**Lab7 - mips-cpu设计  
金泽文 PB15111604**

实验目的：

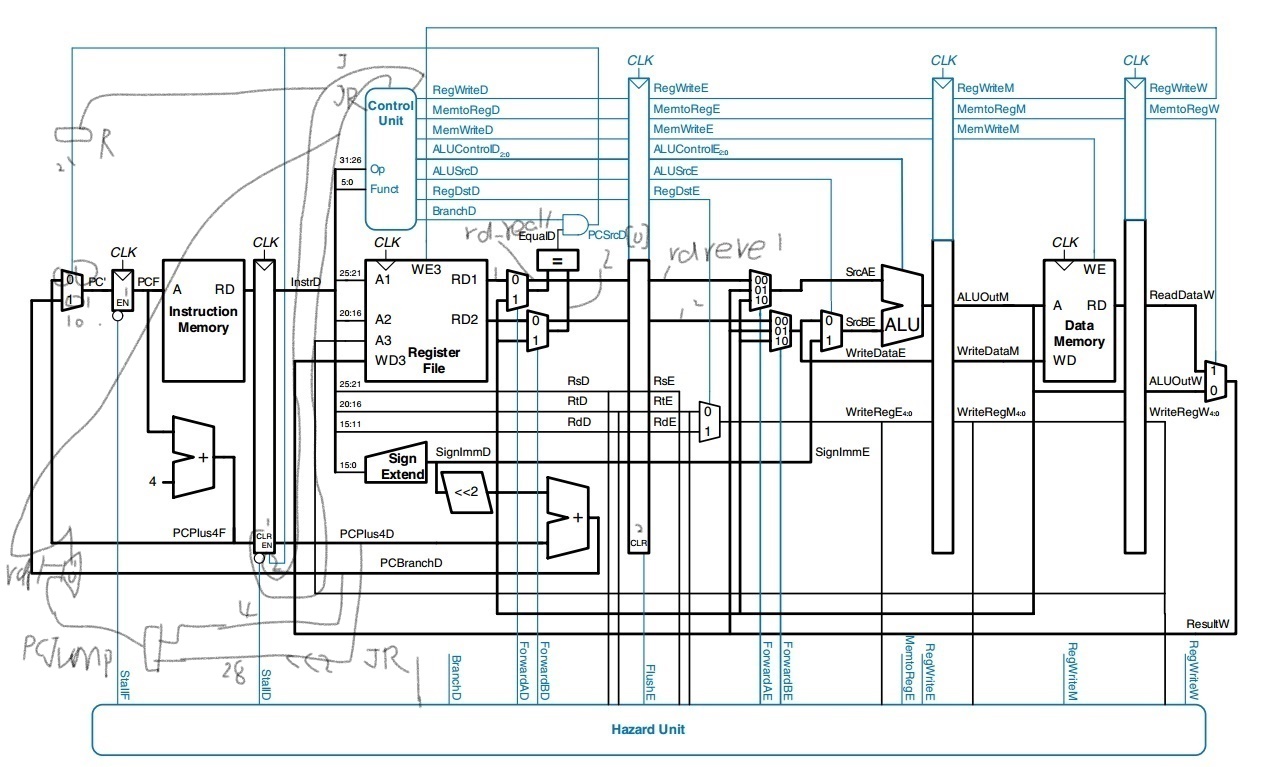
设计更为全面，更为优秀的mips cpu，以加深对Computer Organization and Design的理解。

实验内容：

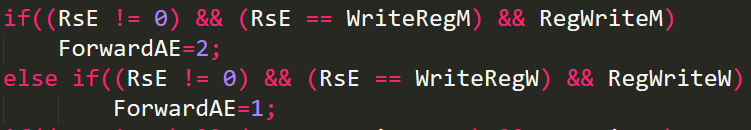
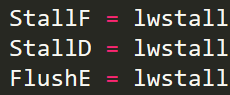
设计并实现流水线、静态分支预测功能，与对16个基本指令基础上的20条指令的处理功能。

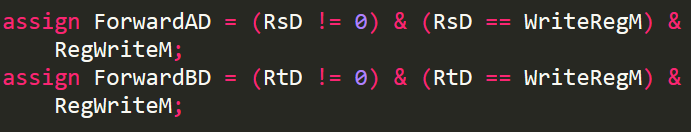
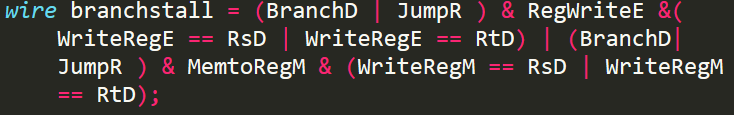
实验分析与设计：

首先是对流水线整体的思考。根据COD书上的内容和**David Money Harris**的《Digital Design and Computer Architecture》上面的讲解，进行设计。

总体按照下图设计。

（原图来自这本书，为了实现j,jr加了一些东西。大体参考了这本书对于单周期cpu的设计。）

为了避免数据冒险，首先要设置前推寄存器ForwardAE，和ForwardBE。判断条件就是，注意要优先判断M，后判断W。同时为了考虑lw对reg的影响，需要考虑stall和flush

 为了避免控制冒险，需要考虑beq等和jr等。

将以上封装在hazard模块中。

对于16条指令（以及额外实现的4条指令），设置control模块，在其中类比lab6针对opcode和funct。针对运算，由于funct对应运算和我们之前实现的alu不一致，所以设置alu\_decoder转码模块。

对于分支预测，我实现了静态分支预测。

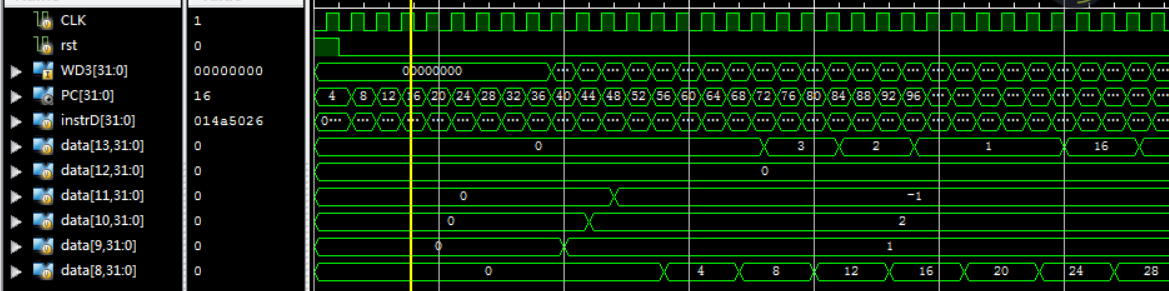
只需要简单的根据寄存器读出来的两个值RD1和RD2是否相等，设置EqualD与control输出的BranchD进行比较即可。另外考虑到J,JR,需要根据control输出的Jump和JumpR进一步判断。

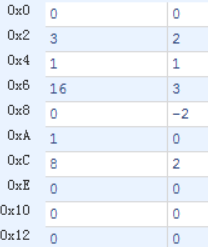
为了针对beq，bgtz，以及加了4条b指令后继续沿用equalD，所以直接设置equalD模块得到equalD结果，用于判断PCSrcD。

另外值得一说的是，为了将单个周期控制在一个时钟周期，所以将寄存器文件的触发沿改为了CLK的下降沿。

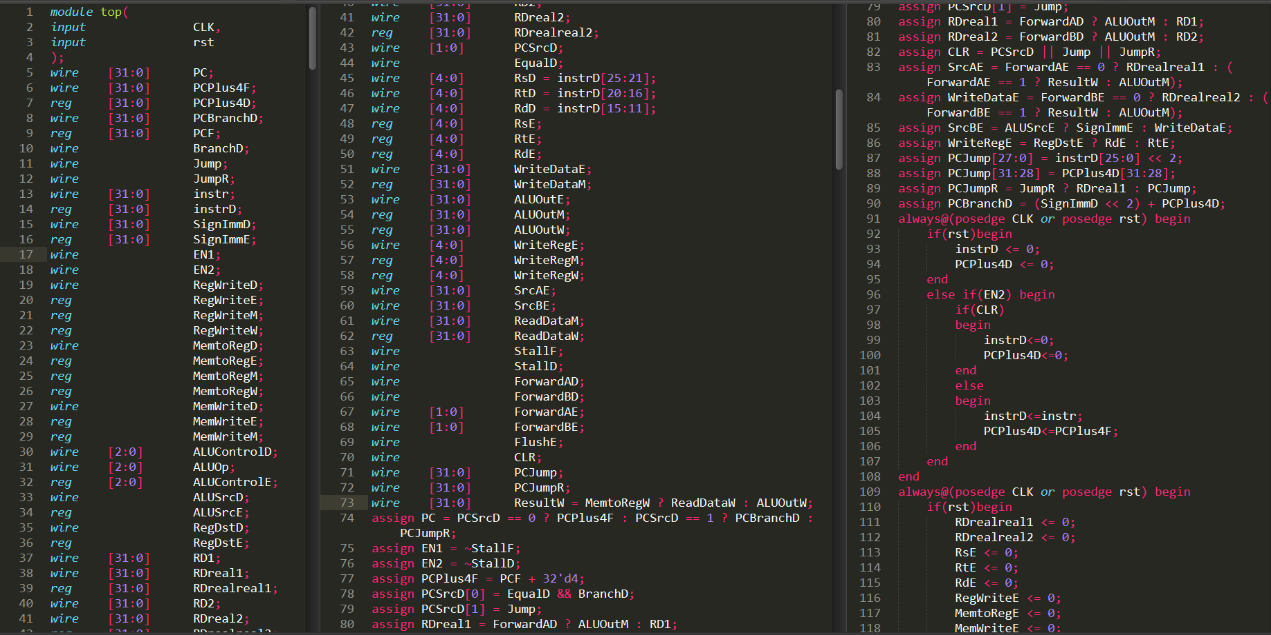
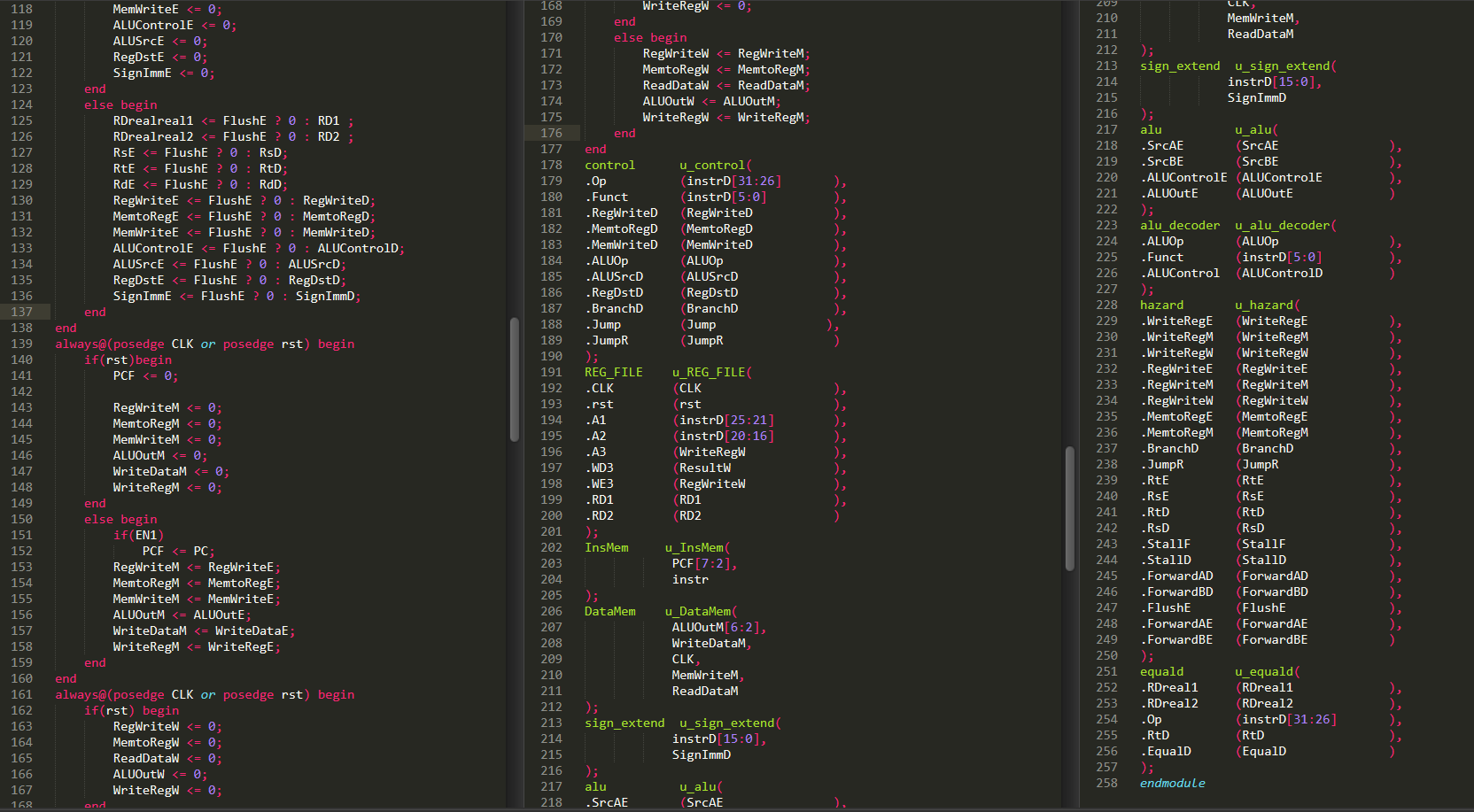
最后，感谢llxx老师和四位助教一个学期以来的知道和陪伴！你们都是坠胖的！

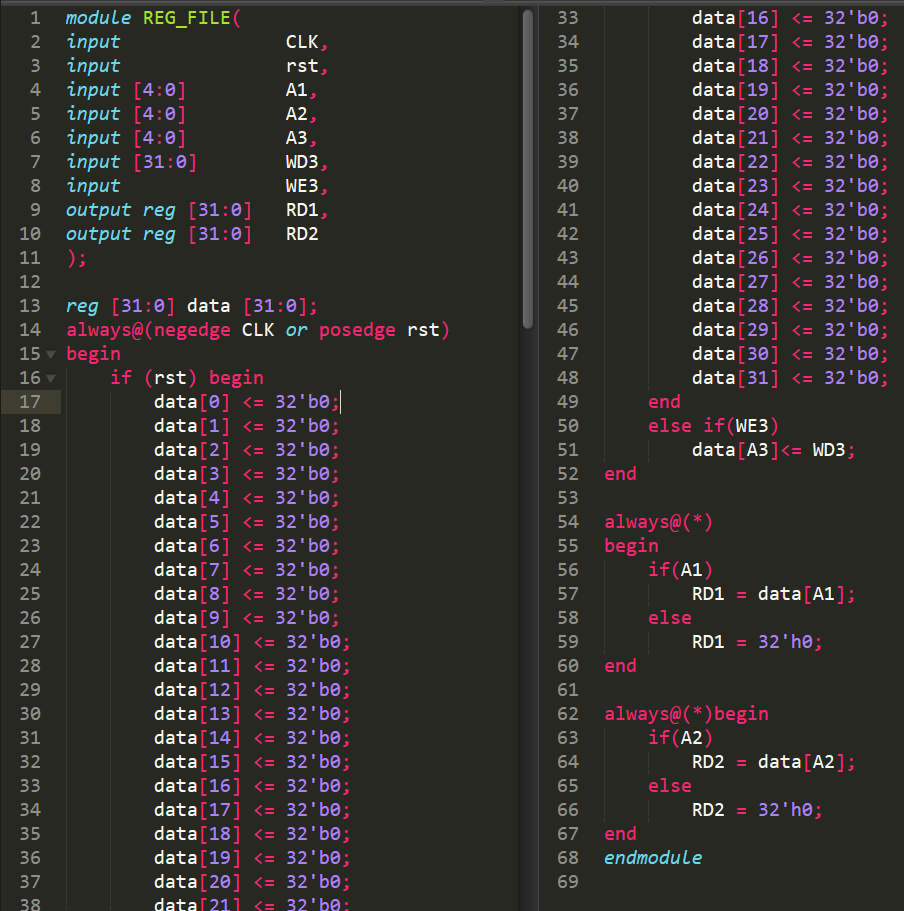
实验结果：

仿真波形：

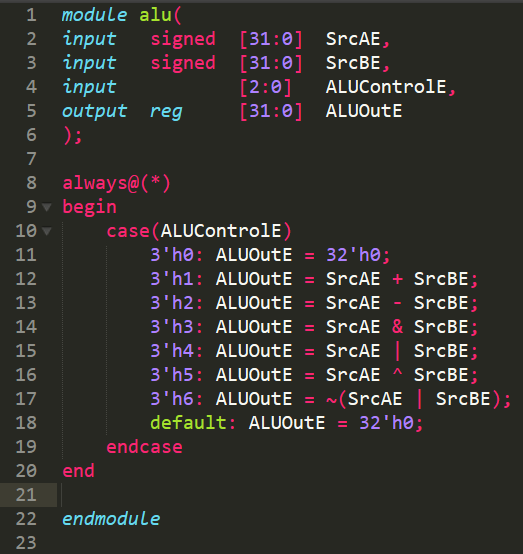
内存：

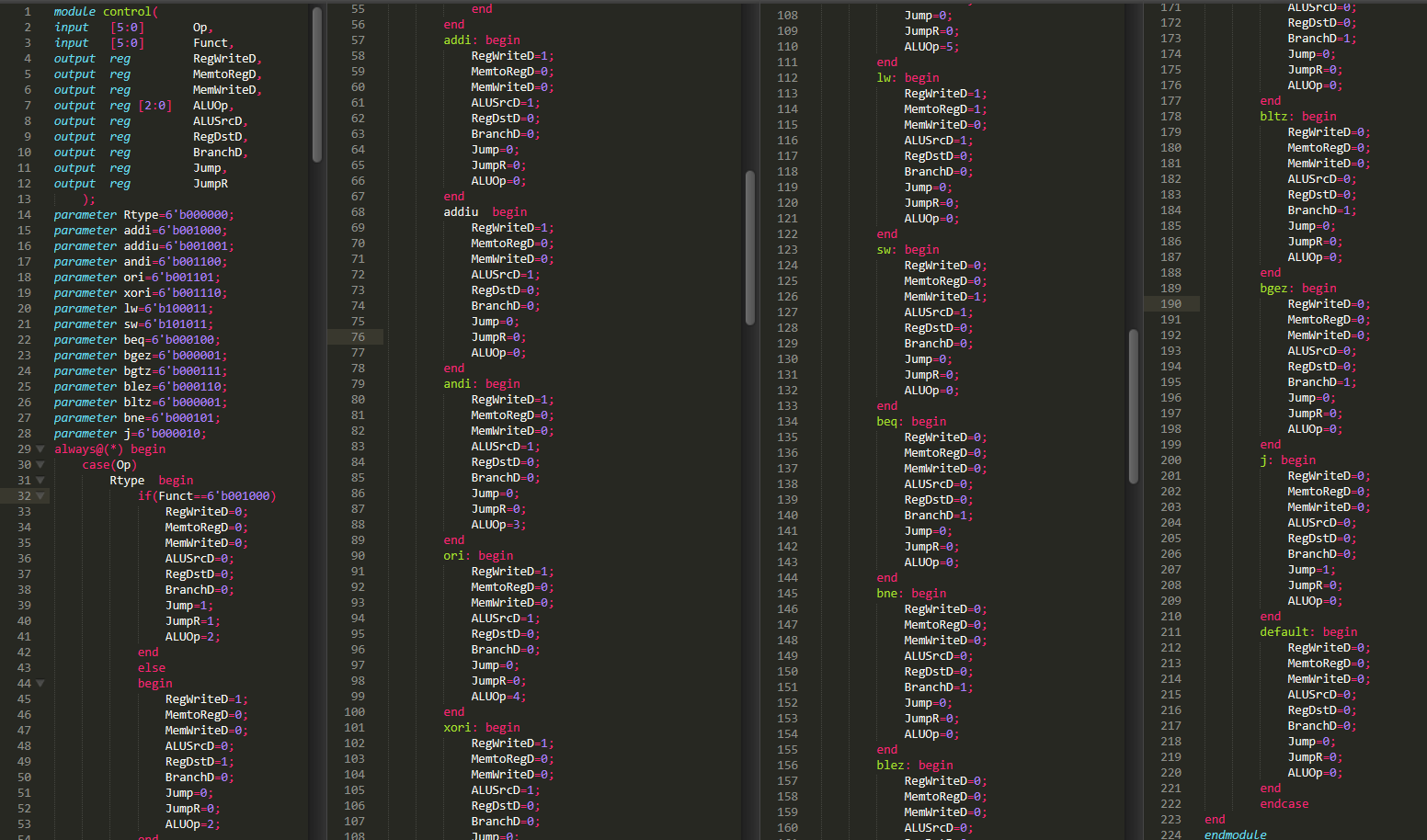
实验代码：

Top：

Reg\_file

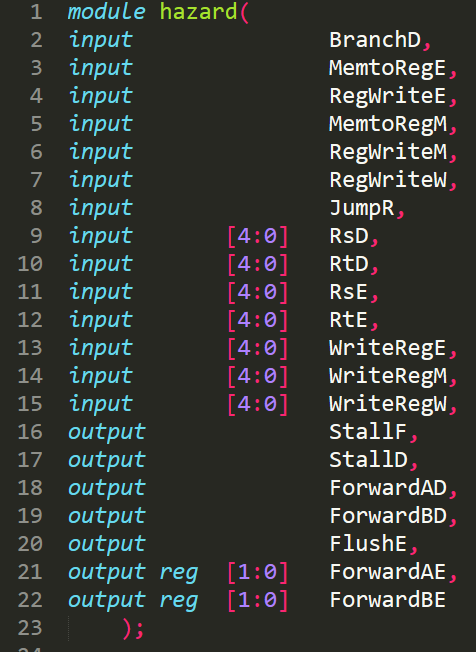
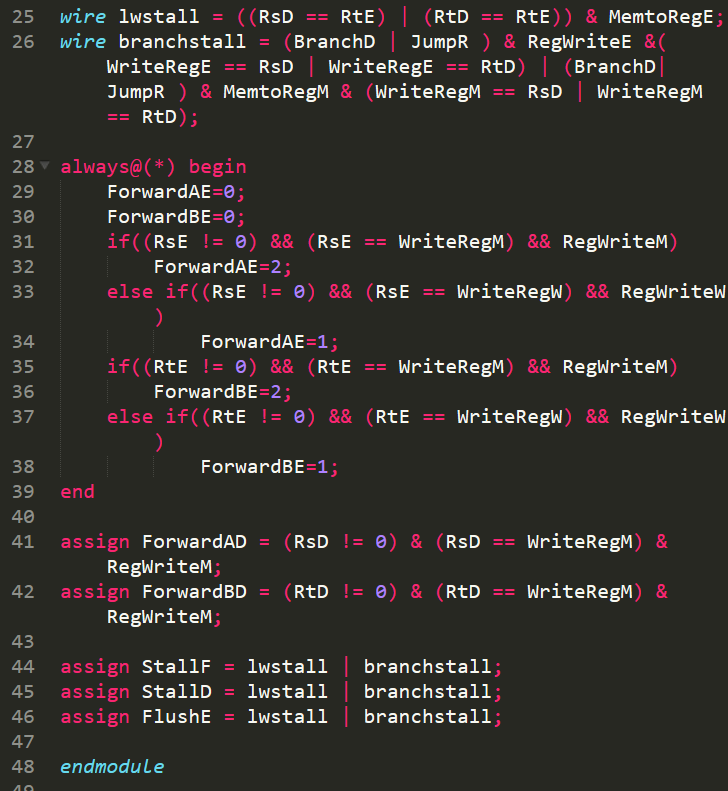
Alu：

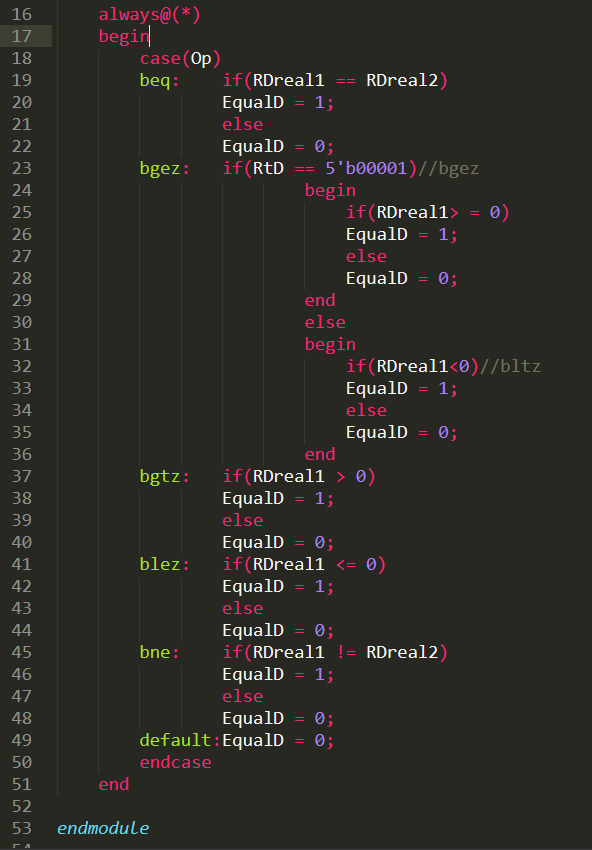
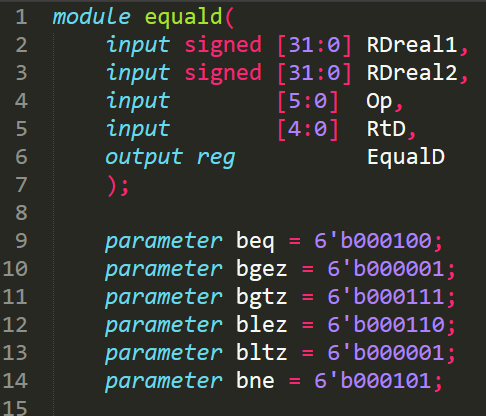


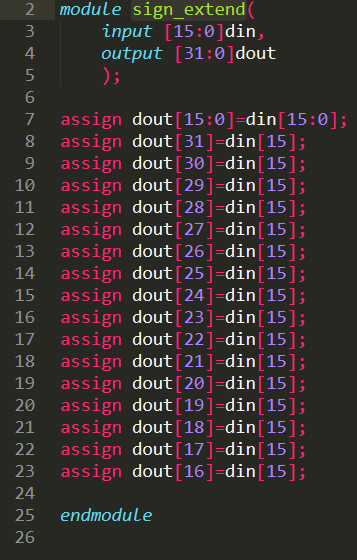
control

alu\_decoder: 

hazard:



equald:

sign\_extend:

sim: 