

第5周作业

3 (1) nop: addi x0, x0, 0 (5) csrrs rd, cycle, x0

(2) ret: jalr x0, x1, 0 (6) addiwn rd, rs, 0

(3) cuupe x6, offset [31:12]

jalr x1, x6, offset [11:0]

(4) addi rd, rs, 0

7 (1) SLTI t3, t1, 0

SLT t4, t0, t1

(2) add t0, t1, t2

BLTU t0, t1, overflow

(3) ARM体系结构中,通过cpsr的状态寄存器反映当前指令的溢出状态
MIPS通过指令触发中断的方式产生溢出信号,通知处理器

8 (1) Op rd, rs1, rs2 \times 0 $\overset{XLZV}{2}$ -1 \times -1 \times

会发生异常,除0的余数为被除数,当检测到除数为0时,触发异常处理流程,可避免计算结果异常并及时发现处理这种错误

(2) 0: 无效操作异常

3: 溢出异常

1: 非规格化操作数异常

4: 下溢异常

2: 除以0异常

5: 舍入误差异常

不会导致处理器陷入系统调用,相反,处理器可以检测这些标志位,并自动或手动地处理异常。

(3) x86: 在32位模式下,将抛出"#DE" (除法错误) 异常;在64位模式下,将抛出"#DE"

ARM: 会抛出"Division by zero"异常,并将cpsr寄存器的Z标志设置 或"#DE"异常
为1,表示结果为0



12 (1) 机器模式 m (2) 机器模式 m (3) 管理员模式 s
(4) 用户模式 u (5) 用户模式 u

```

13 : addi x11, x0, 100, # x11 = 100
    add x10, x0, x0, # x10 = i
Loop: bge x10, x11, exit
      sll x12, x10, 2
      add t0, t0, x12
      add t1, t1, x12
      lw a2, 0(t0)
      lw a3, 0(t1)
      mul a2, a3, t2
      addi x11, x11, 1
      j Loop
exit:
      lw a0, 0(a3)
  
```

```

14 bltu a1, a0, 1f # a2 = c a1 = 'b' a0 = a
    j 2f
1:
    add a2, a1, a0
    2:
    sub a3, a0, a1
  
```

```

15 lw t0, 0(t0)
    addi t1, x0, 3
  
```



↗ lw a, 0[x1]

addi x1, t0, 4

mul x2, t1, 4

addi x2, x2, t0

lw a, 0[x2]

16 lw a0, 0[t0]

lw a1, 0[t1]

sw a1, 0[t0]

sw a0, 0[t1]

17 先令 a0 为 0, a1 为 1, a2 为 30, 当 a0 < 30 时进行以下循环:

将 a1 左移 1 位 ($\times 2$), 将 a0 加 1, 最后结果为 a0 为 30, $a1 \times 2^{30}$

