待补全部分全部以？？？？来说明，请搜索全文并完成。

**Lab 5报告**

学号？？？？

姓名？？？？

箱子号 05

一、实验任务（10%）

在lab4的基础上，添加乘除法指令，并完成可以配套操作的访存指令（ld和st）。此外，完成更多的转移指令，使CPU的指令增加到46条指令。

二、实验设计（40%）

（一）总体设计思路

1. 在译码阶段，复用decoder完成指令的相应译码，并复用部分指令的数据通路和控制通路。

2. 对于乘除法，调用IP核来完成相应的操作。

3. 新增的四条跳转指令功能与已有的beq、bne非常相似，只是跳转的判断条件不同，因此实设计上整体复用原有的数据通路，在其基础上增加对应的判断条件。

4. ？？？？？

（二）重要模块1设计：除法器（以无符号为例）

定制IP核完成无符号除法，我们的工作主要是完成握手信号的连接。

1. 工作原理

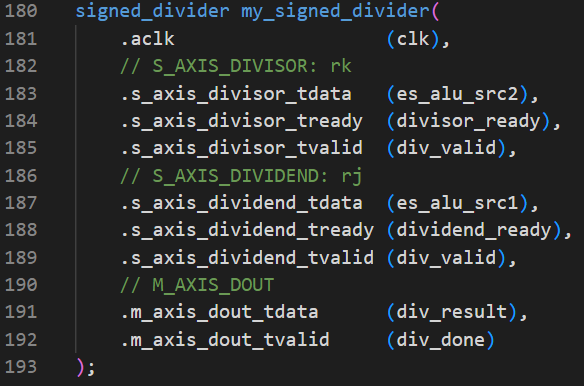


图1：除法器实例化

执行s\_axis\_dividend\_tdata/s\_axis\_divisor\_tdata，将商放在m\_axis\_dout\_tdata的高位，余数放在低位。

1. 接口定义

这一部分的核心是ready，valid和done三种信号的握手，其余信号省略。

当时钟上升沿来临时，将valid置为1，发起握手请求；随后ready信号也会拉高（周期性拉高），在下一个时钟上升沿时，两者均为高则完成握手，data传入IP核，进行运算。

最后，当div\_done拉高时，表明计算结束，阻塞放开，流水线启动。

1. 功能描述

调用IP核完成除法。

1. 重要模块2设计：跳转指令控制信号
2. 工作原理

跳转指令的控制信号如图2所示：

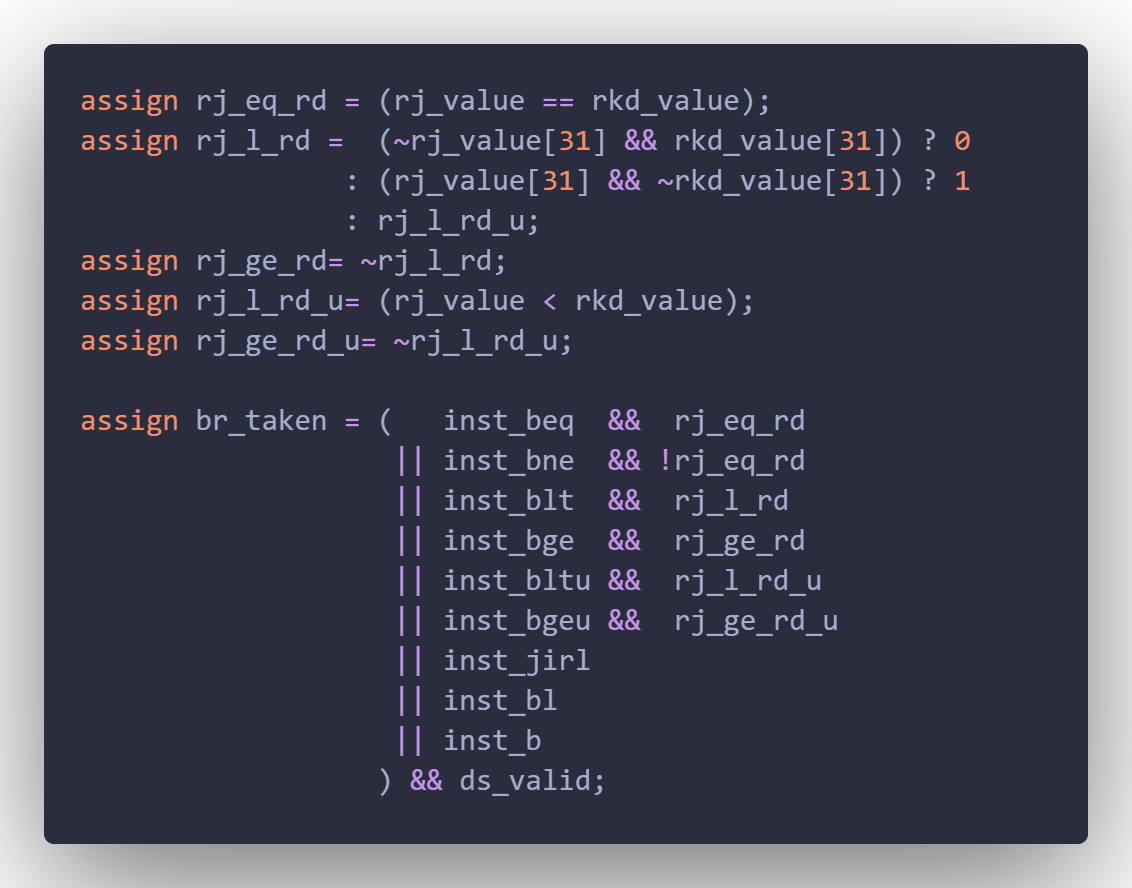


图2：跳转指令的控制信号

本实验新增了四条跳转指令，跳转逻辑分别是无符号小于、无符号大于等于、有符号小于、有符号大于等于。显然只需实现两个小于的判断，再对结果取反即可获得两个大于等于的取值。而有符号小于与无符号小于的比较方式也有所不同。小于号<可以直接对两个数进行无符号比较，但对于有符号比较则无法这样直接。一种方法是采用signed标识符，另一种是考虑两个数字的正负分情况讨论。稳妥起见，这里采用后者实现，得到了四个新增信号的赋值逻辑，进一步更改br\_taken和br\_target的逻辑即可实现相应跳转。

1. 功能描述

通过跳转控制信号判断当前指令是否跳转，完成四条跳转指令的新增

三、实验过程（50%）

（一）实验流水账

2022年9月30号，晚上7点到11点，完成exp10；

2022.10.7 21：00——23：00 实现新增的跳转指令

2022年10月8号，下午两点到四点，完成部分实验报告的书写；

2022.10.9 20：30——22：00完成部分实验报告的书写

（二）错误记录

1、错误1：除法器ip核的时序错误

（1）错误现象

div\_valid未能正确拉高，产生错误。波形会一直前进，无法停下。

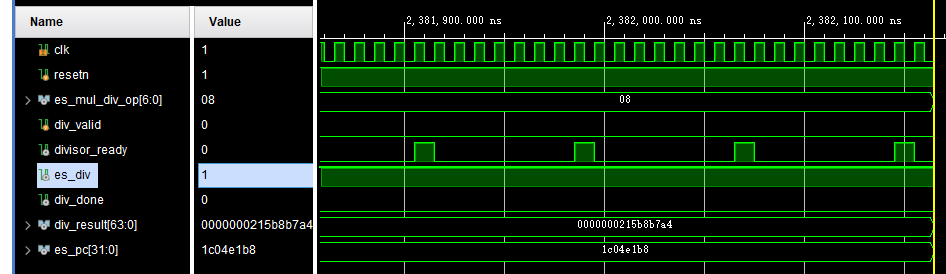


图2：错误1波形截图

（2）分析定位过程

分析div\_valid的来源，是时序的问题，于是开始用纸笔摆弄时序，发现错误原因。

（3）错误原因

分析div\_valid的来源，发现是来源于bus\_r（es\_mul\_div\_op），这需要一次clk才能将bus中的值传入bus\_r，这本是正常的一步。但是由于div\_valid是由时序控制的，其进入条件为(ds\_to\_es\_valid && es\_allowin)，于是会发生下述问题：

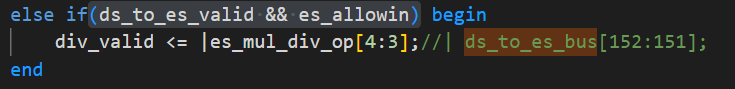


图3：错误1出错代码

在时钟上升沿来临时，ID阶段的div指令将进入EXE阶段，此时，es\_ready\_go的逻辑是组合的，于是在clk后，es\_mul\_div\_op的值成功从bus\_r中读了出来，但是此时es\_allowin也同时被拉低，于是div\_valid的值无法发生变化，不能拉高，最终导致错误。

（4）修正效果

将其直接连上ds\_to\_es\_bus即可，这样便可以解决时序的问题。

（5）归纳总结（可选）

对于axis总线标准以及时序跳变不清晰。

2、错误2：错误简介命名

……

四、实验总结（可选）

供同学们吐槽之用。

……