**计算机组成原理课程设计**

**实验报告**

**单周期处理器设计与实现**

**学号 姓名**

**班级：XXXX**

二零一七年九月十八号

**单周期处理器设计与实现**

1. **设计目的**
2. 了解现代计算机硬件设计的基本流程和方法。
3. 了解典型的 RISC 处理器 MIPS 的体系结构。
4. 了解汇编语言到机器语言到计算机执行软硬的逻辑关系。
5. 掌握处理器的设计原理和方法。
6. 培养寄存器级硬件故障的检错和排错能力。
7. **设计内容**

计单周期控制的处理器，支持 MIPS 指令子集:Lui，Addiu，Add，Lw，Sw，Beq，j，以及一条随机抽取的指令。为了简化模型，指令可以不支持溢出。

1. **设备器材**

操作系统：Windows 10

仿真软件：ModelSim

1. **设计原理及内容**

单周期CPU的特点是每条指令的执行需要一个时钟周期，一条指令执行完再执行下一条指令。

单周期控制器的实现：控制信号在整个指令执行过程中不变，用真值表能反映指令和控制信号的关系。根据真值表就能实现控制器。

单周期的数据通路图：

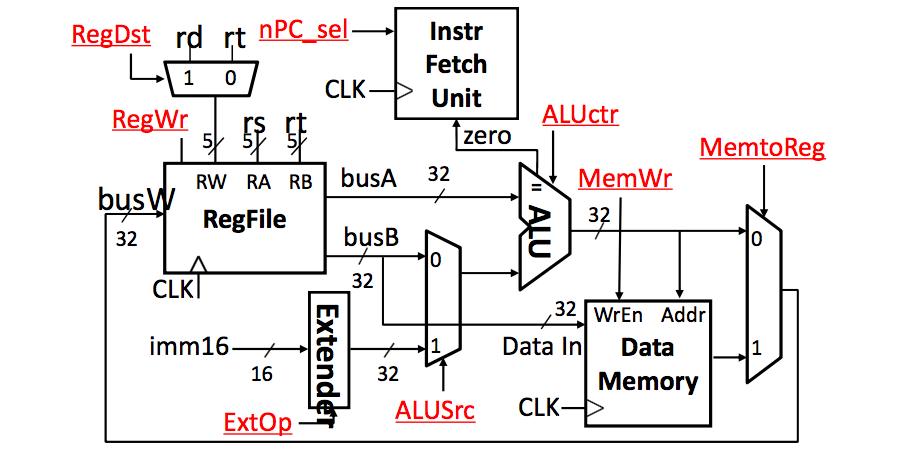


图4-1 单周期处理器数据通路图

1. **设计步骤**

因为老师已经给出了一个支持七条指令的处理器demo，所以我就直接在老师给的demo的基础上直接进行了修改，添加了我抽取到的指令——bgtz。以下步骤是我在理解老师给的程序之后，修改程序所经历的步骤。

## 获得真值表

首先第一步需要根据bgtz的指令说明以及上面的单周期处理器数据通路图来构建真值表。bgtz的指令说明如下：

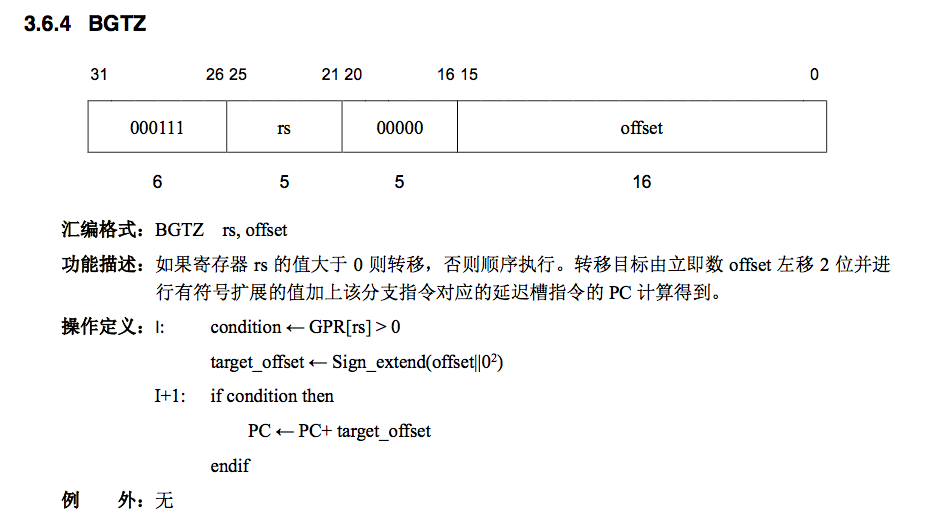


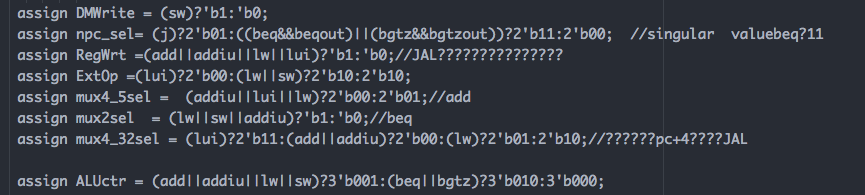
图5-1 bgtz指令说明

结合上面的数据通路图，可以得知bgtz指令影响的控制信号有：nPC\_sel，ExtOp，ALUctr三个控制信号，其中nPC\_sel置为读取offset，ExtOp置为有符号扩展，ALUctr置为减法操作。另外，和beq指令类似，我们还需要添加一个控制信号来判断寄存器rs的值是否大于0，若大于0则置1，否则置0。

## 修改代码

实际修改代码的过程中，我发现ExtOp事实上不用修改，跳转的目标地址的符号扩展已经集成在了nPC模块中，所以我只需要修改两个控制信号：nPC\_sel和ALUctr即可。ctrl模块修改部分：





alu模块修改的部分：



添加一个控制信号来判断跳转条件是否成立。

另外在mips顶层模块中加入bgtzout信号、在ctrl模块中加入bgtz信号即可。

## 实验结果

我们使用了老师提供的Fibonacci数列程序进行了测试，测试结果正确：

1. **设计总结**

本次实验是个人实验，完成一个单周期处理器的设计。因为老师已经给了设计手册以及支持七条指令的一个demo，所以本次实验相对来说还是比较简单的，只要根据老师给出的单周期数据通路图，结合自己抽到的指令，修改每个和抽到的指令有关的控制信号即可，我抽到的指令是bgtz，还需要自己添加一个控制信号来控制跳转条件的输出。实验过程没遇到什么大问题，刚接触ModelSim，使用的不是很熟练，导致调试的时候不是很清楚怎么看想要看的波形，后来经过老师指导，也得到了解决。

经过本次实验，我充分理解到了处理器的工作流程，也算是对理论知识的一次巩固，希望今后还有机会能参与到完整的CPU的设计中去。

1. **参考文献**
2. d1MIPS指令集合和汇编数据通路
3. d2单周期控制和多周期处理器
4. 57条MIPS指令说明
5. MIPS32 Architecture For Programmers Volume II/The MIPS32 Introduction Set
6. 数字设计和计算机体系结构（含目录）