

第二次实验内容

描述

本次实验为实现一个单周期CPU，你需要支持14条指令，见微信群pdf。

```
module Mips(clk,rst);  
  
input      clk ;  
input      rst ;
```

请严格按照此格式来定义端口，包括输入输出名称及位宽、模块命名，否则无法获取输出文件。

请将你的顶层文件及包含Mips module的文件命名为mips.v,并使用include的方式组织多文件编译。提交的时候将整个项目放置到project目录即可。

输入输出说明：

输入

clk: 时钟信号

rst:复位信号，如果该信号为高电平，pc应重置为默认pc。

你的im应从“code.txt”读取指令，该文件与mips.v置于同级目录。

你的dm应从“data.txt”读取数据，该文件与mips.v置于同级目录。

输出

本次实验你需要输出到标准输出。

对于寄存器，当寄存器写入数据时，你需要使用如下指令输出

`$display("reg:%d<=%h",rd,write_data);`,rd为寄存器的编号，write_data为写入的数据。

对于dm，当dm写入数据时，你需要使用如下指令输出`$display("dm:%d<=%h", addr,din);`，addr为写入的地址，din为写入的数据。

例如，当code.txt内容第一行为“24110001”，所对应的汇编指令为`addiu $s1,$zero,1`，即将s1(17号)寄存器的值设置为 $0+1=1$ ，因此，你应当输出`reg:$17<=00000001`。

限制说明

时间限制：5s

内存限制：N/A