实验 6 报告

组员 1：陈飞羽 2018K8009929031
组员 2：彭思叡 2018K8009908040
箱子号：45

一、实验任务（10%）

在 lab5 的 CPU 代码基础上添加指令，具体包括算术逻辑运算类指令 ADD、ADDI、SUB、SLTI、SLTIU、ANDI、ORI、XORI、SLLV、SRAV、SRLV，乘除运算类指令 MULT、MULTU、DIV、DIVU，以及乘除法配套的数据搬运指令 MFHI、MFLO、MTHI、MTLO。

二、实验设计（40%）

（一）总体设计思路

在本设计中，指令 ADD、ADDI、SUB、SLTI、SLTIU、SLLV、SRLV、SRAV 基本复用之前的控制信号及数据通路。同时，为实现指令 ANDI、ORI、XORI，本设计中增加了控制信号 zero_extended_imm 及立即数零拓展机制。

为实现乘除指令和乘除配套转移指令，在 EXE 模块中增加 32 位寄存器 HI 和 32 位寄存器 LO，由 reg 类型实现，并且使用写使能信号 hi_we、lo_we 和写入数据 hi_wdata、lo_wdata 封装两寄存器的写入逻辑。

为实现指令 MULT、MULTU，ID 模块调用了两个 Xilinx IP，分别用于计算有符号乘法和无符号乘法。64 位乘法结果的高 32 位与低 32 位分别写入寄存器 HI 和寄存器 LO。

为实现指令 DIV、DIVU， EXE 模块调用了两个 Xilinx IP，分别计算有符号除法和无符号除法，并修改阻塞控制逻辑，使得除法指令阻塞在执行级等待除法 IP 计算出有效结果。除法 IP 输出 64 位结果，其中商为[63:32]部分，写入寄存器 LO；余数为[31:0]部分，写入寄存器 HI。

为实现指令 MFHI、MFLO，EXE 模块增加三选一，选择 ALUresult、HI、LO 之一传递到 MEM 模块，其他复用之前的数据通路和控制信号。

指令 MTHI、MTLO 复用 EXE 模块的数据通路，完善寄存器 HI、LO 的写入逻辑。

总体设计图如图 1。

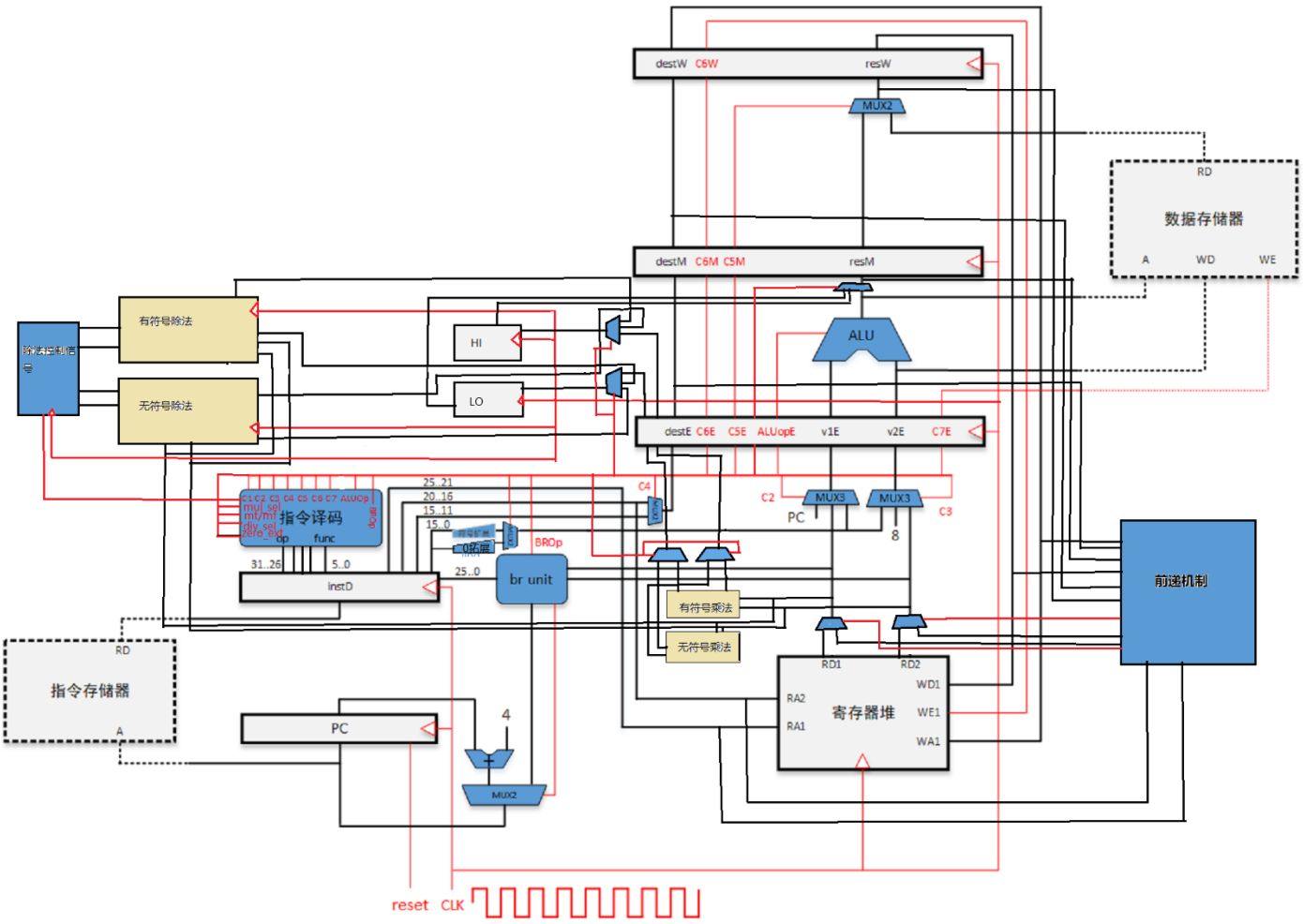

图 1 总体设计图

（二）重要模块 1 设计：DIV/DIVU 模块

1、工作原理

调用 Xilinx IP 实现有符号（或无符号）除法运算。

2、接口定义

名称	方向	位宽	功能描述
Aclk	IN	1	时钟信号
s_axis_divisor_tdata	IN	32	输入除数数据
s_axis_divisor_tvaild	IN	1	除数发送请求信号
s_axis_divisor_tready	OUT	1	除数接收应答信号
s_axis_dividend_tdata	IN	32	输入被除数数据
s_axis_dividend_tvalid	IN	1	被除数发送请求信号
s_axis_dividend_tready	OUT	1	被除数接收应答信号
m_axis_dout_tdata	OUT	64	输出计算结果
m_axis_dout_tvalid	OUT	1	计算结果发送请求信号

3、功能描述

调用该模块时，除数及被除数传入使用握手信号 tvalid、tready 控制。在时钟上升沿到来时，采样到 tvalid、tready 都为 1，则握手成功，除数及被除数被写入该模块的缓存；握手成功后，tvalid 保持为 0 至下一次除法。特别的，除数及被除数的 tvalid 一定要同时拉高，同时拉低。

该模块 AXI 接口上的流控为 Non Blocking 模式，因为本设计为单发射静态流水线 CPU，除法指令会被阻塞在执行级等待除法结果，保证了除法指令随时等待除法结果的输出。

（三）重要模块 2 设计：HI_LO 模块

1、工作原理

封装寄存器 hi 和寄存器 lo 的写入。

2、接口定义

名称	方向	位宽	功能描述
clk	IN	1	时钟信号
reset	IN	1	复位信号
Hi_we	IN	1	寄存器 HI 的写使能
Hi_wdata	IN	32	寄存器 HI 的写入数据
Hi	OUT	32	寄存器 HI 的值
Lo_we	IN	1	寄存器 LO 的写使能
Lo_wdata	IN	32	寄存器 LO 的写入数据
Lo	OUT	32	寄存器 LO 的值

3、功能描述

涉及寄存器 HI、LO 的写入包括六条指令：MULT、MULTU、DIV、DIVU、MTHI、MTLO。修改寄存器 HI、LO 的逻辑实现主体为六选一。

三、实验过程（50%）

（一）实验流水账

10.17 15：00 – 15：30 阅读指令手册，绘制设计图草稿 。

15：30 – 17：00 添加算术逻辑计算指令。

20：00 – 22：00 添加乘除运算指令及乘除配套搬运指令。

22：00 – 23：30 调试 BUG，前 33 个测试通过。

10.18 11：00 – 14：00 调试 BUG，全部测试通过，上板正常工作。

10.19 21：00 – 22：00 绘制硬件设计图。

22：00 – 24：00 撰写实验报告。

10.20 0：00 – 0：50 完成实验报告。

（二）错误记录

1、错误 1：拼写错误

（1）错误现象

报错如图 3。

```
[ 610377 ns] Error!!!
    reference: PC = 0xbfc3ac78, wb_rf_wnum = 0x02, wb_rf_wdata = 0x01519f05
    mycpu    : PC = 0xbfc3ac78, wb_rf_wnum = 0x02, wb_rf_wdata = 0x0151bf45
```

图 2 错误 1 报错

（2）分析定位过程

首先，追溯至 ds_pc=0xbfc3ac78 的时刻，确定该指令为 addi；然后沿该指令的执行过程，查看 15 位立即数、零拓展后的 32 位立即数、rs 寄存器值，发现输入 ALU 的数据正确而 ALU 输出结果错误。

拉取 ALU 的接口信号，发现 ALUop 异常.具体表现如图 4，从图 4 中可以看出 one-hot 编码的 ALUop 中同时有两位为 1。

图 3 错误 1 异常波形

（3）错误原因

将 inst_andi 信号加入 alu_op[4]时，出现拼写错误，误写成 inst_addi。

（4）修正方法

修正拼写错误。

2、错误 2：流水线停滞

（1）错误现象

流水线陷入停滞，trace 打印提示信息如图 5。

```
[21662000 ns] Test is running, debug_wb_pc = 0xbfc2c9e4
[21672000 ns] Test is running, debug_wb_pc = 0xbfc2c9e4
INFO: [Common 17-41] Interrupt caught. Command should exit soon.
[21682000 ns] Test is running, debug_wb_pc = 0xbfc2c9e4
[21692000 ns] Test is running, debug_wb_pc = 0xbfc2c9e4
```

图 4 错误 2 提示信息

（2）分析定位过程

首先，找到流水线陷入停滞的最初时刻，查看 fs_valid、ds_valid、es_valid、ms_valid、ws_valid 信号，发现 ms_valid 及 ws_valid 为 0。然后，找到令 EXE 阶段阻塞的指令，该指令为 div 指令。于是，浏览 div 接口信号，发现除法模块始终未发送除数及被除数的 tready 信号，具体异常如图 6。

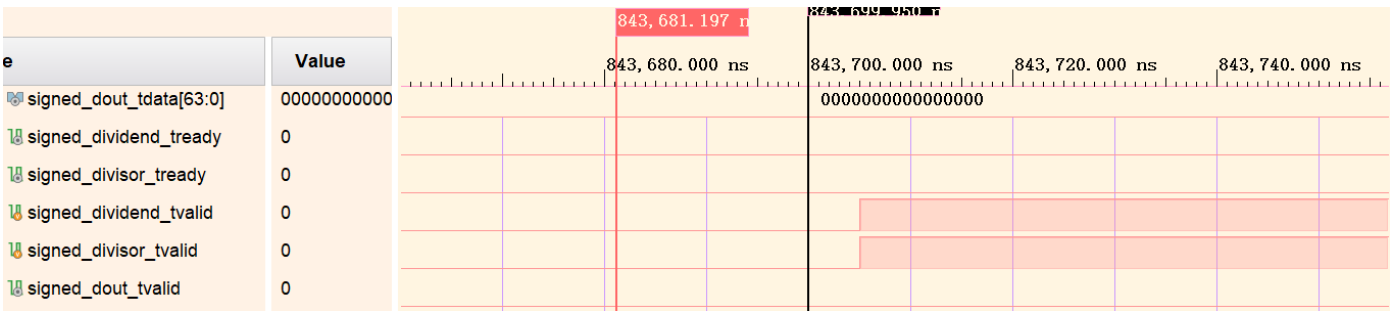

图 5 错误 2 异常波形

（3）错误原因

未将 clk 信号接入 div 模块。

（4）修正效果

将 clk 信号传入 div 模块。

3、错误 3：写入值错误

（1）错误现象

报错信息如图 7。

```
[ 844307 ns] Error!!!
    reference: PC = 0xbfc2ca24, wb_rf_wnum = 0x15, wb_rf_wdata = 0x00000000
    mycpu    : PC = 0xbfc2ca24, wb_rf_wnum = 0x15, wb_rf_wdata = 0x00000002
```

图 6 错误 3 报错信息

（2）分析定位过程

首先，追溯至 ds_pc=0xbfc2ca24 的时刻，发现该指令为 div 指令。然后拉取 div 模块接口信号，发现 tvaild 信号异常，具体如图 8。从图 8 可以看出，在 tready、tvalid 握手成功并拉低后，tvalid 又异常拉高。

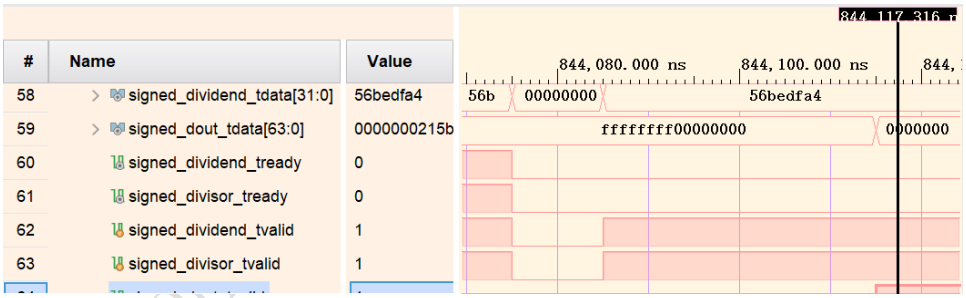

图 7 错误 3 异常波形

（3）错误原因

除法模块握手信号 tvalid 控制逻辑实现错误。

（4）修正方法

引入 div_en 使能信号，重构 tvalid 的控制逻辑。

div_en 为 reg 类型，当处于执行级的指令为 DIV 或 DIVU 时，向 div_en 赋 0，其他时刻赋 1。而令 tvalid 信号在处于执行级的指令为 DIV 或 DIVU 且 div_en 为 1 时，tvalid 赋 1，而 tready 为 1 时，tvalid 赋 0。利用时序逻辑的特性，保证了 tvalid 仅在与 tready 握手成功后的一个时钟周期为 1。

四、实验总结（可选）

本次实验中，通过向 CPU 添加指令，我加深了对 CPU 设计及时序逻辑的理解。