5 段流水 CPU 设计

徐睿 5130379501

2.2.1 实验目的：

1. 理解计算机指令流水线的协调工作原理，初步掌握流水线的设计和实现原理。

2. 深刻理解流水线寄存器在流水线实现中所起的重要作用。

3. 理解和掌握流水段的划分、设计原理及其实现方法原理。

4. 掌握运算器、寄存器堆、存储器、控制器在流水工作方式下，有别于实验一的设计和实现方法。

5. 掌握流水方式下，通过 I/O 端口与外部设备进行信息交互的方法。

2.2.2 实验内容：

1. 采用 Verilog 在 quartusⅡ中实现基本的具有 20 条 MIPS 指令的 5 段流水 CPU设计。

2. 利用实验提供的标准测试程序代码，完成仿真测试。

3. 采用 I/O 统一编址方式，即将输入输出的 I/O 地址空间，作为数据存取空间的一部分，实现 CPU 与外部设备的输入输出端口设计。实验中可采用高端地址。

4. 利用设计的 I/O 端口，通过 lw 指令，输入 DE2 实验板上的按键等输入设备信息。即将外部设备状态，读到 CPU 内部寄存器。

5. 利用设计的 I/O 端口，通过 sw 指令，输出对 DE2 实验板上的 LED 灯等输出设备的控制信号（或数据信息）。即将对外部设备的控制数据，从 CPU 内部的寄存器，写入到外部设备的相应控制寄存器（或可直接连接至外部设备的控制输入信号）。

6. 利用自己编写的程序代码，在自己设计的 CPU 上，实现对板载输入开关或按键的状态输入，并将判别或处理结果，利用板载 LED 灯或 7 段 LED 数码管显示出来。

7. 例如，将一路 4bit 二进制输入与另一路 4bit 二进制输入相加，利用两组分别2 个 LED 数码管以 10 进制形式显示“被加数”和“加数”，另外一组 LED数码管以 10 进制形式显示“和”等。（具体任务形式不做严格规定，同学可自由创意）。

8. 在实现 MIPS 基本 20 条指令的基础上， 实现 Y86 相应的基本指令。

9. 在实验报告中，汇报自己的设计思想和方法；并以汇编语言的形式，提供以上两种指令集（ MIPS 和 Y86） 应用功能的程序设计代码，并提供程序主要

预习内容

1. 实验前仔细阅读 DE1-SOC User Manual 及相关用户应用数据手册，学习并掌握其板载相关资源的工作原理、连接方式、和应用注意事项。

2. 根据课程所讲 5 段流水 CPU 设计原理，提前设计并仿真实现相关设计代码。

实验器材

Altera-DE2 实验板套件 1 套

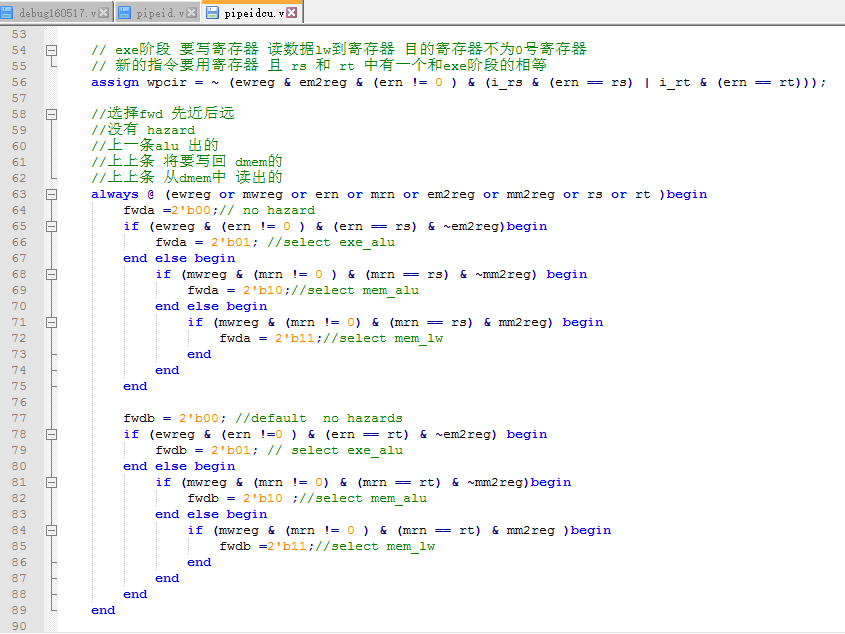
实验过程

Reg：

将左端数据保存于两个阶段间的reg

并在时钟上升沿时 将数据向右端传输

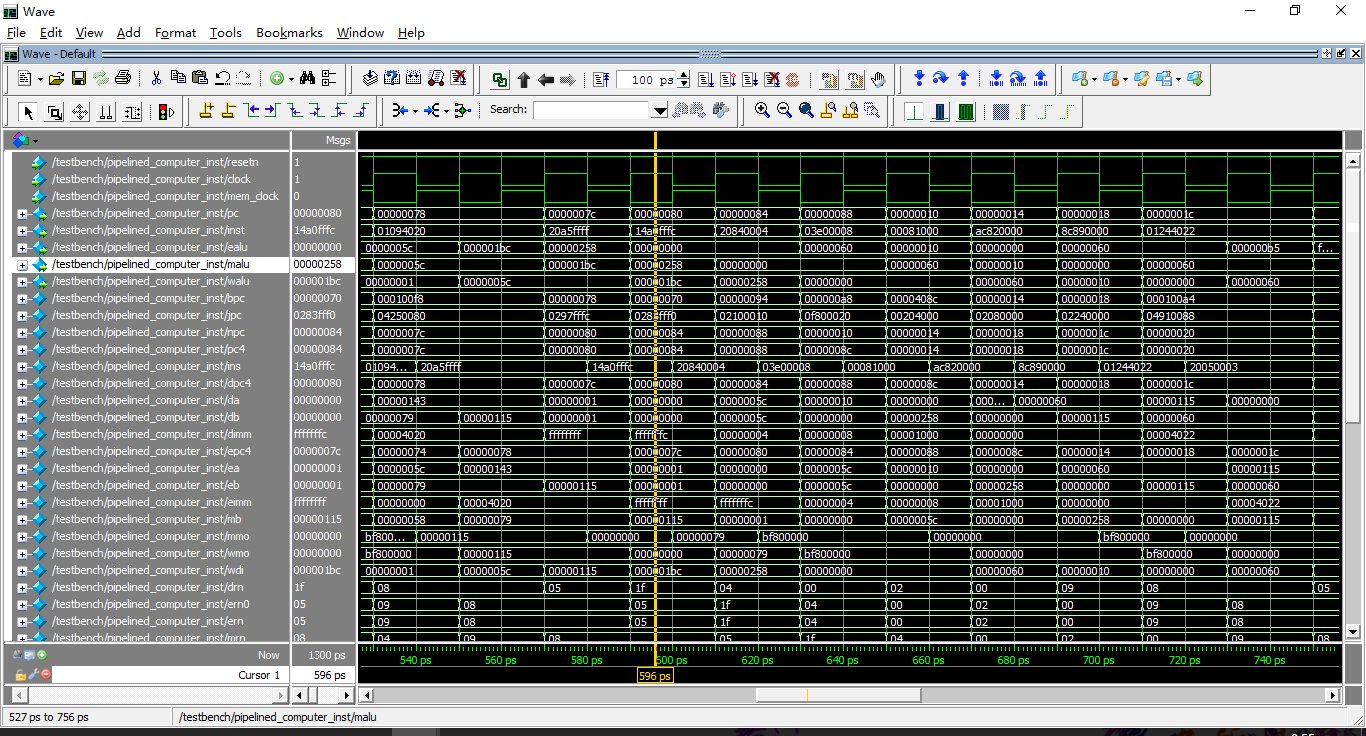
Data hazard



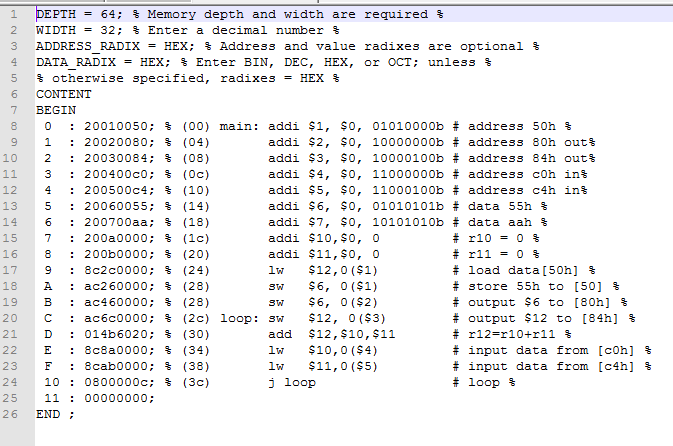
通过wpcir控制前两个阶段的reg

只要没有错误的寄存器和内存写入 线上任意数据都是可以传

Modelsim



I/O



因为跳转的后一条指令一定会被执行 所以最后新增一条空指令

