SI100B EE-Part MiniCPU

RISC-V Background

Yuxuan Li

VSPLab@SIST

2026.05.27

liyx22025@shanghaitech.edu.cn

目录

1 指令集

2 汇编器

3 课程安排

1 指令集

指令集

指令集大致可以分为以下两类

- 复杂指令集（CISC）：以 x86 为代表，主要应用于桌面端。
- 简单指令集（RISC）：以 ARM 为代表，主要应用于移动端和嵌入式设备。

RISC-V 是一个基于精简指令集设计哲学的新兴开源指令集架构。

RISC-V 指令集

RISC-V 指令集可以分为若干个部分

- RV32I ，基础整数指令。
- RV32M，乘除扩展。
- RV32F ，浮点扩展。
- RV32C ，压缩扩展。
- RV32A ，原子扩展。
- RV32V ，向量扩展。

RISC-V 指令集

在该课程中，只要求实现 RV32I 的一个子集，具体而言，有以下 13 条指令

- 算术指令：`add`、`addi`、`sub`、`and`、`andi`、`or`、`ori`、`slt`、`slti`
- 读写指令：`lw`、`sw`
- 跳转指令：`bne`、`jal`

架构与微架构

对于一个指令集而言，处理器可以有多样的电路实现

- 指令集规范：处理器支持运行什么样的指令，称为架构（Architecture）
- 处理器设计：处理器的电路实现，称为微架构（Microarchitecture）

2 汇编器

加法的三种写法

对于一个加法操作

- 高级语言：`x3 = x1 + x2`
- 汇编语言：`add x3, x1, x2`
- 机器语言：`0000000 00010 00001 000 00011 0110011`

编译器和汇编器

- 高级语言 $\rightarrow$ 汇编语言：编译器（Complier）
- 汇编语言 $\rightarrow$ 机器语言：汇编器（Assembler）
- 汇编语言的本质是指令的助记符，汇编指令和机器代码之间是一一对应的，因此，汇编器的设计相对简单，本课程也要求用 Python 自己实现一个汇编器。

3 课程安排

课程安排

1. 第一部分（汇编语言）：RISC-V 的汇编指令是如何使用的?
2. 第二部分（机器语言）：RISC-V 的汇编指令如何转换为对应的二进制编码?
3. 第三部分（电路设计）：RISC-V 处理器的电路应当如何设计以实现这些指令?

参考资料

- Digital Design and Computer Architecture RISC-V Edition
- `https://github.com/liyuxuan3003/ComputerArchitecture/releases`

谢谢大家!