

华中科技大学

2019

计算机组成原理

· 实验报告 ·

专 业： 计算机科学与技术

班 级： CS1704

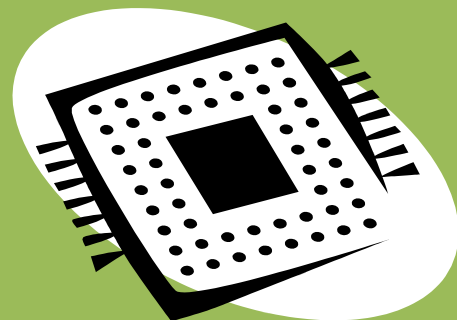
学 号： U201714626

姓 名： 汪清

电 话： 18134687326

邮 件： 2749374330@qq.com

完成日期： 2019/12/21



计算机科学与技术学院

华中科技大学课程实验报告

目录

1 CPU 设计实验.....	2
1.1 设计要求.....	2
1.2 方案设计.....	2
1.3 实验步骤.....	6
1.4 故障与调试.....	14
1.5 测试与分析.....	14
2 总结与心得.....	16
2.1 实验总结.....	16
2.2 实验心得.....	16
参考文献.....	17

1 CPU 设计实验

1.1 设计要求

利用运算器实验中设计的运算器 ALU、寄存器 Regfile、Logisim 中已有的功能部件实现一个单周期 MIPS CPU 设计（8 条指令）、一个多周期 MIPS CPU 微程序控制器设计以及一个多周期 MIPS CPU 硬布线控制器设计。

1.2 方案设计

1.2.1 单周期 MIPS CPU 设计

单周期 MIPS CPU 设计的所有指令均在一个时钟周期内完成，CPI=1，其性能取决于最慢的指令，时钟周期较长。单周期采用哈佛结构，一种将程序指令存储和数据存储分开的存储器结构。哈佛结构是一种并行体系结构，它的主要特点是将程序和数据存储在不同的存储空间中，即程序存储器和数据存储器是两个独立的存储器，每个存储器独立编址、独立访问。因此，单周期的设计中不能设置 AR、DR、IR 等寄存器，运算器和 PC 累加器也分离。

单周期 MIPS CPU 的取指令数据通路如图 1-1 所示。R 型指令、Lw 型指令、sw 型指令等指令的建立过程绘制相应的数据通路，将这些有细微差别的通路综合到一起，绘制出单周期 MIPS CPU 的综合数据通路，如图 1-2 所示。单周期 MIPS 控制器输入有指令字 op 和 func 字段等，输出有多路选择器选择信号、寄存器写使能信号、内存访问控制信号、运算器控制信号、指令译码信号等，其设计如图 1-3 所示。

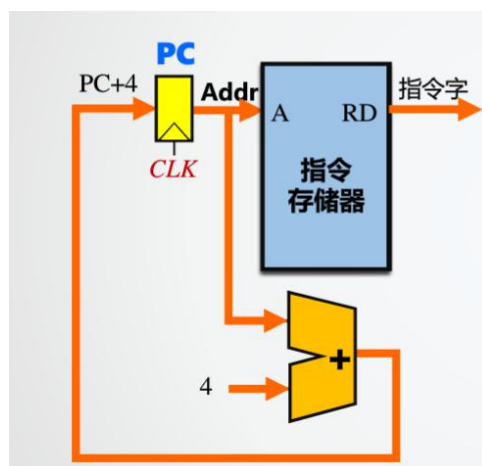


图 1-1 单周期 MIPS CUP 取指令数据通路

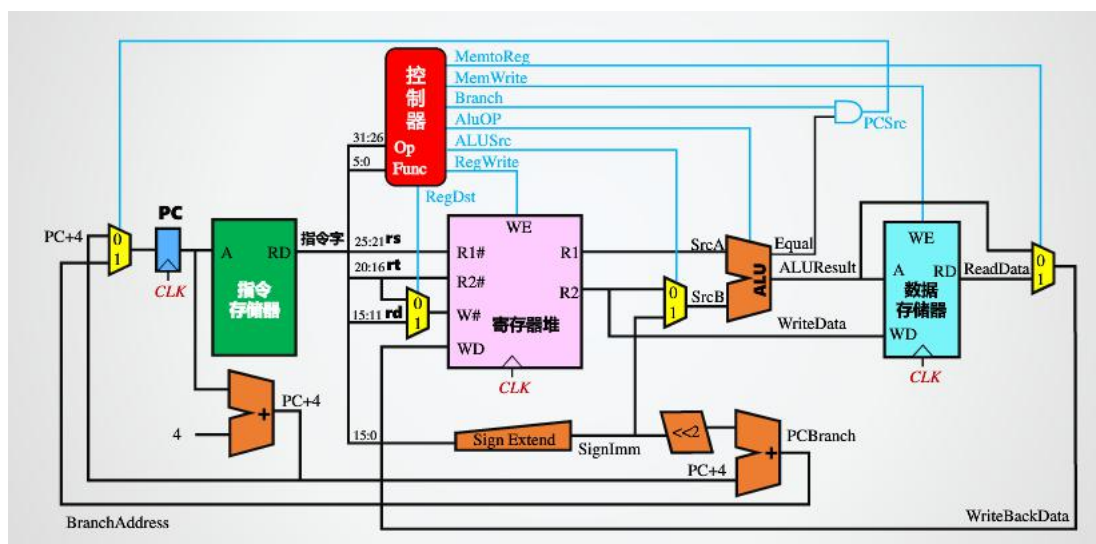


图 1-2 单周期 MIPS 数据通路

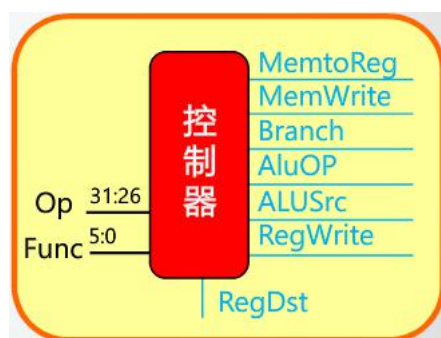
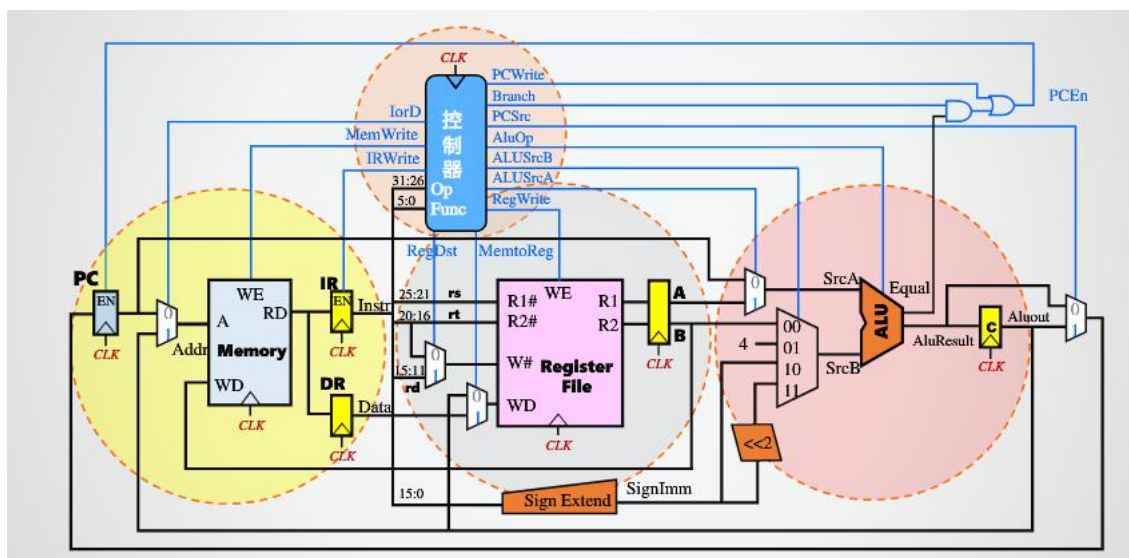


图 1-3 单周期 MIPS CPU 控制器设计

1.2.2 多周期 MIP CPU（微程序）设计

多周期 MIPS CPU 不再区分指令/数据存储器，而是分时使用功能部件，多周期

华中科技大学课程实验报告



多周期 MIPS CPU 取指令阶段 T1 如图 1-5 所示

图 1-4 多周期 MIPS CPU 取指令阶段 T1

华中科技大学课程实验报告

值在运算器中相加，即可得到下一条指令的地址，将其送回 PC。

多周期 MIPS CPU 取指令阶段 T2 如图 1-6 所示

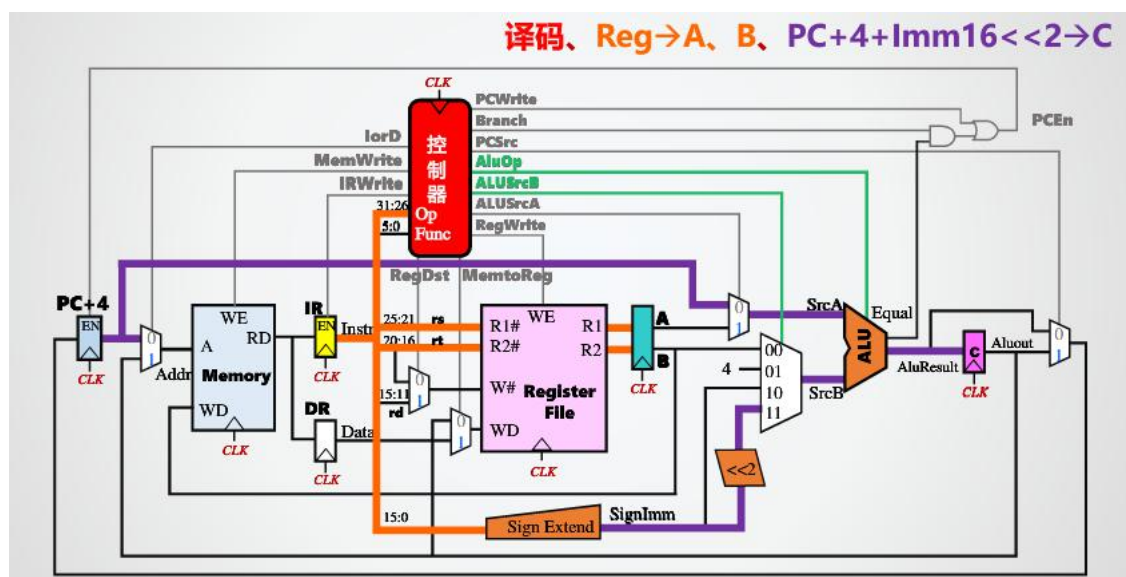


图 1-6 多周期 MIPS 取指令阶段 T2

此阶段将 IR 中的寄存器进行译码，然后取操作数送入寄存器 AB 中，同时还能进行分支地址的计算。

此后，不同形式的指令会在控制器的控制之下，以不同的方式在数据通路进行实现，状态转换图如图 1-7 所示。

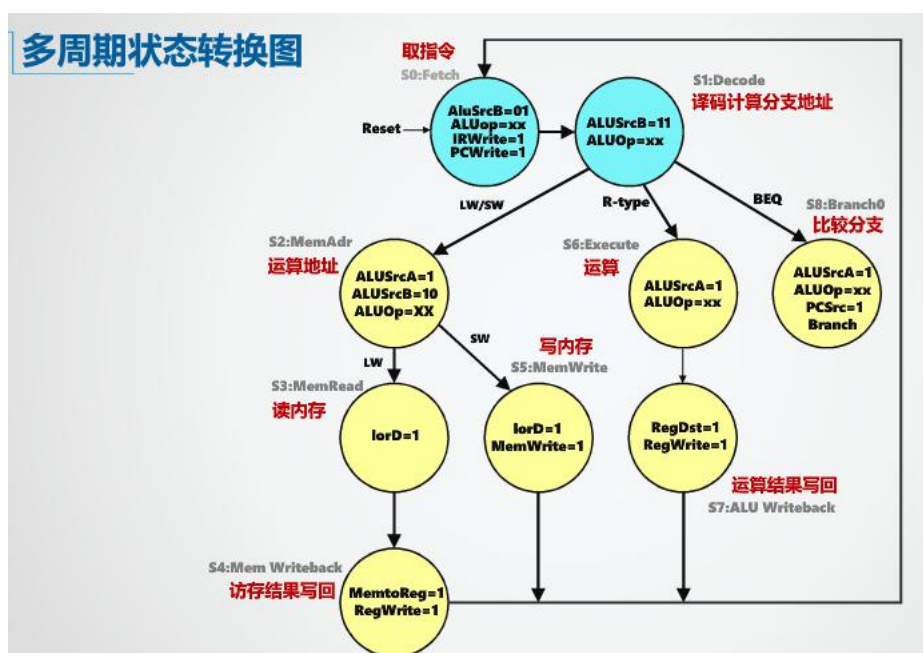


图 1-7 多周期状态转换图

华中科技大学课程实验报告

数据通路的设计并不困难，此次实验的难点在于微程序控制器的设计。这里可以使用老师提供的 excel 表格进行自动生成。填写微地址转移逻辑的自动生成表可以生成地址转移逻辑部件中的电路，填写微指令自动生成表可以将微指令转换成 16 进制的指令形式，并填写到存储控制器中。

1.2.3 多周期 MIPS CPU（硬布线）设计

多周期硬布线的数据通路与微程序相同，区别只在控制器不一样。硬布线使用控制器状态机代替了地址转移逻辑，其内部电路也可以通过老师提供的硬布线控制器状态转换逻辑的自动生成表自动生成，在表格中填写状态转换表，输入现态和指令译码信号，输出次态，表格会自动生成次态逻辑表达式，将其复制到 logisim 的“分析电路”中即可生成 FSM 的电路。而实验的难点也正在填写该表格。

1.3 实验步骤

(1) 单周期 MIPS CPU

参考方案设计，在 logisim 中绘制单周期 MIPS CPU 数据通路如图 1-8 所示

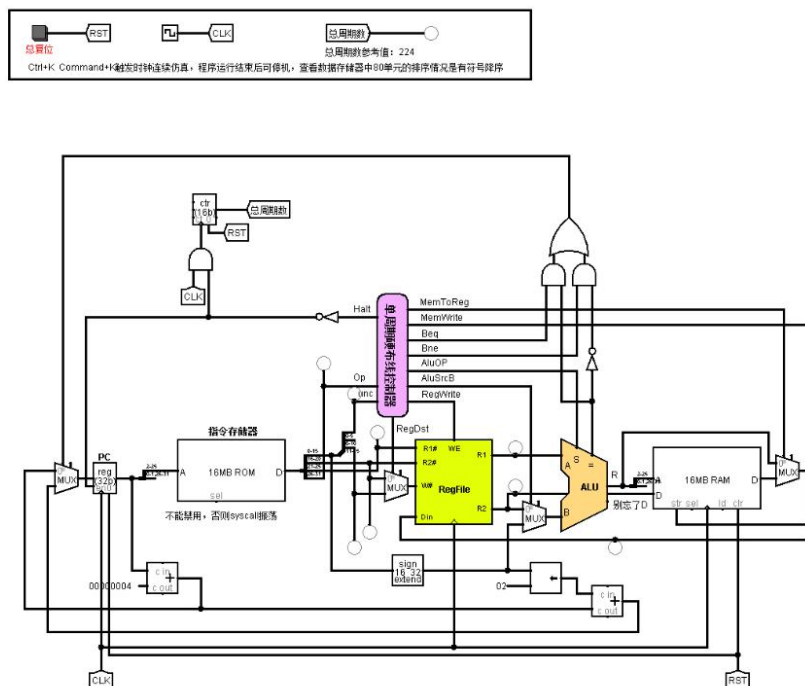


图 1-8 单周期 MIPS CPU 数据通路

华中科技大学课程实验报告

根据单周期硬布线控制器各个端口控制信号的功能，如图 1-9 所示，绘制硬布线控制器的内部逻辑，如图 1-10 所示。指令译码逻辑中指令对应的十六进制编码可以通过查表（MIPS_GREEN_SHEET）得到。

#	控制信号	信号说明	产生条件
1	MemToReg	写入寄存器的数据来自存储器	lw指令
2	MemWrite	写内存控制信号	sw指令 未单独设置MemRead信号
3	Beq	Beq指令译码信号	Beq指令
4	Bne	Bne指令译码信号	Bne指令
5	AluOP	运算器操作控制符	加法，比较两种运算
6	AluSrcB	运算器第二输入选择	Lw指令，sw指令，addi
7	RegWrite	寄存器写使能控制信号	寄存器写回信号
8	RegDst	写入寄存器选择控制信号	R型指令
9	Halt	停机信号，取反后控制PC使能端	syscall指令

1-9 控制信号功能说明

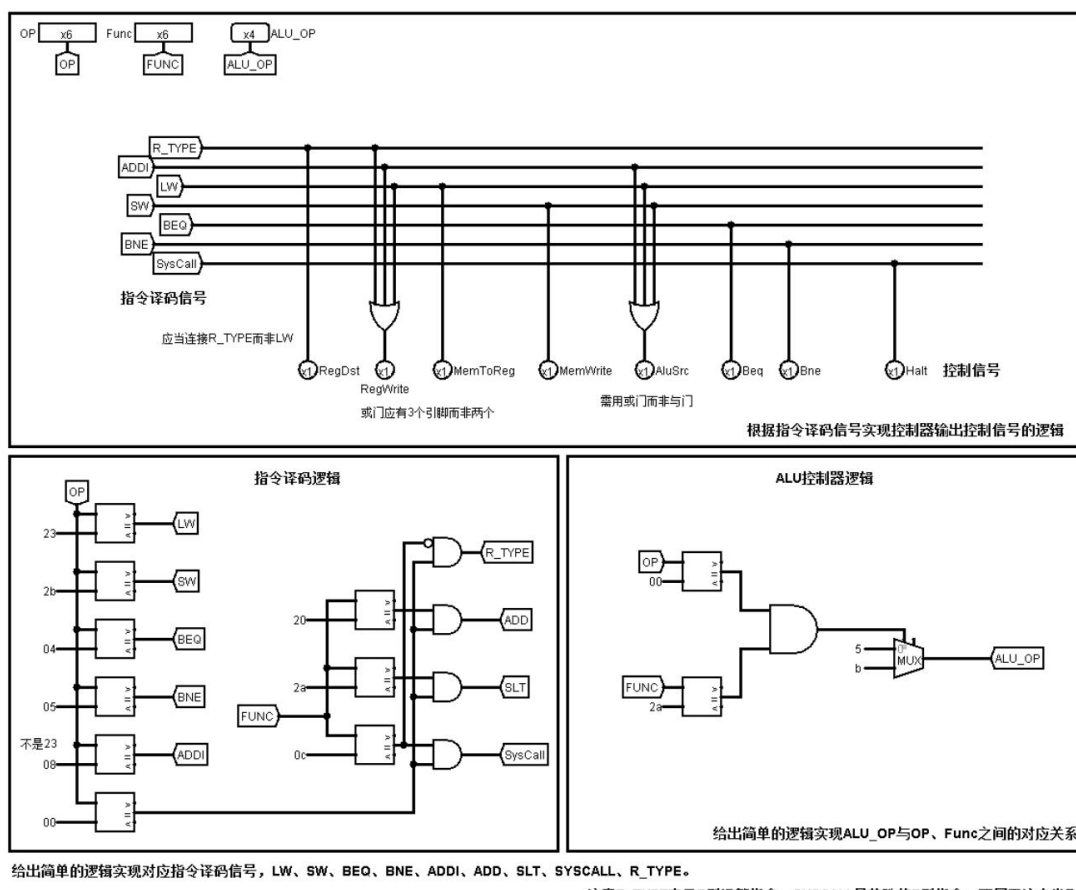


图 1-10 单周期硬布线控制器

(2) 多周期 MIPS CPU（微程序）

参照方案设计，在 logisim 中绘制多周期 MIPS CPU（微程序）的数据通路，如图 1-11 所示。停机 syscall 的微地址为 1101，故将微地址信号送入比较器中

华中科技大学课程实验报告

与 d (1101) 作比较, 如果相等, 检测到停机, 即可以使用计数器对总周期数进行计数。

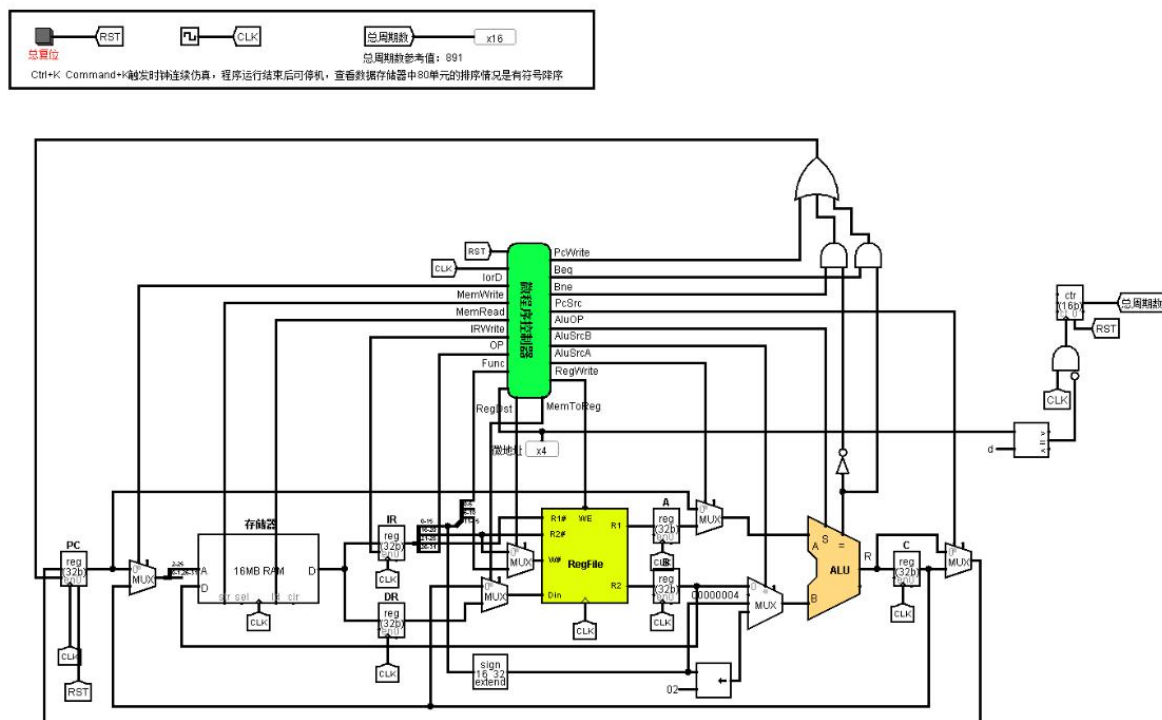


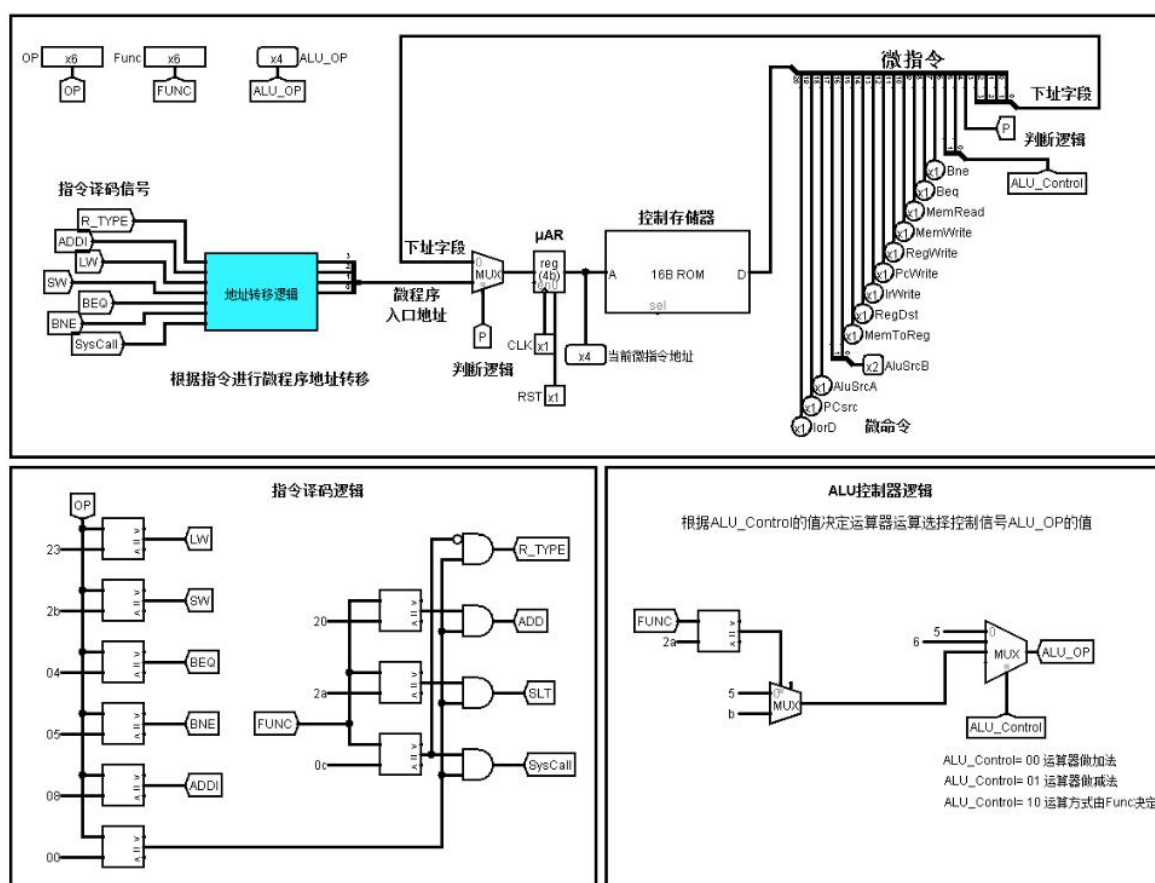
图 1-11 多周期 MIPS CPU (微程序) 数据通路

根据多周期微程序控制器各个端口控制信号的功能, 如图 1-12 所示, 绘制微程序控制器的内部逻辑, 如图 1-13 所示。指令译码逻辑中指令对应的十六进制编码可以通过查表 (MIPS_GREEN_SHEET) 得到。ALU 控制逻辑中, 当 ALU_Control=00 时, 运算器做加法, 当 ALU_Control=01 时, 运算器做减法, 当 ALU_Control=10 时, 运算方式由 Func 决定。地址转移逻辑可以通过表格自动生成, 如图 1-14, 1-15 所示, 将表达式输入 logisim 中可以自动生成地址转移逻辑电路, 如图 1-17 所示。控制存储器中的指令也可以通过表格自动生成, 如图 1-16 所示。

华中科技大学课程实验报告

#	控制信号	信号说明	产生条件
1	PCWrite	PC写使能控制	取指令周期, 分支指令执行
2	IorD	指令还是数据	0表示指令, 1表示数据
3	IRwrite	指令寄存器写使能	高电平有效
4	MemWrite	写内存控制信号	sw指令
5	MemRead	读内存控制信号	lw指令, 取指令
6	Beq	Beq指令译码信号	Beq指令
7	Bne	Bne指令译码信号	Bne指令
8	PcSrc	PC输入来源	顺序寻址还是跳跃寻址
9	AluOP	运算器操作控制符 4位	ALU_Control控制, 00加, 01减, 10由Funct定
10	AluSrcA	运算器第一输入选择	
11	AluSrcB	运算器第二输入选择	Lw指令, sw指令, addi
12	RegWrite	寄存器写使能控制信号	寄存器写回信号
13	RegDst	写入寄存器选择控制信号	R型指令
14	MemToReg	写入寄存器的数据来自存储器	lw指令

图 1-12 控制器控制信号功能



给出简单的逻辑实现对应指令译码信号, LW、SW、BEQ、BNE、ADDI、ADD、SLT、SYSCALL、R_TYPE。

注意R_TYPE表示R型运算指令, SYSCALL是特殊的R型指令, 不属于这个类别

图 1-13 多周期微程序控制器

机器指令译码信号							微程序入口地址				
R_Type	ADDI	LW	SW	BEQ	BNE	SYSCALL	入口地址 10进制	S3	S2	S1	S0
1							7	0	1	1	1
	1						11	1	0	1	1
		1					2	0	0	1	0
			1				5	0	1	0	1
				1			9	1	0	0	1
					1		10	1	0	1	0
						1	13	1	1	0	1

图-14 微程序地址转移逻辑

[illegible]

图 1-15 微程序地址转移逻辑

译码	1	0001	0	0	0	11	0	0	0	0	0	0	0	0	0	00	1	0000	0001100000000000010000	30010
LW1	2	0010	00	0	1	10	0	0	0	0	0	0	0	0	0	00	0	0011	0001100000000000000011	60003
LW2	3	0011	1	0	0	00	0	0	0	0	0	0	0	1	0	00	0	0100	10000000001000000100	100204
LW3	4	0100	0	0	0	00	1	0	0	0	1	0	0	0	0	00	0	0000	0000010001000000000000	8800
SW1	5	0101	0	0	1	10	0	0	0	0	0	0	0	0	0	00	0	0110	0011000000000000000110	60006
SW2	6	0110	1	0	0	00	0	0	0	0	0	0	1	0	0	00	0	0000	1000000000100000000010	100400
R_Type1	7	0111	0	0	1	00	0	0	0	0	0	0	0	0	0	10	0	1000	0010000000000010010000	40048
R_Type2	8	1000	0	0	0	00	0	1	0	0	1	0	0	0	0	00	0	0000	0000001001000000000000	4800
BEQ	9	1001	0	1	1	00	0	0	0	0	0	0	0	1	0	01	0	0000	0110000000001001000000	C0120
BNE	10	1010	0	1	1	00	0	0	0	0	0	0	0	1	01	0	0000	0110000000000101000000	C00A0	
ADDH1	11	1011	0	0	1	10	0	0	0	0	0	0	0	0	00	0	1100	00110000000000000001100	6000C	
ADDI2	12	1100	0	0	0	00	0	0	0	0	1	0	0	0	00	0	0000	0000000001000000000000	800	
NOP	13	1101	1	0	0	00	0	0	0	0	0	0	0	0	0	11	0	1101	100000000000001101101	10006D

第1步： 在第1列安排微程序，通常取指令部分放置在0号单元，同一指令的微程序中的微指令顺序存放

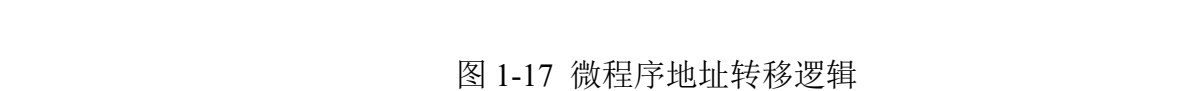
第2步： 填写D到S列的微指令控制信号，注意其中aluSrcB，AluControl为2位，下地址字段4位

第3步： 完成第2步后，最后一列微指令16进制会自动更新

第3步： 将最后一列的16进制编码复制粘贴到Logisim中的控制存储器中

微指令十六进制编码直接复制粘贴到控件中

图 1-17 微指令自动生成表格

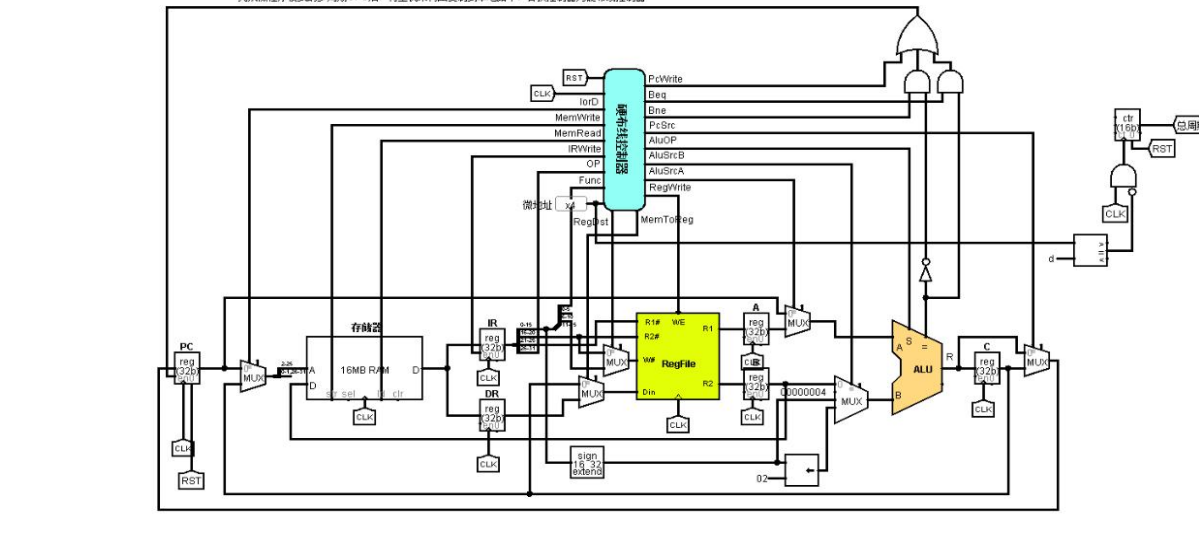


(3) 多周期 MIPS CPU (硬布线)

将多周期 MIPS CPU（微程序）的数据通路复制过来，将微程序控制器改为硬布线控制器，即得到了我们需要的数据通路，如图 1-18 所示。

总复位 RST 总时钟 CLK 总周期数参考值: 801 x16

Ctrl+K Command+K 触发时钟连续仿真, 程序运行结束后可停机, 查看数据存储器中 80 单元的排序情况是有符号顺序



华中科技大学课程实验报告

根据多周期硬布线控制器各个端口控制信号的功能，如图 1-19 所示，绘制硬布线控制器的内部逻辑，如图 1-20 所示。指令译码逻辑中指令对应的十六进制编码可以通过查表（MIPS_GREEN_SHEET）得到。ALU 控制逻辑中，当 $ALU_Control=00$ 时，运算器做加法，当 $ALU_Control=01$ 时，运算器做减法，当 $ALU_Control=10$ 时，运算方式由 Func 决定。控制器状态机的电路可以通过表格自动生成，如图 1-21，1-22 所，将表达式输入 logisim 中可以自动生成状态机 FSM 的组合逻辑电路，如图 1-23 所示。控制存储器中的指令与微程序控制器中相同。

#	控制信号	信号说明	产生条件
1	PCWrite	PC写使能控制	取指令周期，分支指令执行
2	IorD	指令还是数据	0表示指令，1表示数据
3	IRwrite	指令寄存器写使能	高电平有效
4	MemWrite	写内存控制信号	sw指令
5	MemRead	读内存控制信号	lw指令 取指令
6	Beq	Beq指令译码信号	Beq指令
7	Bne	Bne指令译码信号	Bne指令
8	PCSrc	PC输入来源	顺序寻址还是跳跃寻址
9	AluOP	运算器操作控制码 4位	ALU_Control控制，00加，01减，10由Func定
10	AluSrcA	运算器第一输入选择	Lw指令，sw指令，addi
11	AluSrcB	运算器第二输入选择	Lw指令，sw指令，addi
12	RegWrite	寄存器写使能控制信号	寄存器写回信号
13	RegDst	写入寄存器选择控制信号	R型指令
14	MemToReg	写入寄存器的数据来自存储器	lw指令

图 1-19 控制器控制信号功能

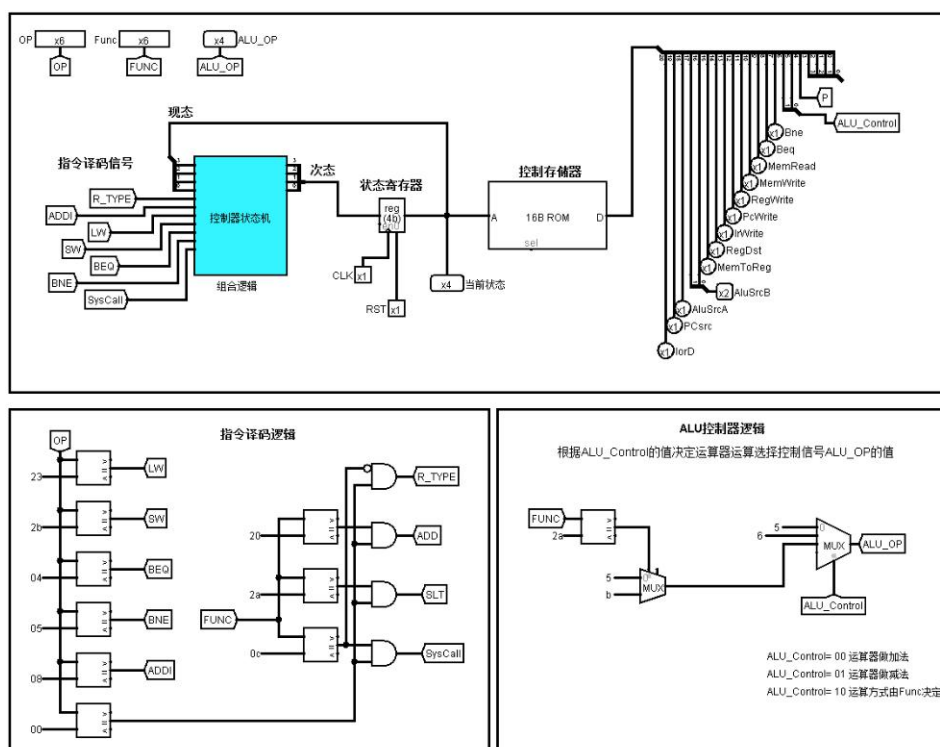


图 1-20 硬布线控制器

华中科技大学课程实验报告

[illegible]

图 1-21 硬布线控制器状态表

[illegible]

图 1-22 硬布线控制器状态表

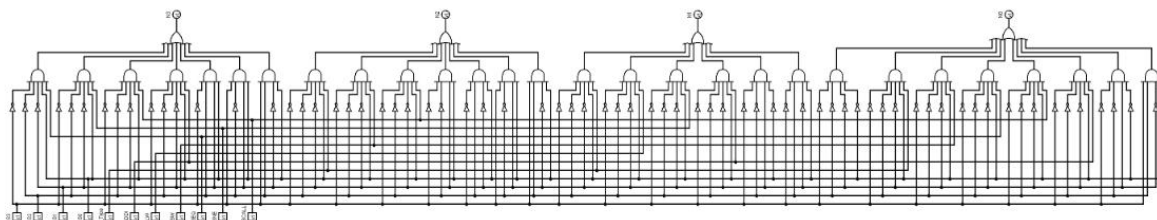


图 1-23 状态机 FSM

1.4 故障与调试

无异常

1.5 测试与分析

1.5.1 单周期 MIPS CPU 的测试

将冒泡排序程序 `sort.hex` 加载到单周期 MIPS CPU 的指令存储器中,然后启用时钟模拟,等到程序执行结束后,观察总周期数与参考值是否相同,如图 1-24 所示。并观察数据存储器地址 80 处的数据是否为 6~1。结果如图 1-25 所示。



图 1-24 单周期 MIPS CPU 运行结束总周期

```
000078 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
000080 00000006 00000005 00000004 00000003 00000002 00000001 00000000 ffffffff
000088 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
```

图 1-25 单周期 MIPS CPU 运行结束后数据存储器中的内容

由此得单周期 MIPS CPU 设计正确

1.5.2 多周期 MIPS CPU（微程序）的测试

将冒泡排序程序 `sort.hex` 加载到单周期 MIPS CPU 的指令存储器中,然后启用时钟模拟,等到程序执行结束后,观察总周期数与参考值是否相同,如图 1-26 所示。并观察数据存储器地址 80 处的数据是否为 6~1。结果如图 1-27 所示。



图 1-24 多周期 MIPS CPU（微程序）运行结束总周期

```
000078 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
000080 00000006 00000005 00000004 00000003 00000002 00000001 00000000 ffffffff
000088 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
```

图 1-25 多周期 MIPS CPU（微程序）运行结束后数据存储器中的内容

由此得多周期 MIPS CPU（微程序）设计正确

1.5.3 多周期 MIPS CPU（硬布线）的测试

将冒泡排序程序 `sort.hex` 加载到单周期 MIPS CPU 的指令存储器中,然后启

华中科技大学课程实验报告

用时钟模拟，等到程序执行结束后，观察总周期数与参考值是否相同，如图 1-28 所示。并观察数据存储器地址 80 处的数据是否为 6~-1。结果如图 1-29 所示。



图 1-28 多周期 MIPS CPU（硬布线运行结束总周期

```
000078 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
000080 00000006 00000005 00000004 00000003 00000002 00000001 00000000 ffffffff
000088 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
```

图 1-29 多周期 MIPS CPU（硬布线运行结束后数据存储器中的内容

由此得多周期 MIPS CPU（硬布线）设计正确

2 总结与心得

2.1 实验总结

本次实验主要完成了如下几点工作：

- 1) 完成方案总结：完成了单周期 MIPS CPU 设计、多周期 MIP CPU（微程序）设计、多周期 MIPS CPU（硬布线）设计
- 2) 功能总结：三个 CPU 都可正确运行 `sort.hex`

2.2 实验心得

我认为此次实验还是比较简单的，学习了 mooc 上的知识后基本就可以实现此次实验，但实验还是非常有意义的，让我从知道组成原理课程中比较抽象的概念到理解原理并实现功能。

参考文献

- [1] DAVID A. PATTERSON(美). 计算机组成与设计硬件/软件接口(原书第 5 版). 北京:机械工业出版社.
- [2] David Money Harris(美). 数字设计和计算机体系结构(第二版). 机械工业出版社
- [3] 谭志虎, 秦磊华, 胡迪青. 计算机组成原理实践教程. 北京:清华大学出版社, 2018 年.
- [4] 秦磊华, 吴非, 莫正坤. 计算机组成原理. 北京:清华大学出版社, 2011 年.
- [5] 袁春风编著. 计算机组成与系统结构. 北京:清华大学出版社, 2011 年.
- [6] 张晨曦, 王志英. 计算机系统结构. 高等教育出版社, 2008 年.

• 指导教师评定意见 •

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： 嵌入签名图片

二、对课程实验的学术评语（教师填写）

三、对课程实验的评分（教师填写）

评分项目 (分值)	报告撰写 (30 分)	课设过程 (70 分)	最终评定 (100 分)
得分			

指导教师签字： _____