

汇编程序设计

李卓 pb19000064

实验目标

熟悉 RISC-V 汇编指令的格式

熟悉 CPU 仿真软件 Ripes, 理解汇编指令执行的基本原理 (数据通路和控制器的协调工作过程)

熟悉汇编程序的基本结构, 掌握简单汇编程序的设计

掌握汇编仿真软件 RARS(RISC-V Assembler & Runtime Simulator)的使用方法, 用该软件进行汇编程序的仿真、调试以及生成 CPU 测试需要的指令和数据文件 (COE)

理解 CPU 调试模块 PDU 的使用方法

实验内容:

1.理解并仿真 RIPES 示例汇编程序

加载 Ripes 示例汇编程序 (Console Printing)→选择单周期 CPU 数据通路→单步执行程序→观察数据通路控制信号和寄存器内容的变化

2.设计汇编程序, 验证 6 条指令功能

Rars 软件设计汇编程序→单步运行程序→人工检查→生成 COE 文件

sw, lw

add, addi

beq, jal

备注: 通过查看数据存储器 and 32 个通用寄存器来实现人工检查

实验过程:

生成斐波那契数列

```
.data
array: .word 1,2
.text
    la t0 array
    lw a0 0(t0)
    lw a1 4(t0)

loop:
    addi t0 t0 4
    addi t1 a0 0
    addi a0 a1 0
    add a1 a1 t1
    sw a0 0(t0)
    jal x0 loop
```

运行结果:

Text Segment					Labels	
Bkpt	Address	Code	Basic	Source	Label	Address
<input type="checkbox"/>	0x00003000	0xffffd297	auipc x5,0xffffffd	4: la t0 array	fls.asm	
<input type="checkbox"/>	0x00003004	0x00028293	addi x5,x5,0		loop	0x00003010
<input type="checkbox"/>	0x00003008	0x0002a503	lw x10,0(x5)	5: lw a0 0(t0)	array	0x00000000
<input type="checkbox"/>	0x0000300c	0x0042a583	lw x11,4(x5)	6: lw a1 4(t0)		
<input type="checkbox"/>	0x00003010	0x00428293	addi x5,x5,4	9: addi t0 t0 4		
<input type="checkbox"/>	0x00003014	0x00050313	addi x6,x5,0	10: addi t1 a0 0		
<input type="checkbox"/>	0x00003018	0x00058513	addi x10,x11,0	11: addi a0 a1 0		
<input type="checkbox"/>	0x0000301c	0x000585b3	add x11,x11,x6	12: add a1 a1 t1		
<input type="checkbox"/>	0x00003020	0x00a2a023	sw x10,0(x5)	13: sw a0 0(t0)		
<input type="checkbox"/>	0x00003024	0xfedff06f	jal x0,0xfffffec	14: jal x0 loop		

Data Segment								
Address	Value (+0)	Value (+4)	Value (+8)	Value (+c)	Value (+10)	Value (+14)	Value (+18)	Value (+1c)
0x00000000	0x00000001	0x00000002	0x00000003	0x00000005	0x00000008	0x0000000d	0x00000015	0x00000022
0x00000020	0x00000037	0x00000059	0x00000090	0x000000e9	0x00000179	0x00000262	0x000003db	0x0000063d
0x00000040	0x00000a18	0x00001055	0x00001a6d	0x00002ac2	0x0000452f	0x00006ff1	0x0000b520	0x00012511
0x00000060	0x0001da31	0x0002ff42	0x0004d973	0x0007d8b5	0x000cb228	0x00148add	0x00213d05	0x0035c7e2
0x00000080	0x005704e7	0x008cccc9	0x00e3d1b0	0x01709e79	0x02547029	0x03c50ea2	0x06197ecb	0x09de8d6d
0x000000a0	0x0ff80c38	0x19d699a5	0x29cea5dd	0x43a53f82	0x6d73e55f	0xb11924e1	0x1e8d0a40	0xcfa62f21
0x000000c0	0xee333961	0xbdd96882	0xac0ca1e3	0x69e60a65	0x15f2ac48	0x7fd8b6ad	0x95cb62f5	0x15a419a2
0x000000e0	0xab6f7c97	0xc1139639	0x6c8312d0	0x2d96a909	0x9a19bd99	0xc7b064e2	0x61ca20bb	0x297a859d
0x00000100	0x8b44a658	0xb4bf2bf5	0x4003d24d	0xf4c2fe42	0x34c6d08f	0x2989ced1	0x5e509f60	0x87da6e31
0x00000120	0xa62b0d91	0x6a057bc2	0x54308953	0xc2360515	0x1a668a68	0xd89c937d	0xaf0321a5	0xc79fb5a2

生成的 coe 文件

fls.txt - 记事本

文件(F) 编辑(E) 格式(O) 查看(V)

```

00000001
00000002
00000003
00000005
00000008
0000000d
00000015
00000022
00000037
00000059
00000090
000000e9
00000179
00000262
000003db
0000063d
00000a18
00001055
00001a6d
00002ac2
0000452f
00006ff1
0000b520
00012511
0001da31
0002ff42
0004d973
0007d8b5
000cb228
00148add
00213d05
0035c7e2
005704e7
008cccc9

```

测试指令：

```
1 .data
2 out: .word 0xff      #led, 0x00000000
3 in: .word 0          #switch
4
5 .text
6 loop:
7 la a0, out           #0x00000000
8 sw x0, 0(a0)         #test sw: 0x00000000
9 addi t0, x0, 0xff    #test addi: 0x000000ff
10 sw t0, 0(a0)
11 lw t0, 4(a0)        #test lw: 0x00000000
12 sw t0, 0(a0)
13 addi t0, x0, 0x01
14 add t0, t0, t0
15 beq x0, t0, loop
16 jal loop
17 |
```

测试结果

register

Registers		
Name	Alias	Value
x0	zero	0x00000000
x1	ra	0x0000002c
x2	sp	0x7fffffff
x3	gp	0x10000000
x4	tp	0x00000000
x5	t0	0x00000002
x6	t1	0x00000000
x7	t2	0x00000000
x8	s0	0x00000000
x9	s1	0x00000000
x10	a0	0x00000000
x11	a1	0x00000000
x12	a2	0x00000000
x13	a3	0x00000000
x14	a4	0x00000000
x15	a5	0x00000000
x16	a6	0x00000000
x17	a7	0x00000000
x18	a8	0x00000000
x19	a9	0x00000000
x20	a10	0x00000000
x21	a11	0x00000000
x22	a12	0x00000000
x23	a13	0x00000000
x24	a14	0x00000000
x25	a15	0x00000000

数据段

Idre	Word	Byte 0	Byte 1	Byte 2	Byte 3
0x0...	0x00000000	0x00	0x00	0x00	0x00
0x0...	0xfd9ff0ef	0xef	0xf0	0x9f	0xfd
0x0...	0xfc500ee3	0xe3	0x0e	0x50	0xfc
0x0...	0x005282b3	0xb3	0x82	0x52	0x00
0x0...	0x00100293	0x93	0x02	0x10	0x00
0x0...	0x00552023	0x23	0x20	0x55	0x00
0x0...	0x00452283	0x83	0x22	0x45	0x00
0x0...	0x00552023	0x23	0x20	0x55	0x00
0x0...	0x0ff00293	0x93	0x02	0xf0	0x0f
0x0...	0x00052023	0x23	0x20	0x05	0x00
0x0...	0x00050513	0x13	0x05	0x05	0x00
0x0...	0x10000517	0x17	0x05	0x00	0x10
-	-	-	-	-	-
-	-	-	-	-	-
-	-	-	-	-	-
-	-	-	-	-	-
-	-	-	-	-	-
-	-	-	-	-	-
-	-	-	-	-	-
-	-	-	-	-	-
-	-	-	-	-	-
-	-	-	-	-	-
-	-	-	-	-	-

Display type: Hex Go to register: Go to section: