小组信息： 胡耘宁12311047 周二下午五六节 王薇老师

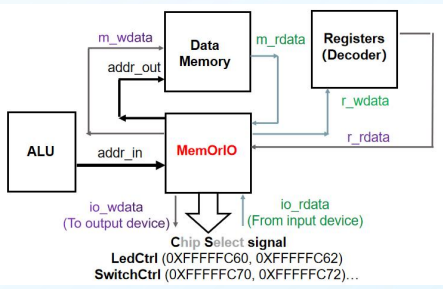
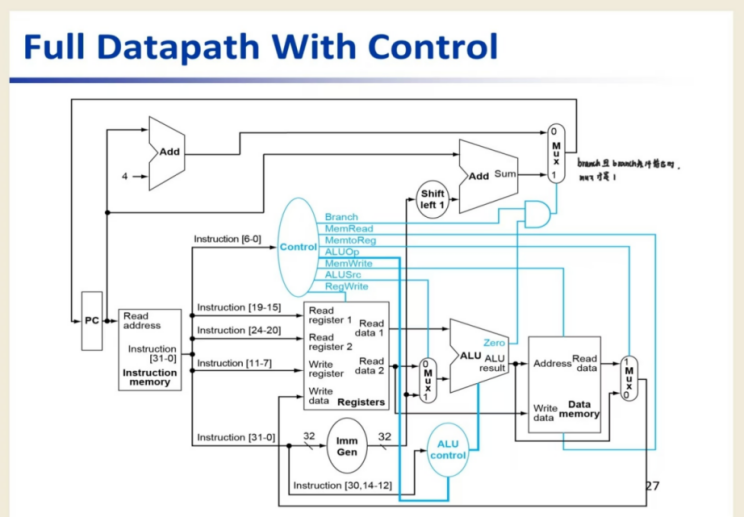
魏宇晴12311043 周二下午五六节 王薇老师

代码规范约定

1. 开展结构化设计：将两个mux设置为单独模块方便例化选择
2. 命名规范：根据课件上的CPU设计图片进行模块命名以及相应的输入输出端口命名
3. 注释要求：对于子模块之间的相互传递的信号标明来自哪个模块或者输出给哪个模块
4. 符号化常量的定义及使用：去抖模块常数化去抖周期

CPU特性说明：单周期，时钟（23MHz），支持的指令集合（R型，I型，S型，B型，J型，U型）

CPU架构设计（内部模块及连线关系）/接口设计说明：



由于接口命名和图片一致，所以可以根据图片清晰看出端口连接的设计

CPU指令与控制信号的关系：

1. Instruction Fetch，根据当前 pc 值读取指令，同时根据 Branch 和 zero 信号决定是否跳转。
2. Controller，根据当前指令解析生成对应的控制信号。
3. Registers，根据控制信号决定是否对寄存器进行写操作。
4. Imm\_Gen，解析指令中的立即数字段，生成扩展后的立即数。
5. ALU，执行算术或逻辑运算。
6. Data\_memory，根据控制信号决定是否读写数据存储器。
7. MemOrIO，判断地址访问是否针对内存或 I/O 设备。

已完成的项目代码以及已搭建的测试场景：

CPU初步设计文件，数码管显示，按钮去抖，限制文件

CPU各个子模块的仿真文件

Instruction\_fetch Instruction\_fetch\_tb

Controller Controller\_tb

Register Register\_tb

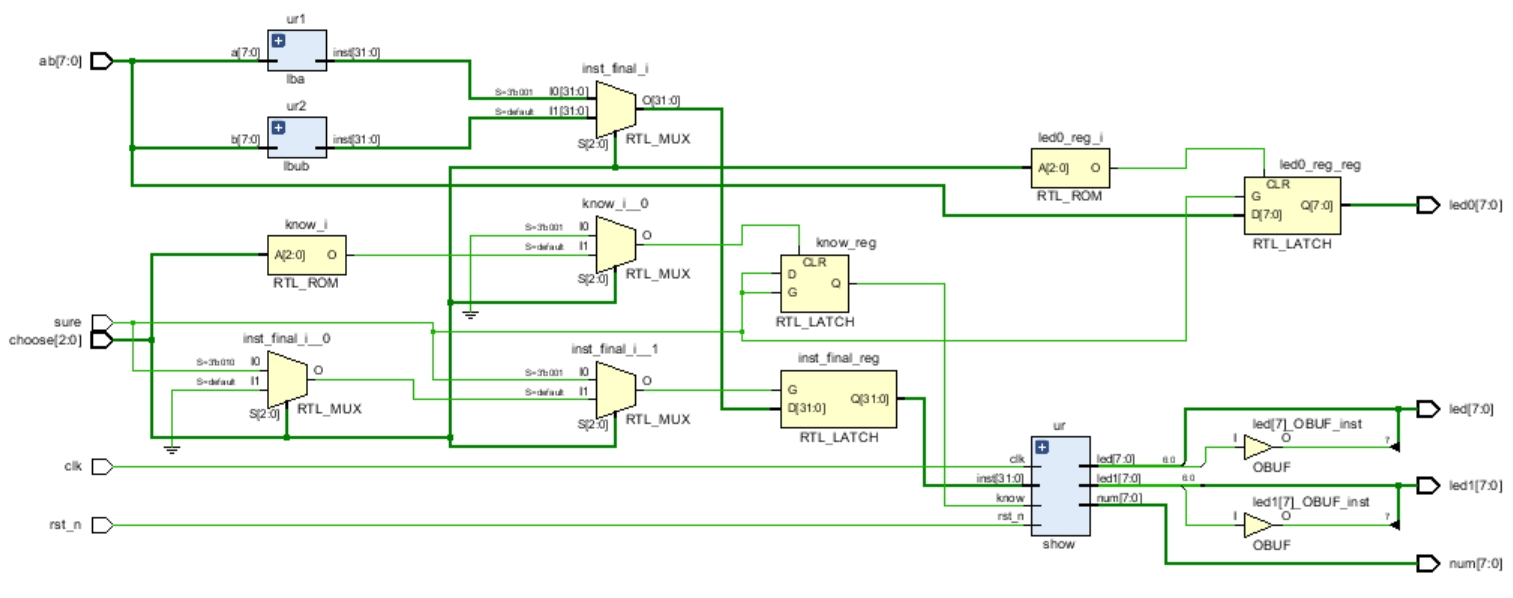
Imm\_gen Imm\_gen\_tb

ALU ALU\_tb

Data\_memmory Data\_memmory\_tb

MemOrIO MemOrIO\_tb

已搭建测试场景1的（0）（1）（2）及其testbench测试



计划使用/开发的工具链：

Vivado：用于对Verilog实现的RISC-V CPU进行综合、布局布线并生成下载到FPGA的bit流文件。

UartAssist：用于串口调试和数据通信。

rars\_27a7c1f：将RISC-V汇编语言代码组装成机器码

rars2coe：将Rars导出的指令和数据内存文本转为FPGA可识别的coe格式文件，用于配置 Vivado 设计的 IP 核。

项目进度：30%

当前的困难或问题：MemOrIO模块的实现

预计最终答辩时间：第十六周

后续计划：进一步完善CPU设计，编写汇编语言文件完成两个测试场景