硬件技术课程设计报告

|  |  |  |
| --- | --- | --- |
| 题 | 目： | 单/多周期 MIPS 指令 CPU 设计 |
| 专 | 业： |  |
| 班 | 级： |  |
| 学 | 号： |  |
| 姓 | 名： |  |
| 电 | 话： |  |
| 邮 | 件： |  |
| 日 | 期： |  |



# XX大学 XX学院 硬件技术课程设计报告

## 目 录

1. **课程设计概述.........................................................................................**
   1. 课设目的 ............................................................................................
   2. 设计任务 ............................................................................................
   3. 设计要求 ............................................................................................
   4. 技术指标 ............................................................................................
2. **总体方案设计.........................................................................................**
   1. 单周期CPU 设计................................................................................
   2. 多周期CPU 设计................................................................................
   3. 扩展功能设计.....................................................................................
3. **详细设计与实现 .....................................................................................**
   1. 单周期CPU 实现 ...............................................................................
   2. 多周期CPU 实现................................................................................
   3. 扩展功能实现.....................................................................................
4. **实验过程与调试 .....................................................................................**
   1. 测试用例和功能测试.........................................................................
   2. 可自行安排章节.................................................................................
   3. 性能分析 ............................................................................................
   4. 主要故障与调试.................................................................................
   5. 实验进度 ............................................................................................
5. **设计总结与心得 .....................................................................................**
   1. 课设总结 ............................................................................................
   2. 课设心得 ............................................................................................

**参考文献.......................................................................................................**

I

# XX大学 XX学院 硬件技术课程设计报告

一、原创性声明

本人郑重声明本报告内容，是由作者本人独立完成的。有关观点、方法、数据和文献等的引用已在文中指出。除文中已注明引用的内容外，本报告不包含任何其他个人或集体已经公开发表的作品成果，不存在剽窃、抄袭行为。

特此声明！

作者签字（电子手写签名）:

II

## 课程设计概述

* 1. 课设目的

掌握控制器设计的基本原理，能利用硬布线控制器的设计原理在 Logisim 平台中设计实现 MIPS 单周期处理器；掌握微程序控制器设计的基本原理，能利用微程序控制器的设计原理设计实现多周期 MIPS 处理器。

* 1. 设计任务

1. MIPS 单周期 CPU 设计
2. MIPS 多周期 CPU 微程序控制器设计
3. MIPS 指令集汇编语言设计
4. 仿真测试
   1. 设计要求

构造单周期和多周期 MIPS 处理器，要求能支持资料中所给的 8 条 MIPS 核心指令，最终设计实现的 MIPS 处理器能运行实验包中的冒泡排序测试程序 sort1.asm，该程序自动在数据存储器的 80~8F 号字单元中写入 16 个数据，然后利用冒泡排序将数据升序排序。实验电路应该能够自动统计指令数目和时钟周期数。

* 1. 技术指标

## 总体方案设计

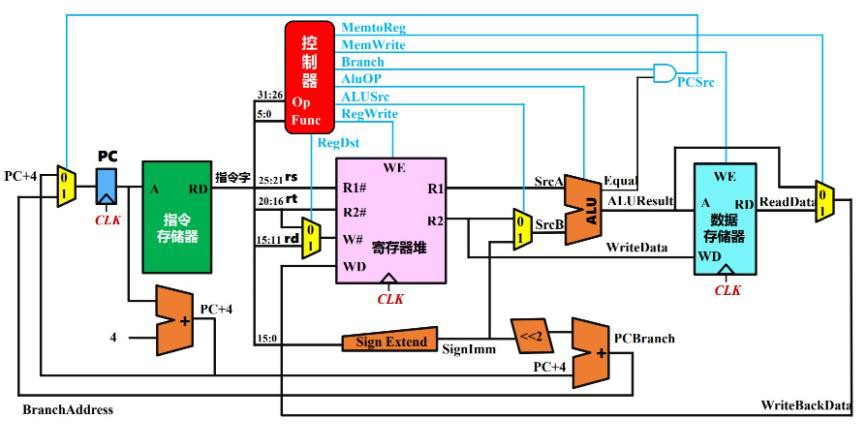
* 1. 单周期CPU 设计

实验目的：掌握硬布线控制器设计的基本原理，并能利用相关原理在 Logisim 平台中设计实现 MIPS 单周期 CPU。

主要任务：绘制 MIPS CPU 数据通路；实现单周期硬布线控制器；基于 8 条 MIPS 核心指令集完成冒泡排序测试程序设计；测试联调与验证

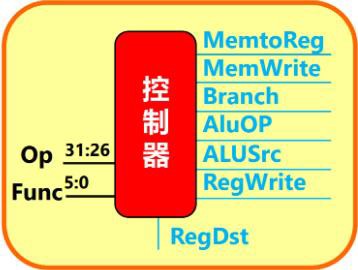
单周期 MIPS CPU 数据通路总框图：

如图，主要利用一个硬布线控制器，指令寄存器，PC 计数器、寄存器堆，ALU 加法器，数据存储器来实现。这里的连接就按照给出的框图进行连接就可以了。

停机是靠位于左上方的计数器，计算周期数。在计数器中设置最大值为 224，当周期达到 224 时即可停机。

（2）设计单周期 MIPS 控制器

首先是指令译码逻辑的设计，该实验只涉及 8 条核心的 MIPS 指令。而这 8 条 MIPS 指令的指令字段已经在附件中给出，并且电路底部文字也给出了关于 SYSCALL 的提示，因此，这部分只需根据相应的 OP 和 FUNC 字段进行简单地逻辑比较就可实现。



其次，是 ALU 控制逻辑的设计，由于该 MIPS CPU 设计中有关的 8 条核心 MIPS 指令中，对于 ALU 运算逻辑单元中只涉及到加法和比较，因此这一部分可以大大简化。只有运行 STL 指令时，需要选择比较运算，其余都是加法运算。

最后，对于控制器输出信号的设计，则要根据硬布线控制器中所包含的 9 中控制信号

进行分析，如下图。主要考虑每种控制信号的产生条件。这里更高的要求是掌握 8 条核心指令集在执行时全部周期中所设计的控制信号，已经使用相应控制信号的作用。

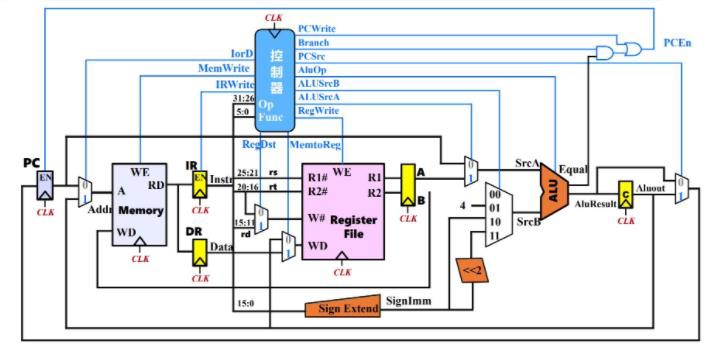


* 1. 多周期CPU 设计

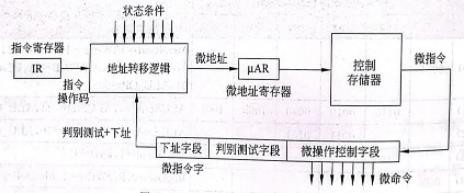
实验目的：掌握多周期 MIPS CPU 设计原理；掌握微程序控制器设计的基本原理；利用微程序控制器的设计实现多周期 MIPS 处理器

主要任务：绘制多周期 MIPS CPU 数据通路；实现微程序控制器；测试联调

多周期 MIPS CPU 数据通路总框图：

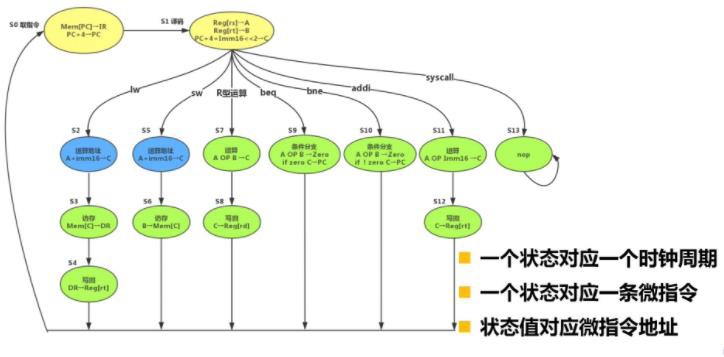


微程序控制器组成框图：主要由控制存储器、地址转移逻辑、微地址寄存器三部分组成。



控制存储器用于存放全部指令的所有微程序，控制存储器的字长等于微指令的长度，其存储容量取决于指令系统；微地址寄存器用于为控制存储器提供微指令地址，初始化时为零，所有控制存储器 0 号单元应该为取指令微指令，微指令寄存器输入来源为地址转移逻辑。；地址转移逻辑，根据指令操作码、外部状态条件、微指令判别测试字段、下址字段共同决定微地址共同决定微地址寄存器的输入。

根据指令状态变换图来完成地址转移逻辑的设计以及整体 CPU 数据通路的设计。



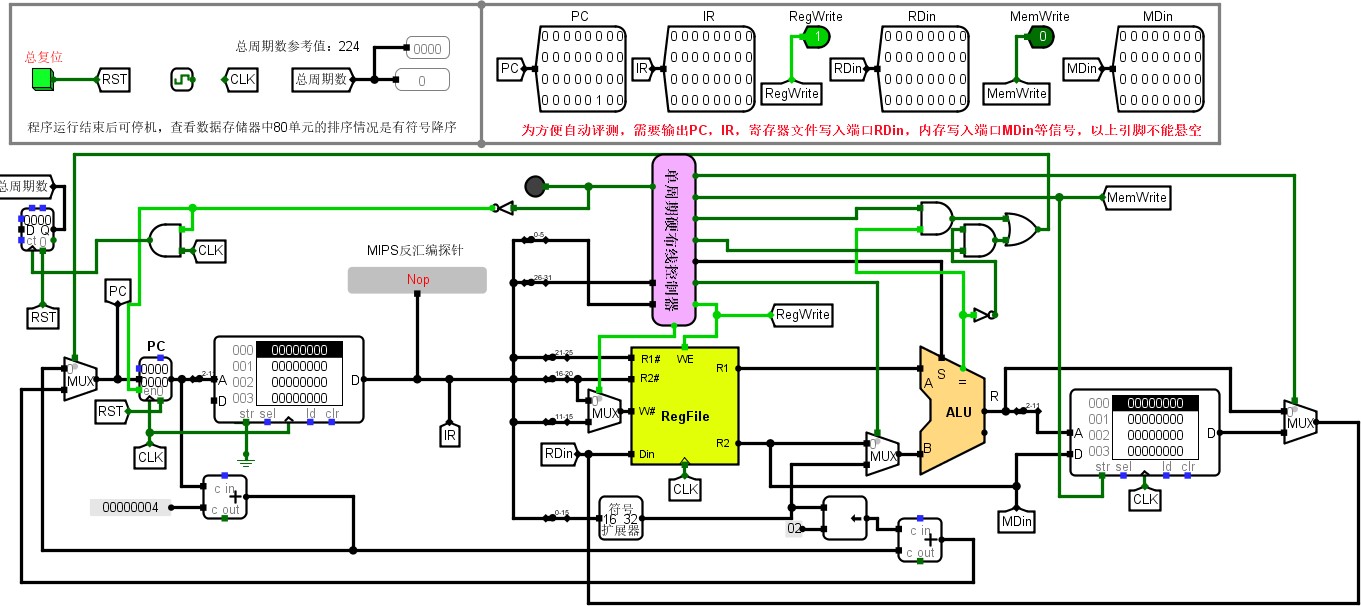
在相应的 EXCEL 表中进行填写，主要是关于机器指令信号及其相关的微程序入口地址的填写。填写完毕之后利用 LOGISIM 的自生成功能来实现电路要求的功能。



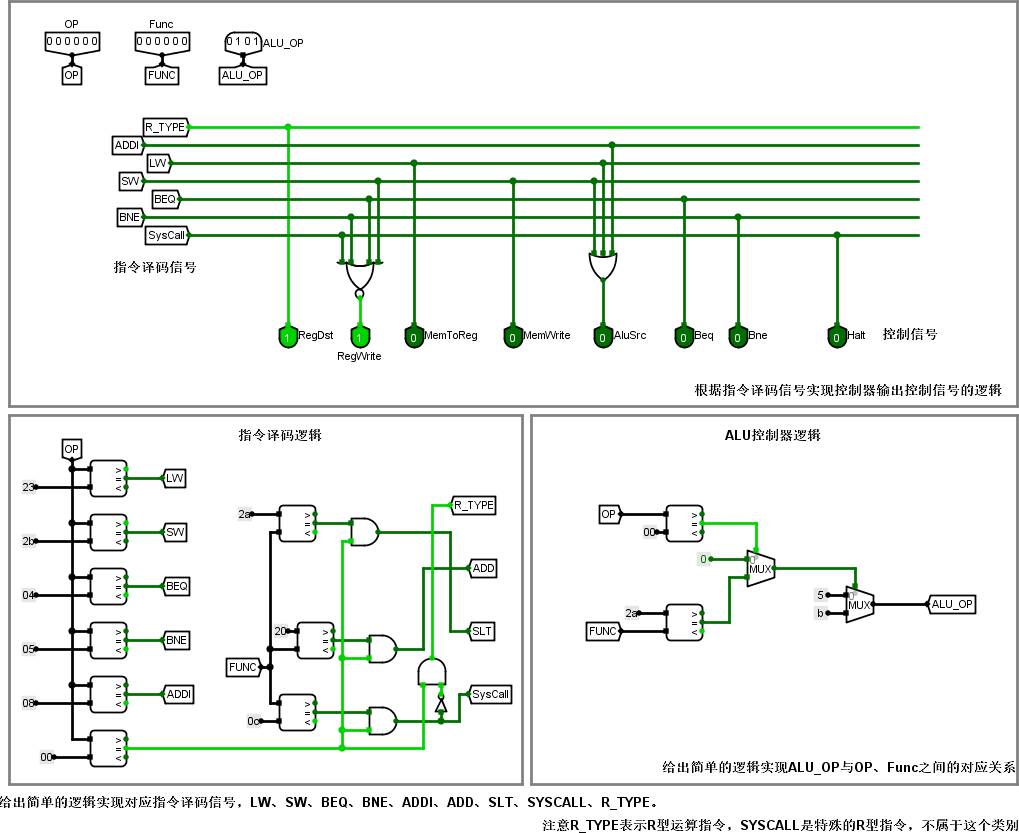
## 详细设计与实现

* 1. 单周期CPU 实现

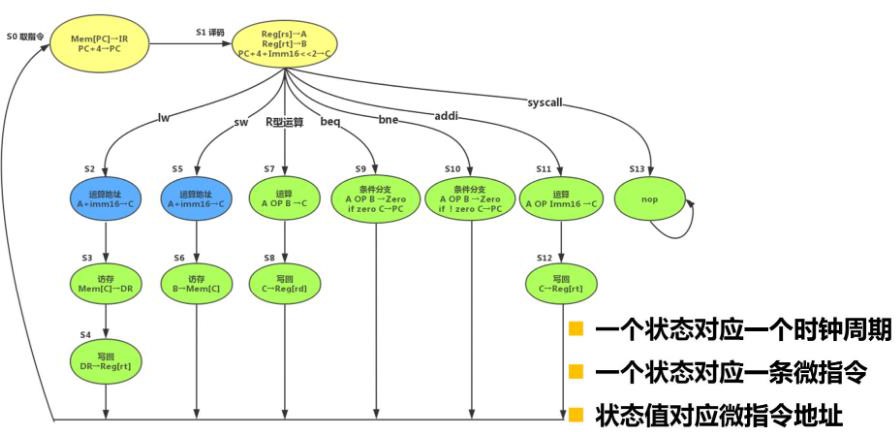
1. 设计硬件布线图
2. 构建 MIPS 主机通路



1. 设计单周期 MIPS 控制器



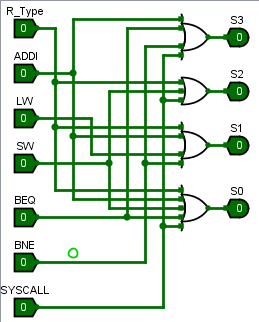
1. 微程序地址转移逻辑设计
   1. 设计原理

主要根据的下面这个指令状态变换图，这个状态图是整个微程序 CPU 设计的核心。根据这个图，在微程序地址转移逻辑自动生成 Excel 表中可以自动生成地址转移逻辑。

在相应的 EXCEL 表中进行填写，主要是关于机器指令信号及其相关的微程序入口地址的填写。填写完毕后，

利用 Logisim 的自生成功能，就可以实现电路要求的功能了。

* 1. **设计硬件布线图**

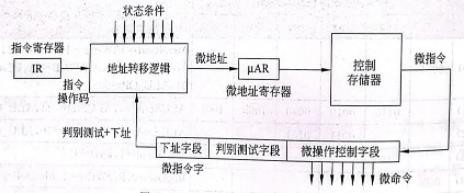


在相应的 EXCEL 表中进行填写机器指令信号及其相关的微程序入口地址的填写。填写完毕后，利用 Logisim

的自生成功能，就可以实现电路要求的功能。

* 1. 多周期CPU 实现

微程序控制器组成框图：主要由控制存储器、地址转移逻辑、微地址寄存器三部分组成。



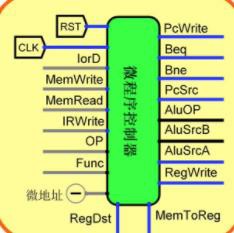
控制存储器用于存放全部指令的所有微程序，控制存储器的字长等于微指令的长度，其存储容量取决于指令系统；微地址寄存器用于为控制存储器提供微指令地址，初始化时为零，所有控制存储器 0 号单元应该为取指令微指令，微指令寄存器输入来

源为地址转移逻辑。；地址转移逻辑，根据指令操作码、外部状态条件、微指令判别测试字段、下址字段共同决定微地址共同决定微地址寄存器的输入。

设计微程序控制器：

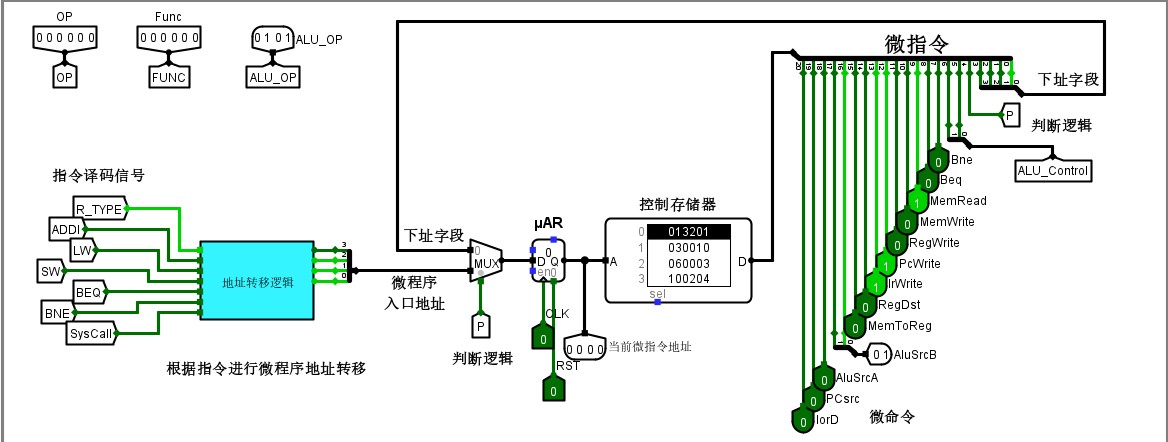
输入信号：指令字 Opcode，func 字段（12 位）；时钟信号、复位信号

输出信号：多路选择器信号；内存访问控制信号；寄存器写使能信号；运算器控制信号、指令译码信号。

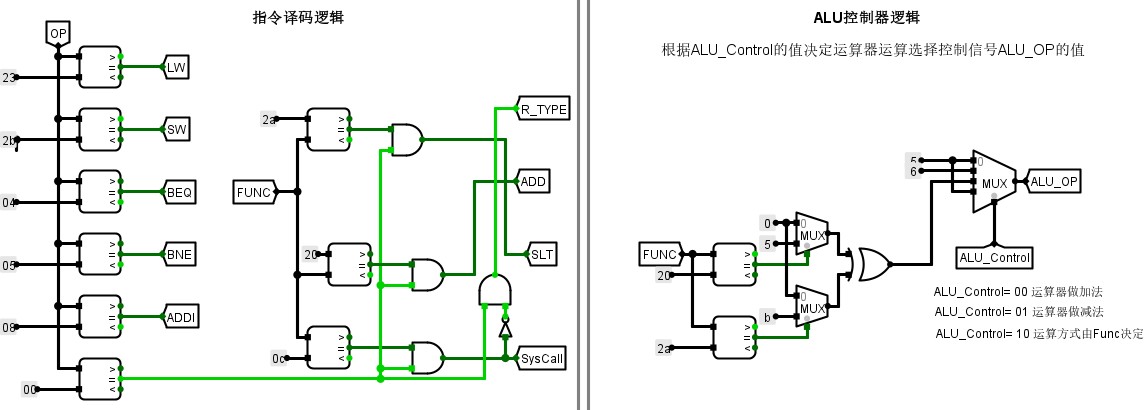


微程序控制器内部架构：

载入微程序，设计地址转移逻辑



完善控制器内部逻辑：首先完成指令译码逻辑、ALU 控制器逻辑



实现微程序地址转移逻辑：填写微程序地址入口表，自动生成微程序地址转移逻辑子电路。

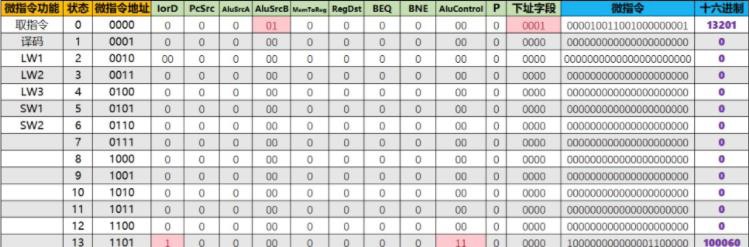


微程序地址入口：



根据状态图构建微程序：状态值→微指令地址

不同状态→微控制信号、P 字段设置、下段地址→微指令→微程序



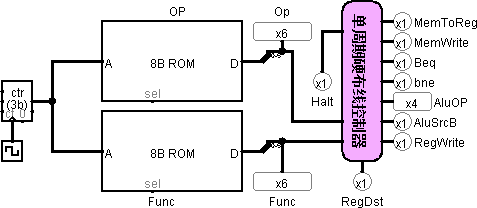


## 实验过程与测试

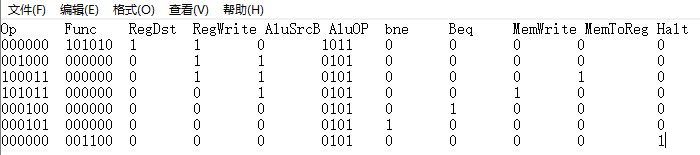
* 1. 测试用例和功能测试

### 、单周期控制信号测试

1. 测试电路如下图，ROM 中依次存放了指令 add、slt、addi、lw、 sw、beq、bne、syscall 的 op 和 func 字段；

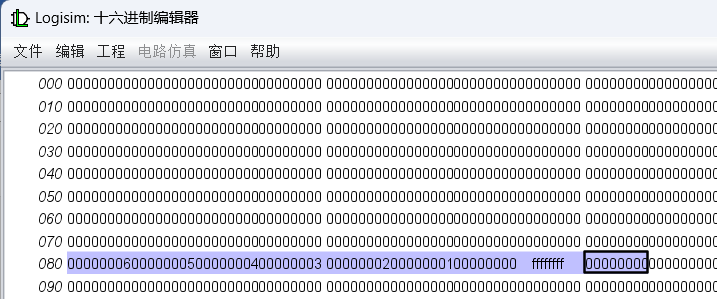


1. 在 Logisim 菜单->模拟->记录器中，将探测器添加记录，并将基数设置为二进制，得到日志记录如下图，经过较对结果符合预期。



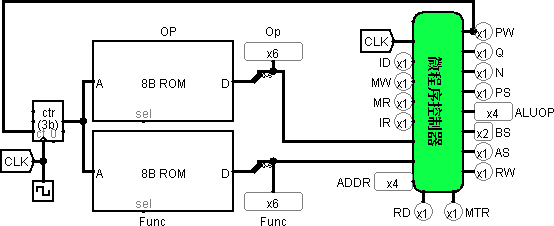
### 单周期 CPU 运行测试程序 sort.hex

测试结果：在 80 号单元处出现 6,5,4,3,2,1,0,ffffffff 的有符号降序数据，如图所示

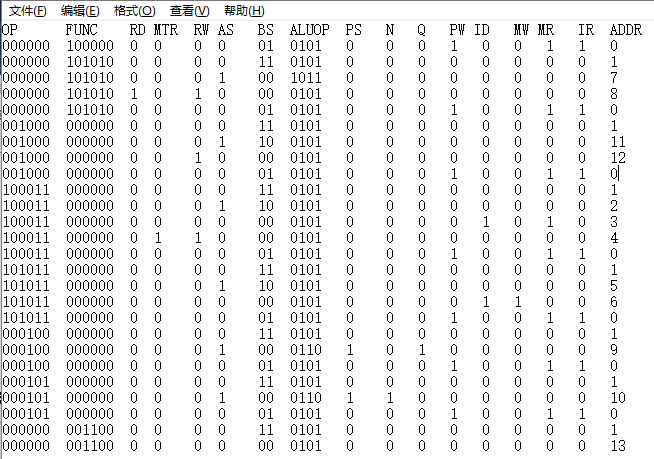


### 、多周期微程序控制信号测试

1. 测试电路如下图，ROM 中依次存放了指令 add、slt、addi、lw、sw、beq、 bne、syscall 的 op 和 func 字段；

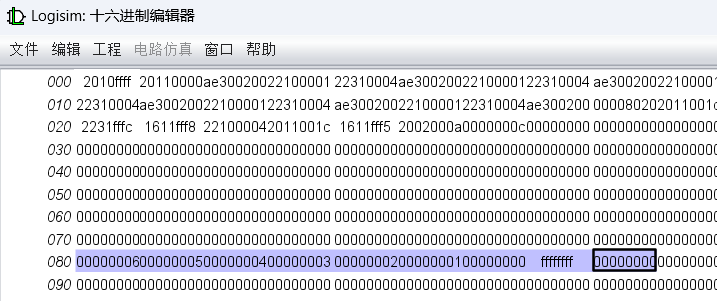


1. 在 Logisim 菜单->模拟->记录器中，将探测器添加记录，并将基数设置为二进制，其中 RD 为 RegDst，MTR 为 MemToReg，RW 为 RegWrite，AS为 AluSrcA，BS 为 AluSrcB，PS 为 PCSrc，ID 为 IorD，MW 为 MemWrite， MR 为 MemRead，IR 为 IRWrite，ADDR 为微程序地址，具体的日志记录如下图，经过比对得知结果正确。

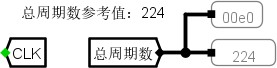


### 、多周期 CPU 运行测试程序 sort.hex

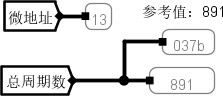
测试结果：在 80 号单元处出现 6,5,4,3,2,1,0,ffffffff 的有符号降序数据，如图所示：



* 1. 性能分析
     1. **单周期 CPU**：测试用例执行完毕后，系统停机，时钟周期数为 224，如图：

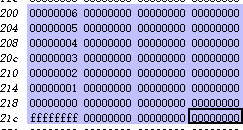


* + 1. **多周期 CPU**：测试用例执行完毕后，系统停机，时钟周期数为 891，如图：

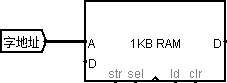


* 1. 主要故障与调试

### 、单周期数据存储器地址错误

故障现象： sort.hex 程序执行完毕后，未在指定的 80 号存储单元形成降序排列的数据，而是在 200 号存储单元形成降序排列数据，而且两个数据地址之差为 4。

原因分析**：**Logisim 中 RAM 只支持一种访问模式，一次访问读出 32 位数据，直接给出字地址使得内存布局错误，如下图：



解决方案：字地址除以 4 即可得到正确的内存布局，即高两位补零作为字节地址送入存储器地址即可

### 、指令问题

故障现象：三级时序中 beq 指令计算周期第二个节拍给出的信号错误。

原因分析：一开始多给了一个 slt 信号，我以为要减一下再判断 equal，但是查看 ALU 电路发现 equal 不需要给 ALU 操作信号。

解决方案：去掉 beq 指令计算周期第二个节拍给出的 slt 信号。

* 1. 实验进度

|  |  |  |  |
| --- | --- | --- | --- |
| 第一周 | 第二周 | 第三周 | 第四周 |
| 回顾计组知识点；观看了慕课视频资料，学习了课程资料内容，搜索整理了 MIPS 指令及汇编的相关知识 | 完成了单周期 CPU 总体设计；完成了单周期 CPU 硬布线控制器及单周期 CPU 总体通路绘制，并通过了仿真  测试 | 了解了多周期 CPU 的结构和原 理，完成了多周期 CPU 地址转移逻辑及的设计；完成了多周期 CPU 总体设  计； | 完成了多周期 CPU 微程序控制器及总体通路绘制，通过了仿真测试；对实验进行了初步总结。 |

## 设计总结与心得

* 1. 课设总结

通过硬件课程设计的学习，我学会了使用工程化方法自动生成组合逻辑电路，及常用的 debug 方法。熟悉了 RISC-V 核心指令在 CPU 中的执行流程，并通过这些指令的实现了解了单总线 CPU 一般指令执行流程的分析方法。了解了 CPU 处理中断的方式以及单总线 CPU 上简单的支持单级中断的硬件设计方法，对计算机技术有了深入的研究。

* 1. 课设心得

在实验中对于课堂上讲授的 CPU 知识有了更深刻的感受，虽然在课堂和书本 上学习了很多东西，但是实际动手做了之后还是很有不一样的感受。

硬件课程设计让我对计算机的硬件组成有了一个总体的认识，同时也和上个学期的数字逻辑、计算机组成原理等课程呼应，让我知道了同步时序逻辑电路的真正作用。在实验的过程中遇到了不少的问题，不过答疑群里面的老师在我们的实验的过程中对于我们提出的问题都给出了很及时和详细的解答，同时群里面也有很多同学们在热心的回答我们的问题，很感谢老师的付出和大家的帮助。

硬件技术课程设计课程评分表

班级：计科四班 学号：202084160431 姓名：徐铭阳

1. 课程设计过程与结果成绩

|  |  |  |
| --- | --- | --- |
| 课程设计过程与结果成绩 | 标准分 | 得分 |
| 在仿真平台上完成单周期CPU 和多周期CPU 的设计和仿真；在 Educoder 平台上对单周期CPU 和多周期 CPU 进行测试，并结果正确 | 90-100 |  |
| 在仿真平台上完成单周期CPU 并在 Educoder 平台上对单周期CPU 进行测试或完成仿真平台上多周期CPU 的设计和仿真。 | 80-89 |  |
| 在仿真平台上完成单周期 CPU 的设计和仿真；能完成部分多周期 CPU 仿真设计，但CPU 未完全实现； | 70-79 |  |
| 在仿真平台上完成单周期CPU 的设计和仿真； | 60-69 |  |
| 不能在仿真平台上完成单周期CPU 的设计和仿真； | 60 以下 |  |

按照课程设计过程与结果成绩要求，按百分制评分，按 50%折算成总评成绩。

1. 课程设计报告成绩

|  |  |  |
| --- | --- | --- |
| 课程设计报告成绩 | 标准分 | 得分 |
| 能很好地完成设计要求的每一部分；方案设计正确；设计分析准确无误、论述透彻清晰严密、书写工整、格式符合要求。 | 90-100 |  |
| 能较好完成设计要求的每一部分；方案设计正确；设计分析基本无错误、论述清晰明白、书写比较工整、格式符合要求。 | 80-89 |  |
| 能较好完成设计要求的每一部分；方案设计能满足要求；论述基本准确，格式基本符合要求。 | 70-79 |  |
| 基本能完成设计要求的每一部分；方案设计基本正确；格式基本符合规范。 | 60-69 |  |
| 只能完成部分设计要求；有方案的设计和分析，但存在明显的抄袭；格式不符合规范。 | 60 以下 |  |

根据课程设计说明书撰写要求，按百分制评分，按 50%折算成总评成绩。总评成绩

|  |  |  |
| --- | --- | --- |
| 课程设计过程与结果成绩\*50% | 课程设计报告成绩\*50% | 总成绩 |
|  |  |  |