## 实验三 MIPS处理器设计实验

## 实验目的

* 进一步加深对运算器、存储器及时序电路的理解。
* 掌握硬布线控制器设计原理。
* 为整机实验以及课程设计做准备。

## 实验环境

Logisim是一款数字电路模拟的教育软件，每一位用户都可以通过它来学习如何创建逻辑电路，方便简单。 它是一款基于Java的应用程序，可运行在任何支持JAVA环境的平台，方便学生来学习设计和模仿数字逻辑电路。Logisim中的主要组成部分之一就在于设计并以图示来显示CPU。当然Logisim中还有其他多种组合分析模型来对你进行帮助，如转换电路，表达式，布尔型和真值表等等。同时还可以重新利用小规模的电路来作为大型电路的一部分。

<http://www.cburch.com/logisim/docs.html>

## 实验内容

利用实验1、2中构建的运算器，寄存器文件等部件以及logsim中其他功能部件构建一个32位MIPS CPU处理器，该处理器应支持下表中所述指令，具体指令功能参见附件中的MIPS标准文档。最终设计完成的CPU能运行教师提供的标准测试程序，程序存储在logisim ROM模块中（指令存储器、数据存储器分开）

（只要求按照教材中讲述的11条指令来实现）

表 1 指令格式

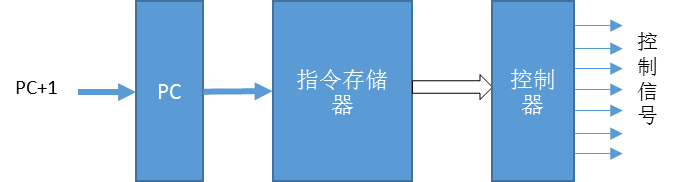
|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **指令** | **格式** | **备注** |
| 1 | Add | add $rd, $rs, $rt | 指令功能及指令格式  参考MIPS32指令集 |
| 2 | Add Immediate | addi $rt, $rs, immediate |
| 3 | Add Immediate Unsigned | addiu $rt, $rs, immediate |
| 4 | Add Unsigned | addu $rd, $rs, $rt |
| 5 | And | and $rd, $rs, $rt |
| 6 | And Immediate | andi $rt, $rs, immediate |
| 7 | Shift Left Logical | sll $rd, $rt, shamt |
| 8 | Shift Right Arithmetic | sra $rd, $rt, shamt |
| 9 | Shift Right Logical | srl $rd, $rt, shamt |
| 10 | Sub | sub $rd, $rs, $rt |
| 11 | Or | or $rd, $rs, $rt |
| 12 | Or Immediate | ori $rt, $rs, immediate |
| 13 | Nor | nor $rd, $rs, $rt |
| 14 | Load Word | lw $rt, offset($rs) |
| 15 | Store Word | sw $rt, offset($rs) |
| 16 | Branch on Equal | beq $rs, $rt, label |
| 17 | Branch on Not Equal | bne $rs, $rt, label |
| 18 | Set Less Than | slt $rd, $rs, $rt |
| 19 | Set Less Than Immediate | slti $rt, $rs, immediate |
| 20 | Set Less Than Unsigned | sltu $rd, $rs, $rt |
| 21 | Jump | j label |
| 22 | Jump and Link | jal label |
| 23 | Jump Register | jr $rs |
| 24 | syscall（display or exit） | syscall | If $v0==10  halt(停机指令)  else  数码管显示$a0值 |

## 实验步骤

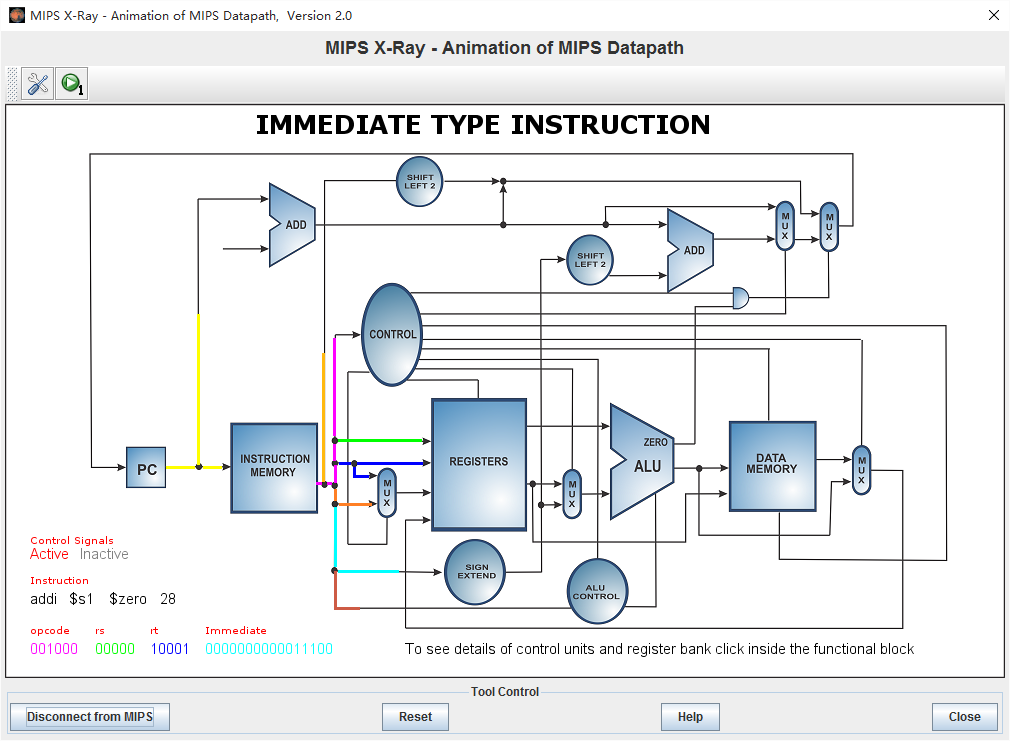
**1、实验准备**

**1、实验步骤**

1. **利用**logisim平台中现有部件构建取指令通路（PC寄存器32位），其中PC由时钟驱动，每个时钟自动完成取值以及PC=PC+4的功能，控制存储器在后续步骤中实现。此处指令存储器选用10位地址线，32位数据线的ROM部件，MIPS32s位地址为字节地址，ROM地址线宽度有限，建议将CPU 32位地址高位部分和字节偏移部分直接屏蔽，数据存储器也参照类似方法处理。



1. 利用综合实验一中封装的32位运算器以及RAM模块构建能满足上述指令系统的简单运算通路，具体指令运算通路可以利用MARS仿真器中的MIPS-XRAY功能查看，如下图：



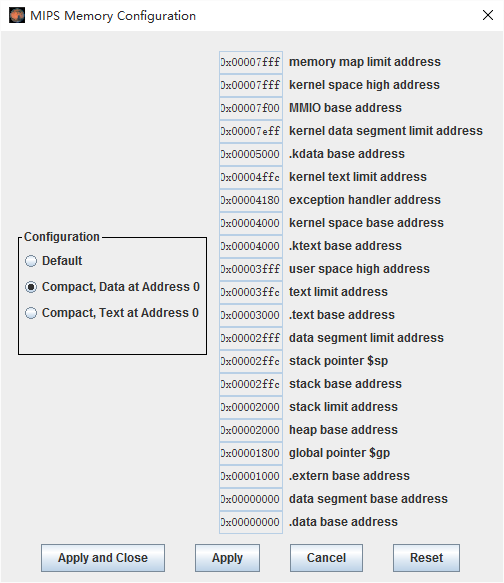
1. 对步骤2构建的不同指令的数据通路进行综合，生成控点（操作控制信号）设计一览表如下：为避免过多的连线，建议控制器输出的信号用隧道标签的方式连接。

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **控制信号** | C’n |  |  |  | C’2 | C’1 | C’0 |
| **有效值** |  |  |  |  |  |  |  |
| **标签**  **（隧道）** |  |  |  |  |  |  |  |
| **功能** |  |  |  |  |  |  |  |

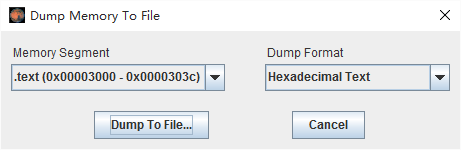
**2、检查要求**

1. 能自动运行如下排序测试程序，
2. 将测试文件“test\_11.hex”通过文件加载到指令存储器，运行该程序，能够得到正确结果。
3. 能单步演示和自动运行

汇编器采用附件包中的mars仿真器，该仿真器功能强大，请主动学习之。注意为了能让mars中汇编的机器码能在logisim中实用，需要设置mars界面中setting的Memory Configration，将内存模式设置为下图的模式，这样数据段起始位置就是0开始的位置。



程序汇编和后可以利用File菜单中的Dump Memory功能将代码段和数据段导出，采用十六进制文本的方式导出到某个文本文件，然后在文件第一行加入“v2.0 raw”即可在logisim中加载到ROM或RAM。



## 实验报告要求

1. 实验目的；
2. 各模块的设计电路和系统的整体电路,对设计要进行详细的分析与说明；
3. 列出操作步骤及顺序,标出重要的开关控制端；给出各控制信号逻辑表达式以及电路。
4. 实验结果的记录与分析；
5. 列出操作步骤及顺序,标出重要的开关控制端；
6. 实验收获和体会；
7. 实验中碰到的问题和解决的方法。

**六、实验报告提交**

请将完成后的cpu.circ文件按以下命名规范命名后作为实验结果提交给班级知道教师当场检查并归档。

* **专业命名规范**

软件工程 SE；计算机科学 CS ；计算机网络CN；

物联网 IT；数字媒体DM；

* **文件命名规范**

专业班级\_学号\_姓名\_cpu.circ

SE1\_1610300101\_张明\_cpu.circ

**注：**本文档有些的不全面、不完整，希望同学们修正。