

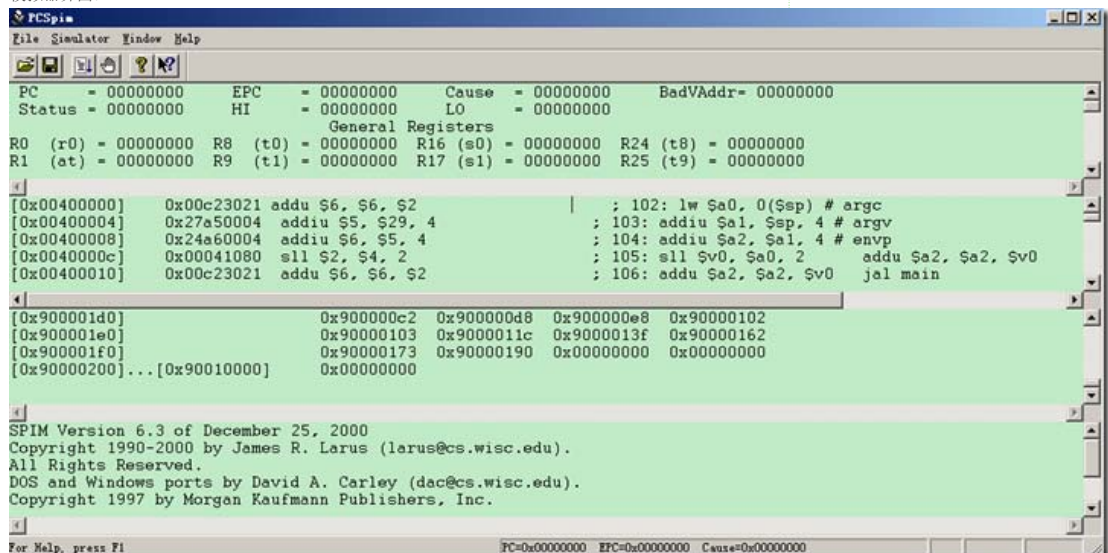
hustlijian

博客园 首页 博文 闪存 新随笔 订阅 XML 管理

posts - 40, comments - 8, trackbacks - 0

MIPS指令学习

- MIPS指令知识的学习：
 - 书本:《计算机组成原理》
 - 网络:
 - [http://gic1.cs.drexel.edu/people/sevy/architecture/MIPSRef\(SPIM\).html](http://gic1.cs.drexel.edu/people/sevy/architecture/MIPSRef(SPIM).html)
 - <http://courses.missouristate.edu/KenVollmar/MARS/Help/SyscallHelp.html>
- 环境配置
 - 从ftp://ftp.cs.wisc.edu目录PUB/SPIM中下载SPIM
 - 运行pcspim.exe解压缩文件
 - 在解压缩后的文件夹中运行SETUP.EXE安装
 - 转到安装目录下运行pcspim.exe
 - 模拟器界面:



- 按照MIPS指令规则编写汇编代码，（后缀名使用asm, s）用PCSpim打开运行。
- 程序结构学习（从hello world开始）
 - 源代码(hello.asm)

```

1: # author : See-See
2: #   text segment
3:   .text
4:   .globl main
5: main:                # execution starts here
6:   la $a0,str          # put string address into a0
7:   li $v0,4            # system call to print
8:   syscall             # out a string
9:   li $v0,10
10:  syscall             # exit
11: #data segment
12:   .data
13: str:  .asciiz "hello world\n"

```

- 效果图



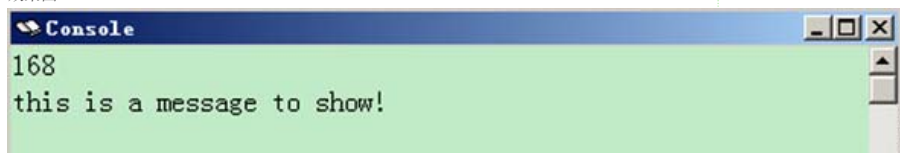
- 程序设计实例
 - 输出实例
 - 源代码

```

1: # author : lijian
2: # date: 2012-01-04 21:00:30
3: # function: use mips output
4:
5:     .data
6: # variable declarations here
7: msg:     .asciiz      "\nthis is a message to show!\n" #declared
for string
8: inter:   .word       168
#declared a interger
9: char:    .byte       'a'
#declared a character
10:
11:
12:     .text
13: main:                # indicates start of code
14:
15: # show interger
16:     li    $v0, 1          # $v0 <= 1
17:     lw    $a0, inter      # $a0 <= inter
18:     syscall
19: # show string
20:     li    $v0, 4          # $v0 <= 4
21:     la    $a0, msg        # $a0 <= msg
22:     syscall
23:
24:     li    $v0, 10         #system call code for exit = 10
25:     syscall              #call operating system to exit

```

- 效果图



- 输入实例
 - 源代码

```

1: # author : lijian
2: # date: 2012-01-04 21:35:31
3: # function: use mips input
4:
5:     .data
6: # variable declarations here
7: msg:     .space 40        #allocate 40 consecutive bytes,
8: msg1:    .asciiz         "\ninput a integer: " # declared for string
9: msg2:    .asciiz         "\ninput a string: "  # declared for string
10: msg3:    .asciiz         "\nyou input: "      # declared for
string
11:     .text
12: main:                # indicates start of code
13:
14: # input integer
15:     li    $v0, 4
16:     la    $a0, msg1
17:     syscall
18:     li    $v0, 5          #read integer
19:     syscall
20:     move   $a1, $v0

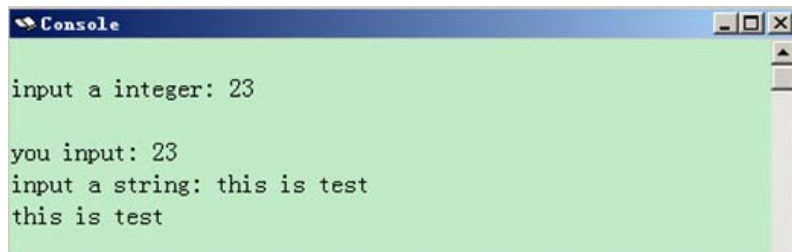
```

```

21:    li    $v0, 4
22:    la    $a0, msg3
23:    syscall
24:    li    $v0, 1
25:    move   $a0, $a1
26:    syscall
27:
28:    li    $v0, 4
29:    la    $a0, msg2
30:    syscall
31:    li    $v0, 8
32:    la    $a0, msg
33:    li    $a1, 40
34:    syscall
35:
36:    li    $v0, 4
37:    la    $a0, msg
38:    syscall
39:
40:    li    $v0, 10                #system call code for exit = 10
41:    syscall                    #call operating system to exit

```

- 效果图



- 代数式运算
 - 源代码

```

1: # author : lijian
2: # date: 2012-01-04 22:09:55
3: # fuction: use mips to count
4:
5:     .data
6: newline:    .asciiz    "\n"
7: line:       .asciiz    "*****\n"
8: head:       .asciiz    "        count numbers    \n"
9: author:     .asciiz    "        author : lijian    \n"
10: version:    .asciiz    "        version: 0.1        \n"
11: first:      .asciiz    "input the first num: "
12: second:     .asciiz    "input the second num: "
13: addString:  .asciiz    " + "
14: subString:  .asciiz    " - "
15: mulString:  .asciiz    " * "
16: divString:  .asciiz    " / "
17: equalString: .asciiz    " = "
18:
19:     .text
20: main:
21: # show menu
22:     li    $v0, 4
23:     la    $a0, line
24:     syscall
25:     li    $v0, 4
26:     la    $a0, head
27:     syscall

```

```
28:    li    $v0, 4
29:    la    $a0, newline
30:    syscall
31:    li    $v0, 4
32:    la    $a0, author
33:    syscall
34:    li    $v0, 4
35:    la    $a0, version
36:    syscall
37:    li    $v0, 4
38:    la    $a0, line
39:    # input first num
40:    syscall
41:    li    $v0, 4
42:    la    $a0, first
43:    syscall
44:    li    $v0, 5
45:    syscall
46:    move   $s0, $v0
47:    # input second num
48:    li    $v0, 4
49:    la    $a0, second
50:    syscall
51:    li    $v0, 5
52:    syscall
53:    move   $s1, $v0
54:    # show add result
55:    li    $v0, 1
56:    move   $a0, $s0
57:    syscall
58:    li    $v0, 4
59:    la    $a0, addString
60:    syscall
61:    li    $v0, 1
62:    move   $a0, $s1
63:    syscall
64:    li    $v0, 4
65:    la    $a0, equalString
66:    syscall
67:
68:    add $a0, $s0, $s1
69:    li    $v0, 1
70:    syscall
71:    li    $v0, 4
72:    la    $a0, newline
73:    syscall
74:
75:    # show sub result
76:    li    $v0, 1
77:    move   $a0, $s0
78:    syscall
79:    li    $v0, 4
80:    la    $a0, subString
81:    syscall
82:    li    $v0, 1
83:    move   $a0, $s1
84:    syscall
85:    li    $v0, 4
86:    la    $a0, equalString
87:    syscall
```

```

88:
89:     sub $a0, $s0, $s1
90:     li     $v0, 1
91:     syscall
92:     li     $v0, 4
93:     la     $a0, newline
94:     syscall
95:
96: # show mul result
97:     li     $v0, 1
98:     move    $a0, $s0
99:     syscall
100:    li     $v0, 4
101:    la     $a0, mulString
102:    syscall
103:    li     $v0, 1
104:    move    $a0, $s1
105:    syscall
106:    li     $v0, 4
107:    la     $a0, equalString
108:    syscall
109:
110:    mul $a0, $s0, $s1
111:    li     $v0, 1
112:    syscall
113:    li     $v0, 4
114:    la     $a0, newline
115:    syscall
116:
117: # show div result
118:    li     $v0, 1
119:    move    $a0, $s0
120:    syscall
121:    li     $v0, 4
122:    la     $a0, divString
123:    syscall
124:    li     $v0, 1
125:    move    $a0, $s1
126:    syscall
127:    li     $v0, 4
128:    la     $a0, equalString
129:    syscall
130:
131:    div $a0, $s0, $s1
132:    li     $v0, 1
133:    syscall
134:    li     $v0, 4
135:    la     $a0, newline
136:    syscall
137:
138:    li     $v0, 10                #system call code for exit = 10
139:    syscall                      #call operating system to exit

```

- 效果图

```

Console
*****
count numbers

author : lijian
version: 0.1
*****
input the first num: 44
input the second num: 22
44 + 22 = 66
44 - 22 = 22
44 * 22 = 968
44 / 22 = 2

```

- 求最大数, 和
- 源代码

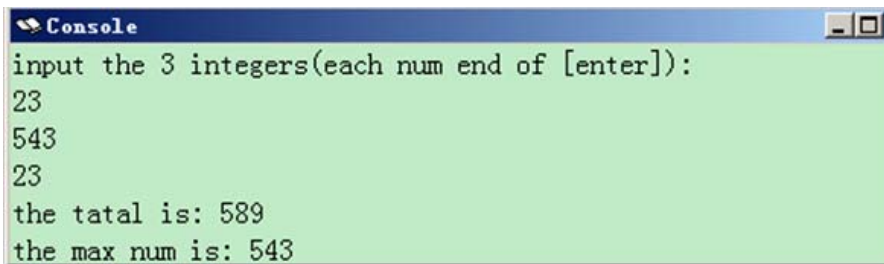
```

1: # author : lijian
2: # data: 2012-01-04 22:38:01
3: # function: find the max num and count the total
4:
5:     .data
6: array:     .space      12           #allocate 12 consecutive bytes
7: msg1:      .asciiz      "input the 3 integers(each num end of
[enter]): \n"
8: msg2:      .asciiz      "the max num is: "
9: msg3:      .asciiz      "the tatal is: "
10: newline:  .asciiz      "\n"
11:
12:     .text
13: main:
14:
15:     la      $t0, array
16:     li      $v0, 4
17:     la      $a0, msg1
18:     syscall
19:     li      $v0, 5
20:     syscall
21:     sw      $v0,($t0) # store the first num
22:     li      $v0, 5
23:     syscall
24:     sw      $v0,4($t0) # stroe the second num
25:     li      $v0, 5
26:     syscall
27:     sw      $v0,8($t0) # store the third num
28:
29:     lw      $s0, ($t0) # get the first num
30:     lw      $s1, 4($t0) # get the second num
31:     lw      $s2, 8($t0) # get the third num
32:
33:     add $s3, $s0, $s1
34:     add $s4, $s2, $s3 # total stored in $s4
35:     li      $v0, 4
36:     la      $a0, msg3
37:     syscall
38:     li      $v0, 1
39:     move     $a0, $s4
40:     syscall
41:     li      $v0, 4

```

```
42:    la    $a0, newline
43:    syscall
44:
45:    blt    $s0, $s1, num2
46:    move   $s3, $s0
47:    j      num3
48: num2:
49:    move   $s3, $s1
50: num3:
51:    bge    $s3, $s2, num4
52:    move   $s3, $s2
53: num4:
54:    li     $v0, 4
55:    la     $a0, msg2
56:    syscall
57:    li     $v0, 1
58:    move   $a0, $s3
59:    syscall
60:    li     $v0, 4
61:    la     $a0, newline
62:    syscall
63:
64:    li     $v0, 10                #system call code for exit = 10
65:    syscall                      #call operating system to exit
```

• 效果图



标签: [MIPS 汇编语言](#)

绿色通道: [好文要顶](#) [关注我](#) [收藏该文](#) [与我联系](#)



[hustlijian](#)

[关注 - 18](#)

[粉丝 - 9](#)

[+加关注](#)

0

0

(请您对文章做出评价)

« 上一篇: [windows shell](#)

» 下一篇: [VC 中 C2275问题解决](#)

posted on 2012-01-04 23:33 [hustlijian](#) 阅读(2172) 评论(0) [编辑](#) [收藏](#)

[刷新评论](#) [刷新页面](#) [返回顶部](#)

注册用户登录后才能发表评论, 请 [登录](#) 或 [注册](#), [访问网站首页](#)。

【推荐】50万行VC++源码: 大型组态工控、电力仿真CAD与GIS源码库

融云, 免费为你的App加入IM功能——让你的App“聊”起来!!



最新IT新闻:

- [陈一舟：人人网即将变成一家社交金融公司](#)
- [Facebook上线媒体内容创建工具Instant Articles](#)
- [乐视体育宣布融资8亿元 成立一年估值达28亿](#)
- [微软今年发售55英寸Surface Hub 针对企业用户](#)
- [阿里云与宝船网合作：超过30万艘船舶数据接入](#)
- » [更多新闻...](#)



最新知识库文章:

- [解析微服务架构（一）单块架构系统以及其面临的挑战](#)
- [说说领域驱动设计和贫血、失血、充血模型](#)
- [贫血，充血模型的解释以及一些经验](#)
- [如何避免软件工程中最昂贵错误的发生](#)
- [面向服务体系和遗留系统](#)
- » [更多知识库文章...](#)

Copyright ©2015 hustlijian Powered By[博客园](#) 模板提供: [沪江博客](#)

昵称: [hustlijian](#)

园龄: [4年4个月](#)

粉丝: [9](#)

关注: [18](#)

[+加关注](#)

2012年1月						
≤	日	一	二	三	四	五
	25	26	27	28	29	30
	1	2	3	4	5	6
	8	9	10	11	12	13
	15	16	17	18	19	20
	22	23	24	25	26	27
	29	30	31	1	2	3

搜索

<input type="text"/>	<input type="button" value="找找看"/>
<input type="text"/>	<input type="button" value="谷歌搜索"/>

常用链接

[我的随笔](#)
[我的评论](#)
[我的参与](#)
[最新评论](#)
[我的标签](#)

我的标签

[汇编语言\(4\)](#)
[汇编语言入门](#) [流程控制](#) [输出十进制数字\(1\)](#)
[汇编语言入门](#) [分支、循环流程控制](#) [十进制输出整数\(1\)](#)
[技巧\(1\)](#)
[爬虫\(1\)](#)
[软件测试\(1\)](#)
[五笔\(1\)](#)
[效率\(1\)](#)
[音乐\(1\)](#)
[职业选择\(1\)](#)
[更多](#)

随笔分类

[Assembly\(5\)](#)
[C\(1\)](#)
[C++](#)
[Unix/Linux\(4\)](#)
[Windows](#)
[软件工具\(4\)](#)
[软件开发\(4\)](#)
[数据结构\(1\)](#)

随笔档案

[2015年4月 \(1\)](#)
[2015年3月 \(3\)](#)

[2014年6月 \(1\)](#)
[2014年3月 \(3\)](#)
[2013年9月 \(1\)](#)
[2013年8月 \(1\)](#)
[2013年5月 \(3\)](#)
[2013年4月 \(4\)](#)
[2012年11月 \(4\)](#)
[2012年10月 \(1\)](#)
[2012年1月 \(1\)](#)
[2011年12月 \(1\)](#)
[2011年11月 \(1\)](#)
[2011年9月 \(1\)](#)
[2011年8月 \(1\)](#)
[2011年7月 \(5\)](#)
[2011年6月 \(3\)](#)
[2011年5月 \(5\)](#)

最新评论

[1. Re:VPS安装记](#)
@天天吃饭呢不要黑！ @_@...
--hustlijian

[2. Re:VPS安装记](#)
赞赞赞！剑神威武！
--天天吃饭呢

[3. Re:汇编语言学习：Debug的使用](#)
有用，学习了。
--shenlei

阅读排行榜

[1. ntp时间服务器安装\(4184\)](#)
[2. AutoHotKey入门使用\(2961\)](#)
[3. MIPS指令学习\(2172\)](#)
[4. 80X86指令总结\(2097\)](#)
[5. keystone命令与client接口学习\(1695\)](#)

评论排行榜

[1. kfs\(kosmosfs\)编译安装运行\(2\)](#)
[2. VPS安装记\(2\)](#)
[3. VC 中 C2275问题解决\(1\)](#)
[4. 汇编语言 文件扩展名修改程序资料\(1\)](#)
[5. 汇编语言学习：Debug的使用\(1\)](#)

推荐排行榜

[1. 如何谋得一个好职位\(2\)](#)
[2. AutoHotKey入门使用\(1\)](#)
[3. 汇编语言 文件扩展名修改程序资料\(1\)](#)
[4. Unix/Linux 实用技巧、箴言\(1\)](#)
[5. 汇编语言学习：子程序设计、子模块调用\(1\)](#)