

实验六-RISC-V 汇编实验

姓名：张云策 学号：3200105787 学院：计算机科学与技术学院

课程名称：计算机系统 1 同组学生姓名：/

实验时间：2021.04.22 实验地点：紫金港东 4-509 指导老师：吴磊

一、实验目的和要求

- 理解 RISC-V 汇编指令
- 掌握 RISC-V 寄存器

二、实验内容和原理

2.1 实验内容

1. 输入样例汇编，运行汇编
2. 单步调试，给出每次 x1 数值变化的截图
3. 回答 x7 最终的数值

2.2 设计模块

2.1 汇编语言

汇编语言（Assembly Language）是任何一种用于电子计算机、微处理器、微控制器或其他可编程器件的低级语言，亦称为符号语言。在汇编语言中，用助记符代替机器指令的操作码，用地址符号或标号代替指令或操作数的地址。在不同的设备中，汇编语言对应着不同的机器语言指令集，通过汇编过程转换成机器指令。特定的汇编语言和特定的机器语言指令集是一一对应的，不同平台之间不可直接移植。

2.2 RISC-V

RISC-V（发音为“risk-five”）是一个基于精简指令集（RISC）原则的开源指令集架构（ISA），简易解释为开源软件运动相对应的一种“开源硬件”。该项目 2010 年始于加州大学柏克莱分校，但许多贡献者是该大学以外的志愿者和行业工作者。与大多数指令集相比，RISC-V 指令集可以自由地用于任何目的，允许任何人设计、制造和销售 RISC-V 芯片和软件而不必支付给任何公司专利费。虽然这不是第一个开源指令与功率效率。该指令集还具有众多支持的软件，这解决了新指令集通常的弱点。RISC-V 指令集的设计考虑了小型、快速、低

功耗的现实情况来实做，但并没有对特定的微架构做过度的设计。

三、 主要仪器设备

在线 RISC-V 汇编模拟器 or 本地 spike 模拟器。

四、 操作方法与实验步骤

4.1 操作方法

将 demo 代码放入模拟器中进行生成。

4.2 实验步骤

略

五、 实验结果与分析

结果：
RISC-V Interpreter

Input your RISC-V code here:

```
1 # riscv
2 initial:
3 lui x6, 0x666
4 addi x1, x0, 2
5 addi x2, x1, 14
6 sw x1, 8(x2)
7 li:
8 ori x3, x1, 4
9 slli x1, x1, 4
10 lw x4, -8(x1)
11 srai x1, x1, 1
12 beq x1, x2, 12
13 addi x6, x6, -1
14 li:
15 addi x1, x0, 2
16 bne x4, x1, 11
17 ..
18
```

Reset

Step

Run

CPU: 32 Hz

[line 11]: srai x1, x1, 1

[line 12]: beq x1, x2, 12

[line 15]: addi x1, x0, 2

[line 16]: bne x4, x1, 11

[line 18]: add x7, x2, x2

No more instructions to run! Press .Reset to reload the code!

Init Value	Register	Decimal	Hex	Binary
0	x0 (zero)	0	0x00000000	0b00000000000000000000000000000000
0	x1 (ra)	2	0x00000002	0b00000000000000000000000000000010
0	x2 (sp)	16	0x00000010	0b0000000000000000000000000000010000
0	x3 (gp)	6	0x00000006	0b0000000000000000000000000000000110
0	x4 (tp)	2	0x00000002	0b000000000000000000000000000000010
0	x5 (t0)	0	0x00000000	0b00000000000000000000000000000000
0	x6 (t1)	6709248	0x00666000	0b00000000011001100110000000000000
0	x7 (t2)	32	0x00000020	0b0000000000000000000000000000010000
0	x8 (s0/fp)	0	0x00000000	0b00000000000000000000000000000000
0	x9 (s1)	0	0x00000000	0b00000000000000000000000000000000
0	x10 (a0)	0	0x00000000	0b00000000000000000000000000000000
0	x11 (a1)	0	0x00000000	0b00000000000000000000000000000000
0	x12 (a2)	0	0x00000000	0b00000000000000000000000000000000
0	x13 (a3)	0	0x00000000	0b00000000000000000000000000000000
0	x14 (a4)	0	0x00000000	0b00000000000000000000000000000000
0	x15 (a5)	0	0x00000000	0b00000000000000000000000000000000
0	x16 (a6)	0	0x00000000	0b00000000000000000000000000000000
0	x17 (a7)	0	0x00000000	0b00000000000000000000000000000000
0	x18 (s2)	0	0x00000000	0b00000000000000000000000000000000

X1 数值变化: start 0 2 2 2 2 32 32 16 16 2 2 final

0

x1 (ra)

2

0x00000002

0b00000000000000000000000000000010

