

MIPS指令系统和MIPS体系结构

计算机体系结构实验一



目录

[1． 实验目的 0](#_Toc511748404)

[2． 实验平台 1](#_Toc511748405)

[3． 实验内容和步骤 1](#_Toc511748406)

2017-4-7

[裴子祥 计科七班 学号2015211921]

[指导老师：黄智濒]

1. 实验目的

(1) 了解和熟悉指令级模拟器。

(2) 熟练掌握 MIPSsim 模拟器的操作和使用方法。

(3) 熟悉 MIPS 指令系统及其特点，加深对 MIPS 指令操作语义的理解。

(4) 熟悉 MIPS 体系结构。

1. 实验平台

实验平台采用指令级和流水线操作级模拟器 MIPSsim。

1. 实验内容和步骤

首先要阅读 MIPSsim 模拟器的使用方法（见附录），然后了解 MIPSsim 的指令系统和汇编语言。

(1) 启动 MIPSsim(用鼠标双击 MIPSsim．exe)。

(2) 选择“配置”—>“流水方式”选项，使模拟器工作在非流水方式下。

(3) 参照 MIPSsim 使用说明，熟悉 MIPSsim 模拟器的操作和使用方法。 可以先载入一个样例程序(在本模拟器所在的文件夹下的“样例程序”文件夹中)，然后 分别以单步执行一条指令、执行多条指令、连续执行、设置断点等的方式运行程序，观察程序执行情况，观察 CPU 中寄存器和存储器的内容的变化。

(4)选择“文件”—> “载入程序”选项，加载样例程序 alltest.asm，然后查看“代码” 窗口，至看程序所在的位置(起始地址为 Ox00000100)。

(5)查看“寄存器"窗口 PC 寄存器的值：[PC]=0x\_\_ 00000000\_\_\_\_\_\_

(6)执行 load 和 store 指令，步骤如下：

①单步执行 1 条指令(F7)。

②下一条指令地址为 0x\_\_\_\_00000004\_\_\_\_\_\_\_\_，是一条\_\_\_\_\_有\_\_\_\_(有，无)符号载入\_\_\_\_\_字节\_\_\_(字 节．半字，字)指令。

③单步执行 1 条指令(F7)。

④查看 R1 的值，[R1]=0x\_\_FFFFFFFFFFFFFF80\_\_\_\_\_\_\_\_\_\_\_\_

⑤下一条指令地址为 0x\_\_\_\_00000008\_\_\_\_\_\_\_\_，是一条\_\_\_\_有\_\_\_\_\_(有，无)符号载入\_\_\_\_\_字\_\_\_\_(字 节．半字，字)指令。

⑥单步执行 1 条指令。

⑦查看 R1 的值，[R1]=0x\_\_\_0000000000000080\_\_\_\_\_\_\_\_\_\_\_。

⑧下一条指令地址为 0x\_\_\_\_0000000c\_\_\_\_\_\_\_\_，是一条\_\_\_\_\_无 \_\_\_\_(有，无)符号载入\_\_\_\_字节\_\_\_\_\_(字 节．半字，字)指令。

⑨单步执行 1 条指令。

⑩查看 R1 的值，[R1]=0x\_\_\_\_0000000000000080\_\_\_\_\_\_\_\_\_\_。

⑾单步执行 1 条指令。

⑿下一条指令地址为 0x\_\_00000014\_\_\_\_\_\_\_\_\_\_，是一条保存\_\_\_\_字\_\_\_\_\_(字节．半字，字)指令。

⒀单步执行 1 条指令(F7)。

⒁查看内存 BUFFER 处字的值，值为 0x\_\_\_\_00000080\_\_\_\_。（内存—>符号表）

(7)执行算术运算类指令。步骤如下：

①双击“寄存器”窗口中的 R1，将其值修改为 2。 ②双击“寄存器”窗口中的 R2，将其值修改为 3。

③单步执行 1 条指令。

④下一条指令地址为 0x\_\_\_00000020\_\_\_\_，是一条加法指令。

⑤单步执行 1 条指令。

⑥查看 R3 的值，[R3]=0x\_\_\_\_0000000000000005\_\_\_\_\_\_\_

⑦下一条指令地址为 0x\_\_\_\_\_\_\_00000024\_\_\_\_\_\_\_\_\_，是一条乘法指令。

⑧单步执行 1 条指令。

⑨查看 L0、HI 的值，[LO]=0x\_0000000000000006\_\_\_\_\_，[HI]=0x\_\_\_0000000000000000\_\_\_\_\_

(8)执行逻辑运算类指令。步骤如下：

①双击“寄存器”窗口中的 R1，将其值修改为 0xFFFF0000。

②双击“寄存器”窗口中的 R2，将其值修改为 0xFF00FF00。

③单步执行 1 条指令。

④下一条指令地址为 0x\_\_\_ 00000030\_\_\_，是一条逻辑与运算指令，第二个操作数寻址 方式是\_\_\_寄存器直接寻址\_\_(寄存器直接寻址，立即数寻址)。

⑤单步执行 1 条指令。

⑥查看 R3 的值，[R3]=0x\_\_ 00000000FF000000 \_\_\_\_

⑦下一条指令地址为 0x\_\_00000034\_\_\_，是一条逻辑与运算指令，第二个操作数寻址 方式是\_\_立即数寻址\_\_\_\_\_（寄存器直接寻址，立即数寻址）。

⑧单步执行 1 条指令。

⑨查看 R3 的值，[R3]=0x\_\_\_\_ 0000000000000000 \_\_

(9)执行控制转移类指令。步骤如下：

①双击“寄存器”窗口中的 R1，将其值修改为 2。

②双击“寄存器”窗口中的 R2，将其值修改为 2。

③单步执行 1 条指令。

④下一条指令地址为 0x00000040\_\_\_\_\_\_，是一条 BEQ 指令，其测试条件是\_\_\_\_相等转移\_\_， 目标地址为 0x0000004c.

⑤单步执行 1 条指令。

⑥查看 PC 的值，[PC]=0x\_\_0000004c\_\_\_，表明分支\_\_成功\_\_\_（成功，失败）。

⑦下一条指令是一条 BGEZ 指令，其测试条件是\_大于等于0转移\_\_，目标地址为 0x\_\_00000058\_\_\_\_

⑧单步执行 1 条指令。

⑨查看 PC 的值，[PC]=0x\_\_\_00000058\_\_\_，表明分支\_\_\_成功\_\_\_\_\_（成功，失败）。

⑩下一条指令是一条 BGEZAL 指令，其测试条件是\_ 大于等于0转移并链接\_\_，目标地址为 0x\_\_00000064\_\_\_\_\_\_

⑾单步执行 1 条指令。

⑿查看 PC 的值[PC]=0x\_\_00000064\_\_\_，表明分支\_\_\_\_成功\_\_\_\_（成功，失败）；查看 R31 的值，[R31]=0x\_\_\_000000000000005C\_\_\_\_\_

⒀单步执行 1 条指令。

⒁查看 R1 的值，[R1]=0x\_0000000000000074\_\_\_\_

⒂下一条指令地址为 0x\_\_\_00000068\_\_，是一条 JALR 指令，保存目标地址的寄存器 为 R1\_，保存返回地址的目标寄存器为 R31\_\_。



⒃单步执行 1 条指令。

⒄查看 PC 和 R3 的值，[PC]=0x00000074\_，[R3]=0x\_000000000000006C\_\_