**实验报告**

2019 年 4 月 30 日 成绩：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 姓名 | 於文卓 | 学号 | 17061833 | 班级 | 17052317 |
| 专业 | 计算机科学与技术 | | 课程名称 | 计算机组成原理课程设计 | |
| 任课老师 | 冯建文 | 指导老师 | 冯建文 | 机位号 |  |
| 实验序号 | 4 | 实验名称 | MIPS汇编器与模拟器实验 | | |
| 实验时间 | 4月30日 | 实验地点 | 1-225 | 实验设备号 | #9 |
| **一、实验目的与要求** | | | | | |
| 1. 实验目的： 2. 学习MIPS指令系统，熟悉MIPS指令格式及其汇编指令助记符，掌握机器指令编码方法 3. 学习MIPS汇编程序设计，学会使用MIPS汇编器将汇编语言程序翻译成二进制文件 4. 了解使用MIPS教学系统模拟器运行程序的方法 5. 实验要求： 6. 按照上述实验内容中的例子，编辑test.asm文件，在PCSpim中打开它，并使用单步执行和连续执行方式运行该程序 7. 将下列程序输入一个R\_CPU\_Test.asm文件，并在PCSpim中单步运行，观察各个寄存器的值，是否和预期的一致 8. 将上述程序的指令代码逐条摘录出来，拷贝至ROM IP核关联文件\*.coe中，以备后续实验使用 9. 撰写实验报告：含执行结果截图、实验结果记录表、实验分析和生成的\*.coe文件内容，以及你对本实验的“思考与探索”部分所作的思考与探索； | | | | | |
| **二、实验设计与程序代码** | | | | | |
| 1. 模块设计说明   （1）使用记事本程序或任何纯文本编辑器，编辑test.asm文件，输入前述内容；  （2）运行PCSpim程序，在其中打开test.asm，先连续执行，输入起始地址0x0040 0000，再单步运行，按照需要在控制台输入2个数据，执行完毕，观察结果。  （3）使用记事本程序或任何纯文本编辑器，编辑R\_CPU\_Test.asm文件，输入规定指令。  （4）在PCSpim程序中打开R\_CPU\_Test.asm，同上单步执行，记录执行结果。   1. 实验程序源代码及注释等   nor $1, $0,$0; #$1=FFFF\_FFFF  sltu $2, $0, $1; #$2=0000\_0001  add $3, $2, $2; #$3=0000\_0002  add $4, $3, $2; #$4=0000\_0003  add $5, $4, $3; #$5=0000\_0005  add $6, $5, $3; #$6=0000\_0007  sllv $7, $6, $2; #$7=0000\_000E  add $9, $5, $6; #$9=0000\_000C  sllv $8, $6, $9; #$8=0000\_7000  xor $9, $1, $8; #$9=FFFF\_8FFF  add $10, $9, $1; #$10=FFFF\_8FFE  sub $11, $8, $7; #$11=0000\_6FF2  sub $12, $7, $8; #$12=FFFF\_900E  and $13, $9, $12; #$13=FFFF\_800E  or $14, $9, $12; #$14=FFFF\_9FFF  or $15, $6, $7; #$15=0000\_000F  nor $16, $6, $7; #$16=FFFF\_FFF0  add $17, $7, $3; #$17=0000\_0010  sllv $18, $8, $17; #$18=7000\_0000  sllv $19, $3, $17; #$19=0002\_0000  sllv $20, $19, $7; #$20=8000\_0000  add $21, $20, $1; #$21=7FFF\_FFFF  or $22, $18, $21; #$22=7FFF\_FFFF  add $23, $20, $22; #$23=FFFF\_FFFF  sub $24, $20, $22; #$24=0000\_0001  sub $25, $22, $20; #$25=FFFF\_FFFF  xor $26, $18, $1; #$26=8FFF\_FFFF  sltu $27, $22, $20; #$27=0000\_0001  sltu $28, $26, $20; #$28=0000\_0000  add $29, $22, $2; #$29=8000\_0000  sub $30, $20, $2; #$30=7FFF\_FFFF  add $31, $11, $26; #$30=9000\_6FF1 | | | | | |
| **三、实验仿真** | | | | | |
| 1. 仿真代码 2. 仿真波形 3. 仿真结果分析 | | | | | |
| **四、电路图** | | | | | |
|  | | | | | |
| **五、引脚配置（约束文件）** | | | | | |
|  | | | | | |
| **六、思考与探索** | | | | | |
| 1. 实验结果记录：       00000827 0001102b 00421820 00622020 00832820 00a33020 00463804 00a64820 01264004 00284826 01215020 01075822 00e86022 012c6824 012c7025 00c77825 00c78027 00e38820 02289004 02239804 00f3a004 0255b025 0296b820 0241d026 02d4d82b 0354e02b 02c2e820 017af820;   1. 实验结论： 2. 问题与解决方案： 3. 思考题：  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | 寄存器 | 内容 | 寄存器 | 内容 | 寄存器 | 内容 |  | | $0 | FFFF\_FFFF | $11 | 0000\_6FF2 | $22 | 7FFF\_FFFF |  | | $1 | 0000\_0001 | $12 | FFFF\_900E | $23 | FFFF\_FFFF |  | | $2 | 0000\_0002 | $13 | FFFF\_800E | $24 | 0000\_0001 |  | | $3 | 0000\_0003 | $14 | FFFF\_9FFF | $25 | FFFF\_FFFF |  | | $4 | 0000\_0005 | $15 | 0000\_000F | $26 | 8FFF\_FFFF |  | | $5 | 0000\_0007 | $16 | FFFF\_FFF0 | $27 | 0000\_0001 |  | | $6 | 0000\_000E | $17 | 0000\_0010 | $28 | 0000\_0000 |  | | $7 | 0000\_000C | $18 | 7000\_0000 | $29 | 8000\_0000 |  | | $8 | 0000\_7000 | $19 | 0002\_0000 | $30 | 7FFF\_FFFF |  | | $9 | FFFF\_8FFF | $20 | 8000\_0000 | $31 | 9000\_6FF1 |  | | $10 | FFFF\_8FFE | $21 | 7FFF\_FFFF |  |  |  | |  |  |  |  |  |  |  | |  |  |  |  |  |  |  |   结果正确  （2）遇到的问题  1. 没有将模拟器设置为Bare Machine导致导入问件时一直报错，后来发现书上有说明要再setting中设置  2. 没有正确设置PC而报错    (3) 其他的汇编模拟器  根据邓添同学在群里分享的，我下载了MARS      可以很方便的生成机器码并且快速导出。要注意的是其不允许代码中出现分号，使用时需要将分号去掉 | | | | | |