices since it's hard to reach real parallelism if IF takes too much time.

References

[1] 手把手教你设计 CPU–RISC-V 处理器篇, 胡振波, 人民邮电出版社,2018

[2] 自己动手写 CPU, 雷思磊, 电子工业出版社,2014