

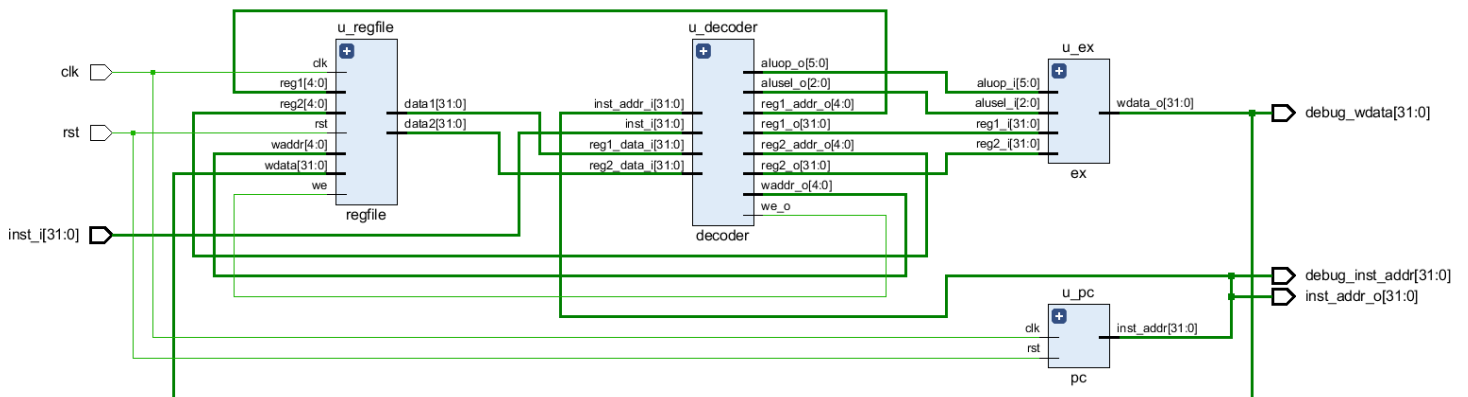
## 第二周任务

### 1. 了解以下关键词：

- a) 程序计数器 pc
- b) 译码器 decoder
- c) 执行器 ALU
- d) 寄存器组 Verilog 中的 mem 类型的使用方法（或掌握任意一种存储器方法皆可）

### 2. 制作一个单周期 CPU（1）：

- a) 说明：
  - i. 本次活动制作的是一个单周期 cpu，指在一个周期内完成一条指令的 cpu（由于是单周期 cpu，其应该由一个时序逻辑模块和多个组合逻辑模块构成）
  - ii. 本次任务包含其中四个部分：程序计数器(pc)、译码器(decoder)、执行器(ex)、寄存器组(regfile)
  - iii. 一条程序执行的流程如下：
    - 1. pc 生成指令地址，然后发送给 cpu 外部存储器，外部存储器返回该地址内存储的指令。
    - 2. decoder 分析指令的构成，根据指令的要求，从 regfile 和指令中包含的立即数中获取 ex 所需要的两个源操作数，判断 ex 要执行的运算类型，判断该指令的运行结果是否要写回 regfile 中，并生成结果要存入的目标寄存器地址。
    - 3. ex 根据 decoder 判断的运算类型对两个源操作数进行运算，产生结果。
    - 4. regfile 根据 decoder 的判断决定是否要接收 ex 产生的结果，如果接收成功，则一条新的数据被写入 cpu 的寄存器中。
- b) 参考结构图（和 demo 的设计匹配）



- c) 需要完成的指令为龙芯杯个人赛中出现的算术运算指令和逻辑运算指令

表 3-1 算术运算指令

指令名称格式	指令功能简述
ADDU rd, rs, rt	加（不产生溢出例外）

2

指令名称格式	指令功能简述
ADDIU rt, rs, immediate	加立即数（不产生溢出例外）
MUL rd, rs, rt	有符号字乘

表 3-2 逻辑运算指令

指令名称格式	指令功能简述
AND rd, rs, rt	位与
ANDI rt, rs, immediate	立即数位与
LUI rt,immediate	寄存器高半部分置立即数
OR rd, rs, rt	位或
ORI rt, rs, immediate	立即数位或
XOR rd, rs, rt	位异或
XORI rt, rs, immediate	立即数位异或

- d) demo 中完成了 addu 和 lui 两条指令，请模仿完成剩余指令，并自行进行样例测试 demo 的 testbench 中保存的是 addu 和 lui 的功能测试指令，程序为斐波那契数列，由于现在没有涉及跳转指令，程序会在指令地址溢出后自动回到开头。

### 3. 参考资料：

- 自己动手做 CPU 雷思磊著
- A03\_“系统能力培养大赛（个人赛）”MIPS 指令系统规范\_v1.00.pdf

### 4. 第二周最后需要完成的目标

- 完成 cpu 的四个部件，并添加算术和逻辑指令，完成仿真。
- 本周和第三周内容不进行验收。
- 关于最终验收，分为两部分。
  - 第一部分，要求自行编写一段程序，并进行仿真（能够体现 cpu 的指令正确性即可）
  - 我们会提供一段功能测试程序，校验部分指令正确性和完成度。（不会超过个人赛指令内容）