计算机系统结构实验 Lab04

朱凡悦 522031910547

May 2024

1 摘要

在 Lab04 中实现了简单的类 MIPS 处理器的几个重要部件：寄存器 (Register)、存储器 (Data Memory) 以及有符号扩展单元 (Sign Extension)，其作用分别是暂时存放一些数据与运算结果、用于较大量存储数据以及对立即数进行有符号拓展。对于存储部件（如寄存器、存储器等），其需要实现读取数据、存放数据和写入数据的功能，并通过软件仿真的形式进行实验结果的验证。

目录

2 实验目的

1. 理解寄存器、数据存储器、有符号扩展单元的 IO 定义
2. Registers 的设计实现
3. Data Memory 的设计实现
4. 有符号扩展部件的实现
5. 对功能模块进行仿真

3 原理分析

3.1 寄存器 (Register)

寄存器是指令操作的主要对象，MIPS 中一共有 32 个 32 位的寄存器，用作数据的缓存。我们设计的寄存器内存储的数据为 32 位二进制数，因此所有的数据接口均接收或发送 32 位二进制数。寄存器的 IO 定义如下图所示。由于不确定 WriteReg, WriteData, RegWrite 信号的先后次序，我们采

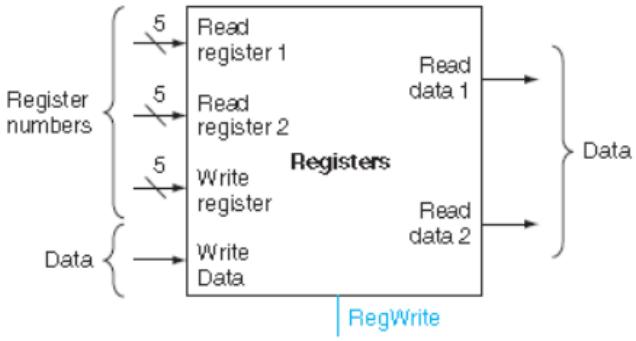

图 1: 寄存器模块的 IO 定义

用时钟的下降沿作为写操作的同步信号，防止发生错误。具体实现代码如下。

```
1    always @(readReg1 or readReg2 or writeReg)
2    begin
3        readData1=(readReg1==5'd0)?32'd0:(regFile[readReg1]);
4        readData2=(readReg2==5'd0)?32'd0:(regFile[readReg2]);
```

```
    end

    always @(negedge Clk)
    begin
        if (regWrite & writeReg!=5'd0) regFile[writeReg]<=writeData;
    end
```

3.2 存储器 (Data Memory)

Data memory 是用来存储运行完成的数据，或者初始化的数据，采用 32 位二进制数作为编号的方式以尽可能贴近现代计算机的设计。与寄存器

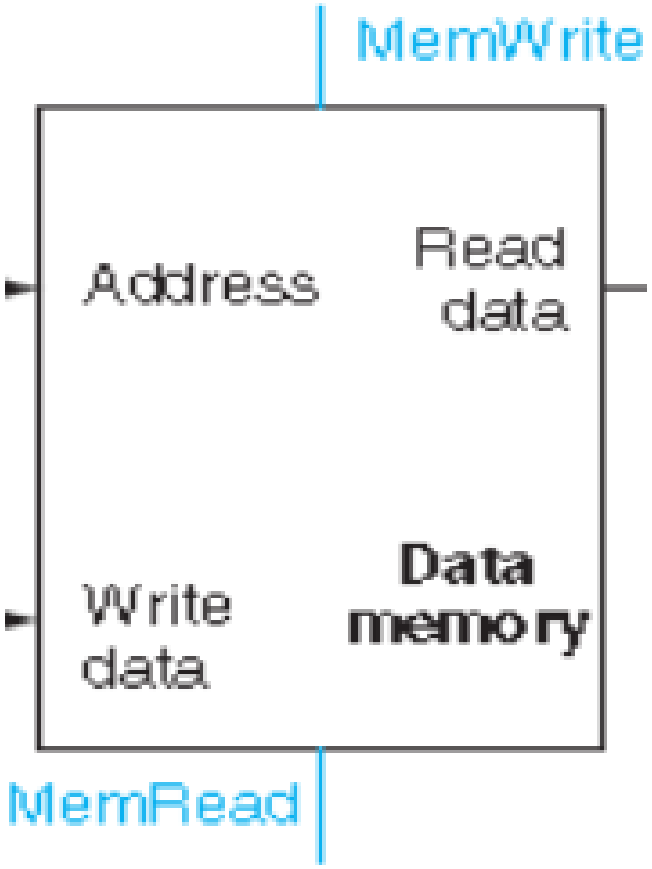

图 2: 内存模块的 IO 定义

类似地，我们让存储器的写数据的操作仅在时钟下降沿进行。具体实现代码如下。

```
    always @(address or memRead or memWrite)
    begin
        readData=(memRead==32'd0)?32'd0:memFile[address];
    end

    always @(negedge Clk)
    begin
       if (memWrite) memFile[address]<=writeData;
    end
```

3.3 有符号扩展单元 (Sign Extension)

将 16 位有符号数扩展为 32 位有符号数，只需要将这个 16 位有符号数的符号位填充在 32 位有符号立即数的高 16 位，再将低 16 位复制到 32 位有符号立即数的低 16 位即可。使用 Verilog 拼接功能可以实现上述需求。

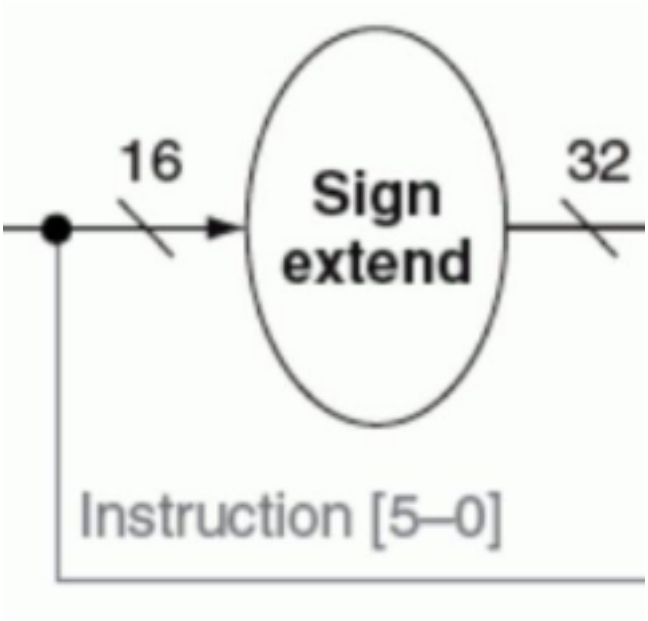

图 3: 带符号扩展单元

具体代码如下。

```
    assign data={{16{inst[15]}},inst};
```

4 功能实现

基于上述原理，易于实现 Register.v, dataMemory.v 和 signext.v。实现后，生成 Register_tb.v, dataMemory_tb.v 和 signext_tb.v 的激励文件用以仿真测试。

5 结果验证

5.1 寄存器 (Register)

在激励文件中对寄存器读写的不同情况进行了测试，测试结果如图 3 所示。

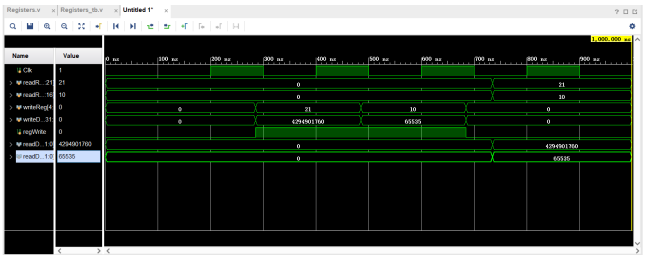

图 4: 寄存器仿真结果

5.2 存储器 (Data Memory)

在激励文件中对存储器读写的不同情况进行了测试，测试结果如图 4 所示。

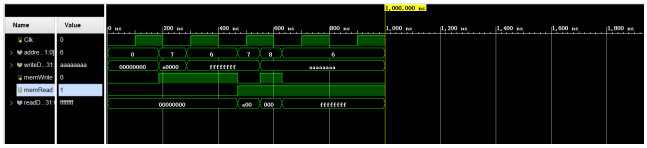

图 5: 存储器仿真结果

5.3 有符号扩展单元 (Sign Extension)

在激励文件中对有符号扩展单元的不同输入都进行了测试，测试结果如图 5 所示。

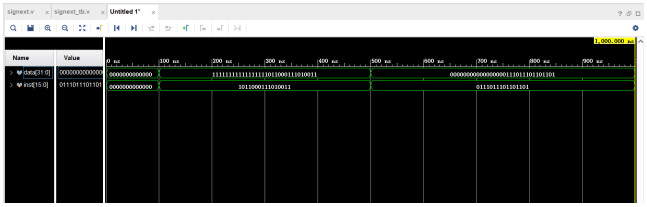

图 6: 有符号扩展单元仿真结果

6 总结与反思

本实验设计并实现了类 MIPS 处理器的三个重要组成部件：寄存器 (Register)、存储器 (Data Memory)、有符号扩展单元 (Sign Extension)，并且通过软件仿真模拟的方法验证了它们的正确性，为后面的单周期类 MIPS 处理器以及流水线处理器的实现奠定基础。