

第一节

2025.2.27

- 1、一个自定义16位RISC CPU的ISA设计。自定义16位RISC CPU的内容包括两方面：
- (一) 16个16位通用寄存器，r0-r15
 - (二) 16条指令及其格式：
 - a) 跳转 `jal rd, imm` ; `jalr rd,rs1,imm`
 - b) 分支 `beq,ble op rs1,rs2,offset`
 - c) 访存 `lb,lw op rd,offset(rs1)`; `sb,sw op rs2,offset(rs1)`。
 - d) 算数和逻辑运算
 - `add,sub,and,or op rd,rs1,rs2`; |
 - `addi,subi,andi,ori op rd,rs1, imm`

16个16位通用寄存器：r0-r15，其中r0为恒0寄存器，r1为返回地址寄存器ra，r2为栈指针寄存器sp，其余为运算寄存器a0-a12，其中a0还作为保存函数参数或返回值。

写具体含义和机器码

机器码顺序：偏移量_目的操作数_源操作数_操作指令

```
1 #目的：认识指令的具体含义并尝试自己写出对应的机器码
2 #机器码顺序：偏移量（imm/offset）_目的操作数_源操作数_操作指令
3
4 0.jal rd,imm:
5 ...
6 # 将PC的值加上2，结果写入rd寄存器，rd默认为r1，同时将PC的值设置为PC加上符号位拓展的imm，即PC=PC+sext(imm)。
```

```

7  #小端模式, small endian mode
8      0000 0100(imm) 0001(rd) 0000(jal)
9  机器码: 16'b0000_0100_0001_0000
10
11  '''
12
13  1.jalr rd,rs1,imm:
14  '''
15  #将PC的值加上2, 结果写入rd寄存器, rd默认为r1, 同时将PC值设置为寄存器rs1的值加上符号位拓展的imm, 即PC=rs1+sext(imm)。
16  # 0100(imm) 0011(rs1) 0001(rd) 0001(jalr)
17  # 机器码: 16'b0100_0011_0001_0001
18  '''
19
20  2.beq rs1,rs2,offset
21  '''
22  如果寄存器rs1(a1:0100)和rs2(a2:0101)的值相等, 那么跳转, 偏移量为offset(6), 否则执行下一条(pc = pc + 2)
23  举例:
24  if a1 == a2:
25      PC = PC + 6
26  else:
27      pc = pc + 2
28  '''
29  机器码: 16'b 0110_0101_0100_0010
30
31  3.ble rd, rs2, offset
32  '''
33  如果寄存器rs1(a1)的值小于等于rs2(a2)的值, 则跳转, 偏移量为offset(6), 否则执行下一条(pc = pc + 2)
34  举例:
35  if a1 <= a2:
36      PC = pc + 6
37  else:
38      pc = pc + 2
39  '''
40  机器码: 16'b0110_0101_0100_0011
41
42  4.lb rd, offset(rs1)
43  '''
44  从寄存器rs1中获得基础地址(a1:0100), a1加上偏移量offset(6)后得到内存地址, 然后将内存地址存放的一字节数据加载到寄存器rd(a2)中
45  举例:
46  ad = a1 + 6
47  a2 = (ad)
48  '''
49  机器码: 16'b0110_0101_0100_0100
50
51  5.lw rd, offset(rs1)
52  '''
53  从寄存器rs1中获得基础地址(a1:0100), a1加上偏移量offset(6)后得到内存地址, 然后将内存地址存放的一个字(2字节)数据加载到寄存器rd(a2)中
54  举例:
55  ad = a1 + 6
56  a2 = (ad)
57  '''

```

```

58  机器码: 16'b0110_0101_0100_0101
59
60  6.sb rs1, offset(rs2)
61  '''
62  从寄存器rs2(a2)中获得基址, a2加上偏移量offset(6)后得到内存地址, 然后将寄存器
    rs1(a1:0100)中存放的一字节数据存放到内存地址中
63  举例:
64  ad = a2 + 6
65  (ad) = a1
66  '''
67  机器码: 16'b0110_0101_0100_0110
68
69  7.sw rs1, offset(rs2)
70  '''
71  从寄存器rs2(a2)中获得基址, a2加上偏移量offset(6)后得到内存地址, 然后将寄存器
    rs1(a1:0100)中存放的一个字(2字节)数据存放到内存地址中
72  举例:
73  ad = a2 + 6
74  (ad) = a1
75  '''
76  机器码: 16'b0110_0101_0100_0111
77
78  8.add rd,rs1,rs2
79  '''
80  将寄存器rs1(r0)和rs2(r1)中的值相加, 然后存放到rd(r2)中
81  add r2, r0, r1
82  举例:
83  r2 = r0 + r1
84  '''
85  机器码: 16'b0010_0001_0000_1000
86
87  9.sub rd,rs1,rs2
88  '''
89  将寄存器rs1(r0)和rs2(r1)中的值相减, 然后存放到rd(r2)中
90  sub r2, r0, r1
91  举例:
92  r2 = r0 - r1
93  '''
94  机器码: 16'b0010_0001_0000_1001
95
96
97  10.and rd,rs1,rs2
98  '''
99  将寄存器rs1(r0)和rs2(r1)进行逻辑并运算, 然后将结果存放到rd(r2)中
100 and r2, r0, r1
101  举例:
102 r2 = r0 & r1
103 '''
104 机器码: 16'b0010_0001_0000_1010
105
106 11.or rd,rs1,rs2
107  '''
108 将寄存器rs1(r0)和rs2(r1)进行逻辑或运算, 然后将结果存放到rd(r2)中
109 or r2, r0, r1
110 举例:
111 r2 = r0 | r1

```

```
112 '''
113 机器码: 16'b0010_0001_0000_1011
114
115 12.addi rd,rs1,imm
116 '''
117 将寄存器rs1(r0)和imm(6)中的值相加, 然后存放到rd(r1)中
118 举例:
119 r2 = r0 + 6
120 '''
121 机器码: 16'b0110_0001_0000_1100
122
123 13.subi rd,rs1,imm
124 '''
125 将寄存器rs1(r0)的值和imm(6)相减, 然后存放到rd(r2)中
126 举例:
127 r2 = r0 - 6
128 '''
129 机器码: 16'b0110_0010_0000_1101
130
131
132 14.andi rd,rs1,imm
133 '''
134 将寄存器rs1(r0)和imm(6)进行逻辑并运算, 然后将结果存放到rd(r2)中
135 举例:
136 r2 = r0 & imm
137 '''
138 机器码: 16'b0110_0010_0000_1110
139
140 15.ori rd,rs1,imm
141 '''
142 将寄存器rs1(r0)和imm(6)进行逻辑或运算, 然后将结果存放到rd(r2)中
143 举例:
144 r2 = r0 | 6
145 '''
146 机器码: 16'b0110_0010_0000_1111
147
```

