单周期MIPS处理器设计报告

姓名： 学号：

1. 数据通路设计

参考讲义图片，完善以下数据通路以支持7条指令。

要求：必须自己绘制该数据通路，可以用铅笔在白纸上手绘拍照贴图，也可以画图工具直接绘制。数据通路中所标注的信号名称必须与代码保持一致（建议Verilog代码中所有信号使用小写字母，不要像下图中大小写混用，因为Verilog语言是区分大小写的，全部使用小写不容易出错）。

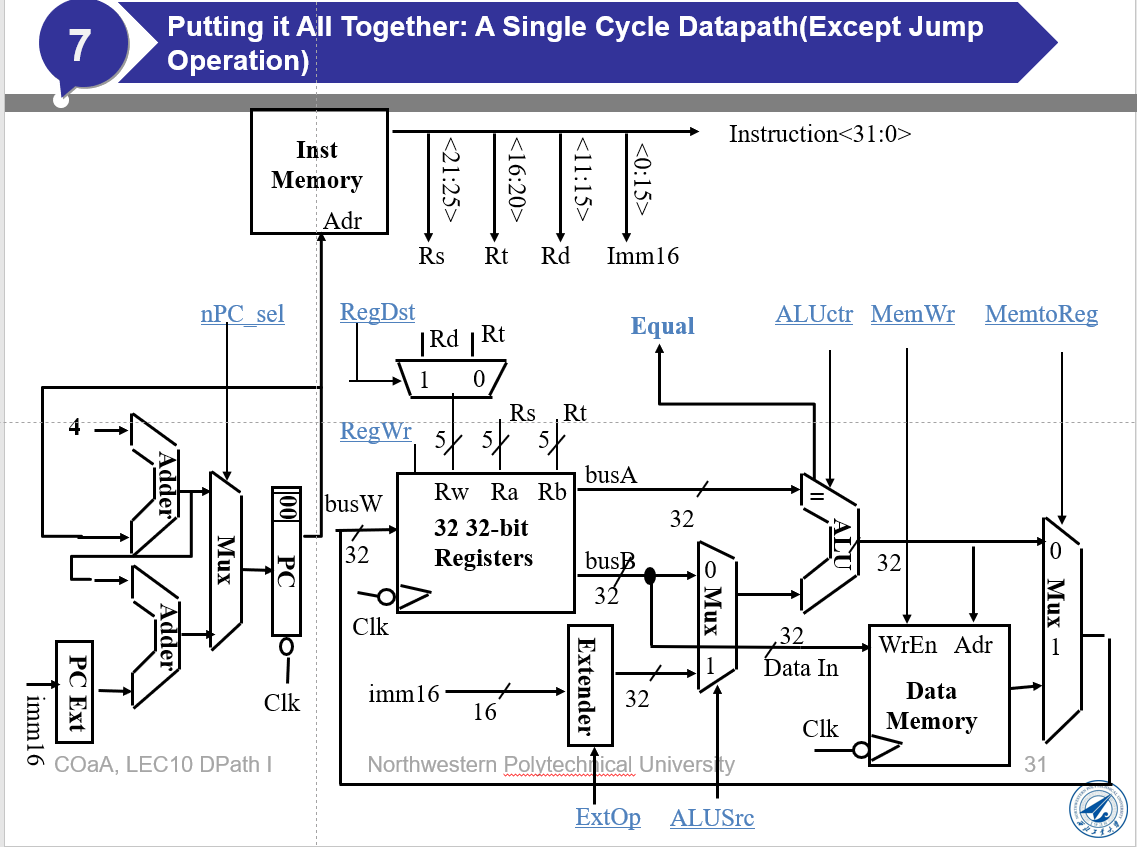


图 1 MIPS处理器的数据通路

1. 控制通路设计

按照数据通路中的控制点，补全下表说明每条指令的控制信号值。

表 1 MIPS处理器的控制信号

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | ADDU | SUBU | ORI | LW | SW | J | BEQ |
| Npc\_sel |  |  |  |  |  |  |  |
| regwr |  |  |  |  |  |  |  |
| regdst |  |  |  |  |  |  |  |
| ALUctr |  |  |  |  |  |  |  |
| …… |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |

1. 仿真结果
2. MARS编写的MIPS汇编程序

MARS工具使用教程：

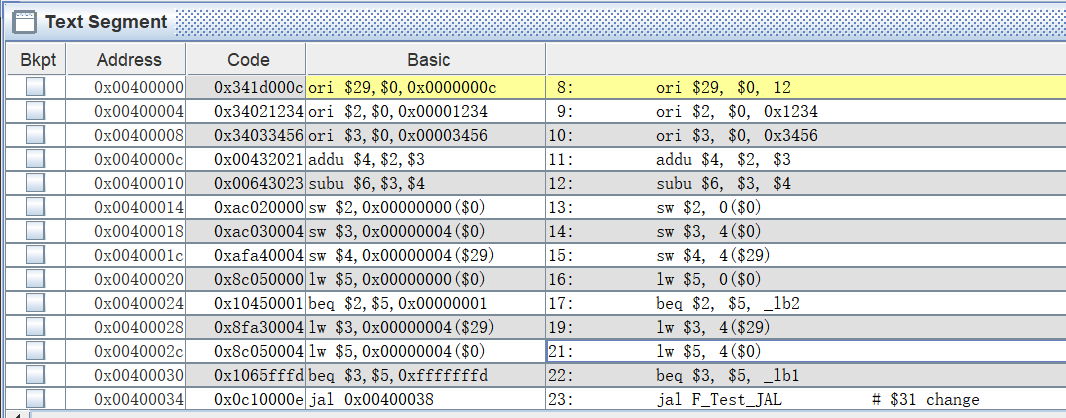
<https://v.youku.com/v_show/id_XMTY3NjcyOTMzMg==.html?f=27854852>

MARS下载地址：

<http://courses.missouristate.edu/KenVollmar/MARS/>

在MARS中汇编通过之后截屏如下：

必须自己按照7条指令自己编写汇编程序，不能完全照搬下面的示范！



1. 仿真波形
2. 实现结果（可选）

Quartus软件熟悉的同学可以粘贴布局布线结果，包括资源占用率、最高主频等实现结果。

1. 评分标准

从设计思路、仿真结果、实现结果等方面进行综合评价，全部满分共计24分。

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 1、设计思路 | 10 | 8 | 6 | 4 | 0 |
| 查看设计文档思路是否清晰，描述是否完整 | 优秀，数据通路描述准确、控制通路表格正确 | 很好，文档较为清楚 | 及格，文档较完整 | 待改进，文档不清或错误较多 | 无文档 |
| 2、仿真结果 | 10 | 8 | 6 | 4 | 0 |
| 主要通过测试激励进行评估，可以用提交者的测试激励，也可以用自己的测试，验证功能是否正确。 | 优秀，7条指令100%验证通过（尽量用自己的测试程序，与提交的测试程序进行交叉验证） | 很好，1~2条错误 | 及格，3条指令错误 | 待改进，50%以上功能点错误 | 基本不对，编译不通过等 |
| 3、实现结果 | 10 | 8 | 6 | 4 | 0 |
| 是否有Quartus布线结果 | - | - | - | 有最高主频、面积占用率等信息 | 无 |

附录：提供代码的说明。

本练习提供了部分源代码，如果你能看懂，最好使用这些代码节省编程工作量。如果你不能看懂，那么可以自行编写代码。提供的代码仅供参考，里面有些代码是多余的功能（例如ALU模块），可以忽略，最终确保7条指令功能正确即可。

Alu.v：一个纯组合逻辑的ALU，操作有aluop确定，其定义在包含的ctrl\_encode\_def.v文件中定义。注意BEQ指令应当使用ALUOp\_EQL操作。

Dmem.v：一个简单的data memory模型。注意其初始化时要读取一个文件dmem.txt，如果你想让data memory开始有数据，那么需要在dmem.txt中添加数据，注意第一行对应的是地址0，第二行对应地址4，以此类推。

Imem.v：一个简单的instruction memory模型。和dmem.v类似，使用imem.txt来进行初始化。也就是说，你必须将MIPS指令的机器码逐行放到imem.txt中来执行。MIPS指令的机器码可以手工编写，也可以使用MARS编程器来产生。

Dmem.v和Imem.v中的addr地址输入要注意，其定义为input [11:2] addr，也就是连接时应不连接mips处理器地址输出的低两位。Addr相当于字地址而不是字节地址。Mips.v中地址定义为output [31:0] imem\_addr，注意在连接时要正确处理连接关系。Dmem和imem只模拟了4K bytes（或1K words），是因为如果建模完整4GB内存，modelsim会消耗大量内存，影响仿真速度。

instruction\_def.v定义了大部分MIPS指令的操作码和function字段，以方便编写译码部分进行指令类型判断。

Regfile.v：两读一写的寄存器文件模型。如果使能debug控制，则可以打印寄存器最新值。

Mips.v：设计顶层。里面实例化了ALU、Regfile等模块。

Tb\_mips.v：Testbench。里面实例化了mips、imem、dmem等模块以对mips模块进行测试。没有把imem/dmem模块放到mips中是为了在使用quartus布线时节约时间，也更加符合memory与cpu是两个独立芯片的实际情况。

注意：imem.txt和dmem.txt的路径在Imem.v和dmem.v中是使用相对路径来指定的，如果你使用modelsim中总是报错找不到这两个文件，可以将这两个txt文件移动到modelsim工程的根目录。