

设每条指令执行时：

1) 相关寄存器的初始状态（机器数）为：

$$x5 = 0xffff\ ffff\ ffff\ 8765, \quad x6 = 0x0000\ 0000\ 0000\ 8765$$

$$x7 = 0xffff\ ffff\ ffff\ 5678$$

2) 相关内存数据区的初始状态（机器数）为：

字节单元地址	字节单元内容
.....	
0x0000 0000 0000 87a7	0x12
0x0000 0000 0000 87a8	0x34
0x0000 0000 0000 87a9	0x56
0x0000 0000 0000 87aa	0x78
0x0000 0000 0000 87ab	0x9a
0x0000 0000 0000 87ac	0xbc
0x0000 0000 0000 87ad	0xde
0x0000 0000 0000 87ae	0xf0
0x0000 0000 0000 87af	0x12
.....	

3) 标号 Label 代表目标指令的地址（目标地址，是字节地址、无符号数）。

请独立分析下列每条指令的执行结果（16 进制机器数），填写下列表格。

1) 寄存器 x5 和 PC 的内容（顺序执行时，下条指令的地址可用  $PC + 4$  表示；跳跃执行时，下条指令的地址可用 Label 表示!）：

序号	汇编指令	结果： $x5 = ?$	结果： $PC = ?$
1	add x5, x6, x7	0xffff ffff ffff dddd	$PC + 4$

2	sub x5, x6, x7	0x0000 0000 0001 30ed	PC + 4
3	addi x5, x6, -20	0x0000 0000 0000 8751	PC + 4
<b>参考：</b> 其中的 immediate 字段是 12 位有符号数，运算时扩展为 64 位			
4	ld x5, 67(x6)	0x12f0 debc 9a78 5634	PC + 4
<b>参考：</b> RISC-V 采用小端模式			
5	sd x5, 67(x6)	0xffff ffff ffff 8765	PC + 4
<b>参考：</b> 并不修改寄存器 x5			
6	lw x5, 67(x6)	0xffff ffff 9a78 5634	PC + 4
<b>参考：</b> 读出的 32 位内容是有符号数，要扩展成 64 位			
7	lwu x5, 67(x6)	0x0000 0000 9a78 5634	PC + 4
<b>参考：</b> 读出的 32 位内容是无符号数，要扩展成 64 位			
8	sw x5, 67(x6)	0xffff ffff ffff 8765	PC + 4
9	lh x5, 67(x6)	0x0000 0000 0000 5634	PC + 4
10	lhu x5, 67(x6)	0x0000 0000 0000 5634	PC + 4
11	sh x5, 67(x6)	0xffff ffff ffff 8765	PC + 4
12	lb x5, 67(x6)	0x0000 0000 0000 0034	PC + 4
13	lbu x5, 67(x6)	0x0000 0000 0000 0034	PC + 4
14	sb x5, 67(x6)	0xffff ffff ffff 8765	PC + 4
15	lui x5, 0x12345	0x0000 0000 1234 5000	PC + 4
<b>参考：</b> 其中的 immediate 字段是 20 位有符号数			
16	and x5, x6, x7	0x0000 0000 0000 0660	PC + 4
17	or x5, x6, x7	0xffff ffff ffff d77d	PC + 4
18	xor x5, x6, x7	0xffff ffff ffff d11d	PC + 4
19	andi x5, x6, 20	0x0000 0000 0000 0004	PC + 4

**参考：** 其中的 immediate 字段是 12 位无符号数，运算时扩展为 64 位

20	ori x5, x6, 20	0x0000 0000 0000 8775	PC + 4
21	xori x5, x6, 20	0x0000 0000 0000 8771	PC + 4
22	sll x5, x7, x6	0x0000 0000 0000 0000	PC + 4
23	srl x5, x7, x6	0x0000 0000 0000 0000	PC + 4
24	sra x5, x7, x6	0xffff ffff ffff ffff	PC + 4
25	slli x5, x7, 5	0xffff ffff ffea cf00	PC + 4

**参考：** 其中的 immediate 字段是 6 位无符号数

26	srli x5, x7, 5	0x07 ff ffff ffff fab3	PC + 4
27	srai x5, x7, 5	0xffff ffff ffff fab3	PC + 4
28	beq x5, x6, Label	0xffff ffff ffff 8765	PC + 4

**参考：** 并不修改寄存器 x5

29	bne x5, x6, Label	0xffff ffff ffff 8765	Label
30	blt x5, x6, Label	0xffff ffff ffff 8765	Label
31	bge x5, x6, Label	0xffff ffff ffff 8765	PC + 4
32	bltu x5, x6, Label	0xffff ffff ffff 8765	PC + 4
33	bgeu x5, x6, Label	0xffff ffff ffff 8765	Label
34	jal x5, Label	PC + 4	Label
35	jalr x5, 100(x5)	PC + 4	0xffff ffff ffff 87c9
36	auipc x5, 0x12345	PC + 0x0000 0000 1234 5000	PC + 4

**参考：** 其中的 immediate (offset) 字段是 20 位有符号数

37	slt x5, x6, x7	0x0000 0000 0000 0000	PC + 4
38	sltu x5, x6, x7	0x0000 0000 0000 0001	PC + 4
39	slti x5, x6, 100	0x0000 0000 0000 0000	PC + 4

**参考：** 其中的 immediate 字段是 12 位有符号数，比较时扩展为 64 位

40	sltiu x5, x6, 100	0x0000 0000 0000 0000	$PC + 4$
----	-------------------	-----------------------	----------

**参考：** 其中的 immediate 字段是 12 位无符号数，比较时扩展为 64 位

2) 内存相关数据区的状态：

	内存字节单元地址最后 8 位（高位部分省略!）							
	0xa8	0xa9	0xaa	0xab	0xac	0xad	0xae	0xaf
sd x5, 67(x6)	0x65	0x87	0xff	0xff	0xff	0xff	0xff	0xff
sw x5, 67(x6)	0x65	0x87	0xff	0xff	0xbc	0xde	0xf0	0x12
sh x5, 67(x6)	0x65	0x87	0x78	0x9a	0xbc	0xde	0xf0	0x12
sb x5, 67(x6)	0x65	0x56	0x78	0x9a	0xbc	0xde	0xf0	0x12