****

计算机组成原理（甲）

实 验 报 告

|  |  |
| --- | --- |
| **学 院** | 网络空间安全学院 |
| **专 业** | 网络工程 |
| **班 级** | 19272401 |
| **学 号** | 19061440 |
| **学生姓名** | F001 |
| **教师姓名** | 袁理峰 |
| **完成日期** | 2020.12.26 |
| **成 绩** |  |
| 实验七 取指令与指令译码实验 （实验名称） | | |
| 1. **实验目的** 2. 学习指令存储器的设计 3. 掌握CPU取指令操作与指令译码的方法和过程 | | |
| 1. **实验原理**   设计一个指令寄存器，只读，物理大小32\*64位  设计PC及其自增电路；  MIPS地址32位，按字节编址，指令存储器：256\*8位  最终目标：设计一个单周期MIPS CPU  在指令周期clk上跳沿，执行取指令操作，在clk下降沿更新PC值。  复位信号rst=1时，PC清零，即指定MIPS CPU从0号主存开始执行程序  生成只读的指令存储器时，使用Memory IP核，同实验5，但是选择Single Port ROM | | |
| 1. **实验环境**   所用电脑的软硬件配置：自己的笔记本电脑、Windows10操作系统  实验所用的软件：ISE design suite | | |
| 1. **主要操作步骤及实验结果记录（不能光截图，要有相应的文字说明）**   （对实验过程中的主要操作步骤进行描述，并随时记录实验过程中观察到的结果，必要时可辅助截图）  任务一：在ISE中使用Memory IP核生成一个Inst\_ROM，当作指令存储器，并且关联一个实验六生成的\*.coe文件  以下为生成ROM的相关操作记录      第一步 第二步 第三步  第四步 第五步 第六步  任务二：编程实现取指令模块，调用Inst\_ROM指令存储器模块，在此基础上编写一个实验验证的顶层模块  由于ISE中不支持UTF-8的编码，因此在这里书写注释。  第25行表示时钟及重置信号  第26行表示Program Counter，即程序计数器  第27行表示PC自增值  第28行表示机器指令码  第31行表示同步清零操作  第32行表示更新PC  第33行表示组合逻辑，暂存PC自增值  第35行是实例化指令存储器IP核  第36行是输入线clka  第37行是地址线6根  第38行是输出线32根  第40行开始是指令分段  第42行是offset或者立即对于I 类型  第43行是J类型的地址  任务四：在Xilinx ISE中创建工程，源编码，然后编译、综合    可以看到，该代码通过了语法检查  任务五：编写激励代码，观察仿真波形，直至验证正确。    图二    图一 图三    图一为激励代码，显然同样通过了语法检查；上图为仿真波形的全部展示，但难以看清。图三是上图的左侧采用了16进制对数据进行展示，图二可以看出每当时钟沿下降时取出相应的地址，实验成功。 | | |
| 1. **实验分析总结及心得**   （结合所学知识对实验过程中观察到的实验结果进行分析总结，以便加深对知识的理解，并总结通过实验学到的知识或技术）  实验中，在复制PCspim的过程中，发现了地址卡在了0296b820中，回忆实验六时，当时也是卡在第22条语句之后程序不再执行，因此我重新回去检查了实验六的程序，发现是实验六时，小金同学给我的代码出现了问题，与书上的代码不一样，最后的十条操作全部都是针对寄存器22进行的操作，因此发生了栈溢出错误，导致程序无法继续执行。因此重新书写了代码，发现情况正常了。  实验中，一开始的跑出来的结果与书上的结果不一样，经过观察后发现存在aaaaaaaa，bbbbbbbb这类特点显著的地址，思考后发现这是实验五的test文件所保存的地址值，检查之后发现在导入时导入错了一个test，因此地址发生了相应的改变。这启示我们在操作时要注意观察目标文件的目录，不要单纯看名字，容易导入成为同名文件但是内容不同。重新导入coe文件之后实验正常。  最后对比取出的指令代码与指令存储器关联文件中的指令码是一致的。判断实验成功 | | |