

实验三 简单模型机的设计与实现

一、实验目的

1. 掌握指令格式、寻址方式、指令系统的设计方法；
2. 掌握 MIPS 指令系统三种指令格式的特点和寻址方式；
3. 深入理解 CPU 的内部结构，主要功能部件的功能和端口功能；
4. 掌握典型指令的执行流程
5. 掌握数据通路建立过程中，控制信号与数据流之间关系；
6. 掌握组合逻辑控制器的设计方法，理解模型计算机的整机工作过程；
7. 提高基于 Verilog HDL 的硬件电路的设计、调试、测试能力。

二.实验设备

1. DE2-70 开发板，下载线缆
2. QuartusII IDE
3. 实验提供的文件见表 1

表 1 源文件清单

序号	文件名	功能简介
1	Misp_cpu.dbf	模型机顶层模块
2	SEG7_LUT.V	七段数码译码模块
3	SEG7_L_8.V	32 位数据 7 段数码数据模块
4	EXT.v	扩展模块
5	Mux.v	数据选择器模块
6	Ctrl_encode_def.v	参数控制定义文件
7	alu.v	运算器模块
8	Dm.v	RAM 模块
9	RF.V	寄存器堆模块
10	im.v	指令存储器模块
11	NPC.v	PC 更新模块
12	PC.V	PC 模块
13	IR.v	IR 模块
14	Ctrl.v	CU 模块

15	cputest.csv	输入输出引脚分配文件
----	-------------	------------

三、实验内容和要求

- 分析表 2 所示汇编程序的代码，完成下列要求。
 - 画出汇编程序的流程图，说明该程序的功能；
 - 写出每条汇编指令对应的机器码和存储地址，并将其存放到指令存储器中。

表 2 汇编指令和机器指令对应关系

机器指令	存储地址	汇编指令
		start: ori \$1,\$0,4
		ori \$2,\$0,1
		ori \$3,\$0,1
		loop1: sw \$2,0(\$2)
		addu \$2,\$2,\$3
		bne \$2,\$1,loop1
		ori \$2,\$0,3
		ori \$1,\$0,0
		loop2: lw \$4,0(\$2)
		addu \$1,\$1,\$4
		subu \$2,\$2,\$3
		bne \$2,\$0,loop2
		sw \$1,0(\$0)
		jal start

2. 分析模型机的基本结构

模型机的顶层结构如图 1-1 至图 1-3 所示，完成如下要求：

- 分析模型机中的主要构成模块，包括程序计数器 PC，PC 更新模块 NPC，指令存储器 IM，指令寄存器 IR，寄存器文件 RF，运算器 ALU、数据存储器 DM 的端口信号的功能，明确各部件的工作原理；分析模型机的总线互联结构，明确各部件间的数据流向，并填写表 3。

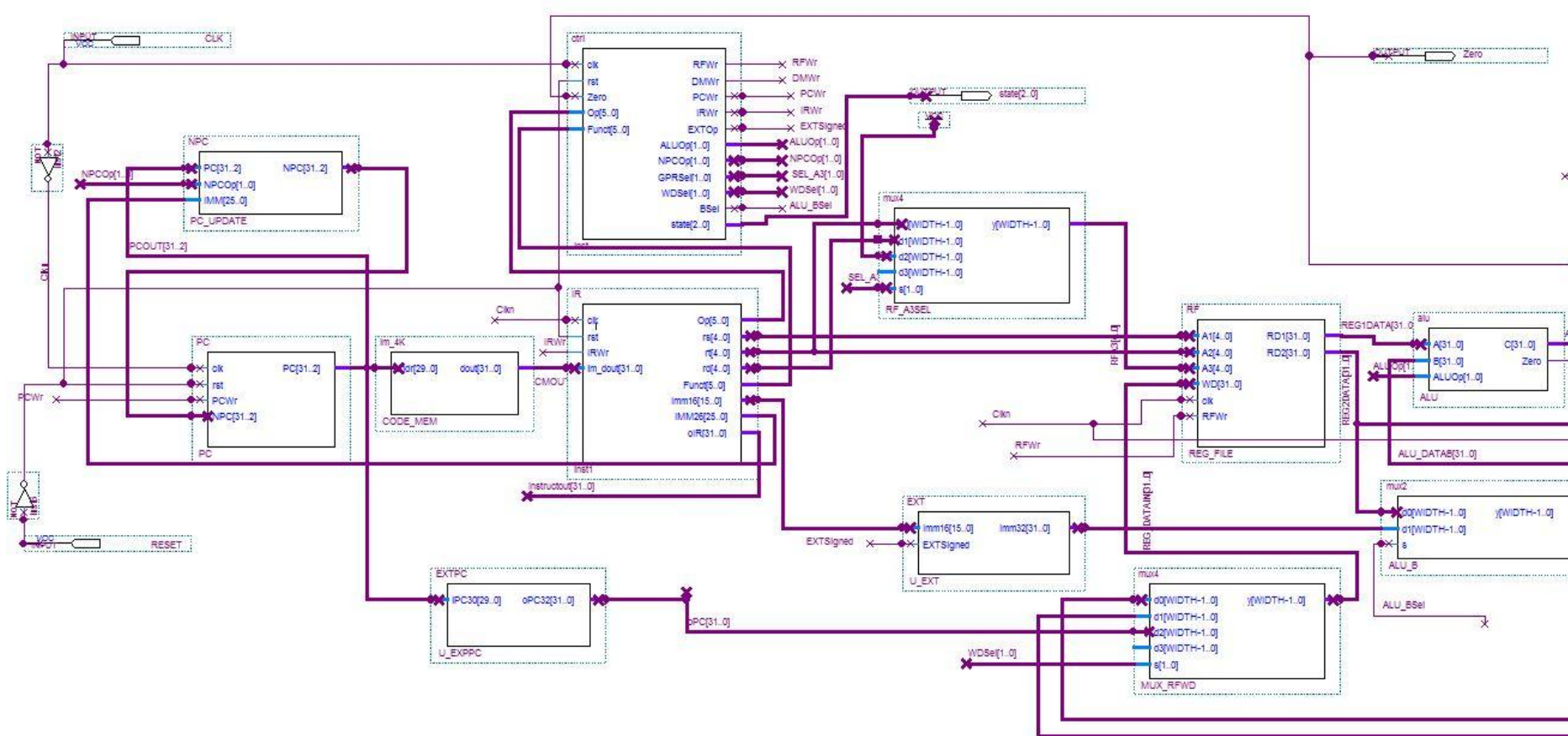


图 1-1 CPU 数据通路图 (1)

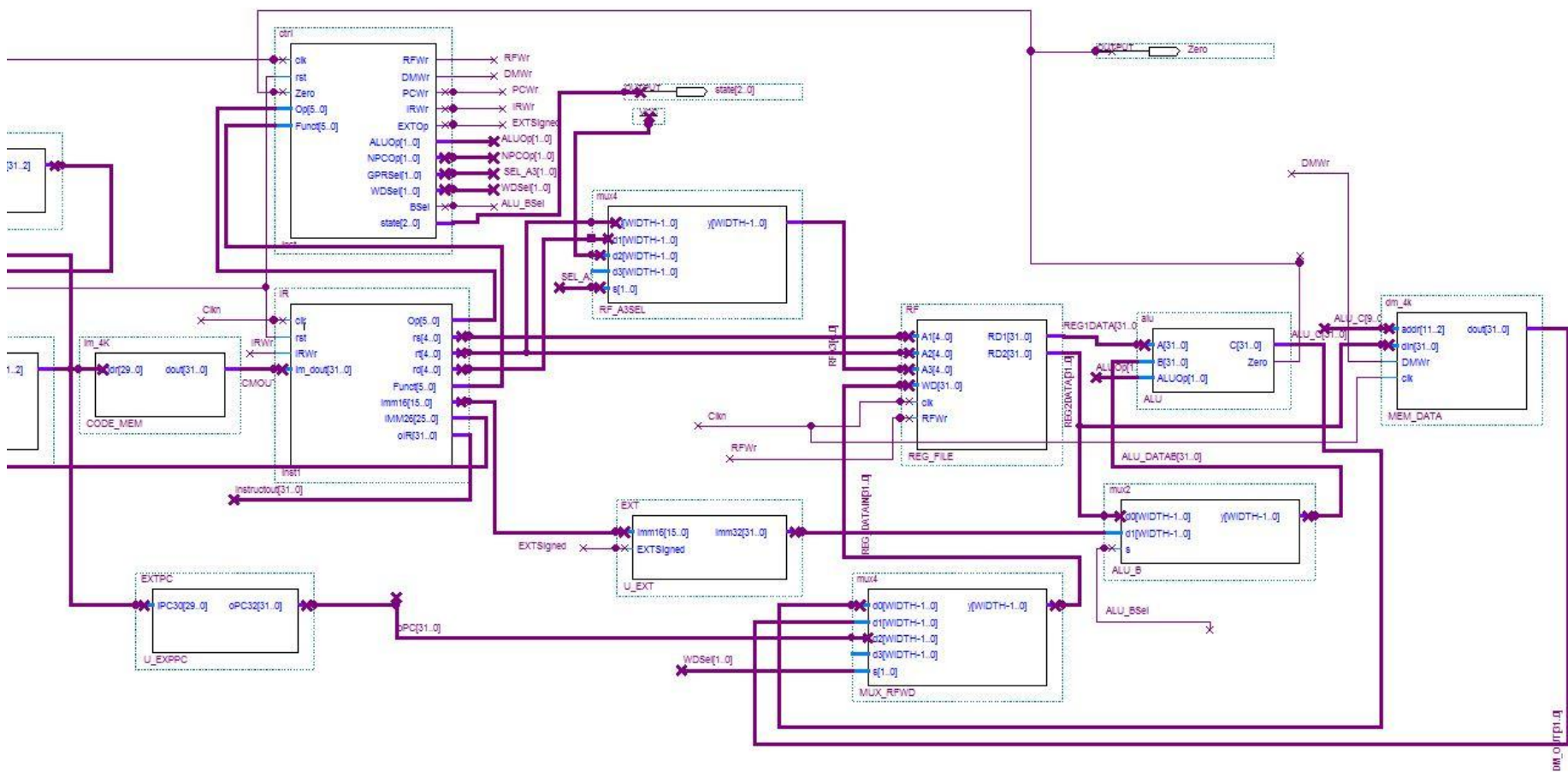


图 1-2 CPU 数据通路图 (2)

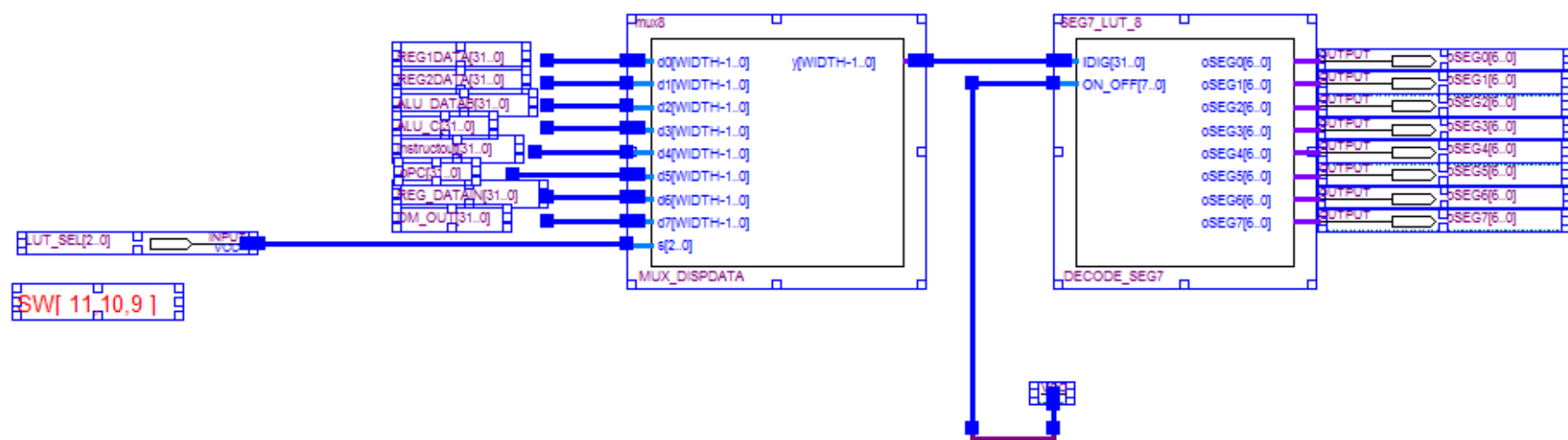


图 1-3 CPU 数据通路图（3）

表 3 主要部件功能及数据流向列表

模块名	功能描述	输入数据来源	输入端口		输出数据目的地	输出端口		控制信号
			信号名称	作用描述		信号名称	作用描述	
PC								
NPC								
IM								
IR								
RF								
ALU								
DM								

(2) 分析系统中的 4 个多路数据选择器，填写表 4

表 4 数据选择器的功能分析

实例化名称	选 择 器 类 型	控制信号名称	输出数据流向	控制信号与输出之间的关系
MUX_DISPDATA	8 选 1	LUT_SEL[2..0]	七段译码器数据输入端	000: REG1DATA[31..0] 001: REG2DATA[31..0] 010: 011: 100: 101: 110: 111:
MUX_RFWD				
ALU_B				
RF_A3SEL				

(3) 分析并说明系统中的 2 个数据扩展器 U_EXTPC 和 U_EXT 的作用

3.设计并实现一个能够执行下列 MISP32 指令的控制器

MISP 指令集: addu, subu, ori, lw, sw, bne, jal

- (1) 写出指令集中每条指令的指令格式，并对指令功能进行说明
- (2) 设计的控制器端口信号如图 2 所示，在端口上标明每个输入和输出信号的作用

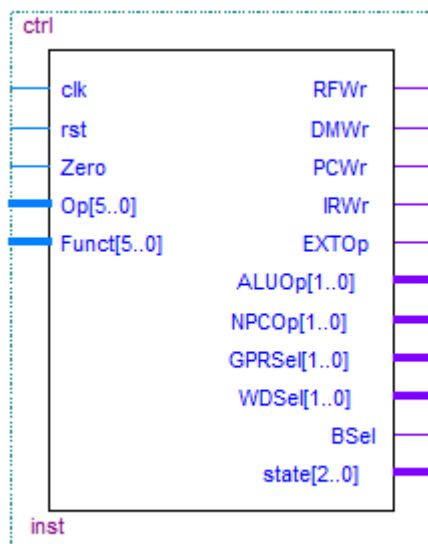


图 2 控制器模块端口引脚图

- (3) 根据指令的功能，分析每条从取指到指令执行的整个过程中所需的控制信号填写表 5。
- (4) 补充完善 Ctrl.v 的 Verilog 代码，使其能够正确执行 MISP 指令集中的 addu, subu, ori, lw, sw, bne, jal 七条指令。

实现过程：将实验内容 1 中的汇编程序对应的机器指令写入指令存储器，并在 QuartusII IDE 环境下，进行系统功能仿真，在波形仿真图中对程序中每条指令的执行过程进行说明。

- (5) 在 DE2-70 开发板上对 CU 的结果进行验证，并观察指令的操作过程中所涉及部件相关数据的变化过程，能够正确的解释指令的执行过程，并对结果的正确性进行分析判断。

四.实验报告要求

1. 分析系统电路图，仔细阅读各模块 Verilog 代码，明确各主要功能模块的功能，理解数据传送通路。按照如下要求画出系统完整的数据通路图。

(1) 使用 A3 纸，横版画出（手画）。

(2) 模块用方块画出，上边外部标出模块名称。方块内部标出模块的所有输入输出信号，每个信号要注明含义。

说明：①对于典型的信号，如 `clk` 等，可以不写出含义；②信号含义的标注方式：可以在模块内部注明，如图 2 的 `REG_FILE` 模块；也可以在外部以列表方式说明模块/信号功能，如图 2 的 `MUX_RFWD` 模块。

(3) 模块间的连接信号线：地址线用蓝色粗线、数据线用黑色粗线、控制线用红色单线画出。传送多种类型信息的信号线用绿色画出。要用箭头表示每组线的传送方向。

(4) 每组地址线、数据线都要标出位数。如图中线上标注的 5 位、32 位。

(5) 每个 控制信号 要标注 3 种信息：外部信号名，连接电路板的资源，控制信号的有效电平。其中，连接电路板的资源要用红色字体写出。

2.填写表 2

3.填写表 3

4.填写表 4

5.对图 2 的端口进行标注

6.填写表 5

7.给出完整的 `Ctrl.v` 的代码

8. 对 CU 的功能仿真波形（可打印）上对中每条指令的执行过程进行说明，分析程序的正确性。