

数字电路实验报告:实验 10 综合实验

曾舒立

一．背景介绍

与大多数指令集相比，RISC-V 指令集可以自由地用于任何目的，允许任何人设计、制造和销售 RISC-V 芯片和软件。虽然这不是第一个开源指令集，但它具有重要意义，因为其设计使其适用于现代计算设备（如仓库规模云计算、高端移动电话和微小嵌入式系统）。设计者考虑到了这些用途中的性能与功率效率。该指令集还具有众多支持的软件，这解决了新指令集通常的弱点。冯诺依曼体系结构是现代计算机的基础。在该体系结构下，程序和数据统一存储，指令和数据需要从同一存储空间存取，经由同一总线传输，无法重叠执行。

根据冯诺依曼体系，CPU 的工作分为以下 5 个阶段：取指令阶段、指令译码阶段、执行指令阶段、访存取数和结果写回。取指令（IF，instruction fetch），即将一条指令从主存储器中取到指令寄存器的过程。程序计数器中的数值，用来指示当前指令在主存中的位置。当一条指令被取出后，程序计数器（PC）中的数值将根据指令字长度自动递增。指令译码阶段（ID，instruction decode），取出指令后，指令译码器按照预定的指令格式，对取回的指令进行拆分和解释，识别区分出不同的指令类别以及各种获取操作数的方法。现代 CISC 处理器会将拆分已提高并行率和效率。执行指令阶段（EX，execute），具体实现指令的功能。CPU 的不同部分被连接起来，以执行所需的操作。访存取数阶段（MEM，memory），根据指令需要访问主存、读取操作数，CPU 得到操作数在主存中的地址，并从主存中读取该操作数用于运算。部分指令不需要访问主存，则可以跳过该阶段。结果写回阶段（WB，write back），作为最后一个阶段，结果写回阶段把执行指令阶段的运行结果数据“写回”到某种存储形式。结果数据一般会被写到 CPU 的内部寄存器中，以便被后续的指令快速地存取；许多指令还会改变程序状态字寄存器中标志位的状态，这些标志位标识着不同的操作结果，可被用来影响程序的动作。在指令执行完毕、结果数据写回之后，若无意外事件（如结果溢出等）发生，计算机就从程序计数器中取得下一条指令地址，开始新一轮的循环，下一个指令周期将顺序取出下一条指令。

本次实验通过设计一个支持 RISC-V 部分指令的 5 级流水线 CPU 并完成一段冒牌排序的汇编代码，通过 CPU 运行此代码来确定 CPU 设计是否正确。

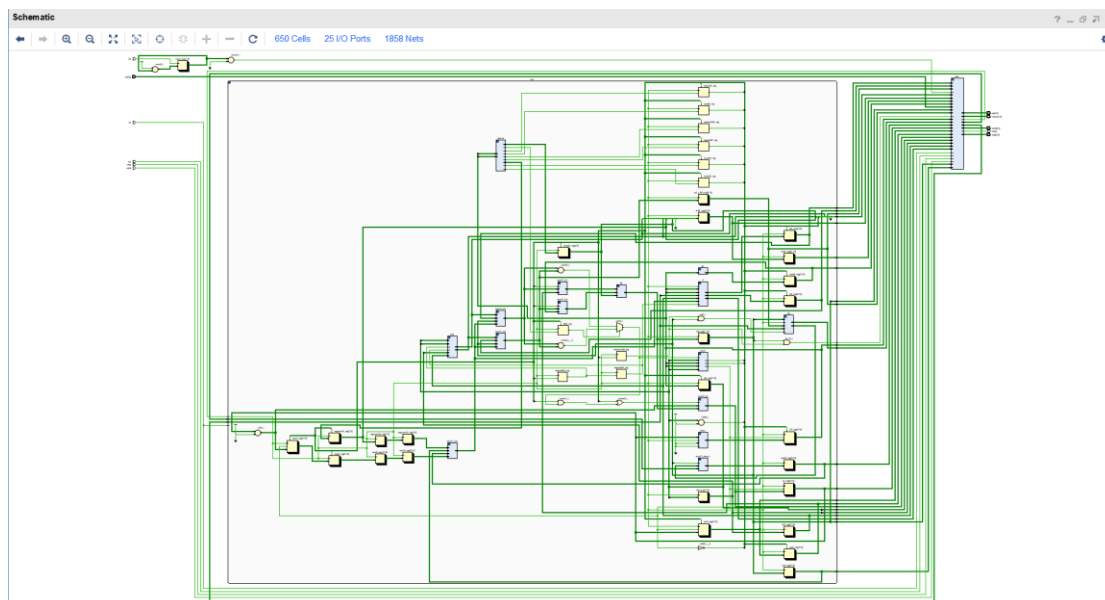
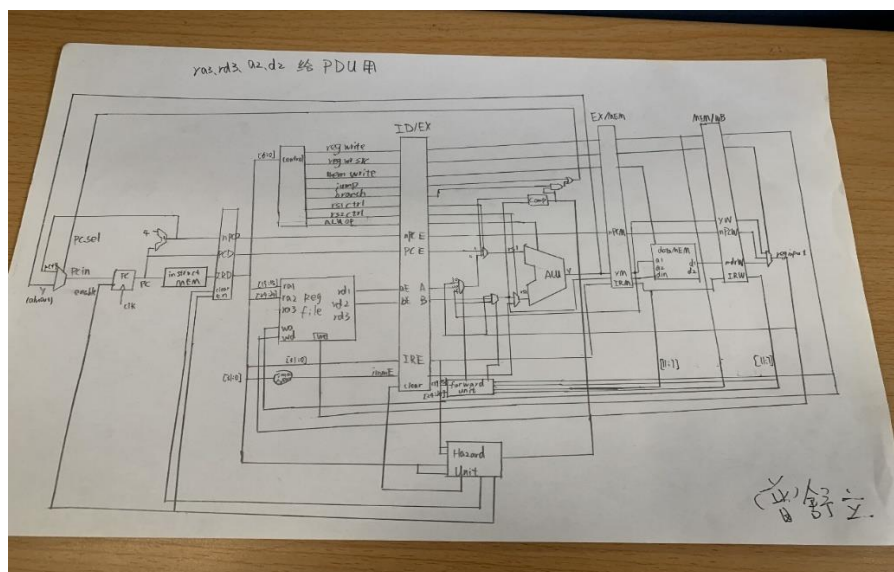
二．实验环境

- Vivado 2020.2
- fpgaol

三．实验目标

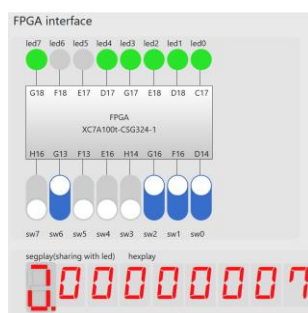
- 完成五级流水线 CPU 的设计
- 完成汇编代码以实现可交互的冒泡排序
- 利用 PDU(Processor Debug Unit，处理器调试单元)来完成综合仿真
- 上板测试并检验冒泡排序是否正确实现

四．数据通路与 RTL 电路

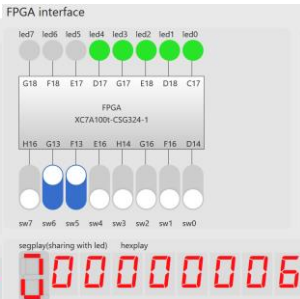


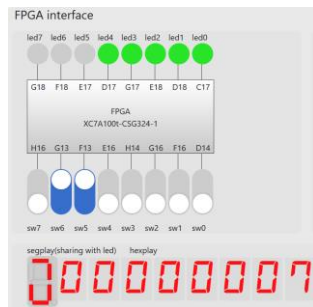
五．下载测试

第一步：指定数据长度为 7



第二步：按 5、3、6、2、7、1、4 的顺序输入数据





六．总结与思考

重温了组成原理的实验，对 CPU 的五段流水线有了更深的认识，弥补了组成原理留下的一些遗憾，难度适中，实验自由度高，学生可发挥空间大。