**深 圳 大 学 实 验 报 告**

**课 程 名 称： 计算机系统(3)**

**实验项目名称： 新增指令实验**

**学 院： 计算机与软件学院**

**专 业： 计算机科学与技术**

**指 导 教 师： 周明洋**

**报告人： 杜良衡 学号： 2022150255 班级： 01B**

**实 验 时 间： 2024.12.13**

**实验报告提交时间： 2024.12.13**

**教务处制**

**一、 实验目标：**

了解RISC-V mini处理器架构，在其基础之上新增一个指令，完成设计并观察指令执⾏。

**二、实验内容**

1) 修改数据通路，新增指令comb rs1,rs2,rd采用R型指令格式，实现将rs1高16位和rs2低16位拼接成32位整数，并且保存到rd寄存器。

2) 在处理器上执行该指令，观察仿真波形，验证功能是否正确。

3）自行设计其他功能指令，并验证设计是否正确

**三、实验环境**

硬件：桌面PC

软件：Chisel开发环境

**四、****实验步骤及说明**

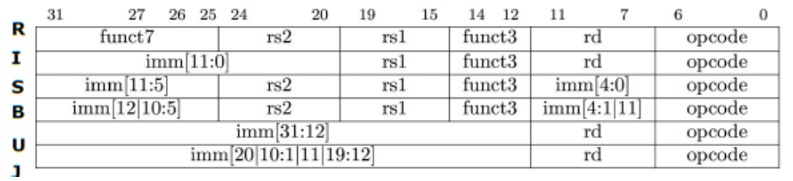
学习Chisel数据通路的Chisel描述，特别是指令译码部分和core核心代码。然后按照下面操作完成指令译码器的修改，以及数据通路的修改，

具体操作如下：按照参考文档完成comb指令的实现，自行设计新指令实现其功能并验证。

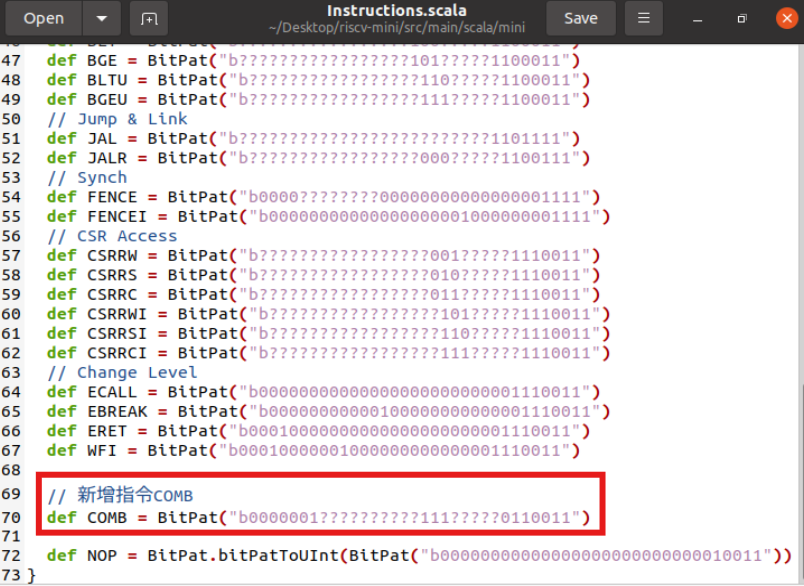
**五、实验结果**

**1. 在Instrutcions.scala文件中添加 comb 指令比特模式串**

comb 为R型指令，riscv的R型指令格式如下：

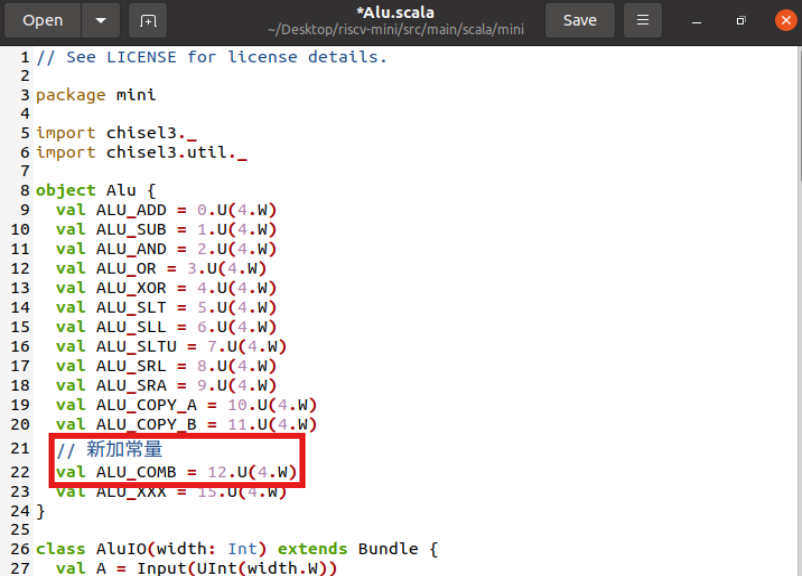


为了避免新加指令与riscv-mini已有指令冲突，这里我将 comb 指令的opcode、funct3和funct7部分设置为0110011、111、0000001。然 后使用 BitPat() 函数设置 comb 指令的比特模式。具体代码如下

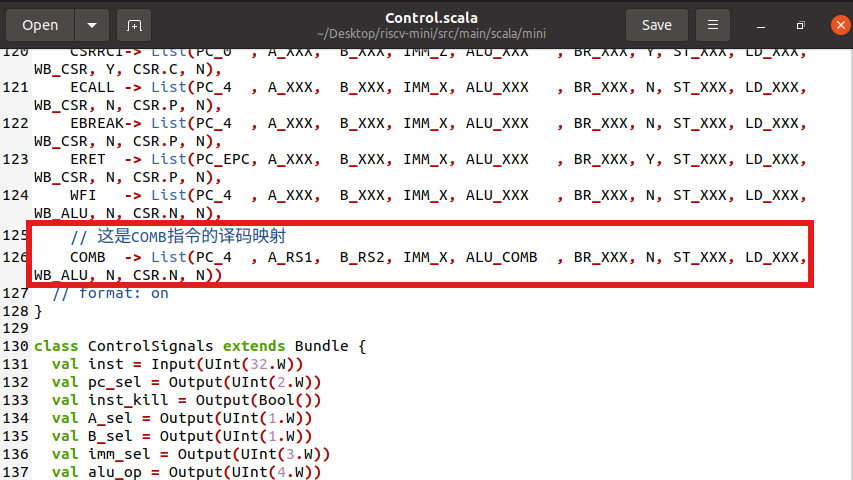


**2. 添加 comb 指令的译码**

comb 指令需要在ALU中将rs1高16位和rs2低16位拼接成32位整数，因此需要在Alu.scala文件中添加常量 码出正确的信号。具体代码如下：

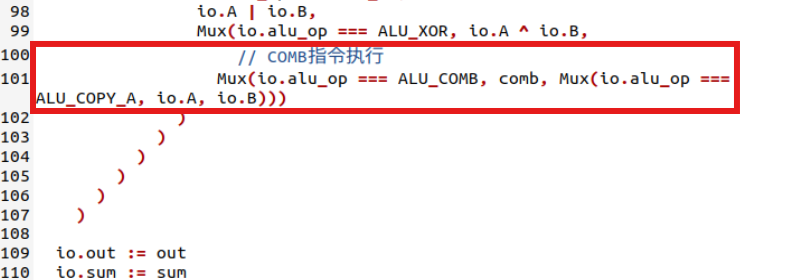


接下来为comb指令添加对应的译码映射。comb指令执行后pc需要加4，并将从寄存器文件中读取的数据rs1和rs2进行拼接操作，然后 将ALU输出的拼接结果写回到寄存器文件中。在Control.scala文件中添加的具体代码如下：



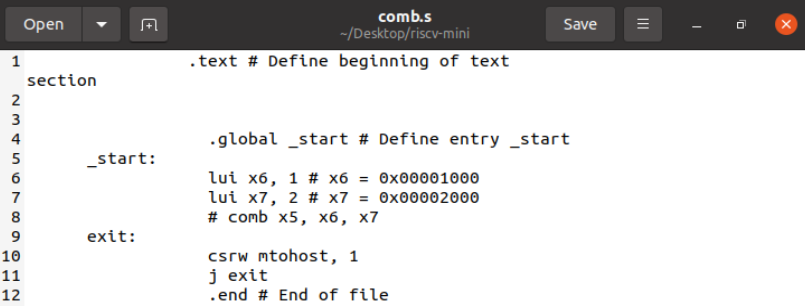
**3. 实现comb指令的执行操作**

在Alu.scala文件添加将rs1高16位和rs2低16位拼接成32位整数的操作，具体代码如下：



**4. 对comb指令进行测试**

首先创建comb.s文件，编写如下的汇编程序：



请注意，因为comb为自己加入的指令，不能被汇编器汇编，所以这里将其注释掉，到后面生成的comb.hex文件中再将comb x5, x6, x7的二进制添加进去。

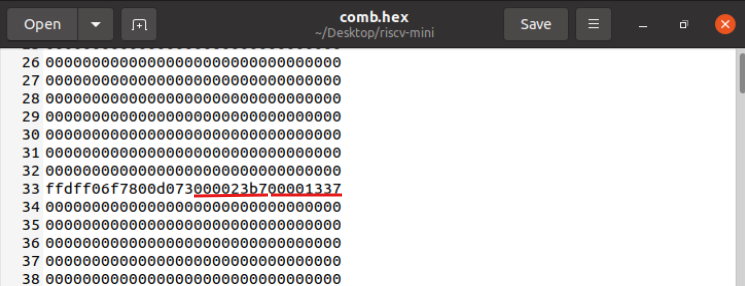
编写完程序后，使用如下命令进行编译：



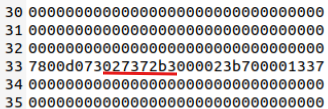
然后使用elf2hx命令将comb二进制文件转换成十六进制：



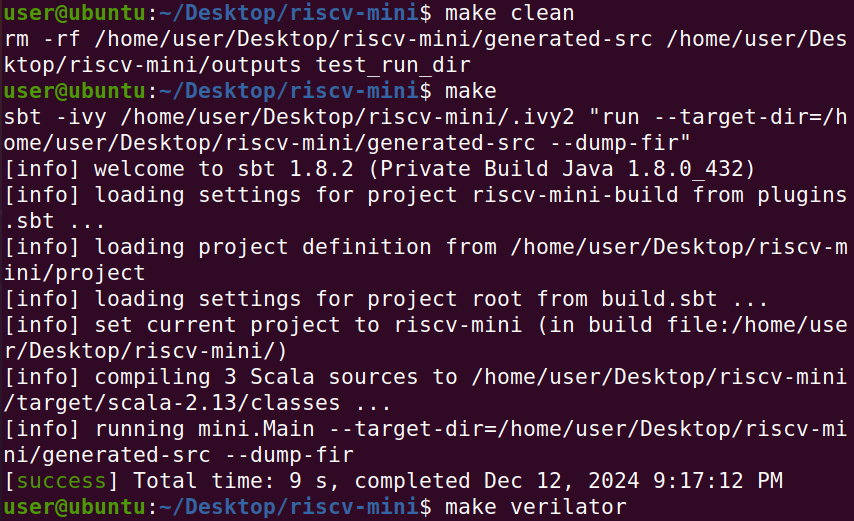
在comb.hex文件中，可以找到 lui x6, 1和 lui x7, 2的机器码对应的十六进制形式：



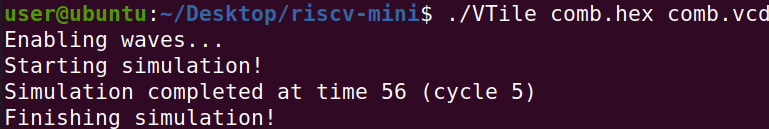
comb x5, x6, x7转换成机器码的十六机制形式为 027372b3。因此处指令存储为小端模式，故我们需要将十六进制数插入到第一个红线 的前面。修改后如下：



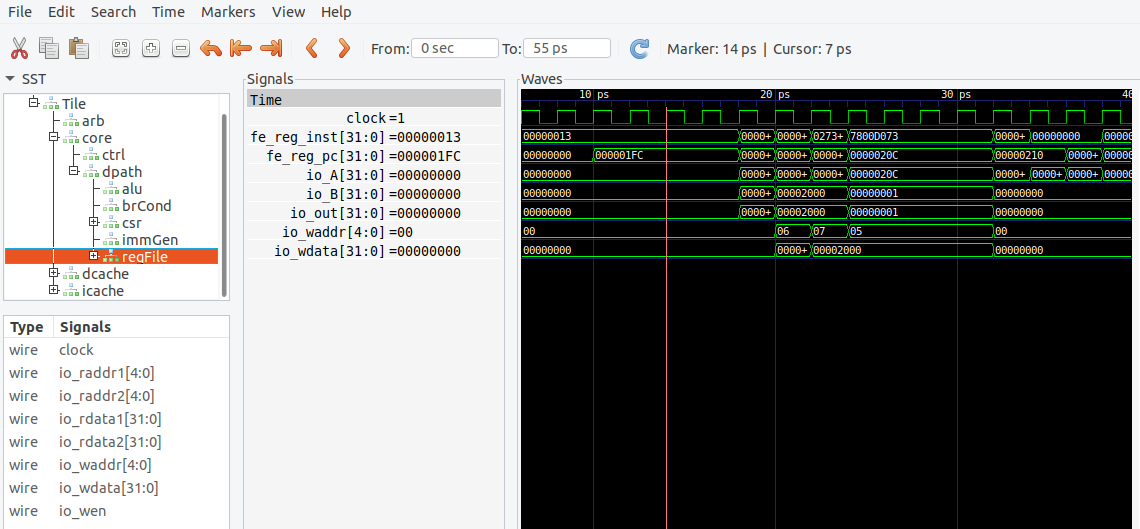
接着需要在主目录下一次执行 make 和 make verilator命令（若之前已经执行过，则在此次操作之前需要执行make clean），执行后会产生VTile可执行文件。



然后执行下面命令，使mini处理器执行新建指令并产生波形文件。



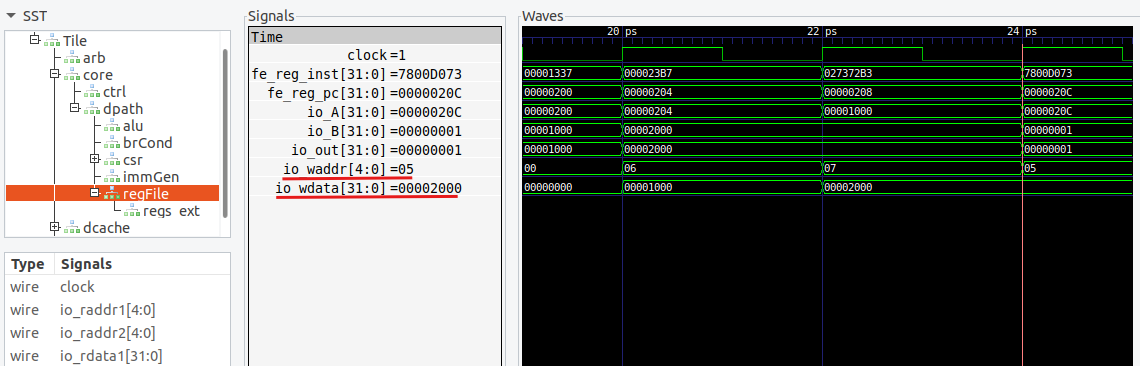
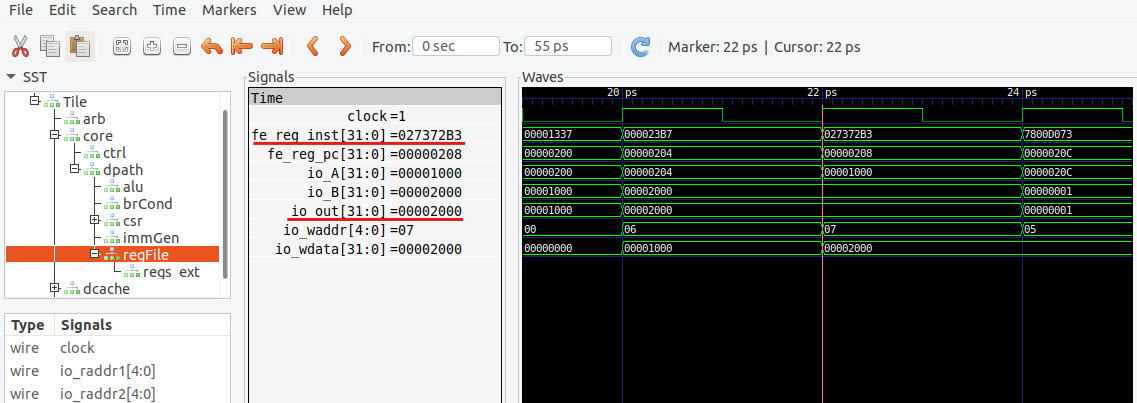
使用GTKWave打开comb.vcd文件，其波形图如下：



指令对应的十六进制形式见下表：



从波形图中可以看出， comb 指令将拼接后的结果0x00002000写回到了5号寄存器中，故该指令执行正常。



**六、实验总结与体会**

通过这次实验，我学会了如何在RISC-V mini处理器的基础之上新增指令：首先要根据risv指令格式，设置该指令各个字段的值，并添加该指令的比特模式，然后设置该指令的译码结果，在ALU中实现该指令的功能。最后让该指令在处理器上执行，并利用vcd波形图验证指令的执行结果。在实验的过程中，我对于RISC-V mini处理器的架构也有了更加深入的了解。

|  |
| --- |
| **指导教师批阅意见：**  **成绩评定：**  指导教师签字：  年 月 日 |
| 备注： |